

### Table of Contents

Table of contents are roughly in order of presentation. Papers in the proceedings are roughly in order of submission. [Whole 2010 proceedings as pdf](#) .

#### I. TUTORIALS

- [10-BRIMS-117 Cognitive Crash Dummies: Here today, look toward tomorrow](#)  
Bonnie John. 85-86.
- [10-BRIMS-009 Systems social science: A design inquiry approach for stabilization and reconstruction of social systems](#)  
Barry Silverman. 18.

#### II. PLENARY SPEAKERS

- [10-BRIMS-001 The shape of things to come: An emerging constellation of interconnected tools for developing the right cognitive model at the right scale](#)  
Wayne Gray. 1-2.
- [10-BRIMS-002 When hatred is bred in the bone: The psychocultural foundations of contemporary terrorism](#)  
Jerrold Post. 3.
- [10-BRIMS-003 Representing human behavior: Where to next?](#)  
LCDR Joseph V. Cohn. 4.
- [10-BRIMS-004 Intertemporal behavior: How people discount the future—  
Experimental data and formal representation](#)  
Robert Axtell. 5.

#### III. SPONSOR PRESENTATIONS

- [10-BRIMS-005 Army Research Lab](#)  
John Lockett. 6-7.
- [10-BRIMS-006 Air Force Research Lab](#)  
Kevin Gluck. 8-9.
- [10-BRIMS-007 Ministry of Defence \(UK\)](#)  
Bharatkumar Patel. 10-13.
- [10-BRIMS-008 NASA](#)  
Brian Gore. 14-17.

#### IV. PAPERS

- [10-BRIMS-119 Reducing the variability between novice modelers: Results of a tool for human performance modeling produced through human-centered design](#)  
Bonnie John. 95-102.
- [10-BRIMS-153 Plan, replan and plan to replan: Algorithms for robust courses](#)

## Report Documentation Page

*Form Approved  
OMB No. 0704-0188*

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE <b>MAR 2010</b>		2. REPORT TYPE		3. DATES COVERED <b>00-00-2010 to 00-00-2010</b>	
4. TITLE AND SUBTITLE <b>Proceedings of the 19th Conference on Behavior Representation in Modeling and Simulation, Charleston, SC, 21 - 24 March 2010</b>				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) <b>BRIMS Society, Inc, 3126 Alexander Pl # 208, Dayton, OH, 45431</b>				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT <b>Approved for public release; distribution unlimited</b>					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT <b>unclassified</b>	b. ABSTRACT <b>unclassified</b>	c. THIS PAGE <b>unclassified</b>			

of action under strategic uncertainty

Maciej Latek & Seyed Mussavi Rizi. 289-295.

- 10-BRIMS-111 Projecting grammatical features in nominals: Cognitive processing theory & computational implementation  
Jerry Ball. 55-62.
- 10-BRIMS-128 Modeling the control of attention in complex visual displays  
Kelly Steelman-Allen & Jason McCarley. 140-145.
- 10-BRIMS-131 Modeling behavioral activities related to IED perpetration  
Lora Weiss, Elizabeth Whitaker, Erica Briscoe & Ethan Trewhitt. 162-169.
- 10-BRIMS-108 Modeling complex social behavior: A system dynamics approach  
John Sokolowski & Catherine Banks. 37-46.
- 10-BRIMS-142 Process modeling for the study of non-state political violence  
Olivier Georgeon, Jonathan Morgan, John Horgan & Kurt Braddock. 240-247.
- 10-BRIMS-138 A hybrid model of ethnic conflict, repression, insurgency and social strife  
Stacy Lovell Pfautz & Michael Salwen. 211-215.
- 10-BRIMS-113 Dynamic data and modeling services suite  
Tina Chau, Alexander Moore, Richard Mullikin & Janet Wedgwood. 65-72.
- 10-BRIMS-103 Identification and application of neurophysiologic synchronies for studying the dynamics of teamwork  
Ronald Stevens, Trysha Galloway, Chris Berka & Adrienne Behneman. 21-28.
- 10-BRIMS-140 Improving usability and integration of human behavior representation engineering across cognitive modeling, human factors, and modeling and simulation best practices  
Sylvain Pronovost & Robert West. 224-231.
- 10-BRIMS-152 Extracting the ontological structure of OpenCyc for reuse and portability of cognitive models  
Bradley Best, Nathan Gerhart & Christian Lebiere. 282-288.
- 10-BRIMS-144 Taxonomy and method for handling large and diverse sets of interactive objects in immersive environment  
David Pietrocola & Barry Silverman. 256-262.
- 10-BRIMS-136 Shifts of critical personnel in network centric organizations  
Craig Schreiber, Kathleen Carley & Jeffrey Hansberger. 195-202.
- 10-BRIMS-151 An agent-based model of conflict in East Africa and the effect of watering holes  
William Kennedy, Atesmachew Hailegiorgis, Mark Rouleau, Jeffrey Bassett, Mark Coletti, Gabriel Balan & Tim Gulden. 274-281.
- 10-BRIMS-143 Resistance is futile: Winning lemonade market share through metacognitive reasoning in a three-agent cooperative game  
David Reitter, Ion Juvina, Andrea Stocco & Christian Lebiere. 248-255.
- 10-BRIMS-126 Modeling the theory of planned behavior from survey data for action choice in social simulations  
Jonathan Alt & Stephen Lieberman. 126-133.

- [10-BRIMS-137 Social identity modeling: Past work and relevant issues for socio-cultural modeling](#) [Best paper award]  
Jonathon Kopecky, Nathan Bos & Ariel Greenberg. 203-210.
- [10-BRIMS-130 Cognitive model exploration and optimization: A new challenge for computational science](#) [Recommended Read]  
L. Richard Moore Jr. 154-161.
- [10-BRIMS-145 Sailing to the model's Edge: Testing the limits of parameter space and scaling](#) [Recommended Read]  
Amy Santamaria & Walter Warwick. 263-269.
- [10-BRIMS-139 Modeling situation awareness for Army infantry platoon leaders using fuzzy cognitive mapping techniques](#) [Best paper award]  
Rashaad Jones, Erik Connors, Mary Mossey, John Hyatt, Neil Hansen & Mica Endsley. 216-223.
- [10-BRIMS-134 Modeling human eye-movements for military simulations](#) [Recommended Read]  
Patrick Jungkunz & Christian Darken. 179-186.
- [10-BRIMS-155 Adapting the taxon-task-taxon methodology to model the impacts of chemical protective gear](#) [Best paper award]  
Shane Mueller, George Anno, Corey Fallon, Gene McClellan, & Owen Price. 304-309.
- [10-BRIMS-129 Collaboration and modeling support in CogLaborate](#) [Best paper award]  
Reuben Cornel, Robert St. Amant & Jeff Shrager. 146-153.

## V. SHORT TALKS

- [10-BRIMS-125 Cognitive flexibility through learning from constraint violations](#)  
Dongkyu Choi & Stellan Ohlsson 118-125.
- [10-BRIMS-122 Human capacity development through simulations: Constructive simulations as a basis for understanding competency requirements in initiative based tactics](#)  
Bruno Emond. 110-117.
- [10-BRIMS-135 Rapid development of intelligent agents in first/third-person training simulations via behavior-based Control](#)  
George Alexander, Frederick Heckel, G. Michael Youngblood, D. Hunter Hale & Nikhil Ketkar. 187-194.
- [10-BRIMS-106 Developing a cognitive model of expert performance for ship navigation maneuvers in an intelligent tutoring System](#)  
Jason Wong, Susan Kirschenbaum & Stanley Peters. 29-36.
- [10-BRIMS-110 Plan ahead: Pricing ITS learner models](#)  
Jeremiah Folsom-Kovarik, Sae Schatz & Denise Nicholson. 47-54.
- [10-BRIMS-157 Subjective logic for composing utility functions from Maslow models](#)  
Nathan Denny. 312-316.
- [10-BRIMS-154 Data-driven coherence models](#)  
Peter Danenberg & Stacy Marsella. 296-303.

- [10-BRIMS-127 Agent frameworks for discrete event social simulations](#)  
Jonathan Alt & Stephen Lieberman. 134-139.
- [10-BRIMS-118 Cultural geography model validation](#)  
Lisa Jean Bair, Eric Weisel & Richard Brown. 87-94.
- [10-BRIMS-141 Procedure design and validation by cognitive task model Simulations](#)  
Tina Mioch, Tomasz Mistrzy & Frank Rister. 232-239.

## VI. SYMPOSIA/PANELS

- [10-BRIMS-159 Human behavior modeling in network science](#)  
Chairs: Sibel Adali & Jeffrey Hansberger; Panelists: William Wallace, Wayne Gray, Ching-Yung Lim & David Hachen. 317-318.
- [10-BRIMS-146 Results and lessons learned from the 2009 DSF Model Comparison Challenge](#)  
Chair: Walter Warwick; Panelists: Varun Dutt, Kevin Gluck & David Reitter. 270-271.

## VII. POSTERS

- [10-BRIMS-112 Modeling a visual search task with a secondary task in IMPRINT](#)  
Carolyn Buck-Gengler, William Raymond, Alice Healy, & Lyle Bourne, Jr. 63-64.
- [10-BRIMS-147 Applying the Human Behavior Architecture in the simulation of a networked command, control and communication structure](#)  
Walter Warwick. 272-273.
- [10-BRIMS-156 Prediction intervals for future performance](#)  
Kelly Addis, Michael Krusmark, Tiffany Jastrzembks, Kevin Gluck & Stuart Rodgers. 310-311.
- [10-BRIMS-161 Evaluating behavior modeling toolsets](#)  
Jennifer Pagan, Beth Wheeler Atkinson & Melissa Walwanis Nelson. 319-320.
- [10-BRIMS-114 Dynamic Data and Modeling Services Suite](#)  
Tina Chau, Alexander Moore, Richard Mullikin & Janet Wedgwood. 73-74.
- [10-BRIMS-166 Policy analysis using Q-learning](#)  
Ceyhun Eksin. 321-326
- [10-BRIMS-115 Tactical behavior composition](#)  
Evan Clark, Joel Eichelberger & Jeffrey Smith. 75-82.
- [10-BRIMS-121 Levy Distributed Search Behaviors for mobile target locating and tracking](#)  
William Lenagh & Prithviraj Dasgupta. 103-109.
- [10-BRIMS-133 Imperfect situation awareness: Representing error and uncertainty in modeling, simulation & analysis of small unit military operations](#)  
Victor Middleton. 170-178.

## VIII. DEMONSTRATIONS

- [10-BRIMS-116 CogTool: Predictive human performance modeling by demonstration](#)

Bonnie John. 83-84.

- o [10-BRIMS-101 Creating realistic human behavior in physical security systems simulation](#)  
Volkan Ustun & Jeffrey Smith. 19-20.

Papers can be cited as:

Georgeon, O. L., Morgan, J. H., Horgan, J., & Braddock, K. (2010). Process modeling for the study of non-state political violence. In *Proceedings of the 19th Conference on Behavior Representation in Modeling and Simulation*, 10-BRIMS-142. 240-247. BRIMS Society: Charleston, SC.

---

## **BRIMS CONFERENCE COMMITTEE 2010**

**Conference Chair:** Tiffany Jastrzembki, Air Force Research Laboratory

### **Executive Committee:**

Joe Armstrong, CAE, Publicity Committee

Sheila Banks, Calculated Insight

Brad Best, Adaptive Cognitive Systems, Technical Program Co-Chair

Brad Cain, Defence Research and Development Canada, Publicity Committee

Andrew Cowell, Pacific Northwest National Laboratory

Nathan Denny, 21st Century Systems, BRIMS Wiki Co-Lead

Uwe Dompke, NATO C3, Publicity Committee

Avelino Gonzalez, University of Central Florida

Coty Gonzalez, Carnegie Mellon University, Paper Review Co-Chair

Jeff Hansberger, Army Research Laboratory,

Troy Kelley, Army Research Laboratory

Bill Kennedy, George Mason University, Technical Program Co-Chair

Christian Lebiere, Carnegie Mellon University, Publicity Committee

Bharat Patel, Defence Science and Technology Laboratory, UK

Frank Ritter, Pennsylvania State University, Technical Program Co-Chair

Barry Silverman, University of Pennsylvania, Tutorial Chair

Lt Col David Sonntag, Asian Office of Aerospace Research and Development, BRIMS

Wiki Lead

Webb Stacy, Aptima, Publicity Committee

Michael Van Lent, SoarTech

Walter Warwick, Alion, Paper Review Co-Chair

Poster & Exhibits Program-Jeanne Eury/Lodestar Group

---

## SPONSORS

The BRIMS Conference thanks the following sponsors for their continued support for this year's Conference.

Air Force Research Laboratory (AFRL)

Army Research Institute (ARI)

Army Research Laboratory (ARL)

Defense Advanced Research Projects Agency (DARPA)

Office of Naval Research (ONR)

Natick Soldier Research, Development, & Engineering Center (NSC)

National Aeronautics and Space Administration (NASA)

UK Ministry of Defence (MoD)

---

[Copyright 2010, BRIMS Society, Inc.](#) Permission is hereby granted to quote any of the material herein, or to make copies thereof, for non-commercial purposes, as long as proper attribution is made and this copyright notice is included. All other uses are prohibited without written permission from the BRIMS Society, Inc.

Additional copies of this material can be obtained from the BRIMS Society, Inc., or from the BRIMS's web site at [www.brimskonference.org](http://www.brimskonference.org).

BRIMS would like to thank the following sponsors...



## WAYNE GRAY

Wayne Gray is a researcher in the fields of computational cognitive modeling, cognitive neuroscience, interactive behavior, cognitive task analysis, cognitive workload, and human error. Since earning his Ph.D. from UC Berkeley he has worked for government and industry research laboratories, as well as universities. He is currently a Professor of Cognitive Science at Rensselaer Polytechnic Institute. Wayne is a Fellow of the Human Factors & Ergonomics Society (HFES), the Cognitive Science Society, and the American Psychological Association (APA). In 2008, APA awarded him the Franklin V. Taylor Award for Outstanding Contributions in the Field of Applied Experimental & Engineering Psychology. He is a past Chair of the Cognitive Science Society and the founding Chair of the Human Performance Modeling technical group of HFES. At present he is the Executive Editor for the Cognitive Science Society's first new journal in 30 years, *Topics in Cognitive Science (topiCS)*.

~~~~~

### *The Shape of Things to Come: An Emerging Constellation of Interconnected Tools for Developing the Right Cognitive Model at the Right Scale*

There are at least three major problems with the current state of cognitive modeling. First, modeling is *too hard and takes too long*. There is a paucity of tools that allow you to set up a cognitive model at the same high level of abstraction that tools such as SPSS™ or SAS™ allow you to set up a complex statistical model for data analysis. Rather, most modeling formalisms require some computer science or mathematics training and typically each new model takes just as long to build as the last model. Second, cognitive modeling seems to engender the “*to a man with a hammer, everything looks like a nail*” syndrome. Once a modeling technique is mastered, too many people try to apply it to every situation whether or not it is the best tool for the current task. Third is *scale inflexibility* and a concomitant *lack of interconnectedness*. Modeling with any given technique locks you into a certain level of analysis. Popping up or down a level of analysis, say from a model of *reading with understanding* to a model of the *perception, eye movements, and memory involved in reading* requires abandoning one model and building another.

I will describe the *shape of things to come* by introducing two modeling tools and the *emerging constellation* that has resulted from their interconnectedness with each other and with the ACT-R (Anderson, 2007) architecture of cognition. The two tools, CogTool (John, Prevas, Salvucci, & Koedinger, 2004) and the *Stochastic Analysis Network Laboratory for Cognitive Modeling* (SANLab-CM, Patton & Gray, 2009) do not require the average user to have a background in computer science or mathematics. In contrast, modeling in ACT-R requires learning a specialized programming language. Although prior computer science or mathematics background is not strictly necessary, few modelers get very far without some training in these disciplines.

CogTool allows the modeler to create *Keystroke Level Models* (KLM, Card, Moran, & Newell, 1983) by demonstrating a sequence of moves in a storyboarded version of the task environment. The KLMs predict the performance times of expert users. It makes these predictions by creating and running a simple *ACT-R* model that uses default ACT-R parameters and the constraints imposed by the task environment.

SANLab-CM is the first tool designed to facilitate the development, manipulation, and comparison of activity network models for cognitive modeling. Examples of this type of modeling include CPM-GOMS (Gray, John, & Atwood, 1993; John, 1990) and the critical-path scheduling of mental processes (Schweickert, 1980; Schweickert, Fisher, & Proctor, 2003). Additionally, SANLab-CM is the first modeling tool that we know of specifically designed to explore the influence of stochasticity on cognitive outcomes. Whereas past CPM-GOMS models enabled the modeler to assign a fixed time to each operation, SANLab-CM enables the modeler to assign means and distributions of times. (Different types of operations may be assigned different default mean times and/or different default distributions. This is a

## WAYNE GRAY

feature, not a limit, as it is possible to assign times and distributions to individual operations.) When the resulting model is run, multiple *critical paths* are produced along with predictions of expected minimum and maximum response times. The utility of SANLab-CM will be demonstrated by comparing SANLab-CM models of Telephone Operator-Customer-Workstation interactions to the nonstochastic models of the same task built by Gray and John (Gray, et al., 1993).

CogTool, SANLab-CM, and ACT-R are interconnected. Whereas SANLab-CM can be used alone, it is possible to build a SANLab-CM model by importing the trace produced by running an ACT-R model. Once imported, SANLab-CM can be used to quickly explore the influence of different distributions (e.g., Gaussian versus gamma), different parameters of the distribution, or (to a limited degree) different designs of the task environment.

Likewise, SANLab-CM can be used in conjunction with CogTool. Running CogTool's simplified ACT-R model produces the KLM's predicted expert performance times. The trace produced by that model can be imported into SANLab-CM. Once in SANLab-CM it can be inspected, edited, manipulated, assigned various distributions, and run to inspect the various critical paths that would be produced by the stochastic activity network.

This is the shape of things to come. CogTool and SANLab-CM require no mathematical or computer science expertise to produce a model. Indeed, whereas SANLab-CM requires cognitive science expertise, CogTool does not. Each of these three tools, CogTool, SANLab-CM, and ACT-R can be used to develop models at different temporal scales so that a modeler who starts with one type of model can quickly develop another. The interconnectedness of SANLab-CM enables an emerging constellation of tools for developing the right model at the right scale.

Anderson, J. R. (2007). *How can the human mind occur in the physical universe?* New York: Oxford University Press.

Card, S. K., Moran, T. P., & Newell, A. (1983). *The psychology of human-computer interaction*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Gray, W. D., John, B. E., & Atwood, M. E. (1993). Project Ernestine: Validating a GOMS analysis for predicting and explaining real-world performance. *Human-Computer Interaction*, 8(3), 237-309.

John, B. E. (1990). *Extensions of GOMS Analyses to Expert Performance Requiring Perception of Dynamic Visual and Auditory Information*. Paper presented at the ACM CHI'90 Conference on Human Factors in Computing Systems, New York.

John, B. E., Prevas, K., Salvucci, D. D., & Koedinger, K. (2004). Predictive human performance modeling made easy *ACM CHI'2004 Conference on Human Factors in Computing Systems* (pp. 455-462). New York: ACM Press.

Patton, E. W., & Gray, W. D. (2009). SANLab-CM: A tool for incorporating stochastic operations into activity network modeling. *submitted for publication*.

Schweickert, R. (1980). Critical-Path Scheduling Of Mental Processes In A Dual Task. *Science*, 209(4457), 704-706.

Schweickert, R., Fisher, D. L., & Proctor, R. W. (2003). Steps toward building mathematical and computer models from cognitive task analyses. *Human Factors*, 45(1), 77-103.

## JERROLD POST

*Dr. Jerrold Post* is Professor of Psychiatry, Political Psychology and International Affairs and Director of the Political Psychology Program at The George Washington University.

Dr. Post has devoted his entire career to the field of political psychology. Dr. Post came to George Washington after a 21 year career with the Central Intelligence Agency where he was the founding director of the Center for the Analysis of Personality and Political Behavior. He played the lead role in developing the "Camp David profiles" of Menachem Begin and Anwar Sadat for President Jimmy Carter and initiated the U.S. government program in understanding the psychology of terrorism. In recognition of his leadership at the Center, Dr. Post was awarded the Intelligence Medal of Merit in 1979. He served as expert witness in the trial in the spring of 2001 for the al Qaeda terrorists responsible for the bombing of the U.S. embassies in Kenya and Tanzania, and, since 9/11, has testified on terrorist psychology before the Senate, the House of Representatives, and the United Nations. He is a widely published author, whose most recent book is "The Mind of the Terrorist: The Psychology of Terrorist from the IRA to al-Qaeda." Dr. Post is a frequent commentator on national and international media on such topics as leadership, leader illness, treason, the psychology of terrorism, suicide terrorism, weapons of mass destruction, Osama bin Laden, Saddam Hussein, Hugo Chavez, Mahmoud Ahmadinejad and Kim Jong Il.

*When Hatred is Bred in the Bone:  
The Psychocultural Foundations of Contemporary Terrorism*

After an introduction to the broad spectrum of terrorist psychology, this presentation will focus on nationalist-separatist and radical Islamist terrorism. We are seeing an increasing broadening and deepening of values and behavior associated with terrorism within mainstream society, as the new heroes and role models are the *shahids*, the martyrs, carrying out acts of suicidal terrorism. These do not represent acts of psychopathologically disturbed youth, but socially valued acts of mainstream individuals responding to powerful social forces. The manner in which radical Islamist leaders have reframed suicide as martyrdom and the social psychology of the assembly line producing suicide bombers will be explicated. The centrality of the core identity of belonging to a valued social movement and the role of the new media in creating a virtual community of hatred will be emphasized. Quotations from interviewed incarcerated terrorists will be used to illustrate the psychology of the terrorists. Implications for counter-terrorism, including the role of psychological operations will be considered.

## LCDR JOSEPH V COHN

LCDR Joseph Cohn is an Aerospace Experimental Psychologist (AEP) in the U.S. Navy's Medical Service Corps and serves as a Program Manager at the Defense Advanced Research Projects Agency (DARPA), in the Defense Sciences Office. His efforts are focused on developing projects that emphasize maintaining human performance/human effectiveness and optimizing the symbiosis between humans and machines. Lcdr Cohn has a doctorate in neuroscience from Brandeis University and a bachelor's degree in biology from the University of Illinois, Urbana-Champaign. He has authored more than 60 publications, served as guest editor on three professional journals, is co-editing a three-volume series of books focusing on all aspects of training system development, and is co-editing a book on warfighter performance. In addition to his military decorations, he received the Navy Modeling and Simulation Award, Training Category, from the ASN (RD&A) Chief Systems Engineer and was chosen as the Potomac Institute for Policy Studies' Lewis and Clark Fellow, exploring the legal and ethical issues associated with using performance enhancing technologies and developing policies and guidelines to ensure their effective —and appropriate—use.

~~~~~

### *Representing Human Behavior: Where to next?*

Advances in neuroscience have contributed to a strong growth in understanding how the human brain effectively processes information leading to behavior. Traditional approaches to representing human behavior for such uses as informing more effective human machine symbiotic systems, have focused on engineering or machine learning techniques to establish couplings between humans and their machines. For example, many of the cognitive architectures that are intended to allow the machine to infer human intention are based on computer processing metaphors, not on actual brain dynamics. This is a partly a result of the levels of technology available to understand and represent the processes through which the human brain transforms information into action. Until very recently, neither the imaging technologies nor the analytic capabilities were available to truly link actual brain activity to behavior. As a result, when one wished to represent human behavior, one was forced to do so using observed behaviors as a starting point, and building predictive models of human behavior on these observed behaviors.

One important goal of neuroscience is to develop techniques for representing the link between observed behavior and underlying neural action. Just as understanding the equations of motion provides a much broader set of capabilities than inferring these equations from a limited set of observations, so too understanding and modeling the dynamics of neural activity as it leads to behavior should provide a much richer and more robust set of models than those based on the actual observed behavior alone. Today, advances in neuroscience and engineering provide the basis for building these 'equations of motion' for the brain and for using brain-based techniques to create and maintain very robust human behavior representations.

## ROBERT AXTELL

Robert Axtell is the Professor and Chair, George Mason University, Krasnow Institute for Advanced Study, Department of Computational Social Science; External Professor, Santa Fe Institute

Dr. Axtell works at the intersection of economics, behavioral game theory, and multi-agent systems computer science. His most recent research attempts to emerge a macroeconomy from tens of millions of interacting agents. He is Department Chair of the new Department of Computational Social Science at George Mason University (Fairfax, Virginia, USA). He teaches courses on agent-based modeling, mathematical modeling, and game theory. His research has been published in "Science," "Proceedings of the National Academy of Sciences USA," and leading field journals. Popular accounts have appeared in newspapers, magazines, books, online, on the radio and in museums. His is the developer of Sugarscape, an early attempt to do social science with multi-agent systems, and co-author of "Growing Artificial Societies: Social Science from the Bottom Up" (MIT Press 1996). Previously, he was a Senior Fellow at the Brookings Institution (Washington, D.C. USA) and a founding member of the Center on Social and Economic Dynamics there. He holds an interdisciplinary Ph.D. from Carnegie Mellon University (Pittsburgh, USA).

---

### Intertemporal Behavior: How People Discount the Future--Experimental Data and Formal Representation

A mathematical formalism is developed for the existence of unique invariants associated with wide classes of observed discounting behavior. These invariants are 'exponential discount rate spectra,' derived from the theory of completely monotone functions. Exponential discounting, the empirically important case of hyperbolic discounting, and so-called sub-additive discounting are each special cases of the general theory. This formalism is interpreted at both the individual and social levels. Almost every discount rate spectrum yields a discount function that is 'hyperbolic' with respect to some exponential. Such hyperbolic discount functions may not be integrable, and the implications of non-integrability for intertemporal valuation are assessed. In general, non-stationary spectra lead to discount functions that are not completely monotone. The same is true of discount rate spectra that are not proper measures. This formalism unifies theories of non-constant discounting, declining discount rates, hyperbolic discounting, 'gamma' discounting, and related notions.

## **ARL submission to BRIMS sponsor panel**

*Mr. John F. Lockett*

U.S. Army Research Laboratory  
Aberdeen Proving Ground, MD 21005-5425

The US Army Research Laboratory (ARL) provides fundamental underpinning research and development for the Army Materiel Command and supplies innovative science, technology, and analysis to enable full-spectrum operations. The Army relies on ARL for scientific discoveries, technologic advances, and analyses to provide Warfighters with capabilities to succeed on the battlefield. The Human Research and Engineering Directorate (HRED) of ARL conducts a broad-based program of scientific research and technology development directed toward optimizing Soldier performance and Soldier-machine interactions to maximize battlefield effectiveness. ARL HRED provides the Army with human factors leadership to ensure that Soldier performance requirements are adequately considered in technology development and system design. Although ARL is not part of the Medical, Personnel, Training and Doctrine, or Test and Evaluation Commands; we collaborate with our colleagues there and throughout the Department of Defense to address Human Systems Integration issues.

ARL HRED high priority research areas include Soldier Performance, Neuroergonomics, Social/Cognitive Network Science, Human Robotic Interaction, and Human Systems Integration. Opportunities and challenges for BRIMS exist in each of these areas. Addressing them entails empirical data collection, development of theoretical frameworks, algorithm development, validation, and usability testing as well as code development. Many of the issues have been presented by sponsors at earlier BRIMS conferences (notably those by Surdu 2007 and Allender 2007<sup>1</sup>) and remain relevant.

The goal of ARL's Soldier performance research is to optimize sensory, perceptual, and physical demands on the Soldier and the Soldier-system to improve survivability, sustainment, efficiency, and performance effectiveness. While much progress has been made on modeling and simulation of human locomotion and to a lesser extent load carriage, challenges remain in representing cooperative team and group tasks. M&S of sensory and perceptual processes exist but compelling cross sensory modality presentations are lacking. Empirical data collection and often as a result BRIMs does not address the combined effects of performance moderators particularly those combinations in which moderators counteract each other at different levels.

ARL's neuroergonomics program seeks to assess Soldier cognitive and neurophysiological function, understand Soldier behavior, and develop non-subjective, operationally relevant cognitive metrics through the translation of laboratory techniques. The goal is to enable the Army to match the capabilities of Soldiers and advanced technologies to maximize investments in systems development. Given recent interest and investment in this area, challenges for BRiMS are well known by the community however additional emphasis should be given to two topics to meet Army needs. BRiMS must be generalizable to militarily relevant settings, conditions and functions i.e. outside the laboratory setting. Also, schema and corresponding BRiMS must be developed to deal efficiently but validly with aggregating from individuals to populations. The Department of Defense may define (aggregate) its members in various ways for example job specialty, rank, mental category, skill level, or gender.

ARL's social/cognitive network science research area involves applying principles from the cognitive, computer, and social network sciences to the conduct of complex dynamic network-enabled operations. Decision makers are not able to use the sheer volume of information available over the network effectively. The goal is to align Warfighter and system capabilities. Specific topics of focus are situation awareness, decision making in environments characterized by information overload, information uncertainty, trust in automation, or joint and multinational operations. Efforts include computer models, tool development, data collection in exercises, and data collection in controlled experimentation. Expected benefits are information to assist the proper design of units and the development of methods to support distributed collaborative planning and decision making at the tactical and operational levels. BRiMS particularly those that are predictive and can underlie intuitive commander planning and decision support tools are of interest to ARL. Social and cultural modeling, as noted in a 2007 BRiMS symposium conducted by Allender and Sutton, continues to be of interest to the Department of Defense. Social and cultural factors should be included across the full spectrum of modeling and simulation research and applications. In this area the emphasis is on using M&S to support on-going operations of all types.

The purpose of ARL's Human Robotic Interaction effort is to reduce workload and improve combat performance for the Soldier-robot team through a better understanding of the human dimension. The expected result is improved interface and adaptive Soldier support technologies scalable to dismounted and

mounted warrior systems in multi-mission environments. In this area, BRiMS is needed as an enabler for exploration, analysis and empirical data collection of concepts for human-robot interaction, human robot teams and interaction with robot-robot teams. M&S that represent perception, management of concurrent tasks, operator control units, adaptive automation, social and cultural norms, and group behaviors are important to this research area.

ARL's mission in Human Systems Integration includes developing tools and analytic methodologies for cost effective insertion of human factors criteria into early acquisition (pre-milestone A) requirements to optimize Soldier-system performance and cost at the systems of systems level. ARL also conducts Soldier-centered analyses to ensure manpower requirements, workload, and skill demands are considered collectively and systematically, avoiding information and physical task overload and taking maximum advantage of aptitudes, individual and collective training, and numbers of Soldiers for an affordable future force. Given this mission, BRiMS is useful in informing system design tradeoff decisions and has proven an effective means of convincing acquisition managers that human factors issues need to be addressed. Improvements in BRiMS already mentioned will help ARL's HSI mission. Attention to verification, validation and accreditation as well as decreasing the resource requirements for using predictive BRiMS will make it more feasible for HSI practitioners to employ this technology. Another aspect of HSI tool development and analysis is the importance of relating human and system component performance to mission performance. To be useful for HSI, BRiMS must be scalable and able to account for the effect of changes in that state of components (including human operators) on mission goals and vice versa. Links that cross classes and application of models are important to decreasing resource requirements for employing M&S and to increasing collaboration with other design fields such as systems engineering.

ARL has recently awarded or will soon award several Collaborative Technology Alliances (CTAs) with Industry and Academia that are expected to advance BRiMS in several of ARL's high priority research areas. A CTA about network science was awarded in September 2009 and two other CTAs – one about Robotics and another about Cognition and Neuroergonomics – are still in competition.

Footnote:

<sup>1</sup> Available online at

<http://brimsconference.org/archives/2007/abstract/07brims-203.htm>

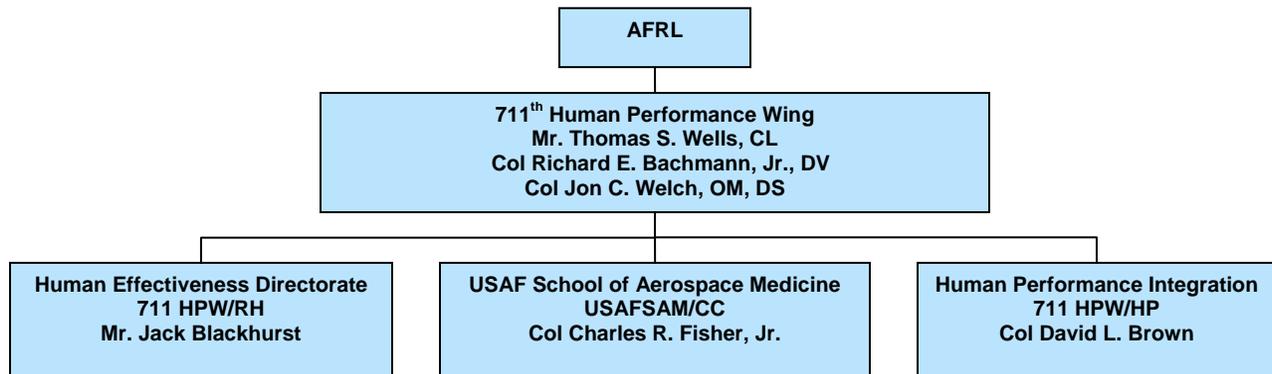
## 711<sup>th</sup> Human Performance Wing

Mr. Thomas S. Wells, SES, Director

The historic activation of the 711<sup>th</sup> Human Performance Wing at Wright-Patterson Air Force Base culminated two years of inspired strategic planning. Standup of the Wing creates the first human-centric warfare wing to consolidate research, education, and consultation under a single organization. The 711 HPW merges the Air Force Research Laboratory Human Effectiveness Directorate with functions of the 311th Human Systems Wing currently located at Brooks City-Base: the United States Air Force School of Aerospace Medicine (including functions of the former Air Force Institute for Operational Health that are merged into USAFSAM) and the 311<sup>th</sup> Performance Enhancement Directorate (renamed Human Performance Integration Directorate).



The Wing's primary focus areas are aerospace medicine, human effectiveness science and technology, and human systems integration. In conjunction with the Navy Aerospace Medical Research Laboratory (NAMRL) moving to WPAFB, and surrounding universities and medical institutions, the 711 HPW will function as a Joint Department of Defense Center of Excellence for human performance sustainment and readiness, optimization and effectiveness research.



The 711th Human Performance Wing mission is *to advance human performance in air, space, and cyberspace through research, education, and consultation*, accomplished through synergies created by the wing's three distinct but complementary entities: the U. S. Air Force School of Aerospace Medicine, the Human Effectiveness Directorate, and the Human Performance Integration Directorate.

## Human Effectiveness Directorate

Mr. Jack Blackhurst, Director

The Human Effectiveness Directorate is leading the Air Force in human-centered research.

## USAF School of Aerospace Medicine

Colonel Charles R. Fisher, Jr., Commander

First-call consultants in aerospace medicine, we find solutions to operational needs of today and tomorrow, and prepare new aeromedical experts for future global challenges.

## Human Performance Integration Directorate

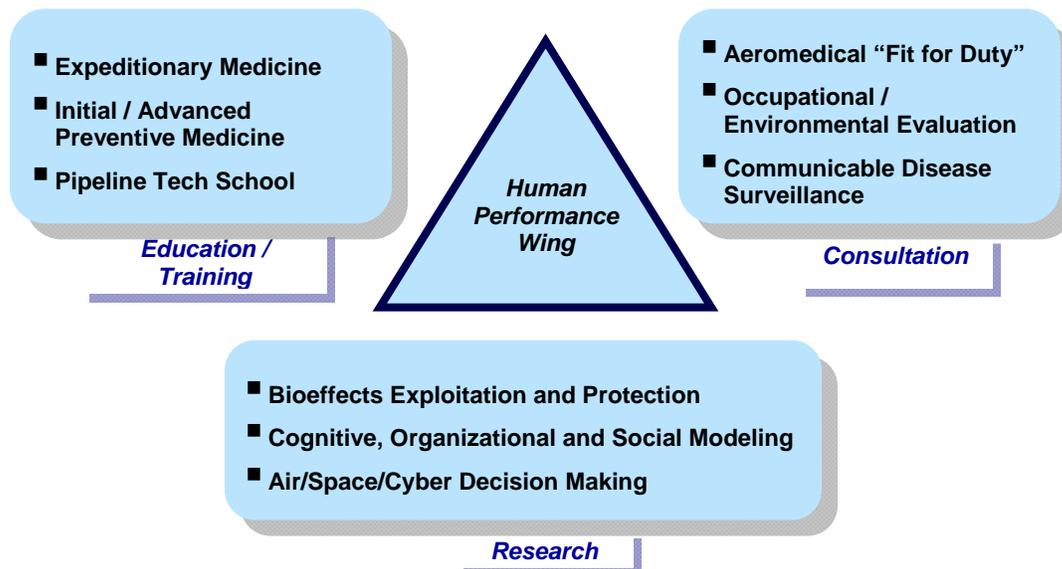
Colonel David L. Brown, Director

HP advocates, facilitates and supports the application of human systems integration principles to optimize operational capabilities.

## BRAC Moves

<b>Relocating from Mesa Research Site, Brooks City-Base TX, and Holloman AFB NM to WPAFB OH</b> <ul style="list-style-type: none"> <li>• 507 Military</li> <li>• 349 Civilian</li> <li>• Total of 856 authorizations to WPAFB</li> </ul>	<b>Related MILCON at WPAFB OH</b> <ul style="list-style-type: none"> <li>• \$238M and 670,000 sq ft</li> <li>• RFP release: Dec 2007</li> <li>• Contract Award: Apr 2008</li> <li>• BOD: 31 May 2011</li> </ul>
<b>AFRL Directed Energy Bioeffects mission to Ft. Sam Houston TX</b> <ul style="list-style-type: none"> <li>• 34 Military</li> <li>• 48 Civilian</li> <li>• Total of 82 authorizations to FSH</li> </ul>	<b>Related MILCON at Ft. Sam Houston TX</b> <ul style="list-style-type: none"> <li>• \$79.5M and 181,000 sq ft</li> <li>• RFP release: Jan 09</li> <li>• Contract Award: Apr 09</li> <li>• BOD: Apr 2011</li> </ul>

*A Human Performance Center of Excellence based on the university model bringing together research, education/training and consultation.*



# Behavioural Representation of People in Contemporary Operating Environments

An UK Overview by Bharatkumar Patel, Dstl, UK MOD  
© Crown Copyright

## 1 Introduction

1.1 This overview focuses on the UK defence need to improve behavioural representation of people in contemporary operating environments in their modelling and simulation capability in order to provide better pre-deployment training and experimentation, and to enhance analysis for better decision-making. It addresses this need through:

- Making Computer Generated Forces Smarter
- Dynamic Social Modelling to improve our decision making and pre-deployment cultural and social training.

## 2 Making Computer Generated Forces Smarter

2.1 The Integrated Human Behaviour Representation (IHBR) programme which was initiated in 2003 seeks to improve the realism and available variability of both Computer Generated Forces (CGF) cognition and behaviour. The initial phase (2003-2005) of the programme explored a means for explicitly differentiating CGF entity '*cognition*' from entity '*behaviour*' and improving CGF entity and unit cognition. The second phase (2006-2008) explored ways of making these improvements in realism and variability of cognition more available to and realisable in the behaviour generation capabilities of legacy, current, and developing CGF systems.

2.2 Given the level of investment in the IHBR programme, and its importance to future CGF application development, the follow-on work will examine and demonstrate how people within Contemporary Operating Environments (COE)<sup>1</sup> can be represented in CGF systems by invoking more realistic, flexible and variable ('smart') behaviours.

2.3 The work will address how to represent all types of people in current and anticipated operational theatres within simulation environments. It

---

<sup>1</sup> A complex overall operational environment with state and non-state players that exists today and in the near future in conflicts of interest, security or war.

is to consider a broad range of factors, including behavioural reasoning, physiological and psychological representation, cultural and societal influencers, and specific threat representations. The demonstration objective is to integrate and assess a comprehensive cognitive system for the purpose of representing all types of people in the COE.

#### 2.4 Specifically the work will:

- Identify and qualitatively evaluate existing and emerging behavioural techniques against defined attributes that would be applicable for representing people in current and anticipated operations, either within CGF systems (e.g. JSAF *ClutterSim* or *CultureSim*, OneSAF *composable-behaviour*, etc), or available as “plug-ins” to other simulation tools (e.g. B-HAVE plug in for VR Forces, AI Implant, CoJACK2, etc)
- Explore the *composable-behaviour* mechanisms available within OneSAF, and demonstrate OneSAF’s ability to represent civilians and insurgents in current COE
- Identify and qualitatively evaluate available Belief, Desire, Intent (BDI) cognitive platforms or architectures (GOTS<sup>2</sup>, COTS<sup>3</sup> open source or freeware), and select and demonstrate the architecture that is most beneficial
- Develop an initial ontology for a couple of CGFs to demonstrate how the same BDI agent plan library can be re-used to drive behaviour in CGFs with very different behaviour repertoires (e.g. VBS2 and OneSAF)
- Demonstrate the ability to integrate smart behaviours in VBS2, initially enabling the expression of subtle, important, culturally-dependent, non-verbal behaviours (including body language) of civilians and insurgents.

### 3 Dynamic Social Modelling

- 3.1 UK is currently developing a research strategy to support Dynamic Social Modelling (DSM) in order to improve cultural and social representation for better decision making and pre-deployment training.

---

<sup>2</sup> Government-Off-The-Shelf

<sup>3</sup> Commercial-Off-The-Shelf

- 3.2 The DSM term is used to describe all software modelling approaches that include social factors. DSM approaches may be incorporated into existing models and simulations or provide stand-alone capabilities to address specific social issues.
- 3.3 A series of workshops and roadmapping exercise were conducted to define the scope of DSM and its relevance and need to support COE.
- 3.4 The output of the workshop recommended a number near term and long term challenges and the strategy for developing and exploiting DSM capability.
- 3.5 The short term requirements identified were for:
  - Operational quick-wins for socio-cultural training and education
  - Development of deployable social factors operational analysis capability.
- 3.6 The long term requirements identified were for a DSM capability comprising a suite of compatible or integrated methods and models that address the full range of effects and cover both military and non-military levers of power. These models would ensure that defence functions are more financially efficient and more effective, through supporting:
  - Training and education
  - Course of action development
  - Policy development
  - Balance of investment decisions.
- 3.7 The strategy to develop and exploit the DSM capability includes the following enablers:
  - Build customer and stakeholder awareness and ownership of DSM
  - Conduct a near-term stocktake of DSM capability
  - Develop internal and external supply base for DSM
  - Ensure availability of data for DSM
  - Establish practical guidance for fit-for-purpose use of DSM
  - Relate DSM developments to COE developments.

## **4 Concluding Remarks**

- 4.1 The key challenge for behavioural representation in COE is timeliness. The methods for human representation in defence models and simulations need to be agile and responsive if they are to be relevant

to COE. Furthermore, they will need to include complex cultural and social dynamic representation.

# The Use of Behavior Models for Predicting Complex Operations

*Brian F. Gore, PhD*

San Jose State University Research Foundation/NASA Ames Research Center

MS 262-4

PO Box 1

Moffett Field, CA 94035-0001

650-604-2542

Brian.F.Gore@nasa.gov

Keywords:

NASA, Ames Research Center, HSI Division, Human Performance Model, Integrated Models, MIDAS v5

**ABSTRACT:** Modeling and simulation (M&S) plays an important role when complex human-system notions are being proposed, developed and tested within the system design process. National Aeronautics and Space Administration (NASA) as an agency uses many different types of M&S approaches for predicting human-system interactions, especially when it is early in the development phase of a conceptual design. NASA Ames Research Center possesses a number of M&S capabilities ranging from airflow, flight path models, aircraft models, scheduling models, human performance models (HPMs), and bioinformatics models, among a host of other kinds of M&S capabilities that are used for predicting whether the proposed designs will benefit the specific mission criteria. The Man-Machine Integration Design and Analysis System (MIDAS) is a NASA ARC HPM software tool that integrates many models of human behavior with environment models, equipment models, and procedural / task models. The challenge to model comprehensibility is heightened as the number of models that are integrated and the requisite fidelity of the procedural sets are increased. Model transparency is needed for some of the more complex HPMs to maintain comprehensibility of the integrated model performance. This will be exemplified in a recent MIDAS v5 application model and plans for future model refinements will be presented.

## 1. Introduction

Complex system integration issues require that the model development process generally follow an iterative design philosophy that collaboratively leverages empirical human data (i.e., either human in the loop, HITL, simulations or real-time measurements) and concurrently feeds information to HITL simulation processes. Many organizations are faced with the goals of completing research as efficiently as possible while maintaining acceptable levels of safety to successfully complete a mission. NASA is no exception. Modeling and simulation techniques, particularly human behavior models, play an important role when complex human-system notions are being proposed, developed, and tested across many of the ten NASA centers. For instance, NASA Johnston Space Center (JSC) utilizes M&S to represent environments, physical structures and equipment components, crew stations, planets and planetary motions, gravitational effects, illumination, human anthropometric and biomechanics, among a host of other domains. NASA Ames Research Center also possesses a number of M&S capabilities ranging from airflow, flight path models (e.g., Airspace Concept Evaluation System, - ACES), aircraft models, scheduling models (e.g., Core-XPRT, Science Planning InterFace to engineering - SPIFe), human performance models (HPMs), and bioinformatics models, among many other kinds of M&S capabilities. One of the many NASA M&S capabilities, an ARC-related HPM

capability termed the Man-Machine Integration Design and Analysis System (MIDAS) is highlighted because of its relevance to the field of human behavior representation.

### 1.1 Human Performance Models (HPMs), Concept Development and Testing

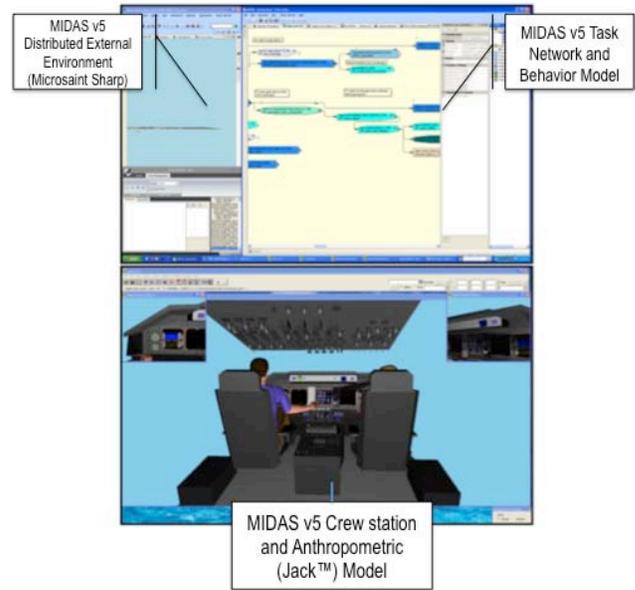
Modeling can play a role in all phases of the concept development, refinement, and deployment process. Hybrids of continuous-control, discrete-control and critical decision-making models represent the 'internal models and cognitive function' of the human operator in complex control systems, and involve a coupling among humans and machines in a shifting and context sensitive environment. These models, known as HPMs, have arisen as viable research options due to decreases in computer costs, increases in representative results, and increases in model validity. They are especially valuable because the computational predictions can be generated early in the design phase of a product, system or technology to formulate procedures, training requirements, and to identify system vulnerabilities and where potential human-system errors are likely to arise. The model development process allows the designer to formally examine many aspects of human-system performance with new technologies to explore potential risks brought to system performance by the human operator (Gore & Smith, 2006). Often this can be accomplished before the notional technology exists for human-in-the-

loop (HITL) testing (Gore, 2000). This method possesses cost and efficiency advantages over waiting for the concept to be fully designed and used in practice (characteristic of HITL tests). Using HPMs in this manner is advantageous because risks to the human operator and costs associated with system experimentation are greatly reduced: no experimenters, no subjects and no testing time (Elkind et al., 1989; Gore, 2000). Hooey and Foyle (2008) outline that HPMs can be used to conduct system robustness testing to evaluate the system from the standpoint of potential deviations from nominal procedures to determine the impact on the performance of the human and the system ("what-if" testing).

### 1.2 The Man-machine Integration Design and Analysis Systems (MIDAS)

MIDAS is a dynamic, integrated human performance modeling and simulation environment that facilitates the design, visualization, and computational evaluation of complex man-machine system concepts in simulated operational environments (Gore, 2008). MIDAS combines graphical equipment prototyping, dynamic simulation, and HPMs to reduce design cycle time, support quantitative predictions of human-system effectiveness, and improve the design of crew stations and their associated operating procedures. HPMs like MIDAS provide a flexible and economical way to manipulate aspects of the operator, automation, and task environment for simulation analyses (Gore, 2008; Gore, Hooey, Foyle, & Scott-Nash, 2008; Hooey & Foyle, 2008).

Gore & Smith (2006) outline that MIDAS links a virtual human, composed of a physical anthropometric character, to a computational cognitive structure that represents human capabilities and limitations. The cognitive component is composed of a perceptual mechanism (visual and auditory), memory (short term, long term-working, and long term), a decision maker and a response selection architectural component. The complex interplay among bottom-up and top-down processes enables the emergence of unforeseen, and non-programmed behaviors (Gore & Smith, 2006). MIDAS can suggest the nature of pilot errors, and highlight precursor conditions to error such as high levels of memory demand, mounting time pressure and workload, attentional tunneling or distraction, and deteriorating situation awareness (SA).



**Figure 1. MIDAS' Environment, Task, and Anthropometric Models.**

MIDAS can be used as a cognitive modeling tool that allows the user to obtain both predictions and quantitative output measures of human performance, such as workload and SA and as a tool for analyzing the effectiveness of crew station designs, information display concepts, operator roles and responsibilities from a human factors perspective (Gore, 2008). MIDAS has proven useful for identifying general human-system vulnerabilities and cross-domain error classes and for recommending mitigation strategies and job re-designs to account for the vulnerable areas, or risks, in system design (Gore & Smith, 2006). Fundamental design issues can therefore be identified early in the design lifecycle, prior to the use of hardware simulators and HITL experiments. In both cases, MIDAS provides an easy to use and cost effective means to conduct experiments that explore "what-if" questions about domains of interest.

### 1.3 The MIDAS User Interface Assists Comprehensibility

MIDAS v5 has a graphical user interface<sup>1</sup> (GUI) that does not require advanced programming skills to use. The GUI brings many of the previously embedded functions to the surface so that the model analyst can observe the underlying structure as well as the model's operation as it is run. The integrated GUI enables the user to build human procedures from MIDAS primitive tasks, create their own tasks, incorporate a series of nested procedures, change the SA context during the simulation and manipulate visual and auditory

<sup>1</sup> MIDAS uses Microsaint Sharp as its GUI which uses the C-Sharp programming language

attributes of equipment components. The MIDAS analyst can organize the human-system interactions visually, thereby greatly improving the model's transparency. Other features of MIDAS v5 include dynamic visual representations of the simulation environment, support for multiple and interacting human operators, distributed simulation, monte-carlo/stochastic performance, HPM timelines, task lists, workload, and SA, performance influencing factors (such as error predictive performance, fatigue and gravitational effects on performance), libraries of basic human operator procedures (how-to knowledge) and geometries for building scenarios graphically (that leverage heavily from Siemens' Jack software).

#### 1.4 MIDAS Approach and Land Applications

The current air traffic control (ATC) system will not be able to manage the predicted two to three times growth in air traffic (JPDO, 2009). The Next Generation Air Transportation System (NextGen) is a future aviation concept that has as its goals to significantly increase the capacity, safety, efficiency, and security of air transportation operations (JPDO, 2009).

MIDAS v5 has been applied to examine a NextGen approach to land concept termed the very closely spaced parallel approach (VCSPA). In order to evaluate this concept, two MIDAS v5 models were generated. The first was a current day Simultaneous Offset Instrument Approach (SOIA) model that contained the current day procedures and the second was a NextGen VCSPA model that contained predictive displays in the cockpit and a modification to the roles and responsibilities of the flight crew and ATC modeled operators. This simulation involved over 500 tasks and culminated in a verifiable model of approach and land operations (vetted by Subject Matter Experts - SMEs). The SA model was augmented within MIDAS to represent how a cockpit crew builds SA of traffic, terrain, and weather information given the accessibility of sources of information. This model effort illustrated the "what-if" simulation capability within MIDAS. The "what-if" approach was completed when MIDAS was exercised with one set of displays and procedure sets designed to represent current day operations and roles followed by a second simulation with an alternate set of displays and procedures encoded to represent the NextGen displays and expected procedures. The model underwent an iterative verification/validation process that included examining: (1) the task sequences and the performance of the model as it executed; (2) the visual fixations, task timings, and workload relative to expected performance given the inputs to the model; and pilot performance according to SME evaluations.

Model comprehensibility is defined as understanding

the relationships that exist among the models being used in an application, the performance of the models in the application, which models are being triggered in the model architecture, and whether the model is behaving as the model analyst would expect. MIDAS v5's comprehensibility was greatly improved with the transparent model architecture (Gore, 2008). The operation of this complex model was verified throughout development and was validated according to SME evaluations. The verification phase of the model was improved given the visibility into the model's operations at any given point in simulation time combined with the cross checking of the jack visualization and the simulation runtime data that was output. The comprehensibility of this model would not have been possible without such a transparent architecture.

This MIDAS v5 effort lead to a greater awareness of potential parameters that should be included in system designs and enabled the research program to visualize the interactions that will be likely in future NextGen operations. It is anticipated that a formal validation approach will be developed and applied to the VCSPA model in an upcoming Federal Aviation Authority (FAA) task. This FAA task will require model refinement and validation, an increased number of alternative closely spaced operations for additional what-if scenarios including alternative pilot roles and responsibilities, and information requirements.

## 2. Conclusion

A number of significant challenges exist for the state of the art in HPMs, two of which will now be highlighted.

Transparency. The first challenge relates to model transparency. Model transparency refers to the ability to comprehend the relationships that exist among the models being used in the simulation, the performance of the models in the simulation, which models are triggering in the model architecture, and whether the model is behaving as the model developer would expect (Gore, 2008). Other researchers refer to this as model traceability, model behavior visibility, model verifiability, and model interpretability (Elkind et al., 1989; Napiersky, Young, & Harper, 2004; Gluck & Pew, 2005; Hooey & Foyle, 2008). Transparency in integrated HPMs is needed to support model verification, validation, and credibility. However, model transparency can be difficult to attain because of the complex interactions that can exist among the cognitive, physical, environment and crew station models, and because the cognitive models embedded within integrated HPMs produce behaviors that are not directly observable. Three types of transparency that the MIDAS researchers have found useful to understand, interpret, and increase the confidence in

the complex models' output include transparency of the input, transparency of the integrated architecture, and transparency of the output (Gore, Hooey, Foyle, & Scott-Nash, 2008). This paper illustrates how the augmentation to the MIDAS GUI has improved model transparency that has led to better model comprehensibility.

Validation. The second challenge facing the HPM community is validation. Validation remains a very large challenge for the HPMs community because statistical validation is oftentimes seen as the Holy Grail for determining whether a model is suitable but when models are deemed statistically valid, they are less generalizable, and less re-usable for applications in new contexts. This places the field of modeling into the conundrum of making models that are statistically valid (correlation,  $r=.99$ ) but that lack the ability to generalize to other tasks or scenarios. When the generalizability of the model is limited, then its value as a cost-effective approach to predict complex human-system interactions is reduced.

Validation is further challenged when modeling future technology concepts where no or little HITL data exists upon which to statistically validate a model (as in the NextGen aviation systems or concepts being designed for the Space program). It is argued that our definition of model validation must be expanded beyond that of statistical results validation to be more representative of a model develop-model verify-model manipulate – model validate iterative process.

### 3. References

- Elkind, J., Card, S.K., Hochberg, J. & Huey B. M. (1989). *Human performance models for computer-aided engineering*. Washington, D.C.: National Academy Press.
- Gore, B.F. (2008). Chapter 32: Human performance: Evaluating the cognitive aspects. In V. Duffy (ed.), *Handbook of digital human modeling*, (pp.32-1-32-18), Boca Raton: CRC Press/Taylor & Francis.
- Gore, B.F., Hooey, B.L., Foyle, D.C., & Scott-Nash, S. (2008). Meeting the challenge of cognitive human performance model interpretability though transparency: MIDAS v5. In the 2<sup>nd</sup> International

Conference on Applied Human Factors and Ergonomics, July 14-17, Las Vegas, Nevada.

- Gore, B.F., & Smith, J.D. (2006). Risk assessment and human performance modeling: the need for an integrated approach. In K.A. Malek (Ed.) *International Journal of Human Factors of Modeling and Simulation*, 1(1), 119-139.
- Gluck, K.A., & Pew, R.W. (2005). *Modeling human behavior with integrated cognitive architectures: Comparison, evaluation and validation*, New Jersey: Lawrence Erlbaum & Associates.
- Hooey, B.L., & Foyle, D.C. (2008). Chapter 12: Advancing the State of the Art of Human Performance Models to Improve Aviation Safety. In D. Foyle, and B. Hooey (eds.), *Human performance modeling in aviation*, (pp.321-349), Boca Raton: CRC Press/Taylor & Francis.
- Joint Planning and Development Office (JPDO). Concept of operations for the next generation air transportation system, Version 3, Joint Planning and Development Office, Washington, DC: JPDO.
- Napierski, D., Young, A., & Harper, K. (2004). Towards a common ontology for improved traceability of human behavior models. In Proceedings of the 13th Conference on Behavior Representation in Modeling and Simulation, Arlington, VA, May.

### Author Biography

**Dr. BRIAN GORE** is a Principal Investigator for San Jose State University Research Foundation, and NASA's Technical Lead for the Man-machine Integration Design and Analysis System (MIDAS) at the NASA Ames Research Center. Dr. Gore manages the MIDAS applications and provides the direction for MIDAS' development.

### Acknowledgement

This work was supported by the Federal Aviation Authority (FAA)/NASA Inter Agency Agreement DTFAWA-10-X-80005 (POCs Dr. T. McCloy, FAA; and Dr. D. Foyle, NASA).

**Systems Social Science -- Tutorial On:  
“Ethno-Political Modeling for Training and Analysis”**

**Barry G. Silverman, PhD**

Electrical and Systems Engineering Department  
University of Pennsylvania, Philadelphia, PA 19104-6315

[basil@seas.upenn.edu](mailto:basil@seas.upenn.edu)

March 2010

**ABSTRACT**

This tutorial explores how a model of models or models library may be useful for profiling and emulating social systems. We begin by exploring challenges for domain specialists, modelers, and social scientists in representing social dilemmas so they may be modeled and simulated. The lack of tools and models for supporting this enterprise are explored as 3 challenges facing the BRIMS community. “Systems social science” is then presented as a meso-scale model of models methodology for design inquiry that synthesizes systems science, agent modeling and simulation, knowledge management architectures, and domain theories and knowledge. The goal is to focus computational science on exploring underlying mechanisms (white box modeling) and to support reflective theorizing and discourse to explain social dilemmas and potential resolutions. To support one in collecting a large library of models, several software design patterns are then explored and illustrated. The tutorial then describes an illustrative agent modeling and simulation library (model of many models from the literature). Two gameworld applications that utilize this library are discussed (a VillageSim and a StateSim). These serve as an example of the new types of instruments useful for systems social science. The conclusions wrapup by reviewing lessons learned about criteria that have guided this research and the types of validity assessment efforts that have been attempted.

**Tutorial Outline:**

- Challenge: 3 Universal Dilemmas (in Human Socio-Cultural Behavior M&S)
  - Domain Specialists’ Challenge
  - Modelers’ Challenge
  - Social Scientists’ Challenge
- Response: Systems Social Science Defined
- Software Design Patterns To Think About (Model View Controller, Model Factory, Model Driven Architecture)
- Example Model of Models Library
- Case Studies: Training & Analysis
- Conclusions, Lessons Learned, Next Steps

**Keywords:** social systems, systems approach, socio-cognitive agents, design inquiry

# Creating Realistic Human Behavior in Physical Security Systems Simulation

*Volkan Ustun*

School of Industrial and Systems Engineering  
Georgia Institute of Technology  
Atlanta, GA, 30332, USA  
ustunvolkan@gmail.com

*Jeffrey S. Smith*

Department of Industrial and Systems Engineering  
Auburn University  
Auburn, AL, 36832, USA  
jsmith@auburn.edu

Keywords:

Human Behavior Representation, Physical Security Systems, Cognitive Simulation, Agent-Directed Simulation

## 1. Introduction

Physical security systems (PSS) are designed to prevent access to a facility by *intruders*, detect the presence of intruders, or facilitate the capture or neutralization of intruders once they are detected, without negatively impacting the intended users of the facility, or *neutrals*. The application domains for PSS include banks, retail stores, schools, airports, subway stations and military installations, where the intention of the intruder can range from simple theft, to kidnap or mayhem to total facility destruction, and intruder mitigation can range from discouraging (in the case of shoplifting, e.g.) to alerting (in the case of burglary, e.g.), to capture and confinement or neutralization (in the case of facility destruction). These systems generally include a combination of physical barriers, human guards, and sensor-based detection systems such as video surveillance systems. Furthermore, the tactics and policies for the security personnel are also integral to the overall PSS. The primary goal here is to assess the effectiveness of a PSS (both the sensor placement and the security policy of the personnel) for detecting intruders and mitigating their impact in compliance with the organization's goals (e.g. deterrence, detection etc.). Other questions of interest that contribute to the primary goal include but are not limited to:

- Is the PSS robust and effective against different tactics used by intruders (e.g. stealth, deceit, and force)?
- What will be the effect of a change in physical security design on intruder behavior?
- What should be the rules of engagement for security personnel to best mitigate the risks imposed by intruders?

The complex interactions among guards, intruders, and neutral entities as well as the interactions between these entities and the environment, complicate analysis of these systems (for instance, a fundamental problem in PSS is to distinguish an intruder from a neutral based on behavior) which is often limited to static "line of sight" and "field of view" models designed to help with camera placement and guard patrol path determination. Existing simulation-based analysis methodologies include only crude and often hard-coded implementations of behavioral responses to predetermined situations for the guards, intruders, and neutrals. This limits the analysis capabilities of these models and makes creating them very time consuming and expensive.

Models for PSS analysis are intended to estimate the system performance in settings which resemble real life situations. A realistic model of human reasoning should incorporate the shortcomings and fallacies of human reasoning as well as its ability to generate quick solutions that are "good enough". Subsequently, realistic and credible simulations of PSS require incorporation of human behavior models that involve situation awareness, cooperative team behavior, planning, and deliberative decision making processes of human agents.

We have demonstrated a proof-of-concept for a novel approach to simulating PSS, comprised of three principle components:

- A spatial model which formally represents the static features of the environment in a simulation-friendly structure;

- An agent-based behavioral framework which realistically represents the decision making activities of the agents using models of perception and heuristics to represent human intuition and decision making; and
- A formal representation of the application domain; for example, which behaviors constitute an intrusion and how an intrusion is detected vary between different domains.

The success of the proposed approach results from the realism and the variety of behaviors generated by the behavioral framework. The behavioral framework is extendable since it uses heuristics to model human intuition. Introduction of different heuristics directly relates to the emergent behavior. In addition, applying these heuristics on the perceived environment (the mental representation of the environment as the agent perceives it) creates interactions and behaviors that are difficult to anticipate in advance. Therefore, even with a limited number of heuristics, it is possible to observe a wide variety of potential activity sequences and interactions between agents that cannot be easily foreseen.

We have discussed the conceptual models for this application in various publications. Ustun (2009) provides the details for the whole computational framework. Ustun et al. (2005) introduces the spatial model. Ustun and Smith (2008) discuss a novel aspect in the agent based behavioral framework. Ustun et al. (2006) has a conceptual introduction to a sample application domain: retail store security systems. Marechal et al. (2009) uses a part of the proposed computational simulation framework in an optimization application.

In this interactive demo, we will demonstrate the several aspects of the proposed computational framework using a poster and a partially live demonstration of the developed computer application. The poster will be primarily used to present the conceptual features and several animations from the sample retail store application will be shown to provide insights on the interesting interactions between the virtual participants of the simulation experiments.

## 2. References

- Ustun, V., Yapicioglu H., Gupta S., Ramesh A., and Smith J.S.. (2005). A Conceptual Architecture for Static Features in Physical Security Simulations. In *Proceedings of the 2005 Winter Simulation Conference*, 958-964.
- Ustun V., Yilmaz L., and Smith J.S. (2006). A Conceptual Model for Agent-based Simulation of Physical Security Systems: A Study on Shoplifting. In *Proceedings of the 2006 ACMSE Conference*, 365-370.
- Ustun V. and Smith J.S.. (2008). Mental Simulation for Creating Realistic Behavior in Physical Security Systems Simulation. In *Proceedings of the 2008 Winter Simulation Conference*, 879-885.
- Ustun V. (2009). Human Behavior Representation in Physical Security Systems Simulation. Ph.D. Dissertation. Auburn University.
- Marechal T.M.A., Smith A.E., Ustun V., Smith J.S., and Lefeber A.A.J. (2009). Optimizing a physical security configuration using a highly detailed simulation model. In Marlin U. Thomas and Adedeji B. Badiru (Eds.), *Handbook of Military Industrial Engineering*. CRC.

## Author Biographies

**VOLKAN USTUN** is a postdoctoral research fellow in the H. Milton Stewart School of Industrial and Systems Engineering at the Georgia Institute of Technology. He has received the B.S. and M.S. degrees in Industrial Engineering from Middle East Technical University (METU) in 1997 and 2000, respectively and the Ph.D. degree in Industrial and Systems Engineering from Auburn University in 2009. Prior to the Ph.D. degree, he has also worked as a research engineer at The Scientific and Technical Research Council of Turkey (TUBITAK). His research interests mainly include discrete-event and agent-based simulation models and frameworks for complex systems.

**JEFFREY S. SMITH** is a professor in the Industrial & Systems Engineering department at Auburn University. Prior to joining Auburn, Dr. Smith was an associate professor in the Industrial Engineering Department at Texas A&M University. He received the B.S. in Industrial Engineering from Auburn University in 1986 and the M.S. and Ph.D. degrees in Industrial Engineering from Penn State University in 1990 and 1992, respectively. In addition to his academic positions, Dr. Smith has held industrial positions at Electronic Data Systems and Philip Morris U.S.A. Dr. Smith is an active member of IIE and INFORMS.

# Identification and Application of Neurophysiologic Synchronies for Studying the Dynamics of Teamwork

*Ronald H. Stevens*

*Trysha Galloway*

UCLA IMMEX Project

5601 W. Slauson Avenue #255

Culver City, CA 90230

Telephone: (310) 649-6568

immex\_ron@hotmail.com, trysha@immex.com

*Chris Berka*

*Adrienne Behneman*

Advanced Brain Monitoring, Inc.

2237 Faraday Avenue, Suite 100

Carlsbad, CA 92008

Telephone: (760) 720-0099

chris@b-alert.com, abehneman@b-alert.com

Keywords:

Teamwork, Neurophysiologic Synchrony, Artificial Neural Networks, EEG

**ABSTRACT** *We describe a process for collecting and combining neurophysiologic signals derived from individual members of a team to develop pattern categories showing the normalized expression of these signals at each second for the team as a whole. The expression of different neurophysiologic synchrony patterns is sensitive to changes in the behavior of teams over time and perhaps to the level of expertise. The utility and limitations of using this approach are demonstrated for three tasks including a team emotion recall research study, an educational study where teams of high school students solved substance abuse simulations and a complex training study where Submarine Officer Advanced Candidate trainees performed submarine piloting and navigation exercises.*

## 1. Introduction

Research on teamwork and cooperative behaviors often adopts an input-process-output framework (IPO). In this model the interdependent acts of individuals convert inputs such as the member and task characteristics to outcomes through behavioral activities directed toward organizing teamwork to achieve collective goals. These activities are termed team processes and include such activities as goal specification, strategy formulation, systems and team monitoring (Marks et al, 2001). Much of this teamwork research has made use of externalized events focusing on *who* is a member of the team, *how* they work together and *what* they do to perform their work. The studies often rely on post-hoc elicitation of the subjective relationships among pertinent concepts. There have been fewer studies looking at the *when* of

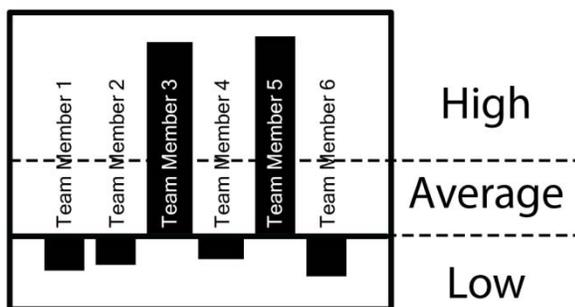
teamwork interactions although the dynamics of team function are known to be complex (Mathieu et al, 2008) with temporal models of teamwork suggesting that some processes transpire more frequently in action phases and others in transition periods (Canon-Bowers et al, 1993; Cohen & Bailey, 1997; Cooke et al, 2003; Mohammed et al, 2000).

Our hypothesis is that as members of a team perform their duties each will exhibit varying degrees of cognitive components such as attention, workload, engagement, etc. and the levels of these components at any one time will depend (at least) on 1) what that person was doing at a particular time, 2) the progress the team has made toward the task goal, and 3) the composition and experience of the team. Given the temporal model of team processes, we believe that the balances of these metrics across the members of the

team will not be random, but will be in rhythm with the team’s changing activities and awareness of the situation. In this study we provide a direct confirmation of this hypothesis.

## 2. What Are Neurophysiologic Synchronies?

We define neurophysiologic synchronies (NS) as the second-by-second quantitative co-expression of the same neurophysiologic / cognitive measures by different members of the team. Figure 2.1 shows an illustration of a neurophysiologic measure being simultaneously detected at a particular point in time from the members of a hypothetical six person team where team members 3 and 5 expressed above average levels of this particular measure while team members 1, 2, 4 and 6 expressed below average levels.



**Figure 2.1.** Example Expression of a Generic Neurophysiologic Measure by Individual Members of a Six-Person Team

## 3. How are Neurophysiologic Synchronies Detected and Analyzed?

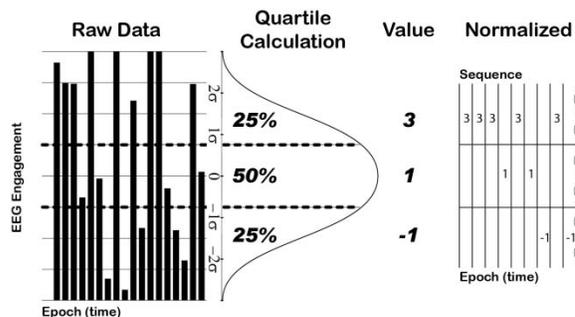
The data processing begins with the eye-blink decontaminated EEG files containing second-by-second calculations of the probabilities of High EEG-Engagement (EEG-E), Low EEG-E, Distraction and High EEG-Workload (EEG-WL) (Levendowski et al, 2001, Berka et al, 2004). Most of the studies to date have used the High EEG-E and EEG-WL metrics.

The EEG engagement (EEG-E) index is related to processes involving information-gathering, visual scanning, and sustained attention (Berka, 2004). EEG-E was derived using a four-class quadratic DFA representing the continuum Sleep Onset, Distraction, Low Engagement, and High Engagement. The four-

class model was constructed using absolute and relative power spectra variables from the 1-40 Hz bins of EEG channels Fz-POz and Cz-POz. The model was created using stepwise regression on a database of over 100 participants under fully rested and sleep-deprived conditions, and validated on an additional 100 subjects.

Three 5-minute baseline conditions were used to derive the DFA coefficients used to individualize the model for each participant: The first 5 min of a 3-choice vigilance task, eyes open paced response task, and eyes closed paced response task. EEG collected during these conditions was used to establish the model for output classes High Engagement, Low Engagement, and Distraction, respectively.

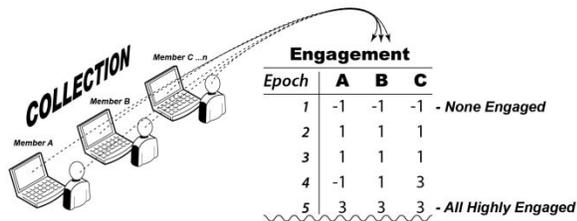
In prior studies with individuals performing complex tasks the raw EEG-E levels were used for studying the problem solving dynamics (Stevens et al, 2007, 2008). Studying team processes using EEG measures; however, requires a normalization step, which equates the absolute levels of EEG-E of each team member with his own average levels. This allows the identification not only of whether an individual team member is experiencing above or below average levels of EEG-E or EEG-WL, but also whether the team as a whole is experiencing above or below average levels.



**Figure 3.1.** Normalization of Neurophysiologic Measures into Quartile Ranges.

In this normalization process (outlined for one individual in Figure 3.1.) the EEG-E levels are partitioned into the upper 25%, the lower 25% and the middle 50%; these are assigned values of 3, -1, and 1 respectively, values chosen to enhance subsequent visualizations. The next step combines these values at each epoch for each team member into a vector representing the state of EEG-E for the team as a

whole, (this is shown for a team of 3 persons in Figure 3.2.).

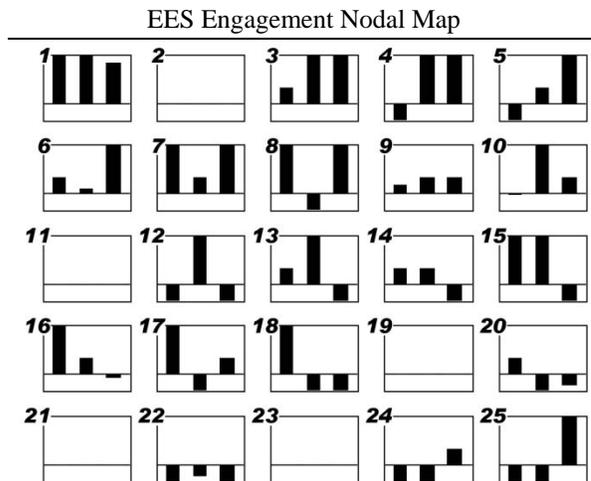


**Figure 3.2.** Creation of Team Performance Vectors. While the process is illustrated for three-member teams it can be expanded to include larger or smaller teams.

The second-by-second normalized values of team EEG-E for the entire episode are then repeatedly (50-2000 times) presented to a 1 x 25 node unsupervised artificial neural network. During this training a topology develops such that the EEG-E vectors most similar to each other become located closer together and more disparate vectors are pushed away. The training results in a linear series of 25 team EEG-E patterns termed neurophysiologic synchronies (NS).

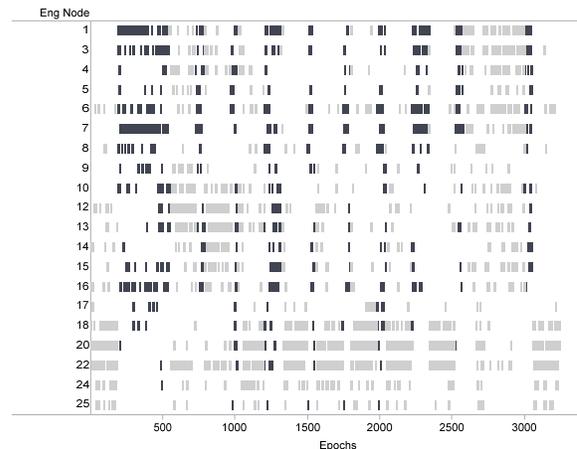
#### 4. A Simple Example: Emotion Recall by a Team

A simple exercise in emotion recall by three team members illustrates the application and applicability of neurophysiologic synchronies for studying the dynamics of teamwork. In this exercise three team members were asked to recall different emotions while wearing an ABM wireless EEG sensor headset. The emotions included anger, grief, hate, joy, romantic love, platonic love, reverence and good learning and bad learning. Each three minute period of emotion recall was separated by 1-2 minutes of rest time before the next emotion was elicited. During both the emotion recall and the rest periods there was minimal talking and the subjects tended to focus on a region of space and / or object. EEG-E and EEG-WL were collected at 1 second epochs, normalized as described in Figures 2.1. & 3.1. and used to train unsupervised ANN. The resulting EEG-E NS patterns are shown in Figure 4.1. The most common NS was pattern 22 representing the epochs where all individuals expressed low levels of EEG-E and this was followed by node 20 where only individual #1 showed elevated EEG-E.



**Figure 4.1.** Neurophysiologic Synchronies for EEG-E and EEG-WL During Emotion Recall

The time course of EEG-E expression for the session is shown in Figure 4.2. at each second of the exercise. The epochs in black indicate resting periods and those in gray indicate recall of emotions.



**Figure 4.2.** Neurophysiologic Synchronies for EEG-E During Emotion Recall

Neurophysiologic Synchronies # 20 and 22 were associated with most of the emotion expression shown during epochs 600-2500 and these were characterized by below normal expression of EEG-E by all members of the team. The exceptions to this pattern were for the emotions anger and hate. During these epochs individual #2 showed above average expression of EEG-E while individuals 1 & 3 were still average / below average in EEG-E expression. These NS were also not associated with the Resting period or the Unknown periods; the Unknown period was a resting

period that was extended for 7 minutes. The epochs where 2 or more members of the team showed elevated EEG-E levels were primarily found during the resting periods.

Thus, in a simple teamwork task with little interaction among the team members a consistent pattern of NS expression could be observed which varied with the properties of the task. Interestingly, periods of low EEG-E expression were associated with the active portion of the task suggesting that these low levels do not indicate lack of engagement, but rather the lack of external involvement of each individual.

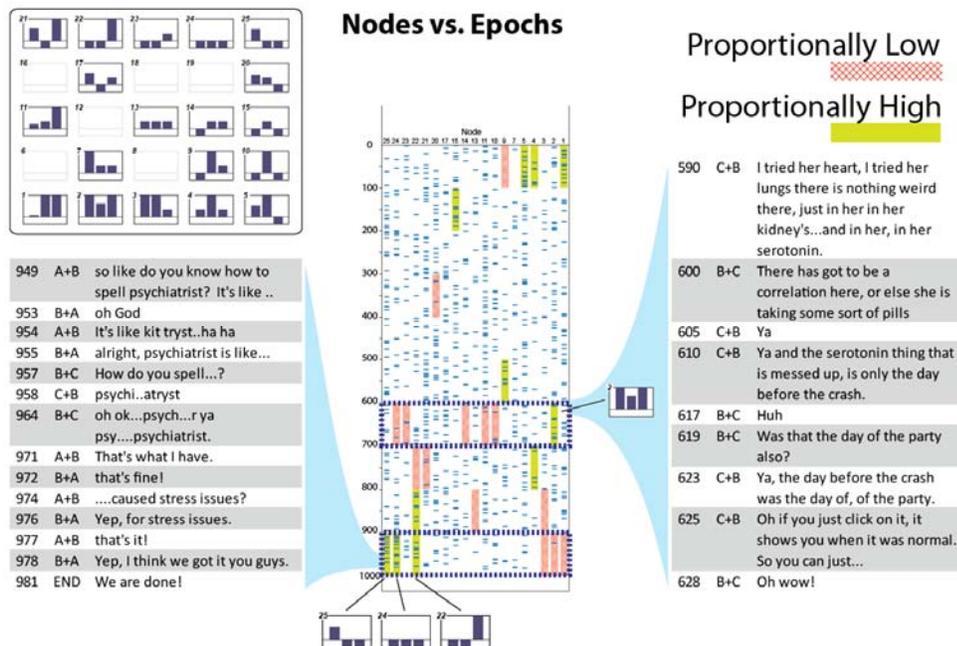
From the perspective of neurophysiologic synchronies and teamwork, the emotion recall results are important as they show that the different members of the team *consistently* entered a particular neurophysiologic state during the elicitation of emotions and they consistently exited that state during the rest periods. This was observed both for EEG-E and EEG-WL although it was more pronounced with the EEG-E. As the team was not engaged in verbal communication, it also indicates that the state that was entered into during emotion recall was not dependent on active communication among the team members but was more related to the internal representation of the task being generated by each of the team members. Thus NS expression may be a reflection of the internal state

of team members and of the team as a whole.

### 5. A More Complex Teamwork Simulation: Substance Abuse Decision Making.

The second task represents an educational activity where teams of three high school students explored an online IMMEX™ problem space where the goal was to make a decision whether the simulated person should seek help for substance abuse. One member of the team accesses physiologic and neurophysiologic data, one member examined social issues such as school / job performance, difficulties with the law, interactions with peers, etc, and the third person leads the group interactions and guided the decision.

During the task audio and video recordings were made of each student enabling a reconstruction of team member actions and the interactions of the group, allowing a mapping of NS expression to team events. An example of this mapping for one of six groups is shown in Figure 5.1. Here two segments of the team discussions are highlighted, one where EEG-E levels were low and another where they were high. During the period where EEG-E NS was low the team conversation focused on determining how to spell 'psychiatrist' whereas when high, the team was involved in a formulation of a final decision.



**Figure 5.1.** Mapping Different NS Expressions to Collaboration Events and Discussions. The NS patterns for the group are shown in the upper left corner and their expression is shown for each epoch. The highlighted segments represent areas where particular NS patterns are expressed at higher or lower levels by cross tabulation. Two segments of the discussions are highlighted where particular NS were either high or low.

## 6. A Very Complex Teamwork Simulation: Submarine Piloting and Navigation

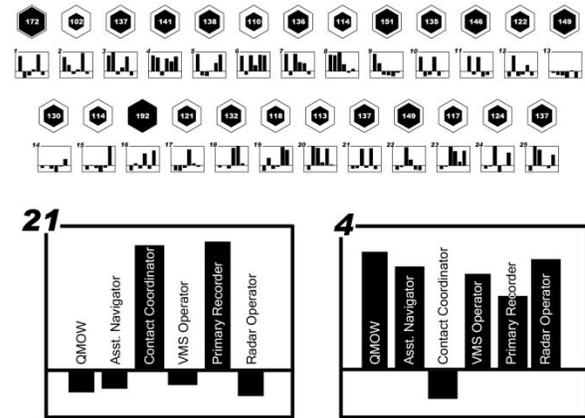
The final example shows the application of the approach to a very complex training task which is the safe piloting of a submarine. These studies were conducted with navigation training tasks that are integral components of the Submarine Officer Advanced Course (SOAC) where Junior Officers train to become department heads and ship drivers.

The task the trainees performed is a high fidelity Submarine Piloting and Navigation (SPAN) simulation that contains dynamically programmed situation events which are crafted to serve as the foundation of the adaptive team training. Such events in the SPAN include encounters with approaching ship traffic, the need to avoid nearby shoals, changing weather conditions, and instrument failure. There are also task-oriented cues to provide information to guide the mission, and team-member cues that provide information on how other members of the team are performing / communicating. Finally there are adaptive behaviors that help the team adjust in cases where one or more members are under stress or are not familiar with aspects of the unfolding situation.

Each SPAN session begins with a briefing detailing the navigation mission including a determination of the static position of the ship; weather conditions; potential hazards; and overall plan of the mission. This section is followed by the simulation which can last from 20 – 60 minutes or more. The simulation is then paused and a debriefing session begins that helps teams monitor and regulate their own performance based on the dimensions of teamwork deemed critical for effective team performance: From a cognitive perspective this teamwork task is complex, requiring not only the monitoring of the unfolding situation and the monitoring of one’s work with regard to that situation, but also the monitoring of the work of others.

Each neurophysiologic synchrony shows a pattern of EEG-E for each member of the team and provides a snapshot of the overall team engagement. As an example, NS 21 indicates a pattern where the Contact Coordinator (Position 3) and Primary Recorder (Position 5) are highly engaged and the other 4 team

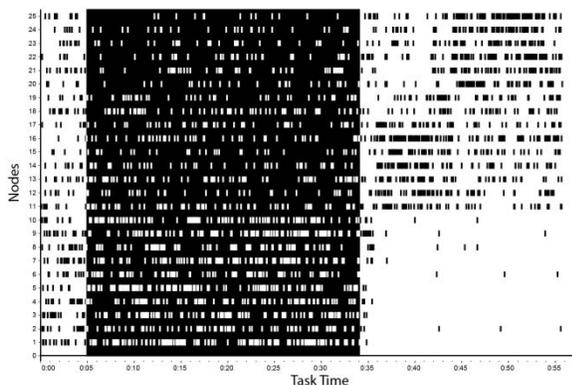
members are at below average levels of engagement (Figure 6.1). Node 4 indicates a pattern where the Contact Coordinator (Position 3) is below average in EEG-E expression and the team members at the other positions have high levels.



**Figure 6.1.** The Neurophysiologic Synchrony and Frequency Map for a Submarine Piloting and Navigation Team. The neurophysiologic synchrony patterns are shown by the histograms in the boxes representing each neural network node, and the frequency of occurrence of each neurophysiologic synchrony is shown by the degree of fill in the hexagons. An expanded view of patterns 21 and 4 are shown in the lower portion of the figure.

The neurophysiologic synchronies so defined, can then be applied to explore multiple dynamics of teamwork such as: 1) Does the quantitative and qualitative expression of NS patterns change with varying task demands? 2) Is the team’s convergence toward shared situation awareness reflected in NS patterns? 3) Do preferred NS patterns change with team experience?

The following example shows how the expression of different neurophysiologic synchrony patterns changes over the course of a SPAN task by one team (Figure 6.2.) with the pre-briefing epochs (0-4 minutes), simulation epochs (4-35 minutes), and the debriefing epochs (35-55 minutes) highlighted.



**Figure 6.2.** Distribution of Neurophysiologic Synchrony Patterns during a SPAN Performance. The NS expressed at each second of the session are plotted vs. the task time. The initial segment on the left is the briefing period, the darkened section in the middle is the simulation itself, and the final segment to the right is the de-briefing segment.

The most noticeable difference was the near absence of NS 1-10 expression during the debriefing section; instead these were replaced by NS 11-25 which are those NS where the majority of team members expressed low EEG-E levels. These appeared as soon as the debriefing began, and it is interesting that they are expressed infrequently during the simulation suggesting a difference in team coordination across these two task segments. After several minutes of the debriefing there was elevated expression of NS 21-25 which represents moments where the team members, especially the contact coordinator, are expressing above average levels of EEG-E.

The differences between the pre-briefing and the simulation are less striking, perhaps due to the relatively short briefing period, but statistical comparisons (cross tabulation) showed that NS 1, 9 and 10 were underrepresented during this segment (this is where the common feature is the Navigator and Primary Recorder have high EEG-E levels) and synchrony 16 was over represented (this is where the VMS and Radar Operators had elevated EEG-E). These results suggest that neurophysiologic synchronies can change rapidly in response to changing task situations and that the changed synchrony patterns can persist over periods of 10 minutes or more.

## 7. Discussion

One of the challenges for extending the measurement of team behavior is the development of unobtrusive and real-time measures of team performance that can be practically implemented (Salas et al, 2008). We believe that the approach we have described begins to address some of these challenges and can be applied to a wide variety of team tasks.

Neurophysiologic synchronies represent a low level data stream that can be collected and analyzed in real time and in realistic settings. Our goal for studying NS expression is to be able to rapidly determine the functional status of a team in order to assess the quality of a teams' performance / decisions, and to adaptively rearrange the team or task components to better optimize the team. The neurophysiologic measure we have used for this study is a measure of engagement in the sense that high levels represent a state of external awareness while low levels better represent an introspective state.

The usefulness of this approach will depend on the cognitive indicator chosen. In parallel studies we have similarly modeled an EEG-derived measure of workload and the NS with the same teams show very different dynamics from those described here with EEG-E. An important challenge will be relating the dynamics of any new cognitive measure to the team task to best determine what aspects of team cognition are being measured.

Three examples were presented, one from a research perspective, one from an educational perspective, and one from a training perspective. In all three examples extended periods of time (minutes or more) were observed where NS patterns were preferentially expressed.

Analogous to the long memory phenomena embedded in some communication and other data streams (Gorman, 2005), there may also be information contained in the sequence of the neurophysiologic stream over longer time frames which may reflect more aspects of team cognition rather than individuals' immediate concerns with the task. Some suggestion that may be so comes from earlier autocorrelation studies where positive autocorrelations can be observed over 20 seconds or more (Stevens et al, 2009).

The second and third studies with the high school students in classrooms and the SOAC trainees in the SPAN similarly demonstrated that the techniques can be practically implemented in a variety of real-world situations. These studies also indicate that the approach can be flexibly scaled from three-person teams to teams with at least six team members.

Combined, these findings suggest that neurophysiologic indicators measured by EEG may be useful for studying team behavior not only at the milliseconds level, but at more extended time frames.

## 8. References

- AlZoubi, O., Calvo, R. A., & Stevens, R. H. (2009). Classification of EEG for Affect Recognition: An Adaptive Approach. *Lecture Notes in Artificial Intelligence*. A. Nicholson, X. Li (Eds.) Springer-Verlag Berlin Heidelberg. 5866, pp 52-61.
- Berka, C., Levendowski, D. J., Cvetinovic, M. M., Petrovic, M. M., Davis G. et al. (2004). Real-Time Analysis of EEG Indexes of Alertness, Cognition, and Memory Acquired With a Wireless EEG Headset. *International Journal of Human-Computer Interaction*. Lawrence Erlbaum Associates, Inc. **17**(2), 151-170.
- Cannon-Bowers, J. A., Salas, E., & Converse, S. A. 1993. Shared mental models in expert team decision making. In J. N. J. Castellan (Ed.), *Current issues in individual and group decision making*: 221-246. Hillsdale, NJ: Lawrence Erlbaum.
- Cohen, S. G., & Bailey, D. E. 1997. What makes teams work: Group effectiveness research from the shop floor to the executive suite. *Journal of Management*, **23**: 239-290.
- Cooke, N. J., Kiekel, P. A., Salas, E., & Stout, R. 2003. Measuring team knowledge: A window to the cognitive underpinnings of team performance. *Journal of Applied Psychology*, **7**: 179-199.
- Levendowski, D.J., Berka, C., Olmstead, R.E., Konstantinovic, Z.R., Davis, G., Lumicao, M.N., Westbrook, P. (2001). Electroencephalographic indices predict future vulnerability to fatigue induced by sleep deprivation. *Sleep* **24** (Abstract Supplement): A243-A244.
- Marks, M. A., Mathieu, J. E. & Zaccaro, S. J. (2001). A Temporally Based Framework and Taxonomy of Team Process. *The Academy of Management Review*. Vol. 26, No. 3. pp 356-376.
- Mathieu, J., Maynard, M. T., Rapp, T., & Gilson, L. (2008). Team Effectiveness 1997-2007: A Review of Recent Advancements and a Glimpse Into the Future. *Journal of Management*. Vol. 34, No. 3. Pp. 410-476.
- Mohammed, S., Klimoski, R., & Rentsch, J. (2000). The Measurement of Team Models: We Have No Shared Scheme. *Organizational Research Methods*, **3**, 123-165.
- Salas, E., Cook, N. J., Rosen, M. A. (2008) On Teams, Teamwork, and Team Performance: Discoveries and Developments. *Human Factors: The Journal of the Human Factors and Ergonomics Society* Vol. 50 (3): 540-547.
- Stevens, R. H., Galloway, T., and Berka, C. (2007). Allocation of Time, Workload, Engagement and Distraction as Students Acquire Problem Solving Skills in "Foundations of Augmented Cognition", 4<sup>th</sup> Edition, D. Schmorrow, D. Nicholson, J. Drexler and L. Reeves (eds). pp. 128-137.
- Stevens, R. H., Galloway, T., and Berka, C. (2007). EEG-Related Changes in Cognitive Workload, Engagement and Distraction as Students Acquire Problem Solving Skills. *Lecture Notes in Computer Science. User Modeling 2007*. Springer Berlin / Heidelberg. Volume 4511/2007, pp 187-196.
- Stevens, R. H., Galloway, T., and Berka, C. (2007). Exploring Neural Trajectories of Scientific Problem Solving Skill Acquisition. *Lecture Notes in Computer Science. Foundations of Augmented Cognition*. Springer Berlin / Heidelberg. Volume 4565/2007, pp 400-408.
- Stevens, R. H., Galloway, T., and Berka, C. (2007). Integrating Innovative Neuro-educational Technologies (I-Net) into K-12 Science Classrooms. *Lecture Notes in Computer Science. Foundations of Augmented Cognition*. Springer Berlin / Heidelberg. Volume 4565/2007, pp 47-56.
- Stevens, R. H., Galloway, T., and Berka, C., & Sprang, M. (2009). Can Neurophysiologic Synchronies Be Detected during Collaborative Teamwork?

*Proceedings: HCI International 2009*, July 19-24, San Diego, CA. pp. 271-275.

Stevens, R. H., Galloway, T., Berka, C., Johnson, R. & Sprang, M. (2008). Assessing Student's Mental Representations of Complex Problem Spaces with EEG Technologies. *Proceedings: Human Factors and Ergonomics Society 52nd Annual Meeting*, September 22-26, 2008, New York.

## 9. Acknowledgments

This work was supported by The Defense Advanced Research Projects Agency under contract number(s) NBCHC070101, NBCHC090054. The views, opinions, and/or findings contained in this article/presentation are those of the authors and should not be interpreted as representing the official views or policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the Department of Defense.

Special thanks to John Stallings for preparing the illustrations and to the Officers and Staff of the Submarine Learning Center for their participation in these studies, to Dr. Marcia Sprang and the students at Esperanza High School for their continued participation, and to Dr. Rafa Cavalio for the emotion recall studies.

Author contact: immex\_ron@hotmail.com

## 10. Author Biographies

**RON STEVENS, PH.D.** is a Professor of Microbiology, and member of the Brain Research Institute at the UCLA School of Medicine. He is the director of the internet-based IMMEX problem solving project which has engaged over 150,000 students and teachers in computational education and professional development activities that span elementary school through medical school. Most recently (2007) Dr. Stevens received the 'Foundations of Augmented Cognition' award from the Augmented Cognition Society. His current interests are the use of machine learning tools and electroencephalography (EEG) to model the acquisition of scientific problem solving skills.

**CHRIS BERKA**, CEO and Co-Founder of Advanced Brain Monitoring has over 25 years experience managing clinical research and developing and commercializing new technologies. She is co-inventor

of seven patented and seven patent-pending technologies and is the principal investigator or co investigator for grants awarded by the National Institutes of Health, DARPA, ONR and NSF. She has 10 years experience as a research scientist with publications on the analysis of the EEG correlates of cognition in healthy subjects and patients with sleep and neurological disorders.

**TRYSHA GALLOWAY** directs the EEG studies for The Interactive Multi Media Exercises (IMMEX™) laboratory and is co-author on eight peer reviewed published studies. Her research interests blend the population based advantages of probabilistic performance modeling with the detection of neurophysiologic signals to help personalize the learning process in complex education and training activities.

**ADRIENNE BEHNEMAN** is a Project Manager at Advanced Brain Monitoring. Since 2007, she has played a key leadership role in the Accelerated Learning, RAPID and ANITA projects. She is interested in the development of neuroscience-based tools to enhance training and education. Her current focus is on researching the psychophysiology of expertise in domains including marksmanship, deadly force decision making and team function, as part of the Accelerated Learning project.

# Developing a Cognitive Model of Expert Performance for Ship Navigation Maneuvers in an Intelligent Tutoring System

Jason H. Wong<sup>1</sup>, Susan S. Kirschenbaum<sup>1</sup>, Stanley Peters<sup>2</sup>

<sup>1</sup>Naval Undersea Warfare Center, Newport, RI

<sup>2</sup>Stanford University, Stanford, CA

jason.h.wong@navy.mil, susan.kirschenbaum@navy.mil, peters@csl.stanford.edu

## Keywords:

expert performance, ship navigation, perceptual heuristics, intelligent tutor, cognitive task analysis

**ABSTRACT:** *The goal of this project is to develop a cognitive model of expert ship-handling performance. This model was integrated with an intelligent tutoring system and an immersive visual simulation used by the U.S. Navy. This intelligent tutor and expert cognitive model (written in a Java-based version of ACT-R) provides feedback to the student based on the student actions in order to reduce workload on the instructors. The nature of ship navigation and the requirements for the intelligent tutor presented unique challenges for development. This paper describes how the resulting cognitive model balances a need for expert performance while compensating for student error, uses perceptual heuristics when the ACT-R vision module is not feasible, and how these and other issues affected model development. Future plans for system test and evaluation are also discussed in the context of improving training.*

## 1. Project Overview

The Conning Officer Virtual Environment (COVE) is a ship-handling simulation system used by the U.S. Navy to train officers in how to complete ship navigation maneuvers (known in the U.S. Navy as ship-handling “evolutions”). These can include docking a ship, getting a ship underway, or twisting a ship about its axis. This training occurs after students undergo classroom instruction, so this simulation provides a hands-on practice environment for novices. COVE, which is based on the Virtual Ship software (Computer Sciences Corporation, 2009), is used to provide students with ship-handling training without the cost or risk to equipment of at-sea exercises. One downside to this system is that an expert instructor is required to constantly monitor progress and provide feedback, no matter how basic the exercise.

In order to reduce the overall workload on instructors, the goal of this project was to develop a system consisting of a set of new components that interact with each other. One component is an intelligent tutor (Bratt, Schultz & Peters, 2007) that monitors student progress and provides appropriate feedback. The second component, and the subject of this paper, is a cognitive model (developed using a Java-based implementation of ACT-R; Harrison, 2009) of expert performance. This model is designed to represent expert performance in various ship navigation evolutions to provide a point of comparison against the actions taken by the student.

The requirement that the model represent human performance led to the selection of ACT-R (Anderson, et al., 2004; Anderson, 2007) as choice of cognitive architecture to implement the specific cognitive and perceptual operations used in completing an evolution. The use of a cognitive architecture guides the creation of a system that represents human cognition (and its limits) instead of a computer-based algorithmic solution that ignores the constraints of cognition.

The expert model was designed to provide the tutor with a sense of how an expert would perform the navigation evolution, including the actions taken, rules followed, and perceptual cues that are used. The entire system would then be able to give feedback to the student based on the actions taken and visual cues examined. While some cognitive models have been developed to operate with other components, few have been developed to support an intelligent tutoring system, and this presents a unique set of challenges.

## 2. Description of System Components

The task environment that the cognitive model operates in consists of multiple pieces. The primary component is the COVE simulation software itself. The simulation strives for realism in many important areas (Smallman & St. John, 2005), including elements in the visual environment such as hydrodynamics, weather, currents, piers, buoys, and ships. Some ships are also modeled in high-fidelity; that is, the physics of the engine and rudder are accurately modeled instead of the ship following a

simple speed and course. All of these elements are rendered in an immersive environment that can be displayed on a single monitor, in a more complex multiple monitor setup, or using a head-mounted display. A screenshot of the rendered scenario can be seen in Figure 1 (top). The multi-monitor setup is complete with head-tracking, control through voice recognition, text-to-speech capability, and a separate instructor console for monitoring performance.

COVE scenarios are created using specially designed software that includes detailed real-world ports and realistic ships that are placed in the environment. Ships can be given a set of waypoints to follow, new physical objects can be added, and the weather can be changed using the scenario creator (Figure 1, bottom).



Figure 1.

The student interacts directly with the COVE simulation, issuing verbal commands, listening to responses and status reports, and viewing the environment and the ship under their command (known as “ownship”). The intelligent tutoring system adds two components into this dynamic – an intelligent tutor and expert model (Figure 2). The tutor

monitors student progress and compare the student’s actions with those of the expert model in order to provide feedback. The expert model needs to accomplish various navigation evolutions and inform the tutor as to what actions were taken and why.

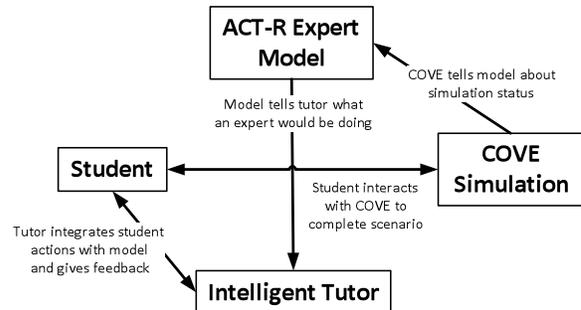


Figure 2.

This intelligent tutor/cognitive model system is designed to be implemented in the complex multi-monitor COVE simulators, which presented many challenges, including how vision is accomplished. The ACT-R vision component can only handle a single display, so a software solution was created to compensate for this shortfall and will be described in further detail later in the paper.

### 3. Ship Navigation Maneuvers

Several different ship navigation tasks varying in complexity were modeled. One basic evolution is intersecting a range, where a ship is transiting and must make a turn to intersect a new heading. While this may seem trivial, there is much skill in knowing when to begin the turn, how hard to take the turn, and when to ease off the engines and rudder. Another basic evolution is twisting a ship in a box, which involves rotating the ship on its pivot point without moving the ship forwards or backwards. This is difficult because students often do not have previous experience performing this kind of maneuver, and managing the engines and rudder so that the ship does not move laterally is a challenge.

Advanced navigation evolutions are also going to be modeled. To a great extent, these use more basic evolutions as building blocks (Rigeluth, 2007). For example, getting underway from the dock involves twisting the ship away from the pier, transiting forward and then making a turn to go out to sea. There is more to keep track of with these complex tasks, but they still use basic maneuvers at their core.

The intelligent tutoring system was not designed to replace the current ship-handling curriculum already in place. Instead, the system would augment training by supporting the scaffolding approach already taken by the course: begin by mastering simple maneuvers, then grow those into more complicated ones over the length of the training. The tutoring system supports both simple and complex maneuvers, so students can use the system throughout the course.

All of these tasks are difficult for students to grasp due to a number of factors. The hydrodynamics of maneuvering a ship can be difficult to understand, since students rarely have prior experience with ship-handling. Additionally, the tools available to affect the ship's speed and heading (rudder, port and starboard engines, and a tugboat in some cases) work differently when used in different combinations. Finally, there is often a lag between issuing a command (e.g., "All engines ahead full") and observing the effect of that command (e.g., increased speed), so a comprehension of cause-and-effect can take time to develop.

Another factor is that there are many paths to accomplish the same goal. One expert may attempt to increase the rate of ownship turn by increasing engine speed while another may instead decide to set the rudder farther over. Both options are correct, and it was important to capture all the possibilities. Also, different experts may teach their preferred method of accomplishing a task, increasing the necessity of the cognitive model and tutor to accommodate all the action paths available to the student. Finally, if the student deviates from a given parameter, the expert model must still function even if the action was not one of an "expert."

Implementing these ship navigation maneuvers in a cognitive model was difficult due to the perceptual nature of these maneuvers. Intersecting a range requires starting a turn, assessing speed through visual cues such as motion parallax (the apparent displacement of objects caused by a change in observer position), and lining up two separate range markers to ensure that the ship is in the ideal position in a harbor channel. These perceptual judgments often occur in the form of heuristics. An example heuristic used by baseball outfielders is that they will keep a constant visual angle between themselves and the ball instead of performing complex calculations (McBeath, Shaffer & Kaiser, 1995). These heuristics also apply to ship navigation and were implemented into a cognitive model. Determining how these strategies are

used was derived from a combination of expert interviews and observing ship-handling performance.

## 4. Expert Model Development

ACT-R was a natural choice of cognitive architecture due to the requirement of cognitive plausibility of the expert ship-handling model. Due to the necessity for the cognitive model to communicate with COVE and the intelligent tutor, the model was developed in Java-based jACT-R (Harrison, 2009) instead of Lisp-based ACT-R. Using Java increased compatibility with other system components and was more easily modified by those unfamiliar with Lisp, since Java is a more accessible language. jACT-R was designed to be as similar to ACT-R as possible, especially in terms of retaining the aspects of cognitive plausibility in the architecture.

The primary focus of developing a cognitive model for this project is the accurate modeling of several factors. One factor involves the possible actions that an expert could perform in order to maneuver the ship, and another is the perceptual monitoring and scanning behaviors that takes place to ensure successful completion of a navigation task.

### 4.1 Task Analysis Foundations for the Model

In order to develop a cognitive model of expert ship navigation, subject matter experts from the Naval Surface Warfare Officers School were consulted over multiple sessions. One phase of information collection involved watching students practicing using COVE and examining the feedback that instructors provided them. By analyzing the tone (positive or negative) and content of the feedback (pre-action advise or post-action critique), an understanding was developed of what aspects of ship-handling were emphasized and evaluated by human instructors. This influenced the development of the intelligent tutor as well as the expert model. For example, it became quickly apparent that the use of perceptual cues was critical to success. Also, a majority of the feedback came after an action was taken, so the student had to be allowed to make a mistake first.

Another phase of information collection centered around how course instructors performed various ship-handling maneuvers. Experts were interviewed and observed performing these tasks in the COVE simulator. These sessions were analyzed and distilled into cognitive task analyses. These took the form of a

traditional task analysis (which lists a sequence of observable tasks). Additionally, internal cognitive processes were also taken into account (Zachary, Ryder & Hicinbothom, 1999) and included.

The task analysis framework known as GOMS (goal, operator, method, selection; Card, Moran & Newell, 1983; John & Kieras, 1996) was selected for the task analysis because of the hierarchical nature of the tasks. There is a specific order as to which events happen, so representing tasks as a series of goals and sub-goals provided a great deal of benefit when translating these task analyses into cognitive models. For some navigation evolutions, GOMS-like task analyses were already completed (Grassi, 2000), so they were integrated into this project.

However, navigation maneuvers do not lend themselves perfectly to GOMS modeling. GOMS does not take into account the perceptual cues that are used in ship navigation. As an example, Figure 3 shows two range markers (the orange and white boards) that serve as a visual cue for ownship heading when they are lined up with the bow jackstaff. While GOMS is able to support a goal such as “Monitor speed until desired heading achieved,” there are a number of perceptual cues that indicate heading (such as range markers) that may be used in various combinations, and GOMS does not include a method for incorporating these visual cues.



**Figure 3.**

Due to this shortfall, a Critical Cue Inventory (CCI) was created to support a list of perceptual cue descriptions that could be used to accomplish a goal. The CCI could also include heuristics as to when a particular visual cue is more likely to be used, which

aided in building the expert cognitive model. An example truncated CCI used for determining the rate of swing of the bow can be found in Table 1.

**Table 1.**

<b>Critical Cue Inventory for: Determine Rate of Swing of Bow</b>	
<b>CUE</b>	<b>DESCRIPTION</b>
Jackstaff	Examine the rate of swing of the jackstaff compared to a fixed environmental object. Used when there is a physical landmark present.
Rate of Turn indicator	Interpret the Rate of Turn visual indicator in the COVE instrument cluster.
Change in heading	Determine how quickly the heading is changing over time using the various heading indicators. Used when there is a lack of landmarks.

#### 4.2 Goal Stack Component

The cognitive model built in jACT-R used many standard components in ACT-R models, including the goal and retrieval buffers. Nonetheless, there are several noteworthy characteristics of the model that arose from project requirements. The first is the implementation of a goal stack that drives the entire execution of the cognitive model. Due to the hierarchical nature of navigation evolutions, it is only natural to create a chunk that can hold multiple goal levels in the goal buffer. Various productions push and pop goals from the stack, and the state of the goal stack is checked during the conflict resolution process to determine which production to fire next.

Certain steps in a navigation evolution must occur at a specific time, and properly utilizing the goal stack assisted with this need. Knowing when a turn is complete, for example, requires monitoring ownship heading or lining up the jackstaff with an environmental object. The production to stop the turn (i.e., “approaching-heading-NOW-stop-turn”) needs to fire when the goal stack matches specific conditions. The bottom of the goal-stack needs to match the basic goal of “make-turn,” and a goal at the top needs to match the goal of “monitor-until-desired-heading-reached.” Once these conditions are met, the production “pops,” or removes, the old goal from the stack and “pushes” on a new goal which, presumably, would stop the turn by shifting the rudder or slowing

the engines.

This implementation aids in creating generic productions that are needed by many different higher-level goals and may occur multiple times throughout an evolution (e.g., “activate-rudder”). The entire goal stack does not need verification – instead, the production only needs to check the top goal. None of the other goals below need to be checked – for example, it does not matter whether a lower-level goal is “make-turn” or “twist-ship.” Either way, the rudder must be activated. Another advantage from being able to check specific goals within the goal stack is that multiple possible action paths to complete a task are easily implemented.

Figure 4 contains a jACT-R pseudocode example that checks the goal stack. The top production is generic and may be called multiple times in an evolution. This production also does not need to verify the entire goal stack. The bottom production must occur at a specific time and checks each level of the goal stack.

```

<!-- This generic production only needs to
ensure the top goal is to issue the engine order -->
<production name="issue-starboard-engine-order">
  <conditions>
    <match buffer="goal" >
      <slot name="goal-2" equals="issue-starboard-
engine-order"/>
  ...

<!-- This specific production ensures the top goal
matches the desired goal and the other goals also
match (or are clear) -->
<production name="monitor-speed-heading-until-
turn-time">
  <conditions>
    <match buffer="goal" >
      <slot name="goal-1" equals="ownship-ahead"/>
      <slot name="goal-2" equals="monitor-until-turn-
time"/>
      <slot name="goal-3" equals="clear"/>
      <slot name="goal-4" equals="clear"/>
  ...

```

**Figure 4.**

### 4.3 Use of Perceptual Cues

There are many visual cues in the environment that an expert uses to properly execute ship maneuvers. It was necessary to pull this information from the COVE simulation directly instead of going through the jACT-R vision module, but it was critical to maintain cognitive plausibility for vision in the model, so the software pulling information from the COVE simulation must act “behind the scenes” to fill a

jACT-R buffer that is accessible to the model. Even a basic subgoal, such as visually scanning to assess ship status (speed, heading, etc.) required accurate cognitive modeling. Experts will often alternate between paying attention to the environment and to the ship status indicators while executing a navigation evolution, and cycling between these objects takes place frequently. This behavior was assessed in experts through interviews and head-tracking within the COVE system.

This scanning behavior was inserted into the model so the expert model’s current awareness of the situation correctly reflects the experience of a human expert. From this, the intelligent tutor can detect if the student is exhibiting similar scanning behavior. If this was not the case, the tutor can issue prompts to the student to check speed, heading, rudder status, and other important parameters.

Another example that demonstrates the criticality of the vision system is the monitoring of specific perceptual thresholds (e.g., to know when to begin and end turns, when ownship is far enough away or close enough to the dock, etc.). Experts do not intently stare at one location in the environment waiting for this threshold to be passed, nor are they able to focus on more than one area at once. Instead, scanning behavior is used (again derived from interviews and head-tracking), and the expert model needed to accurately capture this behavior.

The standard vision module within jACT-R is able to gather visual information from a display using attentional and imagery constructs, and locations are represented by their x- and y-positional screen coordinates. This vision scheme has been implemented successfully in heavily perceptual tasks such as driving (Salvucci, Boer & Liu, 2001). However, the COVE simulation is too complex to use the relatively basic jACT-R vision module. This is because the visual scene is distributed amongst multiple monitors and computers, which the vision module cannot handle. Instead, information must be passed directly from the simulation to the Java core of jACT-R using a client-server model. This information is then inserted into a buffer created for this model.

The solution to this problem was to implement “vision” through external software. COVE generates the rendered environment and keeps track of some environmental objects (e.g., piers), the environmental conditions (e.g., current and wind speed), and the

status of all the ships (e.g., speed and course of ownship, tugs, etc.). The simulation does not keep track of objects such as buoys, and the locations of those objects had to be measured manually and inserted into a separate database. Together, these components possessed the information that the expert cognitive model would otherwise try and obtain through more traditional means of vision. It was more efficient and easier to retrieve the necessary information about the environment directly from COVE instead of attempting to adapt the jACT-R vision module.

A technical necessity for the entire tutoring system was the separation of various system components. The COVE simulation needed to remain a separate entity. While the expert model is an important piece of the intelligent tutor, the tutor itself also needed the option to run as a standalone component. Due to the need for separation between each system component, software bridges were built to interface between the COVE software, the Java core of jACT-R, and the intelligent tutor. The bridge between COVE and jACT-R requests information from COVE and then fills a custom jACT-R buffer that is accessible by the cognitive model. This buffer was programmed as an Eclipse IDE plug-in and imported into jACT-R.

A simple example will help illustrate this process. In the case of monitoring ownship speed, a human expert would look down at a console (on a monitor separate from the rendered environment) and read off the speed. For the cognitive model to do this, it would request the speed from the COVE-ship-state buffer (similar to requesting a particular chunk from the retrieval buffer) first. This buffer is refreshed by the COVE/Java bridge, which periodically queries the COVE software as to the state of the simulation, which includes ownship speed information.

This software bridge allows for the simulation of many of the visual cues utilized by human experts but is pulled directly from the simulation. Therefore, the perceptual cues and heuristics used by human experts are still present in the cognitive model because the software bridge is abstracted away from the model. This abstraction allows for the ability of the model to accomplish something akin to traditional vision in a cognitively plausible manner.

#### 4.4 Cognitive Plausibility

Defining cognitive plausibility for this expert

cognitive model was different from many other models. The purpose of the expert model within the tutoring system is to act as an “answer key” to compare against student actions. Therefore, the expert is not supposed to commit errors. For example, there is no need to learn new actions, nor is there need for millisecond accuracy in cognitive function. Also, memory decay was not implemented. Instead, the expert model implemented visual scanning behavior between the environment and ownship status indicators to refresh memory. This reflects student behavior because they are often told not to trust their own memory.

A plausible expert, especially one that is used as a yardstick to measure human students against, should always perform an evolution as flawlessly as possible. However, an expert model that is part of a tutoring system must also be able to adapt to student behaviors, which are not always optimal. The first iteration in creating a model for any ship-handling evolution represented optimal performance of a maneuver. Once this was achieved, multiple action paths were built out from this single path. For example, the expert model knows the optimal distance from a range in which to make a turn, and this behavior is the model default. If the student overshoots this range, the expert model was designed to compensate for the error. This action path may result in using a greater amount of rudder than is typically called for. While suboptimal, it was important that the model possess these behaviors both for some degree of cognitive plausibility and to be a useful components of the intelligent tutor.

One area where it was especially important to maintain cognitive plausibility was in visual scanning behavior. If a computer program was written that did not take into account the limits of human cognition, then a student would be compared to a computer instead of a simulated human expert. While a computer could monitor multiple information streams at once, this would not reflect human cognition. Instead, a reflection of expert human behavior would require scanning multiple sources of data in a serial manner.

## 5. Empirical Validation

Traditional validation of cognitive models seeks to match human performance with that of the model, typically on a temporal scale. For example, an accurate cognitive model of visual search should generate target detection times that are similar to

human reaction times. Here, we are attempting to match the perceptual actions of an expert instead. The actions taken by the model do need to occur with some degree of temporal accuracy, but the millisecond modeling accuracy of jACT-R is not necessary for this application.

An important first step of validation involves demonstrating a complete system to the instructors who will be using the system for instruction. This has already been done with the standalone intelligent tutor system, which only contained rudimentary knowledge about ship-handling (e.g., maximum limits on speed). The system was well-received by instructors, who noted that feedback on perceptual components of the task would add to the utility of the final product.

Further validation steps have not occurred but are currently being planned. One important step in this model validation plan will examine how the entire tutoring system performs when a novice student uses the simulation. This will be tested on actual students taking a ship-handling course, but can also be tested on non-student novices. The critical data to collect from these experiments is the performance of the expert model. This is to ensure that the model was able to traverse the multiple action paths in response to student performance. For example, if a novice stops a turn too late, the model should react by shifting the rudder in the opposite direction. If the model had direct control of the ship, this mistake should not have happened in the first place. However, the model does not have control and must compensate for many errors that a novice can make. This will require many novices and hours of testing, but will serve to make a more robust model.

A critical final test of the intelligent tutor and expert model system is to determine how training is improved with use of this system. Improvement will be measured along several factors, including performance in ship maneuvering (measured across several variables such as time to completion and deviation from optimal channel position), amount of training retention, and number of human instructor hours required during training. The hope is that the tutor can increase the number of students that a single instructor can supervise while maintaining the same level of (or improving) training effectiveness.

## 6. Conclusions

While there have been other projects that have

integrated a cognitive model into a larger framework, these have mostly focused on training applications by creating a simulated teammate to work with other humans (Scolaro & Santarelli, 2002; Ball, et al., 2009). The project described here also works within a larger framework, but not in a team context. Instead, the model represents a single expert that changes its behavior in direct response to student actions.

This project presents a unique application of the jACT-R cognitive architecture in many ways. The requirements for the project necessitated the development of an expert cognitive model that needed to balance cognitive plausibility with near-flawless expert performance, perceptual heuristics without actual vision, and multiple action paths with an emphasis on tutoring. As intelligent tutoring systems are becoming increasingly popular, it is important to understand how cognitive modeling can add to these systems in a useful way.

While an expert model “answer key” cannot make mistakes such as memory retrieval failures, the model must compensate for student errors in order to remain useful. This required a far more extensive knowledge gathering period in order to explore task performance more fully, and also requires a greater testing period to ensure that many practical possibilities for behaviors that deviate from the optimum are accounted for.

Intelligent tutoring systems are often used in complex environments, which requires ACT-R models to perceive information that cannot be retrieved through the current, primitive vision module. Instead, software bridges must interface between the simulation and ACT-R itself, but the simulation environment must remain abstracted away from the model to maintain cognitive plausibility. Overall, cognitive modeling has a great deal to offer intelligent tutoring systems, and an optimal methodology to create these models is currently being shaped.

## 7. References

- Anderson, J. R. (2007). *How Can the Human Mind Occur in the Physical Universe?* New York: Oxford University Press.
- Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review*, *111*(4), 1036-1060.
- Ball, J., Myers, C., Heiberg, A., Cooke, N., Matessa,

- M., & Freiman, M. (2009). The Synthetic Teammate Project. *18th Conference on Behavior Representation in Modeling and Simulation*, Sundance, UT.
- Bratt, E. O., Schultz, K., & Peters, S. (2007). Challenges in Interpreting Spoken Military Commands and Tutoring Session Responses. *Grammar Engineering Across Frameworks 2007*, Stanford, CA.
- Card, S., Moran, T., & Newell, A. (1983). *The Psychology of Human-Computer Interaction*. Hillsdale, NJ: Lawrence Erlbaum Associates.
- Computer Sciences Corporation. (2009). Virtual Ship [Computer Software]. Retrieved December 1, 2009.
- Grassi, C. R. (2000). A task analysis of pier-side ship-handling for virtual environment ship-handling simulator scenario development. Masters Thesis at the Naval Postgraduate School.
- Harrison, A. M. (2009). jACT-R [Computer Software]. Retrieved December 1, 2009, from <http://jactr.org/>.
- John, B. E., & Kieras, D. E. (1996). The GOMS Family of User Interface Analysis Techniques: Comparison and Contrast. *ACM Transactions on Computer-Human Interaction*, 3(4), 320-351.
- McBeath, M. K., Shaffer, D. M., & Kaiser, M. K. (1995). How baseball outfielders determine where to run to catch fly balls. *Science*, 268(5210), 569-573.
- Reigeluth, C.M. (2007). Order, first step to mastery: An introduction to sequencing in instructional design. In F. Ritter, J. Nerb, E. Lehtinen, & T. O'Shea (Eds.), *In Order to Learn: How the Sequence of Topics Influences Learning*. New York, NY: Oxford University Press.
- Salvucci, D. D., Boer, E. R., & Liu, A. (2001). Toward an Integrated Model of Driver Behavior in a Cognitive Architecture. *Transportation Research Record*, 1779, 9-16.
- Scolaro, J., & Santarelli, T. (2002). Cognitive Modeling Teamwork, Taskwork, and Instructional Behavior in Synthetic Teammates. *11th Conference on Computer Generated Forces and Behavioral Representation*, Orlando, FL.
- Smallman, H. S., & St. John, M. (2005). Naive Realism: Misplaced Faith in Realistic Displays. *Ergonomics in Design*, 13(3), 6-13.
- Zachary, W. W., Ryder, J. M., & Hicinbothom, J. H. (1999). Building Cognitive Task Analyses and Models of a Decision-making Team in a Complex Real-time Environment. In J. M. Schraagen, S. F. Chipman, & V. J. Shalin (Eds.), *Cognitive Task Analysis* (pp. 365-383). Mahwah, NJ: Lawrence Erlbaum Press.

## Acknowledgments

The Office of Naval Research program of Training, Education, and Human Performance sponsored this research. Thanks to Alden Daley, Jia Huang, and Anthony Harrison for jACT-R and coding support.

## Author Biographies

**DR. JASON H. WONG** is a Human Factors Scientist with the Naval Undersea Warfare Center. He received his Ph.D. in 2009 from George Mason University and applies his research to projects that integrate cognitive theory and system design. He examines the human-computer interaction aspects of complex systems, creates improved submariner training methodologies, and develops cognitive models to simulate human performance.

**DR. SUSAN S. KIRSCHENBAUM** is an Engineering Psychologist in the Combat Systems Department of the Naval Undersea Warfare Center Division, Newport, Rhode Island and the Human Systems Integration (HSI) Lead for NUWCDIVNPT. She has continuing interests in expertise and in the effects of uncertainty and other information variables on decision making. Applications for her research are in the design of systems for decision support, human-computer interaction, and display design.

# Modeling Complex Social Behavior: A System Dynamics Approach

John A. Sokolowski, Ph.D.

Catherine M. Banks, Ph.D.

Virginia Modeling, Analysis and Simulation Center

Old Dominion University

1030 University Blvd.

Suffolk, VA 23435

757-686-6232, 757-686-6224

[jsokolow@odu.edu](mailto:jsokolow@odu.edu), [embanks@odu.edu](mailto:embanks@odu.edu)

**Keywords:** human behavior modeling, system dynamics, causal loop diagrams, stocks and flows, insurgency, IRA

**Abstract:** *System Dynamics (SD) will be used to facilitate a holistic representation of the British counter-insurgency (COIN) in Ireland with a view to events and relationships from a macro to micro perspective. SD modeling facilitates assessment of cause and effect factors, direct and indirect variables, and corresponding and correlative relationships of insurgency as a complex system. The model characterizes the relationships between and among inorganic and organic factors, i.e., events and human behavior / response. The purpose of the study is to better understand what served to unite the Irish insurgency, self rule, and what would have moderated the British COIN. Resultant model iterations allow for in depth analysis of case studies to explore hypothetical scenarios and what if questions.*

## 1. Introduction

The Systems Dynamics (SD) modeling paradigm is used for analyzing complex systems in many different areas. This modeling technique characterizes causal and correlative relationships between and among inorganic and organic factors, *i.e.*, events and human behavior / response. Specifically, SD facilitates a holistic representation of those events, and it can progress from the macro to micro perspective. SD also allows for sensitivity and statistical output analysis. Resultant model iterations allow for in depth analysis of case studies to explore hypothetical scenarios and *what if* questions.

The paper uses an SD to model complex social systems. First, is a discussion of SD as a modeling paradigm and the development of causal loops and stock and flows. SD will be used to explore the evolution and escalation of civil uprising (1916) and war (1919-1921) in Ireland specific to the relationship between and among the protagonists during this period. The significance of the research comes in the form of an analysis of the models and their outputs with comments on the model's function in explaining and understanding the case study.

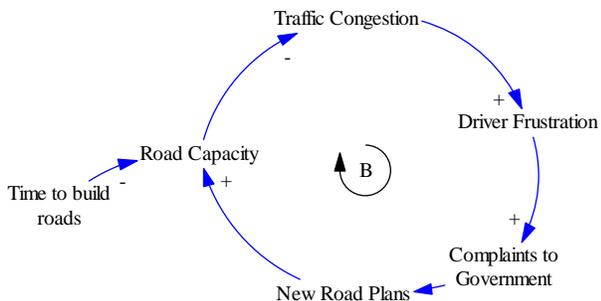
## 2. System Dynamics

SD is a methodology for modeling and subsequently studying complex systems such as those found in political or other social systems as entities that maintains their

existence through the mutual interaction of their parts (Forrester, 1991). The methodology consists of:

- Identifying a problem or system to be modeled
- Developing a hypothesis to explain the cause of the problem or the behavior of the system
- Developing a model to capture causes/ behaviors
- Validating the model to show that it reproduces the real-world behavior
- Devising possible solutions to the problem or modification of the behavior
- Testing these solutions in the model to show the possible outcome or impact of the proposed solution

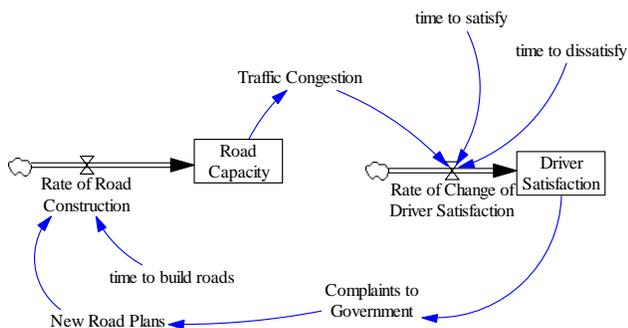
SD models are defined and represented by causal loop diagrams (that serve to identify factors and their relationships to explain how the system behaves) and stock and flow diagrams. Both are a critical in the modeling process as they serve as the foundation for capturing / explaining how the system behaves. Figure 2.1 is a causal loop diagram of factors that influence highway road construction.



**Figure 2.1 Causal loop diagram for road construction**

Each arrow in the diagram represents a causal link between two variables (e.g., as traffic congestion increases driver frustration increases; in turn causing more complaints to government; which leads to new road plans and more road capacity; with more road capacity traffic congestion decreases). The plus signs indicate the *effect variable* changes in the same direction as the *cause variable*. A negative sign (-) indicates an opposite change. The minus sign in the center of the loop shows this as a balancing relationship (the loop continues to feed on itself in a negative manner causing an exponential decrease in traffic congestion). Opposite this is a reinforcing loop indicated by a central plus sign. These behaviors describe an important concept in SD: *feedback loops*, which refer to situations where variable X affects variable Y and Y in turn affects X possibly through a chain of causes and effects. Studying these links independently to predict how the system will behave is not possible as only the study of the system with its multiple feedback loops connected to one another will lead to proper results.

Causal loop diagrams provide a conceptual model of how the system behaves. To turn the model into a functional simulation of the system requires translating the causal loop diagram into a stock and flow representation. *Stocks* are system variables whose values can be accumulated over time. *Flows* are the rate variables that govern the changes to the stock levels. Figure 2.2 is a stock and flow diagram for the traffic congestion example.



**Figure 2.2 Stock and flow diagram for traffic congestion**

In Figure 2 the rectangular boxes represent the stocks; here interest is placed on how road capacity and driver satisfaction change over time. The large arrows represent the flows with a valve symbol characterizing an adjustable rate of road construction and driver satisfaction. The other variables control these rates and thus the levels of each stock variable.

### 3. British Counter-Insurgency and the Easter Rising 1916

Civil uprising and insurgency are appropriate case studies to model as they are complex social systems that can be represented using SD. This study on Ireland looks at violence at the turn of the 20<sup>th</sup> century: the Easter Rising of 1916 and the Anglo-Irish War of 1919-1921. The following is a succinct discussion of these events.

In 1912 the British House of Commons passed the *Home Rule in Ireland Act*. If approved Home Rule could serve to split northern and southern Irish. The Protestant-Unionist-Loyalist-Irish of the northeast resisted this measure believing they would become a minority population among the Catholic-Nationalist-Gaelic-Irish of the south. To combat this measure of devolution **Unionists** organized as a group of militant rebels, the **Ulster Volunteers**, men who had no qualms about taking-up arms against the southern Irish or the King.

The Ulster Volunteers were countered in the south by **Nationalist** supporters of the Act who in 1913 organized, took arms, and called themselves the **Irish Volunteers**. The Act was never implemented due to the outbreak of World War I in 1914. Britain’s commitments in this war gave way to the call for Allied support among citizens of the empire to include all Ireland; and many Irish enlisted.

Members of the predominant **Irish Parliamentary Party** (IPP) hoped to use this gesture of war support as a bargaining chip in that when the war was over arguing the institution of Home Rule based on the show of Irish goodwill and support for the allies. Not all Irish agreed with this political tactic; in fact, many in the south opposed fighting the war in general, and more specifically fighting the war for Britain. Concurrently, another organization, more radical in its ideals and approach to Irish self-rule, began to prepare for a domestic revolt against British governance in Ireland. The **Irish Republican Brotherhood** (IRB), a secret society that came to be the most radical expression of nationalism, along with other Irish Volunteers planned an insurrection to establish an Irish Republic (Kostick, 1996).

In 1915 the Nationalists under the direction of charismatic leader Michael Collins p, with the help of Irish-American ties in New York, arranged for a shipment of arms from

Germany for the following spring. Both the Germans and the Volunteers knew that an Irish uprising aimed at the British could benefit both the German offensive in Europe and the uprising to lay claim to a Republic of Ireland.

An arms exchange was foiled as the German ship bringing arms was intercepted off the Kerry coast. The arms seizure signaled two things: the British were now aware of the covert activity taking place between the Irish and enemies of the realm and the Volunteers knew that without arms their planned Uprising was futile. Still, it was decided to go ahead with the Uprising as the Nationals sought to strike with what they had before the British had an opportunity to regroup and respond. Thus, a handful of Volunteers, reconciled to failure and willing to lose their lives, proceeded with the uprising. Some even believed a *blood sacrifice* was needed regardless of the odds against a victory (Walsh, 2009).

On 24 April 1916 approximately 150 Volunteers marched into the Dublin's General Post Office and ordered the staff to leave. The Volunteers took advantage of three things: Britain's overseas commitments, Ireland's tie to the Catholic Church and its condemnation of the war, and the threat of conscription.

The Easter Rising resulted in 1,351 wounded, 318 killed, 179 buildings destroyed, 3,430 men interned, and 92 death sentences (Kostick, 1996). The Rising lasted 6 days because it took that much time for British authorities to flood the city with troops. In Britain, the Rising was viewed as a stab in the back and it was believed that the Irish Volunteers were assisting the Germans. As such, British military policy and reprisals created many martyrs.

British reprisals in the form of execution and severe treatment of any associated with the Rising effectively changed the mood of Irish Nationalists, civilians and Volunteers, as they became more amenable to a radical means to an end (Auguseijn, 1996). In fact, the failed rebellion resulted in an emotional response by the Nationalist population and it accomplished precisely what the Volunteers sought, a revived civilian support for an Irish Republic (Hart, 2003).

### 3.1 Modeling the Easter Rising

The above narrative highlights the cause, ideologies, events, protagonists, and results of the incident. These can be dissected to construct the causal loops and stocks and flows.

The modeling effort begins with an analysis of the above narrative following the SD methodology outlined earlier in the paper. The task is to develop a model of the Anglo-Irish insurgency. In analyzing the above events one can see that the majority of the Irish citizens preferred self-

rule because of their dissatisfaction with British dominance. This dissatisfaction was caused by the imposition of British rule and British culture, which was different than the Gaelic-Catholic heritage that had been suppressed. This socio-cultural factor was a catalyst for ripening longstanding conditions causing a call to insurgency and the 1916 Rising. These are variables that can be used to begin the causal loop diagram. This segment of the loop is shown in Figure 3.1.

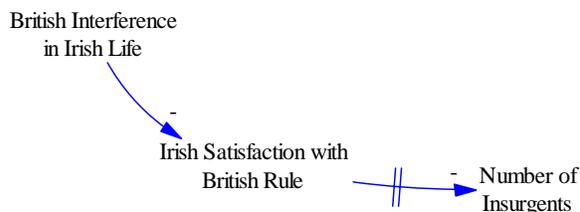


Figure 3.1 Initial causal loop diagram segment

Here, British interference in Irish life caused dissatisfaction with British rule leading to a growing number of insurgents.

As the number of insurgents grew so did the threat of violent incidents. This culminated with the takeover of the Post Office on Easter Monday 1916. The British were now under pressure to respond. British soldiers retaliated with many acts of killing and brutality, which only caused the perception of more interference by the British on Irish civilian life. This chain of events will allow additions to the causal loop diagram of Figure 3.1 and it completes a reinforcing loop that continues to feed the rise of the insurgency in Figure 3.2.

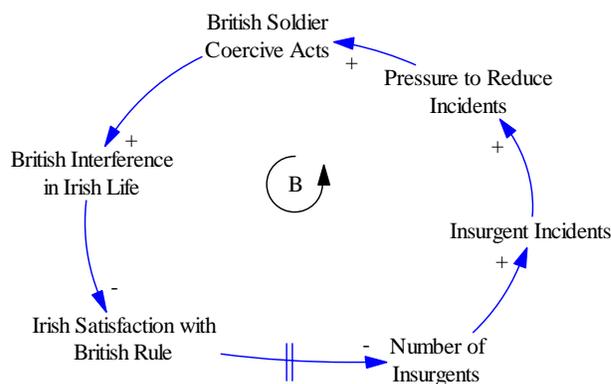


Figure 3.2 Insurgency creation loop

Because dissatisfaction with British rule does not instantaneously create new insurgents, a delay symbol (two parallel lines) was added to the segment connecting

*Irish Satisfaction with British Rule with Number of Insurgents* to indicate this delay. This model now sets the stage for the Anglo-Irish War.

#### 4. British Counter-Insurgency and the Anglo-Irish War 1919-1921

Modeling the next phase of the case study calls for a revised explanation of the ideology and intent of insurgency, the shifting populations among the protagonists, the structured tactics of the insurgents, and the formal counter-insurgency policy implemented. Importantly, since these are continuous events, the model must note the tipping points that result in desired or proscribed outcomes of the protagonists.

For purposes of modeling the Anglo-Irish War the designations will follow as such: the **post-1916 Nationalists** are outraged by British reprisals after the Easter Rising and are much more amenable to using nefarious acts in a tit-for-tat environment; some Volunteers changed their ideology and consider themselves **Republicans** seeking a free Ireland with no political stipulations. By 1920 these Irish Volunteers reorganize and become the **Irish Republican Army (IRA)** under Michael Collins (Fitzpatrick, 1998). Coupled with a Republican posture, the IRA sought autonomy over the entire state and escalated the rebellion via guerrilla tactics throughout the pre-war, from post-Rising 1916 through 1919, and then during the heated battle which began in 1920 until the truce of June 1921.

The executions that immediately followed the Easter Rising served to shift the support of many civilians to the Republican cause. All but one of the leaders of the Rising lost their lives. Irish politics now shifted: Parliamentary elections held in 1918 placed the IPP in low esteem and gave way to overwhelming wins for Sinn Fein (Augusteijn, 1996). This public support was the impetus for a provisional government, an Irish Parliament (Dail) which convened on 21 January 1919. It is with this self-proclaimed government that Collins reorganized the Volunteers into the IRA who swore allegiance to both the Republic and the Dail.

The IRA was perceived by some members of the Dail to possess a mandate for war against the British. As such, the IRA began a methodical campaign of guerrilla warfare by first targeting British soldiers. It benefited from public support in waging this campaign for the years between 1916 and 1918 were bloody; many Irish families suffered from British brutality. The most significant event during this period was the *anti-conscription campaign*.

By April 1918 conscription of Irishmen was enacted and it yielded much ill-will on the part of all Irish. As such,

*conscription was the catalyst to a united cause.* Many strikes and an anti-conscription rebellion resulted in the designation of 13 counties as Special Military Areas with large numbers of British troops deployed to keep the peace. At the close of 1918 this number exceeded 100,000 (Walsh, 2009). Sinn Fein membership increased from 66,000 in December 1917 to over 100,000 members in April 1918 (Hopkinson, 2002). Two things are significant regarding this crisis: 1) conscription was the catalyst to a united cause among civilians and Volunteers, and the Church contended that the Irish people had a right to resist; 2) the British were hard pressed by the various tactics (labor strikes and guerrilla operations) used in the resistance.

In March 1920 support was brought in to buffer RIC losses and the escalation of violence in the form of the **Black and Tans**. The British government placed 7,000 Tans under the administration of the Royal Irish Constabulary (RIC). The Tans conducted their affairs like a para-military force. A second quasi-military force was introduced that same summer, the **Police Auxiliary Cadets**. They, too, were to bolster the RIC, control the Tans, and avoid military conflict. They numbered 2,215 and were all too often just as bad as the Tans in their mistreatment of civilians; however, they focused on the IRA. By end of 1921 there were 17,000 RIC officers and 80,000 British troops in Ireland (Kostick, 1996). Collins estimated IRA membership during the war was 100,000 nominally, with 15,000 actively serving, and 3,000 who can be trusted to be drawn up at any time. The IRA benefited by the widespread civilian support throughout the counties primarily in civilian refusal to provide any information to the British.

As the war escalated two incidents took place that brought the conflict to levels beyond which the protagonists could tolerate: the 21 November 1920 killings that became infamously known as *Bloody Sunday* (the simultaneous assassination of fourteen officers, in eight Dublin locations). *Bloody Sunday* represented the microcosm of the whole conflict in respect to the role of intelligence, appalling violence, revenge, and propaganda. No set of incidents was so decisive in changing British attitudes of the Anglo-Irish War as corpses of assassinated British officers taken in succession through the streets of London to a massive funeral in Westminster Abbey (Hopkinson, 2002).

In the aftermath of *Bloody Sunday*, attacks on property of Sinn Fein sympathizers became a regular occurrence with thirty-three documented cases and the destruction of 191 houses (Hopkinson, 2002). IRA arrests abounded: 1,478 in January increased to 2,569 in March a final total of 4,454 in July. On 25 May 1921 the *Burning of the Custom House* in Dublin resulted in additional political

damage for the Parliament and continued guerilla attacks against British forces. The British military saw the worst casualties during the summer of 1921 with forty-eight killed (Hopkinson, 2002). Internally, confusion existed in the form of military authority over police authority and the relationship of Martial Law to Civil Law. By July 1921 Parliament called for an end to the Anglo-Irish stalemate via a truce.

#### 4.1 Modeling the Anglo-Irish War

The synopsis of the Anglo-Irish War depicts the continued effort by the Irish insurgents to affect their will on the British government and the various actions taken by the British to counter that effort. The British employed military responses to get control of the insurgency and to destroy it continuing their interference with Irish life in an effort to end insurgency. This portion can now be added to the causal loop diagram of Figure 3.2 to represent the insurgent suppression loop. This update is shown in Figure 4.1.

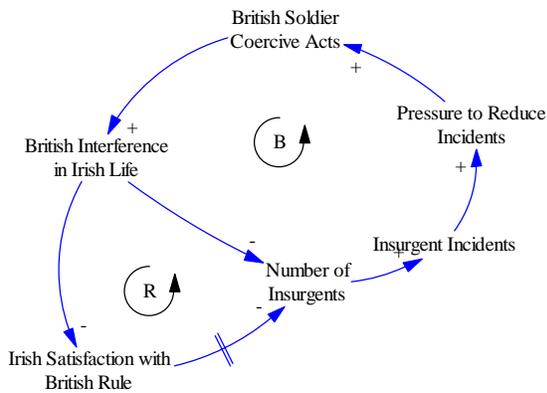


Figure 4.1 Addition of insurgent suppression loop

As a response to the continued pressure by the Irish insurgents to affect their will on Ireland, the British government felt pressure to regain control of the situation. This pressure came both from the *internal* violent acts that insurgents perpetuated and from *external* world opinion of the situation. As a result, Britain committed an increasing number of soldiers and other law enforcement personnel in an attempt to quell the violence and regain control of the situation. Figure 4.2 shows the addition of a British perception loop and its affect on British troop levels in Ireland.

Thus far the model has accounted for the major cause and effect relationships influencing the Irish insurgency for the period 1916 through 1921. With these relationships in place a stock and flow diagram can be constructed and a simulation developed to replicate this situation. With this

an investigation of *what-if* scenarios can be conducted to see what may have produced more favorable results.

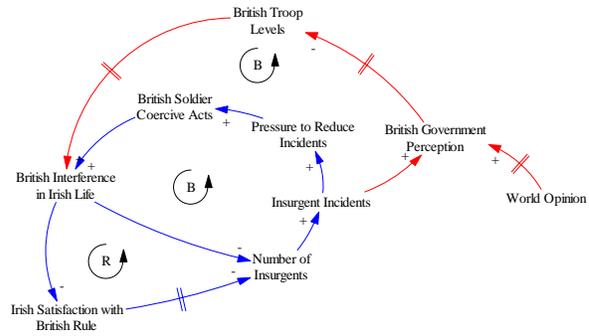


Figure 4.2 Addition of British perception loop

To begin the stock and flow diagram one must decide what variables to track from a quantitative standpoint. For the purpose of this example the *Number of Insurgents* and *Irish Satisfaction with British Rule* will be principal variables of interest. The *Number of Insurgents* is affected by an insurgent creation rate, an insurgent loss rate, and an insurgent retirement rate. *Irish Satisfaction with British Rule* is governed by the change in their satisfaction level. This initial stock and flow diagram is shown in Figure 4.3.

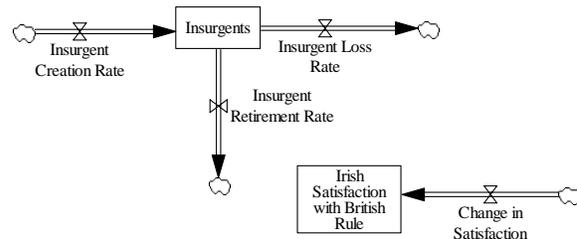
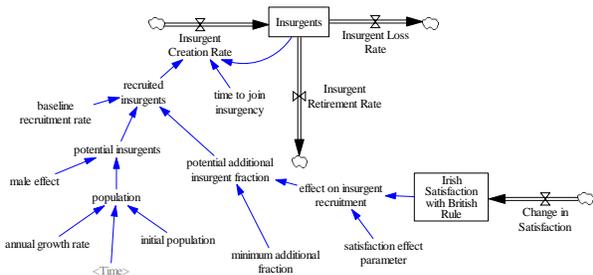


Figure 4.3 Initial stock and flow diagram

From the causal loop diagram of Figure 4.3 one can see that the rate at which insurgents are created is dependent upon Irish satisfaction level. However, it is also dependent upon the tendency of a small portion of the general Irish population to be drawn to an insurgency because of its inherent disposition. This would account for a core group of people who would be part of an insurgency no matter what the circumstances. The number in this group is dependent on the size of the population and the fraction of that population that would be predisposed to insurgency. This number would then be added to that portion of the population affected by British

rule thus providing the overall contribution to the insurgent creation rate. Figure 4.4 shows the addition of these factors to the initial stock and flow diagram.



**Figure 4.4 Stock and flow diagram showing affect on insurgent creation rate**

The population is dynamic, that is it grows over time at some annual growth rate from an initial base population. Thus, the growth must be accounted for in a dynamic model of this type. In the Irish insurgency case, the active insurgents were mostly male, so the population figure must be adjusted to account for this demographic.

Figure 4.4 provides a graphical representation of the variables controlling the insurgent creation rate. Underlying each of these variables is a numeric value or equation that implements the computation necessary to simulate the insurgency. For example the equation to compute *population* would be:

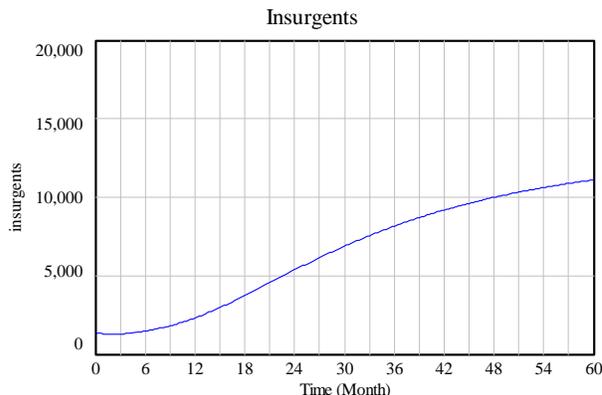
$$population = initial\ population * (1 + annual\ growth\ rate)^{time}$$

The other variables are computed in a similar manner.

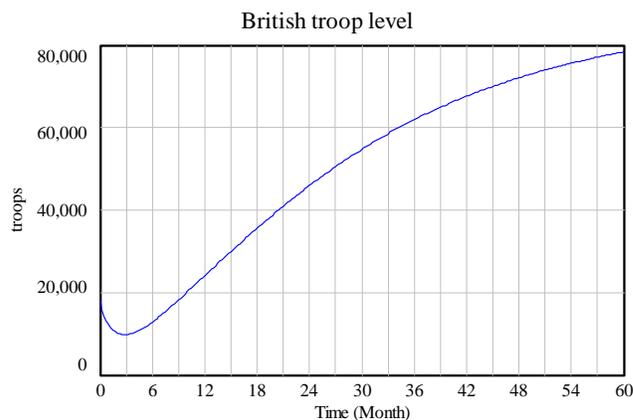
One can continue to build the entire stock and flow diagram in a manner as outlined above using the final causal loop diagram of Figure 4.2. The complete model is shown in Figure A.1 at the end of this article. A similar approach was taken by Anderson in his approach to capturing the dynamics of this insurgency (Anderson, 2006).

With a completed model, step 4 of the System Dynamics process requires validation so that its output is a proper reflection of the real-world system. (Several formal methods exist for validation, see Petty, 2009.) For this model, variables such as Irish population, insurgent levels, and British troop levels were compared to historical values. Model parameters were adjusted to achieve calibration against historical results. At this point the model is an accurate reflection of the Irish insurgency during the period of time under study. One can then run the simulation to obtain model output reflective of the insurgency behavior. Figures 4.5 and 4.6 provide graphs

of simulation results of insurgent level and British troop level.



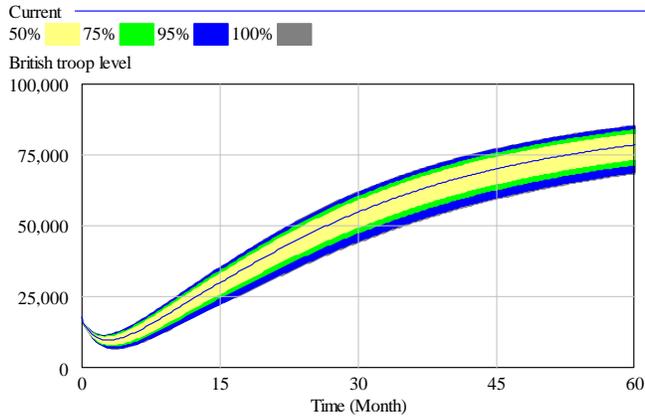
**Figure 4.5 Irish insurgent level 1916 – 1921**



**Figure 4.6 British Forces in Ireland 1916 – 1921**

As noted above some model parameters were adjusted to calibrate performance. It is important to know how sensitive the model output is to make changes in these parameters. Model results may be relatively insensitive to some parameter changes indicating that precise values for them may not be significantly important. Small changes in other parameters may cause a dramatic change in output, thus having more exact values for them becomes significant to model accuracy.

One model parameter that was manipulated to match British troop levels with historic values was *troop factor*. For the results in Figures 4.5 and 4.6 this value was set at 0.15. If this value was allowed to uniformly vary between 0.10 and 0.20 what impact would that have on troop level? Figure 4.7 shows the output of this sensitivity analysis based on 200 runs of the model.



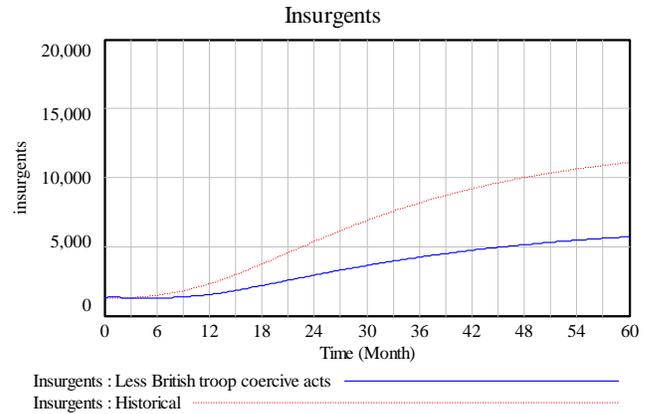
**Figure 4.7 Sensitivity analysis for troop factor**

The shaded areas of the graph represent confidence intervals for *British troop level* given the assumed random variation. This indicates that *British troop level* is relatively insensitive to small changes in this parameter.

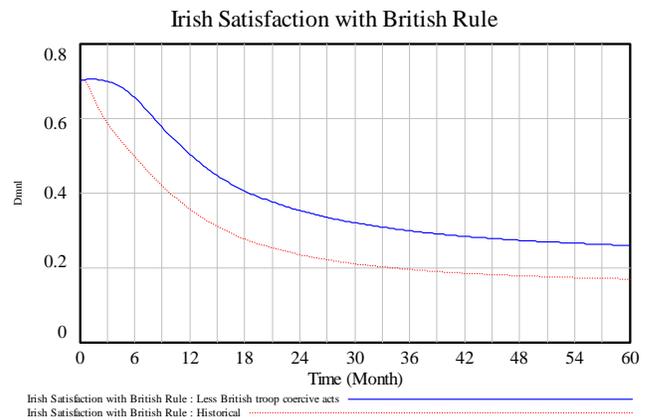
#### 4.2 What-if Analysis

Per steps 5 and 6 of the System Dynamics modeling process, simulation facilitates exploring different outcomes of a situation by changing particular model parameters. This capability is significant for social systems such as this one since these types of systems often times cannot be experimented on or readily manipulated as they can be in a simulation. Starting with a calibrated model that closely replicates historical results one can see how changes in policy would have possibly affected the outcome of the historical event.

The case study reflects brutal treatment by the British on Irish insurgents; this spilled over to the general Irish population. If the British would have adopted a less brutal approach what impact might that approach have had on the outcome? To investigate this scenario one can reduce the *max coercive acts* parameter, which governs the number of coercive acts committed by each British soldier on a monthly basis. The historical result was based on a value of 0.2 for this parameter. Suppose the British government implemented a policy that better controlled how the soldiers behaved and the number of acts was reduced to 0.1 acts per soldier. Figure 4.8 shows the affects of this policy. Figure 4.9 shows effects on Irish satisfaction with British rule.



**Figure 4.8 Effect of less coercive acts by British troops on insurgent level**



**Figure 4.9 Effect of less coercive acts by British troops on Irish satisfaction**

From a cause and effect analysis, fewer coercive acts resulted in less dissatisfaction with British rule, which resulted in a lower insurgent creation rate. It is also helpful to look at the ratio of British troops to insurgents for both the historical case and the hypothesized fewer coercive acts case. Figure 4.10 shows the British troop levels for both cases.



**Figure 4.10 British troop levels for historical and what-if cases**

At the end of the conflict this ratio was 7.05 troops to insurgents. With fewer coercive acts on the part of the British troops this ratio was computed to be 11.7. This change is due to the fact that fewer acts of troop harassment or brutality reduces distress in the Irish community, thus lowering support or need for the IRA. Therefore, there are fewer men who desire join the insurgency. With this higher troop to insurgent ratio one could postulate that a safer environment existed in Ireland thus making the Irish population more at ease and more benevolent towards the occupying British forces. As a benefit to Britain, fewer troops would be required to suppress insurgent activity lowering the cost of the counter-insurgency. This draws attention to the importance of troop behavior in these types of operations.

## 5. Conclusions

SD was used to explore the evolution and escalation of the insurgency events in Ireland via observing causal loop relationships to determine more precisely how the behavior / relationship of the British to the Irish incited discontent. The initial stock and flow data from the Easter Rising was included as part of a larger SD model of the Anglo-Irish War. The output of that model provided a computational explanation of insurgent activity incited by tit-for-tat nefarious acts on the part of all protagonists.

The analysis and *what if* discussion yielded commonsense conclusions; however, it also had the added benefit of being able to determine exactly how much of a draw down or decrease in British troops and/or modification in troop behavior is needed to change social behavior among Irish civilians as well as affect insurgency recruitment / sustainability. This is a very useful tool in social science research relative to human behavior modeling for it allows social science modelers to work toward *estimating*

*the odds of being correct* rather than *getting predictions right*. It also addresses the difficulty of representing social science knowledge analytically and the challenge of expressing approximate knowledge in understandable terms independent of any computer programming language, mathematical formalism, or disciplinary background (Davis, 2009).

## 6. References

- Anderson, E. G. *A Preliminary System Dynamics Model of Insurgency Management: The Anglo-Irish War of 1916-21 as a Case Study*. In Proceedings of the 2006 International System Dynamics Conference, Nijmegen, The Netherlands.
- Augusteijn, J. *From Public Defiance to Guerrilla Warfare: The Experience of Ordinary Volunteers in the Irish War of Independence 1916-1921*. Dublin: Irish Academic Press, 1996.
- Davis, PK. *Representing Social-Science Knowledge Analytically in Social Science for Counterterrorism: Putting the Pieces Together*, Edited by Paul K. Davis, Kim Cragin. The RAND Corporation, 2009.
- Fitzpatrick, D. *The Two Irelands 1912-1939*. Oxford: Oxford University Press, 1998.
- Forrester, JW. *System Dynamics and the Lessons of 35 Years*. MIT Report D-4224-4, 1991.
- Hart, P. *The IRA at War 1916-1923*. Oxford: Oxford University Press, 2003.
- Hopkinson, M. *The Irish War of Independence*. Montreal: McGill-Queen's University Press, 2002.
- Kostick, C. *Revolution in Ireland: Popular Militancy 1917-1923*. London: Pluto Press, 1996.
- Petty, M. *Verification and Validation in Principles of Modeling and Simulation: A Multidisciplinary Approach*, Sokolowski, J. A., Banks, C. M. (eds.), Hoboken: John Wiley and Sons, Inc., 2009.
- Walsh, P. *The Irish Civil War: A Military Study of the Conventional Phase*. Paper delivered to NYMAS, CUNY Graduate Center, N.Y.

## Author Biographies

**JOHN SOKOLOWSKI** is a research professor and Director of Research at Old Dominion University's Virginia Modeling, Analysis & Simulation Center. His research interests include human behavior modeling, multi-agent system simulation, and computational representation of social systems.

**CATHERINE BANKS** is an Assistant Professor of Research at the Virginia Modeling, Analysis, and Simulation Center in Suffolk, Virginia. Her current research focuses on modeling states and their varied

histories of revolution and insurgency, political economy  
and state volatility.

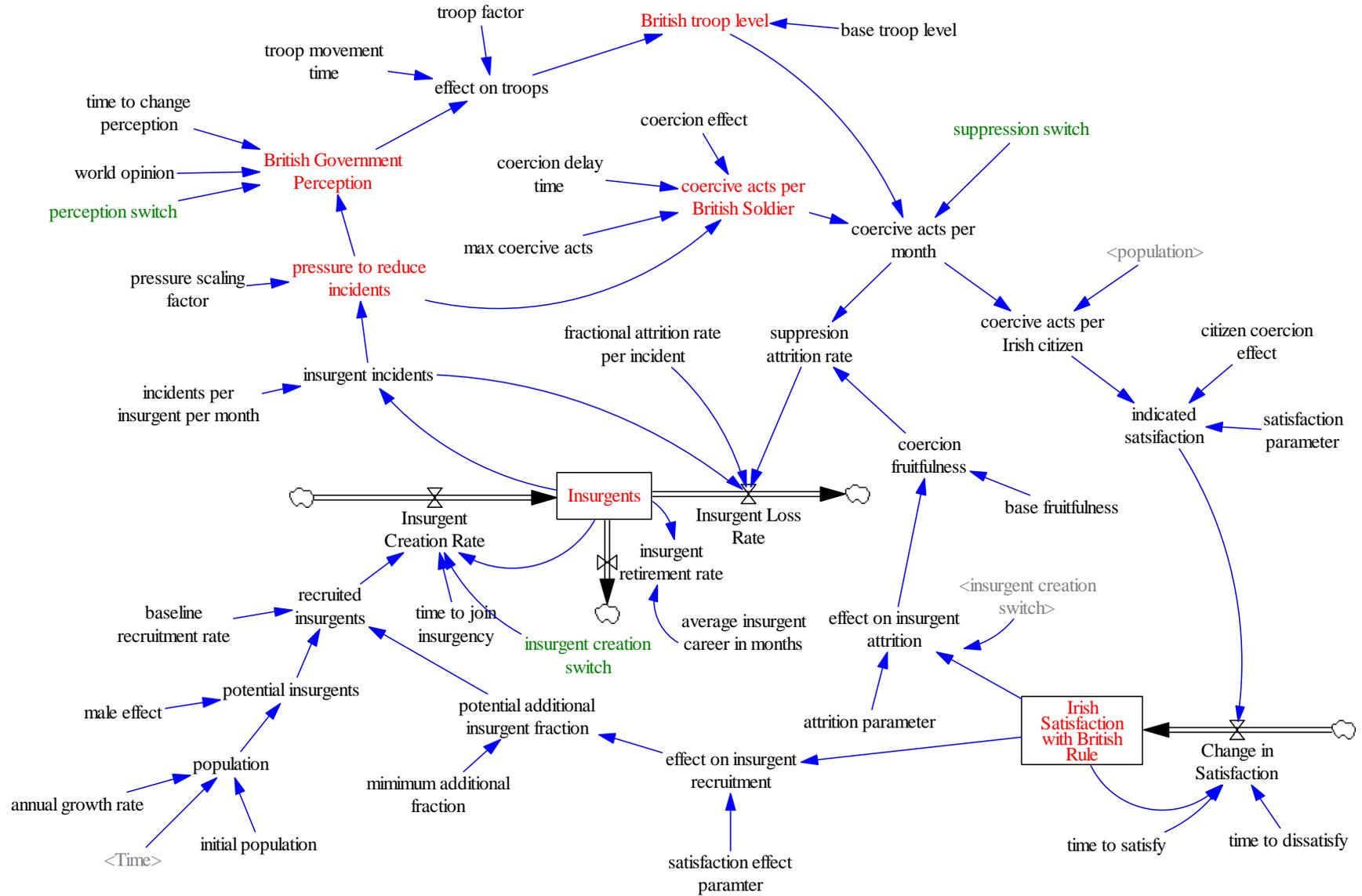


Figure A.1 Complete Irish insurgency stock and flow diagram

## Plan Ahead: Pricing ITS Learner Models

Jeremiah T. Folsom-Kovarik

Sae Schatz

Denise Nicholson

University of Central Florida

Institute for Simulation and Training

3100 Technology Parkway

Orlando, FL 32826

jfolsomk@ist.ucf.edu, sschatz@ist.ucf.edu, dnichols@ist.ucf.edu

Keywords:

intelligent tutoring systems, learner models, development costs

**ABSTRACT:** *Intelligent tutoring systems (ITSs) are highly adapted to individual learners, and therefore their learner models are central to their operation and account for a large fraction of their development costs. Different learner model architectures may have different development costs, but those costs are not widely reported in the literature. This paper presents individual reports from an anonymous questionnaire sent to ITS professionals in September 2009. The respondents estimated the development costs of recent ITSs and their associated learner models. The resulting data aligns with and amplifies published accounts, as well as contributing new cost information about model types that have not previously appeared in the literature.*

### 1. Introduction

In an intelligent tutoring system (ITS), personalized treatment makes teaching and training more effective. ITSs adapt their interactions to individual learners by estimating users' traits, states, or misconceptions in a learner model. Since adaptation and personalization play defining roles in ITSs, the learner model is key to every new system. Practitioners will benefit from an open discussion of what to expect when developing different types of models.

Following Snow and Swanson (1992), this paper divides personalization in an ITS into macroadaptation and microadaptation. *Macroadaptation* describes changes the ITS makes prior to a learning episode based on pre-task measures or historical data, which can include problem selection or ordering. *Microadaptation* describes changes during a learning episode based on ongoing performance or behavioral assessment, which can include giving the learner custom hints and feedback.

Several competing model architectures support ITS adaptation, and published accounts reviewed in this paper suggest that different model types might have different impacts on cost. To the extent that model types support macroadaptation and microadaptation, they can all be appropriate choices for an ITS. One factor that could help practitioners choose a learner model is its development cost. Controlling the cost of a model can make more resources available for other development tasks or help maximize the project's return on investment.

This paper compares anecdotes about the cost to develop different learner model architectures, as one important considera-

tion among many in designing a new ITS. In the rest of this section, the model types being considered are introduced and information about their development in the published literature is reviewed. The remainder of this paper describes a questionnaire of the ITS community that solicited additional anecdotes focused on development costs. Section 2 explains the questionnaire method, section 3 describes the results, and section 4 provides some interpretation of these results.

#### 1.1 Model architectures

This section introduces six learner model architectures common in the ITS field. These architectures form the separate categories described in this paper.

An *overlay* metaphor describes the earliest and simplest learner models, such as those in Scholar (Carbonell, 1970), PLATO West (Burton & Brown, 1976), and Wusor II (Carr, 1977). An overlay model is conceptually like a checklist of all the knowledge and skills an ITS must impart. The ITS records learners' competencies as a subset, or overlay, of the ideal checklist. It gives a novice no checkmarks and a perfect expert a checkmark for each item on the list. Successful or unsuccessful performance in the tutor grows or shrinks the overlay. *Differential* models are a subset of overlay models that apply the "checklist" approach but do not require learners to master all expert knowledge to satisfy learning requirements.

Although overlay models can encode novices' lack of expert skills or knowledge, novices do not simply lack knowledge. Often, they also possess incorrect knowledge that an ITS should specifically identify and correct. *Buggy* or *perturbation*

learner models include information about possible misconceptions or *bugs*. Model builders can either generate possible misconceptions automatically by systematically breaking rules in a cognitive theory, (e.g., Brown & VanLehn, 1980; Burton, 1982), or can let subject-matter experts list likely misconceptions (e.g., Johnson, 1990).

An extension of early buggy-model ITSs is the *cognitive tutor*. Like in buggy models, cognitive tutors model misconceptions as breaks in a cognitive model, but they also specify an algorithm, *model tracing*, for matching observed mistakes to the underlying misconceptions. The learner model in a cognitive tutor is a set of *production rules*, grounded in cognitive theory, that mirror the mental steps the learner makes while working—for example, selecting a theorem to apply in a geometry proof (Anderson, 1993). Model tracing tries different productions together to see which could have produced the learner's observed behavior. The granularity of the production rules supports detailed microadaptation but does not readily enable macroadaptation. To compensate for this, modern cognitive tutors typically also use a second learner model for macroadaptation, such as an overlay (Corbett & Bhatnagar, 1997) or Bayesian model (Baker, Corbett, & Aleven, 2008).

Model-tracing tutors revolve around a detailed cognitive model describing how learners work and learn. One way of building an ITS with similar performance, but with less cognitive science, is with *example tracing* (Koedinger, Aleven, Heffernan, McLaren, & Hockenberry, 2004). Instead of general cognitive rules that apply to any problem, example-tracing tutors let builders write example solutions for each problem. Specific errors can still trigger specific remediations, but only when examples of those errors are programmed ahead of time.

Another way to avoid reconstructing hidden mental events is to use a *constraint-based* model (Ohlsson, 1992). These models are collections of constraints, i.e., boundary conditions that describe incorrect problem states. Tutors based on constraints allow learners to interact freely with the system until something happens that requires correction. Uniquely among the learner models, constraint-based models assume that behaviors they do not recognize are correct—not wrong—and that learners are “innocent until proven guilty” (Mitrovic, Koedinger, & Martin, 2003, p. 320). Like production-rule models, constraint-based models can be paired with an overlay model to control macroadaptation, for example by inferring unmastered skills from constraint violations (Martin & Mitrovic, 2002).

Finally, *classifiers* can also play the role of a student model. A classifier as a learner model typically sorts individual learners into groups. These groupings can be similar to assessments from overlay or buggy models, but unlike typical overlay or buggy models, classifiers use more principled methods of interpreting observations as evidence, and potentially can update many model estimates with each assessment. Classifiers that have been used as learner models include *Bayesian networks*

(e.g., Arroyo, Woolf, & Beal, 2006; Conati & Zhou, 2004; Luckin & du Boulay, 1999), *finite-state automata* (e.g., Stottler, Fu, Ramachandran, & Vinkavich, 2001), *decision trees* (e.g., Cha et al., 2006; McQuiggan, Mott, & Lester, 2008), *neural networks* (e.g., Castellano, Mastronardi, Di Giuseppe, & Dicensi, 2007), and ensemble methods (e.g., Hatzilygeroudis & Prentzas, 2004; Lee, 2007). Although there are many different kinds of classifiers, in at least some practical situations they are approximately equivalent in their performance (McQuiggan et al., 2008; Walonoski & Heffernan, 2006).

## 1.2 Published accounts

Although development cost is an important consideration for practitioners making an ITS operational, it is only irregularly reported in the academic literature. This section gathers reports that authors volunteered in published academic sources. The common metric for reporting ITS costs in these sources is the ratio of ITS development time in person-hours to user interaction time in hours per individual. Reporting costs in a ratio format makes figures more comparable across different ITSs that may undertake more or less complex tutoring tasks.

Cognitive tutors and model-tracing algorithms have been the subject of both significant research and also operationalization (Koedinger, Anderson, Hadley, & Mark, 1997). Initial publications on the first cognitive tutors reported cost ratios between 1000:1 and 100:1 to build an entire ITS (Anderson, 1993). As another example within this range, an algebra tutor had a 200:1 ratio for the whole system (Koedinger et al., 2004). Building cognitive tutors in the future may be easier because specialized authoring tools are in development. A preliminary study of a new authoring tool showed a 40% reduction in effort that could make future cognitive tutors more cost-effective (Aleven, McLaren, Sewall, & Koedinger, 2006).

Example tracing models were created as a response to the high development cost of using the model-tracing approach, and a preliminary study showed that cost ratios were only 23:1 for an entire example-tracing ITS (Koedinger et al., 2004). Furthermore, example tracing is more straightforward than model tracing for nonprogrammers, and novices could use it to build a whole ITS with a cost ratio of 40:1 (Razzaq et al., 2008).

Constraint-based tutors were also designed to require less development effort than cognitive tutors, because the tutor can still give meaningful results without a complete set of constraints or in domains for which it is difficult to write exhaustive production rules (Mitrovic et al., 2003). The first ITS based on constraints had a 220:1 cost ratio for building the learner model only (Mitrovic & Ohlsson, 1999). Since then, new authoring systems have let novices create a simple tutor or reimplement an existing ITS about as quickly as experts had previously (Martin, Mitrovic, & Suraweera, 2008; Mitrovic et al., 2006; Suraweera, Mitrovic, & Martin, 2007).

Constraints and production rules have also been directly compared on the cost of developing the same learner model. In one study, an expert in model-tracing built a cognitive tutor to teach the same domain as an existing constraint-based tutor. The two tutors were approximately equal in complexity and presumably in development cost (Mitrovic et al., 2003). In another study, a single team built new constraint-based and model-tracing tutors to teach the same task. They found that the constraint-based tutor took four times as long to implement because of extra effort to learn the more complex architecture. Excluding their learning time, the team found that model tracing took slightly more time to implement, but the two architectures nonetheless required approximately the same effort (Kodaganallur, Weitz, & Rosenthal, 2005).

While precise cost figures have not typically been published for overlay models, buggy models, or classifiers, some studies have explored these development experiences. For example, studies of buggy models suggest that generating a complete misconception list can be a long or even unending task because different misconceptions are prevalent in different populations. (Payne & Squibb, 1990; VanLehn, 1982). Theory also warns about potential high costs of Bayesian models. Initializing Bayesian networks can require precise expert estimates or large amounts of empirical data, although it is possible to start using the model with initial settings and refine it during use (Conati & Maclaren, 2005). The design effort grows quickly with complexity, so that a Bayesian network with just 40 inputs would be difficult to initialize, and its estimates would be highly suspect (Ott, Imoto, & Miyano, 2004).

The research community has produced limited reports on development time, including a comparison of the same team developing two equivalent model types and a comparison of experts in their respective architectures developing equivalent models. However, publication of development cost estimates remains sparse, with only a few estimates published for some model types and none at all for other widely used architectures. The rest of this paper helps to address these gaps in the published knowledge.

## 2. Method

### 2.1 Questionnaire

To increase knowledge of learner model development costs, an anonymous questionnaire was emailed to ITS community members in September of 2009. Because of space restrictions, only the parts of the questionnaire that produced data used in this paper are reproduced in the appendix. However, a full version of the questionnaire is available in (Folsom-Kovarik, Schatz, & Nicholson, in preparation).

The questions answered in this paper describe participants' experiences on the last ITS each person worked on that is ready or almost ready to interact with learners. This makes

practitioners' memories more recent and also helps ensure the data presented reflect current modeling and authoring technology. Participants were asked to estimate the development effort in person-hours for the ITS as a whole and also for the learner model or models specifically. To calibrate the complexity of the ITS being described, participants were also asked the amount of time one learner would be expected to engage with the ITS. All questions were optional.

Participants were asked thirty additional questions relating to previous experiences with building specific model types. Because of low response rates and space limitations those questions are not discussed in this paper.

### 2.2 Participants

The questionnaire was emailed to all 63 attendees of the 2009 Army Research Institute Workshop on Adaptive Training Technologies and to an additional 88 authors of publications cited in a survey of the ITS field (Folsom-Kovarik et al., in preparation) who did not attend the workshop. Eleven participants responded anonymously. The responses give a varied anecdotal view of the development costs for different student models in the current state of the field.

Participants in the study came from diverse backgrounds. Of the eleven participants, five people were academics, three worked in industry, and two worked in government or military positions. Three people had worked on one or two ITSs, three had worked on three to five ITSs, and four had worked on six ITSs or more. Three people had worked on ITSs for three to six years and seven had worked on ITSs for seven years or more. One participant did not share any demographic data.

## 3. Results

### 3.1 Model architectures in current ITS development

Out of eleven participants, nine reported that the ITS he or she worked on most recently used a single learner model. Two reported using two learner models, and none reported using more than two constructs. The models participants used included representatives from five of the six architecture categories described in this paper. Example tracing was not represented. Note that the mention of a model type in this section *does* indicate current ITS research or development is using that architecture, but failure to mention a type does *not* indicate whether that architecture is in common use or not.

### 3.2 Development cost ratios

This section relates individual experiences with building different model types. The data are recent, since they represent participants' descriptions of the last project they completed. As elsewhere in this paper, cost is reported as a ratio reflecting the number of development person-hours spent to create one hour of individual instruction.

**Table 3.2.1:** Individual reports of *macroadaptation* models' development cost in relation to ITS teaching time.

Model Architecture	Cost Ratio
Overlay	24:1
Decision trees (Classifier)	30:1
Knowledge tracing	48:1
Model tracing	100:1
Overlay	667:1
Knowledge tracing	1375:1

Table 3.2.1 describes the cost of models supporting macroadaptation from six respondents who estimated both development time and instruction time. Table 3.2.2 gives the same information for microadaptation, as described by seven respondents. All participants in the study stated that they used microadaptation in their ITSs, and all but one used macroadaptation as well. Although macroadaptation costs were more variable, a two-tailed T-test did not find support in these responses for a significant difference between the cost of developing macroadaptation versus microadaptation.

Certain model types were represented more than once in the responses. Although these responses may come from different participants describing the same project, the likelihood is low because there was no instance when the details from one participant substantially matched another participant's response.

**Table 3.2.2:** Individual reports of *microadaptation* models' development cost in relation to ITS teaching time.

Model Architecture	Cost Ratio
Overlay	24:1
Knowledge tracing	48:1
Behavior transition networks (Classifier)	50:1
Differential model (Overlay)	100:1
Constraint-based model	100:1
Buggy model	133:1
Knowledge tracing	450:1

Table 3.2.3 shows seven responses relating the cost of building an entire ITS, not just the learner model, to the hours of instruction provided. Each ITS is described by the model types the respondents used. The next section relates the cost of model development to the cost of system development.

**Table 3.2.3:** Individual reports of an *entire ITS's* development cost in relation to its teaching time, showing models used.

Model Architecture	Cost Ratio
Classifiers	250:1
Constraint-based model	333:1
Overlays	400:1
Knowledge tracing *	500:1
Model tracing and differential models	600:1
Knowledge tracing *	2000:1
Overlay and buggy models	5333:1

In Table 3.2.3, two respondents (marked with an asterisk) stated that they used knowledge tracing but did not affirm using model tracing. Since knowledge tracing refers to a way of using a second learner model in conjunction with a cognitive tutor, it may be that these ITSs also used model tracing.

### 3.3 Learner model cost as a percentage of ITS cost

Eight participants reported development cost estimates for both a tutoring system as a whole and its learner model. Costs in this section are absolute values, so some new responses can be used that did not appear in the previous section because they lacked instruction time estimates. Taken as an aggregate, these responses show how much of an ITS's cost goes toward building its learner model.

Responses indicated that, in general, a learner model accounts for about a third of the cost of an ITS, with a mean reported ratio of 33%, a median of 31%, and a standard deviation of 28 percentage points. The responses were overall consistent, so that dropping one low and one high outlier brought the standard deviation to 9 percentage points. The low outlier used an overlay model, and the high outlier used knowledge tracing.

## 4. Discussion

### 4.1 Interpretations

Although the responses gathered in this survey provide valuable anecdotal insights, there are too few responses to apply a detailed statistical analysis. However, individual responses suggest some interesting trends. One interesting fact is the high variability of cost estimates when more than one participant described the same model type. The large differences might be attributable to modeling tasks related to the architecture, such as learning to use a new model type, or unrelated, such as spending more time eliciting knowledge from subject-matter experts. Unfortunately, this study cannot determine how much of the variation in cost reports was attributable to the different model types.

Although combining the conflicting cost reports as an average might give a better view of the effort a model requires under many different circumstances, it would be misleading to aggregate such sparse data. Instead, it is more useful to use the most favorable estimate for each model type as a best-case scenario. Since there is no upper limit on the development effort anyone can expend on any model, examining the lowest or best case instead helps show whether it is at least possible to spend low amounts of time.

The best-case cost estimates for building a learner model alone cluster into two groups. One group of models has a cost ratio of 50:1 or lower, while the other group has a cost ratio between 100:1 and 133:1. The very high cost estimates in the results are not best-case scenarios because other participants reported lower estimates for the same model categories. The

model types in the low-cost group include overlays, classifiers, and knowledge-tracing models (which are typically implemented with a Bayesian or overlay model). The model types that cost more include buggy models, constraint-based models, and the production-rule models in cognitive tutors. Considering best-case scenarios only, these model types cost between two and 5.5 times as much as the low-cost models.

**Table 4.1.1:** *Best-case scenario* model costs, as determined by finding the lowest cost ratio reported for each model category.

Model Architecture	Cost Ratio
Overlays	24:1
Classifiers	30:1
Knowledge tracing	48:1
Constraint-based model	100:1
Production-rule model (model tracing)	100:1
Buggy model	133:1

Estimating the cost of building a whole ITS, not just a learner model, makes values in this study comparable to published estimates of this figure. The costs of model-tracing tutors and constraint-based tutors reported in this study are approximately equal to figures published in the academic literature.

Using the reasoning discussed above in this section, the two whole-ITS cost ratios over 1000:1 in Table 3.2.3 do not represent best-case scenarios because there are lower cost estimates with the same model types. The remaining values in that table are all on the same order of magnitude and even the highest estimate, 600:1 for a model tracing cognitive tutor, was only 2.4 times as high as the lowest estimate. Although these estimates are quite close to each other, the responses do suggest that changing the learner model might halve or double the development time of the entire ITS.

The different responses in Table 3.2.3 also suggest an ordering of system development costs by learner model type. Using classifiers as learner models may lead to the fastest ITS development. This confirms intuitions that classifiers, as off-the-shelf tools, are easy to use and do not require publications about their development effort.

Surprisingly, tutoring systems using overlay models fell in the middle of the pack at best, despite the low cost of overlay models compared to other types in this study. However, this unexpected result may be due to the cost of knowledge elicitation on the two projects in question, rather than any costs directly associated with overlay models.

Considering whole-system costs, constraint-based systems are somewhat easier to develop than cognitive tutors, a conclusion which concurs with published anecdotes. The best-case costs of building a tutor with model-tracing or knowledge-tracing are higher than that of a constraint-based tutor, despite the fact that considering the learner model alone, constraints cost the same or more (see Table 4.1.1). A possible factor that might

contribute to this difference is that constraint-based systems can work with less precise learner models, which might lead to less effort in creating specific hints and remediations for many different errors (Mitrovic et al., 2003). Cognitive tutors, with their model tracing and knowledge tracing algorithms, took the most effort of any ITS to build, confirming the intuition that led to constraint-based modeling and example tracing.

#### 4.2 Limitations

Limitations of this study include a small population size, possible selection bias, and possible lack of consideration in forming estimates. Although the number of responses reported in this paper is comparable to the number of related publications from the academic community, that number does not yet reach levels that would allow a detailed statistical analysis. Furthermore, participants were not invited randomly, and invitees with certain characteristics may have been more or less likely to respond. Finally, ITS researchers who include development costs in publications can support their figures with careful records, while respondents in this study had to estimate costs after the fact. Because of these limitations, responses in this paper should be viewed as anecdotes rather than predictions of future performance. Although this study presents anecdotal evidence, it is still valuable input into choosing a learner model architecture if the limitations are understood.

#### 5. Conclusion

This paper has presented anecdotal evidence concerning the development cost of learner models in ITSs. ITSs focus on personalization for every user, and this study showed that their learner models often account for about one third of their development cost. Different learner models have different costs to develop. In this study, eleven ITS practitioners from industry, academia, and military organizations shared their valuable experiences to provide anecdotal evidence about those costs.

The anecdotes in this paper, which align with the few published experiences previously available, suggest that certain learner models can be easier to build than others. Overlay models and classifiers used as learner models have the lowest development costs. With current authoring tools, constraint-based learner models are approximately as expensive to build as production-rule models. Buggy learner models are the most expensive to develop. The differences in model costs are also reflected in smaller but still noticeable differences in the cost of the entire ITS.

This study only addresses learner model development costs. It may be the case that more expensive learner models produce such good cognitive fidelity (Neches, Langley, & Klahr, 1987), effects on learning outcomes, or other benefits that they justify their cost or more. The authors of this paper are currently in the process of exploring this new data on model cost in relation to ITS benefits, that is, return on investment.

## 6. References

- Aleven, V., McLaren, B. M., Sewall, J., & Koedinger, K. R. (2006). *The cognitive tutor authoring tools (CTAT): Preliminary evaluation of efficiency gains*. Paper presented at the 8th International Conference on Intelligent Tutoring Systems.
- Anderson, J. R. (1993). *Rules of the Mind*. Hillsdale, NJ: Lawrence Erlbaum Associates.
- Arroyo, I., Woolf, B. P., & Beal, C. R. (2006). Addressing cognitive differences and gender during problem solving. *Technology, Instruction, Cognition & Learning*, 4, 31–63.
- Baker, R. S. J., Corbett, A. T., & Aleven, V. (2008). *More Accurate Student Modeling Through Contextual Estimation of Slip and Guess Probabilities in Bayesian Knowledge Tracing*. Paper presented at the 9th International Conference on Intelligent Tutoring Systems.
- Brown, J. S., & VanLehn, K. (1980). Repair theory: A generative theory of bugs in procedural skills. *Cognitive Science: A Multidisciplinary Journal*, 4(4), 379–426.
- Burton, R. R. (1982). Diagnosing bugs in a simple procedural skill. In D. Sleeman & J. S. Brown (Eds.), *Intelligent Tutoring Systems* (pp. 157–183). London: Academic Press.
- Burton, R. R., & Brown, J. S. (1976). *A tutoring and student modelling paradigm for gaming environments*. Paper presented at the SIGCSE-SIGCUE joint symposium on Computer science education.
- Carbonell, J. (1970). AI in CAI: An artificial-intelligence approach to computer-assisted instruction. *IEEE Transactions on Man-Machine Systems*, 11(4), 190–202.
- Carr, B. P. B. (1977). *Wusor II: A Computer Aided Instruction Program with Student Modelling Capabilities*. Massachusetts Institute of Technology, Cambridge, MA.
- Castellano, M., Mastronardi, G., Di Giuseppe, G., & Dicensi, V. (2007). *Neural Techniques to Improve the Formative Evaluation Procedure in Intelligent Tutoring Systems*. Paper presented at the IEEE International Conference on Computational Intelligence for Measurement Systems and Applications.
- Cha, H. J., Kim, Y. S., Park, S. H., Yoon, T. B., Jung, Y. M., & Lee, J. (2006). *Learning styles diagnosis based on user interface behaviors for the customization of learning interfaces in an intelligent tutoring system*. Paper presented at the 8th International Conference on Intelligent Tutoring Systems.
- Conati, C., & Maclaren, H. (2005). *Data-driven refinement of a probabilistic model of user affect*. Paper presented at the 10th International Conference on User Modeling.
- Conati, C., & Zhou, X. (2004). *Building and evaluating an intelligent pedagogical agent to improve the effectiveness of an educational game*. Paper presented at the 9th International Conference on Intelligent User Interfaces.
- Corbett, A. T., & Bhatnagar, A. (1997). *Student Modeling in the ACT Programming Tutor: Adjusting a Procedural Learning Model With Declarative Knowledge*. Paper presented at the 6th International Conference on User Modeling. from um.org
- Folsom-Kovarik, J. T., Schatz, S., & Nicholson, D. (in preparation). *Return on investment: A practical review of ITS student modeling techniques*. Orlando, FL: University of Central Florida.
- Hatzilygeroudis, I., & Prentzas, J. (2004). Using a hybrid rule-based approach in developing an intelligent tutoring system with knowledge acquisition and update capabilities. *Expert Systems with Applications*, 26(4), 477–492.
- Johnson, W. L. (1990). Understanding and debugging novice programs. *Artificial Intelligence*, 42, 51–97.
- Kodaganallur, V., Weitz, R. R., & Rosenthal, D. (2005). A comparison of model-tracing and constraint-based intelligent tutoring paradigms. *International Journal of Artificial Intelligence in Education*, 15(2), 117–144.
- Koedinger, K. R., Aleven, V., Heffernan, N., McLaren, B., & Hockenberry, M. (2004). *Opening the door to non-programmers: authoring intelligent tutor behavior by demonstration*. Paper presented at the 7th Annual Intelligent Tutoring Systems Conference.
- Koedinger, K. R., Anderson, J. R., Hadley, W. H., & Mark, M. A. (1997). Intelligent tutoring goes to school in the big city. *International Journal of Artificial Intelligence in Education*, 8(1), 30–47.
- Lee, C. S. (2007). Diagnostic, predictive and compositional modeling with data mining in integrated learning environments. *Computers & Education*, 49(3), 562–580.
- Luckin, R., & du Boulay, B. (1999). Ecolab: The development and evaluation of a Vygotskian design framework. *International Journal of Artificial Intelligence in Education*, 10(2), 198–220.
- Martin, B., & Mitrovic, A. (2002). *Automatic problem generation in constraint-based tutors*. Paper presented at the 6th International Conference on Intelligent Tutoring Systems.
- Martin, B., Mitrovic, A., & Suraweera, P. (2008). ITS domain modelling with ontology. *Journal of Universal Computer Science*, 14(17), 2758–2776.
- McQuiggan, S. W., Mott, B. W., & Lester, J. C. (2008). Modeling self-efficacy in intelligent tutoring systems: An inductive approach. *User Modeling and User-Adapted Interaction*, 18(1), 81–123.
- Mitrovic, A., Koedinger, K. R., & Martin, B. (2003). *A comparative analysis of cognitive tutoring and constraint-based modeling*. Paper presented at the 9th International Conference on User Modeling.
- Mitrovic, A., & Ohlsson, S. (1999). Evaluation of a constraint-based tutor for a database language. *International Journal of Artificial Intelligence in Education*, 10(3–4), 238–256.
- Mitrovic, A., Suraweera, P., Martin, B., Zakharov, K., Milik, N., & Holland, J. (2006). *Authoring constraint-based tutors in ASPIRE*. Paper presented at the 8th International Conference on Intelligent Tutoring Systems.
- Neches, R., Langley, P., & Klahr, D. (1987). Learning, development, and production systems. In D. Klahr, P. Langley & R. Neches (Eds.), *Production system models of learn-*

ing and development (pp. 1–53). Cambridge, MA: MIT Press.

- Ohlsson, S. (1992). Constraint-based student modeling. *Journal of Artificial Intelligence and Education*, 3(4), 429–447.
- Ott, S., Imoto, S., & Miyano, S. (2004). *Finding optimal models for small gene networks*. Paper presented at the Pacific Symposium on Biocomputing.
- Payne, S. J., & Squibb, H. R. (1990). Algebra mal-rules and cognitive accounts of error. *Cognitive Science: A Multi-disciplinary Journal*, 14(3), 445–481.
- Razzaq, L., Patvarczki, J., Almeida, S. F., Vartak, M., Feng, M., Heffernan, N. T., et al. (2008). The ASSISTment Builder: Supporting the Life Cycle of ITS Content Creation. *IEEE Transactions on Learning Technologies*, 2(2), 157–166.
- Snow, R. E., & Swanson, J. (1992). Instructional psychology: Aptitude, adaptation, and assessment. *Annual Review of Psychology*, 43(1), 583–626.
- Stottler, R. H., Fu, D., Ramachandran, S., & Vinkavich, M. (2001). *Applying a Generic Intelligent Tutoring System (ITS) Authoring Tool to Specific Military Domains*. Paper presented at the Interservice/Industry Training, Simulation & Education Conference.
- Suraweera, P., Mitrovic, A., & Martin, B. (2007). *Constraint Authoring System: an empirical evaluation*. Paper presented at the 13th International Conference on Artificial Intelligence in Education.
- VanLehn, K. (1982). Bugs are not enough: Empirical studies of bugs, impasses and repairs in procedural skills. *Journal of Mathematical Behavior*, 3(2), 3–71.
- Walonoski, J. A., & Heffernan, N. T. (2006). *Detection and analysis of off-task gaming behavior in intelligent tutoring systems*. Paper presented at the 8th International Conference on Intelligent Tutoring Systems.

## Appendix: Questionnaire

Because of space restrictions, only the parts of the questionnaire that produced data used in this paper are reproduced in this appendix. However, a full version of the questionnaire is available in (Folsom-Kovarik et al., in preparation).

### A recent ITS

Please describe the intelligent tutoring system (ITS) you worked on most recently that is ready, or nearly ready, to interact with students.

1. For the ITS you worked on most recently, approximately how many different student models did it use? *{No explicit student model, 1, 2, 3 or more modeling components}*
2. What student model type or modeling algorithm did the system use to SELECT MATERIAL to present? What did the system use to RESPOND TO ERRORS? If the system used

more than one student model, please describe ONE model for each adaptation type.

Selecting or ordering material: *{Choose one or free response}*

Adapting corrections or hints: *{Choose one or free response}*

*Did not use student modeling*

*Overlay model*

*Differential model*

*Perturbation model*

*Bug or bug-part library*

*Model tracing*

*Knowledge tracing*

*Example tracing*

*Other production-rule model*

*Constraint-based model*

*Case-based model*

*Finite-state automata*

*Behavior transition networks*

*Decision trees*

*Neural networks*

*Neurule system*

*Bayesian networks*

*Other (fill in below)*

For the following questions, feel free to answer with an estimate, a range, or even an order of magnitude.

Please measure work in person-hours: each person working full-time for one week contributes about 40 person-hours, and one person working full-time for a year contributes about 2000 person-hours.

3. About how much work, measured in person-hours, did it take to create the ITS? How much of that time was spent working on the student models?

The whole ITS: *{Free response}*

The primary student model for MATERIAL SELECTION: *{Free response}*

The primary student model for HINTS AND FEEDBACK: *{Free response}*

4. Approximately how much additional time, measured in person-hours, was saved by reusing work from other projects?

The whole ITS: *{Free response}*

The primary student model for MATERIAL SELECTION: *{Free response}*

The primary student model for HINTS AND FEEDBACK: *{Free response}*

5. Did your team use any authoring tools to help build the ITS?

The whole ITS: *{Yes, No}*

The primary student model for MATERIAL SELECTION: *{Yes, No}*

The primary student model for HINTS AND FEEDBACK: *{Yes, No}*

6. (Optional) If so, which authoring tools did you use? *{Free response}*

7. When the project was finished, how many hours of instruction per student did the ITS provide? *{Free response}*

8. Are there any other comments you'd like to include about the student models in this ITS, how their design was determined, the model-building process, or anything else? *{Free response}*

### *Demographic information*

As with all the questions in this survey, these questions are optional and you may leave any of them blank.

38. What type of organization do you work for? *{Industry, Government, Academic}*

39. Approximately how many adaptive education or training systems have you been involved with researching or creating? *{0, 1-2, 3-5, 6+}*

40. Approximately how long have you been involved with the research or development of adaptive technologies for education or training? *{N/A, 1-2 years, 3-6 years, 7+ years}*

### **Acknowledgements**

This work is supported in part by the Office of Naval Research Grant N0001408C0186, the Next-generation Expeditionary Warfare Intelligent Training (NEW-IT) program. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the ONR or the US Government. The US Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

### **Author Biographies**

**JEREMIAH T. FOLSOM-KOVARIK** is a Research Assistant with the UCF-IST Applied Cognition and Training in Immersive Virtual Environments (ACTIVE) lab. He is a graduate student working toward a computer science PhD. His research with ACTIVE focuses on improving ITSs with novel user-system interactions and learner modeling opportunities.

**SAE SCHATZ, Ph.D.** is a Research Associate with the UCF-IST ACTIVE laboratory. Dr. Schatz performs applied research in scenario-based training, adaptive instruction, individual differences, and cultural modeling. She is experienced with both social and technological sciences, including human factors, human-computer interaction, educational psychology, computer programming, and instructional technology. Before joining ACTIVE, she served as an instructor in the UCF Digital Media Department, where she taught design philosophy, graphic arts, and web development.

**DENISE NICHOLSON, Ph.D.** is the Director of the UCF-IST ACTIVE laboratory. Her research focuses on human systems modeling, simulation and training includes virtual reality, human-agent collaboration, and adaptive human systems technologies for Department of Defense applications. She joined the university in 2005 with over 18 years of government service ranging from bench-level research at the Air Force Research Lab to leadership as the Deputy Director for Science and Technology at the U.S. Navy's NAVAIR Training Systems Division. She has authored more than 70 technical publications, and is coeditor of the three-volume work *The Handbook of Virtual Environments for Training and Education*, released November 2008.

# Projecting Grammatical Features in Nominals: Cognitive Processing Theory & Computational Implementation

*Jerry T. Ball*

Air Force Research Laboratory

6030 S. Kent St

Mesa, AZ 85212

(480) 988-6561

Jerry.Ball@mesa.afmc.af.mil

Keywords:

grammatical feature, nominal, incremental, interactive, pseudo-deterministic, language comprehension

**ABSTRACT:** *The cognitive processing theory and computational implementation of a linguistic theory of the representation and projection of grammatical features in nominals is described. The processing of nominals is part of a larger model of language comprehension implemented in the ACT-R cognitive architecture. The model combines a serial, pseudo-deterministic processing mechanism for building linguistic representations—implemented within ACT-R’s production system—with a parallel, activation and selection mechanism for choosing between alternatives—implemented as an interaction between ACT-R’s procedural (production) and declarative memory (DM) systems.*

## 1. Introduction

This paper describes an extension to a model of human language comprehension which incorporates grammatical features within nominals to support the binding of pronouns, anaphors and elliptical arguments, and to facilitate reference resolution. The language comprehension model has been under development in the ACT-R cognitive architecture (Anderson, 2007) since 2002 (Ball, 2003; Ball, 2007b; Ball, Heiberg & Silber, 2007) and is capable of handling a broad range of grammatical constructions. A key commitment is development of a model which is at once functional and cognitively plausible. We believe that adherence to well-established cognitive constraints may actually facilitate the development of a functional model by pushing development in directions that are more likely to be successful. Although there may be short-term costs associated with adherence to cognitive constraints, we expect, and have already realized, longer-term benefits (Ball et al., submitted). The dual commitment to functionality and plausibility distinguishes this research from most research in computational linguistics and computational psycholinguistics.

The language comprehension model is a key component of a larger synthetic teammate model (Ball, et. al, 2009) which includes language generation, dialog management and task behavior components, in addition to language comprehension. These components interface to each other through a situation representation component. The major components of the synthetic teammate are all being developed within ACT-R. The main objective of the synthetic teammate project is to develop cognitive agents capable of being integrated into team training simulations without detriment in training. To achieve this goal, the

cognitive agents must be capable of closely matching human behavior across a range of cognitive capacities.

## 2. Linguistic Theory

The underlying linguistic theory is an adaptation of X-Bar Theory (Chomsky, 1970; Jackendoff, 1977) called Bi-Polar Theory (Ball, 2007a). In Bi-Polar Theory, there are four primary phrase internal grammatical functions: *head*, *specifier*, *complement*, and *modifier*. With respect to nominals or noun phrases (NPs), the typical head is a noun like “pilot” and the typical specifier is a determiner like “the” as in “the pilot”. We reject the functional head hypothesis (Abney, 1987) which treats “the” as the head and “pilot” as a complement, aligning instead with Culicover & Jackendoff’s (2005) “Simpler Syntax”. The specifier and head—the most basic elements of a nominal—constitute the two poles of Bi-Polar Theory. At a minimum, a nominal will contain a specifier, a head, or both. The typical modifier—which is not required—is either an adjective like “old” which occurs between the specifier and head as in “the old pilot” or a prepositional phrase like “in the airplane” which occurs after the head as in “the pilot in the airplane”. There are few true complements in nominals and they will not be considered in this paper. We prefer the terms nominal or object referring expression to NP, since the head of a nominal is not necessarily a noun—the head may be empty (e.g. “the red” in “I like the red” in reference to a red object) or it may contain a word or phrase that is not a noun (e.g. “running” in “the running of the bull” or “giving to the poor” in “his giving to the poor is nice”).

It is a key claim of this research that words and phrases functioning as specifiers and modifiers—in addition to

heads—may project grammatical features to encompassing nominals. Grammatical features may be redundantly encoded in words and phrases fulfilling different grammatical functions. At the level of the nominal, the projected grammatical features are collected into a set without duplicates. Redundantly encoded grammatical features may occasionally conflict or a grammatical feature may be unspecified—without the expression being ungrammatical—necessitating mechanisms for handling conflicts and accommodating unspecified features.

The primary grammatical features include definiteness, number, animacy, gender, person and case. The definiteness feature is most closely associated with determiners like “the” and “a”, demonstrative pronouns like “this” and “that” and quantifiers like “all” and “some”. There are (at least) four possible values: *universal* (e.g. “all” in “all books”), *definite* (e.g. “the” in “the book”), *indefinite* (e.g. “a” in “a book”), and *negative* or *zero* (e.g. “no” in “no books”). The number, animacy and gender features are most closely associated with nouns. The possible values for number are *singular*, *mass* (a subtype of *singular*) and *plural*. The possible values for animacy are *human* (a subtype of *animate*), *animate* and *inanimate*. The possible values for gender are *male* and *female*. There is no *neuter* gender in English. With a few exceptions, only human (or animate) nouns are encoded for gender. Plural and mass nouns, but not singular count nouns, are also *indefinite*. For example, the singular count noun “man” is *singular*, *human* and *male*; the plural count noun “rocks” is *indefinite*, *plural* and *inanimate*; and the singular mass noun “rice” is *indefinite*, *singular* and *inanimate*. The grammatical features person and case are only associated with a small number of personal, possessive and reflexive pronouns (e.g., “I” is *first* person, *subjective* case; “me” is *first* person, *objective* case; “he” is *third* person *subjective* case; “him” is *third* person, *objective* case). All reflexive pronouns are objective case (e.g. “myself” is *first* person *objective*, “himself” is *third* person, *objective*) and all possessive pronouns are genitive case (e.g. “my” is *first* person, *genitive*, “hers” is *third* person, *genitive*). There are actually two genitive forms in English, one which functions as a specifier (e.g. “my” in “my book”) and one which functions like a pronoun (e.g. “mine”). Although we use the term “case” to describe the genitive, it differs from subjective and objective case in important respects, especially in its specifier function.

To be grammatical, a nominal normally requires an indication of definiteness, typically provided by the specifier, and an indication of number, typically provided by the head. For example, in “the book”, “the” is *definite* and “book” is *singular*. Since pronouns, proper nouns, and plural and mass nouns also provide an indication of definiteness, they can occur alone as nominals (e.g. “he” is *definite* and *singular*, “John” is *definite* and *singular*,

“books” is *indefinite* and *plural* as in “books are fun to read”). On the other hand, singular count nouns do not provide an indication of definiteness and do not normally occur alone in nominals (e.g. “\*book is fun to read”).

A key aspect of language comprehension is determining the referents of nominals. The set of grammatical features projected to the nominal provides the grammatical basis for determining the referent, and is especially important for determining co-reference. For example, given the input “The man kicked the ball. She ran to first base.” the nominal “the man” indicates that an object of type man is being referred to that is somehow salient in the context of the utterance. This salience is indicated by the *definite* feature of “the”. Likewise for “the ball”. On the other hand the occurrence of “she” is problematic. Pronouns normally indicate co-reference to a previously introduced referent. However, the *female* gender of “she” is inconsistent with the *male* gender of “the man” and the *human* animacy of “she” is inconsistent with the *inanimate* feature of “the ball”. There is no previously mentioned referent to which the pronoun can co-refer.

Besides their importance for reference determination, grammatical features facilitate language comprehension in other ways. For example, interpreting the classic “flying planes are dangerous” vs. “flying planes is dangerous” depends on number agreement between the subject “flying planes” and the auxiliary verb “is” vs “are” with “flying planes” being ambiguous between a reading in which the head “planes” projects the feature *plural*, and a reading in which the head “flying” leads to construal of the expression as *singular*. Likewise, determining the meaning of “the book I gave the man” and “the man I gave the book” hinges on the animacy of “book” and “man”, interacting with the ditransitive verb “give” which prefers an animate indirect object and an inanimate direct object.

Although grammatical features can be extremely useful for language comprehension, they are only useful to the extent that there is grammatical evidence that they exist. It makes little sense to treat common nouns as having case or person features since there is no grammatical marking for these features in English. For example, “the man” can occur as the subject or object as in “the man kicked the ball” and “the horse kicked the man”. Including a case feature for common nouns simply introduces an ambiguity that must be resolved by the context in which the noun occurs—the noun itself provides no such indication. With respect to person, all common nouns could be treated as *third* person by analogy with *third* person pronouns which are grammatically distinct, coupled with claims that subject-verb agreement in English is based on both number and person. However, Ball (submitted) argues that subject-verb agreement in English is based strictly on number, with the exception of the *first* person pronoun “I” and present tense

verbs (e.g. “I am hungry”), making a *third* person feature for common nouns grammatically unnecessary.

We adhere to the basic principle that **where there is no grammatical distinction, there is no grammatical feature**. Without grammatical evidence, there is simply no basis for learners of English to learn the feature. Although most pronouns are marked for case and person in English, common nouns are not. Insisting that all nouns have case and person features to capture a (universal) generalization over nouns and pronouns, is counter-productive—the grammatical generalization introduces unnecessary ambiguity which does not facilitate comprehension. Knowledge of language involves representations or constructions at multiple levels of abstraction, with the most specific constructions that match a given linguistic input carrying most of the weight for language comprehension.

### 3. Psycholinguistic Theory

There is extensive psycholinguistic evidence that human language processing is essentially incremental and interactive (Gibson & Pearlmutter, 1998; Altmann, 1998; Tanenhaus et al., 1995; Altmann & Steedman, 1988). Garden-path effects, although infrequent, strongly suggest that processing is essentially serial at the level of phrasal and clausal analysis (Bever, 1970). Lower level processes of word recognition suggest parallel, activation-based processing mechanisms (McClelland & Rumelhart, 1981; Paap et al., 1982). At the level of phrasal and clausal analysis, humans appear to deterministically pursue a single analysis which is only occasionally disrupted, requiring reanalysis. One of the great challenges of psycholinguistic research is to explain how humans can process language effortlessly and accurately given the complexity and ambiguity that is attested (Crocker, 2005). As Boden (2006, p. 407) notes, deterministic processing “would explain the introspective ease and speed of speech understanding”, but a purely deterministic, incremental processing mechanism would more frequently make incorrect local choices requiring reanalysis than is evident in human language processing. Marcus (1980) proposed a lookahead mechanism to improve the performance of a deterministic, yet monotonic, processor, bringing it into closer alignment with human performance. However, there is considerable evidence that humans immediately determine the meaning of linguistic inputs (cf. Tanenhaus et al., 1995; Altmann & Mirkovic, 2009) which is inconsistent with extensive lookahead, delay or underspecification—the primary serial and monotonic mechanisms for dealing with ambiguity. As Altmann & Mirkovic (2009, p. 605) note “The view we are left with is a comprehension system that is ‘maximally incremental’; it develops the fullest interpretation of a sentence fragment at each moment of the fragment’s unfolding”. Not only is there not extensive lookahead, delay

or underspecification, the human language processor engages in “thinkahead”, predicting what will come next rather than waiting until the succeeding input is available before deciding on the current input.

To capture the essentially incremental nature of human language processing, we adopt a *serial, pseudo-deterministic* processor that builds linguistic representations by integrating compatible elements, relying on a non-monotonic mechanism of *context accommodation* to handle cases where some incompatibility that complicates integration manifests itself. Context accommodation makes use of the full context to make modest adjustments to the evolving representation or to construe the current input in a way that allows for its integration into the representation. Context accommodation need not be computationally expensive (i.e., a single production may effect the accommodation, just as a single production may effect integration without accommodation). In this respect, context accommodation is not a reanalysis mechanism that disrupts normal processing; rather, it is part and parcel of normal processing. Reanalysis mechanisms need only kick in when context accommodation fails and larger adjustment is needed. Further, as will be shown below, context accommodation can give the appearance of parallel processing in a serial processing mechanism, blurring the distinction between serial and parallel processing.

The mechanism of context accommodation is most closely related to the *limited repair parsing* of Lewis (1998). Context accommodation may be viewed as a very modest form of repair. According to Lewis (1998, p. 262) “The putative theoretical advantage of repair parsers depends in large part on finding simple candidate repair operations”. The mechanism of context accommodation provides evidence for this theoretical advantage.

To capture the essentially interactive nature of human language processing, we propose a probabilistic, context-sensitive mechanism for activating alternatives in parallel and selecting the most highly activated alternative. This *parallel, probabilistic* mechanism selects between competing alternatives, but does not build any structure—building structure is the function of the incremental integration mechanism. At each choice point, the parallel, probabilistic mechanism uses all available information to activate and select the preferred alternative, and the serial, pseudo-deterministic mechanism integrates the preferred alternative into the evolving representation. Use of the full local context supports selection of alternatives that are likely to be correct, allowing the serial integration mechanism to be largely deterministic. However, the local context is not always consistent with the global context and locally preferred choices sometimes turn out to be globally dispreferred. The mechanism of context accommodation allows the processor to adjust the evolving representation to accommodate the subsequent context, without lookahead,

backtracking or reanalysis. Only when the context accommodation mechanism breaks down do more disruptive reanalysis processes become necessary. The use of the term *pseudo-deterministic* to describe the basic processing mechanism reflects the integration of parallel, probabilistic activation and selection mechanisms and context accommodation with what is otherwise a serial, deterministic processor.

#### 4. Cognitive Processing Theory

ACT-R (Adaptive Control of Thought—Rational) is a computational implementation of a general cognitive architecture developed to model a broad range of cognitive capacities (Anderson, 2007). It consists of a production system combined with a declarative memory system and includes modest perceptual-motor capabilities for interacting with a computer. There is no distinct language subsystem within ACT-R (nor does the language comprehension model introduce such a subsystem). In ACT-R, a single production executes at a time, providing a serial bottleneck for processing, however, which production is selected for execution is determined by a parallel, utility selection mechanism. Similarly, declarative memory (DM) retrieval returns a single DM chunk, but selection of the chunk relies on a parallel, spreading activation mechanism. ACT-R is thus a hybrid serial, parallel architecture.

The language comprehension model—called Double-R (for Referential and Relational)—builds linguistic representations of referential and relational meaning based on the linguistic input, surrounding context and prior knowledge. The model uses ACT-R's production system to build representations, combined with ACT-R's declarative memory (DM) system to select grammatical constructions which are used to build these representations. Grammatical constructions (including word level constructions) are stored in DM and retrieved on the basis of spreading activation from the linguistic input and the prior context. The spreading activation mechanism interacts with the production system via a retrieval production which specifies the type of construction to be retrieved and the current goal. The single grammatical construction which matches the retrieval template and is most consistent with the linguistic input, prior context and current goal is retrieved. Separate integration and/or build productions determine how to integrate the retrieved construction into the evolving representation, either via integration into an existing representation or projection of a novel representation.

At the processing of each word in a linguistic input, humans typically succeed in identifying the word, determining the correct grammatical function of the word, and integrating the word into the evolving linguistic representation. The likely way this is accomplished is by using all available information—be it lexical, syntactic, semantic or pragmatic—to make the correct grammatical choice. This

implies a highly context sensitive, parallel determination of the grammatical function of the current word (consistent with constraint-based theories), followed by the serial and deterministic integration into (or projection of) the evolving representation (an aspect of processing ignored—or at least de-emphasized—by most constraint-based theories). At each choice point, all information is considered in parallel in making the best choice, but once a choice is made, processing proceeds serially and deterministically forward until the next choice point.

In the processing of nominals, this means that the processing of each word leads to recognition of the word, determination of the appropriate phrase internal grammatical function of the word, projection of a higher level phrasal unit or integration of the grammatical function into an existing higher level phrasal unit, and projection of grammatical features from the grammatical function to the higher level unit. For example, in the processing of “the man”, the processing of the word “the” leads to recognition of the determiner “the”, determination of its grammatical function as a specifier, projection of a nominal construction, and projection of the grammatical feature *definite* to the nominal construction. The subsequent processing of “man” leads to recognition of the noun “man”, determination of its grammatical function as a head, integration of the head into the nominal construction projected by “the” and projection of the grammatical features *singular* (number), *human* (animacy) and *male* (gender) to the nominal construction. It is important to note that the determiner “the” projects a nominal construction. Not only do determiners project grammatical features, but they project nominal constructions and determine the category of the construction (functioning like a head in this respect). On the other hand, in the absence of a determiner (and projected nominal) a *plural* or *mass* noun can also project a nominal construction. For example, in “rice is good for you”, the mass noun “rice” projects a head which in turns projects a nominal construction (in the absence of a nominal construction projected by a determiner), and projects the grammatical features *indefinite* (definiteness), *singular* (number), and *inanimate* (animacy) to the nominal.

When the projection of grammatical features results in a conflict, blocking or overriding mechanisms—specific instances of context accommodation—come into play. The blocking and overriding mechanisms occur within the current context, making full use of the context to determine the appropriate projection of grammatical features. As an example of feature blocking, consider the nominal “the books”. The *definite* feature of “the” projects to the nominal and blocks projection of the *indefinite* feature of “books”. As an example of feature overriding consider the nominal “that dog”. The *inanimate* feature of “that” is overridden by the *animate* feature of “dog”. Grammatical evidence that “that” carries the feature *inanimate* is provided by expressions like

“I like that” in which “that” cannot normally be used to refer to an *animate* object.

Determination of the grammatical function of a word has important representational and processing implications. For example, in the processing of “that” in “that man”, if “that” functions as a specifier and projects a nominal, then when “man” is processed, “man” can simply be integrated as the head of the nominal. In this case, “that” behaves like a typical determiner. However, if “that” functions as the head—behaving instead like a typical pronoun, then when “man” is processed, “man” must be accommodated by shifting “that” into the specifier function to allow “man” to function as the head. Whether or not “that” is encoded in the mental lexicon as a determiner, a pronoun (including relative pronoun), or both, is likely to depend on the history of use of the word. Regardless of which form is retrieved, the language processor must be capable of accommodating the alternative use. Given that the function of “that” cannot be fully determined until the subsequent input is processed (assuming an incremental processor without lookahead), retrieval mechanisms are likely to retrieve the most frequent form (unless the prior context is somehow able to bias retrieval of the alternative form). This basic fact is often overlooked in grammatical treatments which ignore processing considerations. Thus, it is often suggested that “that” in “that man” is a (demonstrative) determiner, whereas, “that” in “that is nice” is a (demonstrative) pronoun. For this to be the case, determining the part of speech of “that” would need to be delayed until after the subsequent input is processed, or ignoring processing, given the syntactic context surrounding “that”.

A similar mechanism is needed in the incremental processing of noun-noun combinations. For example, in the processing of “the altitude restrictions”, when “altitude” is processed it can be integrated as the head of the nominal projected by “the”, but when “restrictions” is subsequently processed, “altitude” must be shifted into a modifier function to allow “restrictions” to function as the head.

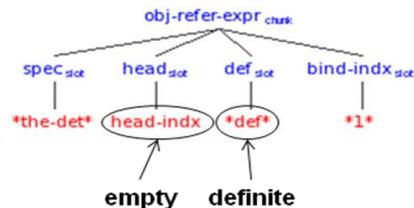
### 5. Computational Implementation

The language comprehension model contains a capability to display the representations that are generated from the linguistic input in a tree format (Heiberg, Harris & Ball, 2007). In the model, nominals are called object referring expressions (abbreviated “obj-refer-expr”). The use of the term “object **referring expression**” indicates that the representations are linguistic, but not purely syntactic, and highlights the importance of the referential dimension of meaning. The terminal nodes may contain words, but do not contain anything like abstract concepts or word senses. To more fully represent the meaning of the object referring expression, it must be mapped to a non-linguistic representation of the object to which it refers (within the situation representation). This mapping will not be discussed

in this paper, but it is noted that the mapping is facilitated by the nature of the linguistic representations as compared to typical syntactic representations.

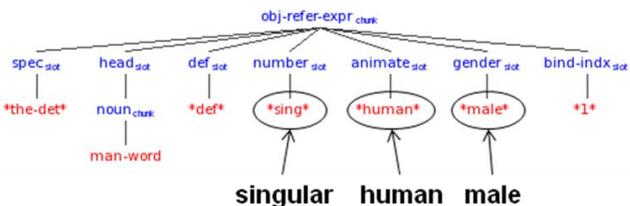
The processing of the nominal “the man” is shown below:

“the” →



The word “the” is identified as a determiner (abbreviated “\*the-det\*”) that projects an object referring expression with “the” functioning as the specifier (abbreviated “spec”). The object referring expression chunk has a head slot. The value “head-idx” indicates that this slot does not yet have a value. The object referring expression chunk has a definiteness slot (abbreviated “def”) which has the value *definite* (abbreviated “\*def\*”). This value was projected from “the”. Finally, the object referring expression has a “bind-idx” slot which contains the index “\*1\*”. This index supports the binding of pronouns, traces and anaphors in more complex linguistic expressions. It should be noted that the tree representations are simplified in various respects. In particular, the grammatical feature slots of the individual lexical items are not displayed. Further, only some slots without values are displayed. For example, the head slot is displayed even if it doesn’t have a value, but grammatical feature slots and modifier slots (pre and post-head) without values are not displayed.

“the man” →



The processing of the word “man” leads to its identification as a noun and integration as the head of the object referring expression projected by “the”. “Man” projects the grammatical features number, animate (i.e., animacy), and gender with the values *singular*, *human*, and *male* to the object referring expression.

The processing of pronouns like “his” and “her” introduces interesting challenges for an incremental processor. Consider the processing of “his book” (diagrams on page 7). The possessive pronoun/determiner “his”—treated as a possessive pronoun (abbreviated “poss-pron”) by the model—projects a possessive object specifier (abbreviated “poss-obj-spec”) which is a special type of object referring

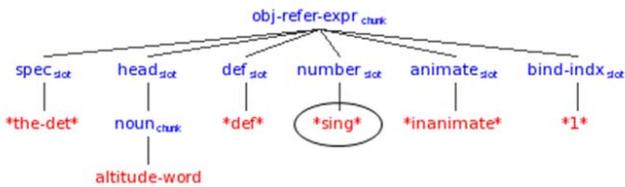
expression that functions as a specifier. In addition to the grammatical features typical of nouns and determiners, the features person and case with the values *third* and *genitive* (abbreviated “\*gen\*”) are projected to the possessive object specifier. The possessive object specifier in turn projects a higher level object referring expression and functions as the specifier. The *definite* feature of the possessive object specifier is projected to the higher level object referring expression. Note that there are two distinct bind indexes to support co-reference to either object referring expression. The word “book” is recognized as a noun and integrated as the head of the higher level object referring expression projected by “his”. The features *singular* and *inanimate* are projected to the higher level object referring expression. Overall, the object referring expression refers to an object of type book. Reference to this object is facilitated by inclusion of the possessive pronoun “his” which provides a *reference point* (cf. Taylor, 2000) for identifying the referent of the overall expression.

The pronoun “her” differs from “his” in that it is both a personal pronoun and a possessive determiner (e.g., “I like her” vs. “I like her book”). Whereas “her” alone functions as a personal pronoun, establishing a single referent, “his” alone does not. In “I like his”, “his” is functioning as a possessive pronoun, not a personal pronoun. Possessive pronouns, unlike personal pronouns, establish dual referents via a separate reference point. Note that “his” unlike “her” is both a possessive determiner and possessive pronoun (“hers” is the possessive pronoun form of “her”). At the processing of the word “her”, it is treated as a personal pronoun and functions as the head of the projected object referring expression, but if “her” is followed by “books”, a higher level object referring expression is projected and “her” is shifted into a specifier function, so “books” can function as the higher level head (projection of the *indefinite* feature of “books” is blocked). As a personal pronoun, “her” also projects case and person features with the values *objective* (abbreviated \*obj\*) and *third*. From a processing perspective, the primary difference between “his” and “her” is that “his” immediately projects a higher level object referring expression and functions as a specifier within the higher level expression—setting up the expectation for a head—whereas “her” does not (see diagrams on next page).

The possessive pronoun “hers” differs from “his” in that there is no expectation for the occurrence of a head in the higher level object referring expression (i.e., “hers” cannot be a possessive determiner as in “\*hers book”). This is indicated by marking the head of the higher level object referring expression as “\*implied\*” (a similar approach is adopted in the treatment of the implied subject of imperative statements) (see diagram on next page).

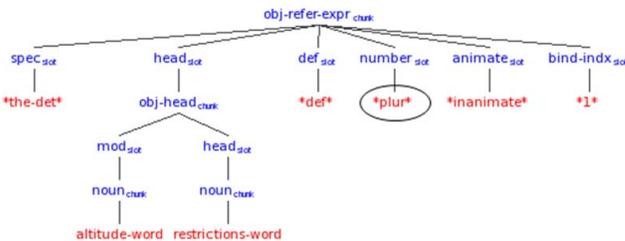
As a final example, consider the processing of “the altitude restrictions”. The processing of “the” is as before.

“the altitude” →



The word “altitude” is identified as a noun and integrated as the head of the object referring expression projected by “the”. “Altitude” also projects the grammatical features *singular* and *inanimate*. In parallel, “altitude” projects an object head structure with pre- and post-head modifier slots (see “obj-head” below showing pre-head “mod” and “head” slots). The capability of the model to build structures in parallel is extremely limited. In this case, the object head is projected in parallel but does not get integrated into a higher level structure unless needed to support subsequent processing. Integration of “altitude” (the noun) as the head is the minimum structure needed at this point in processing.

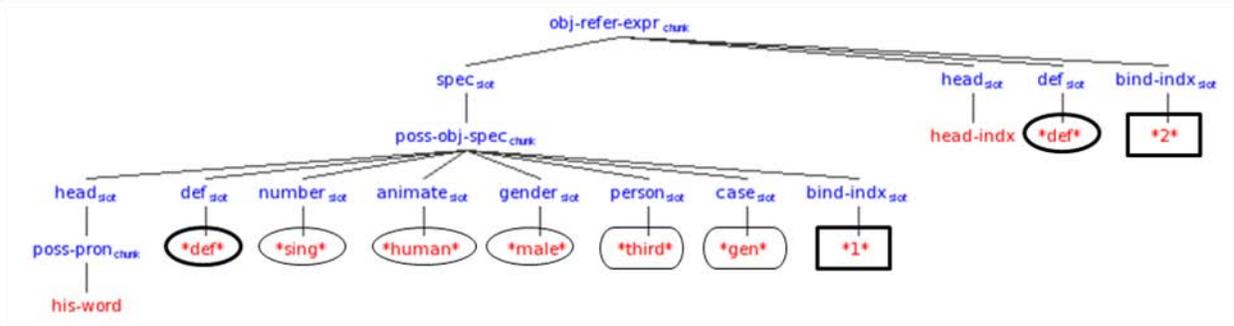
“the altitude restrictions” →



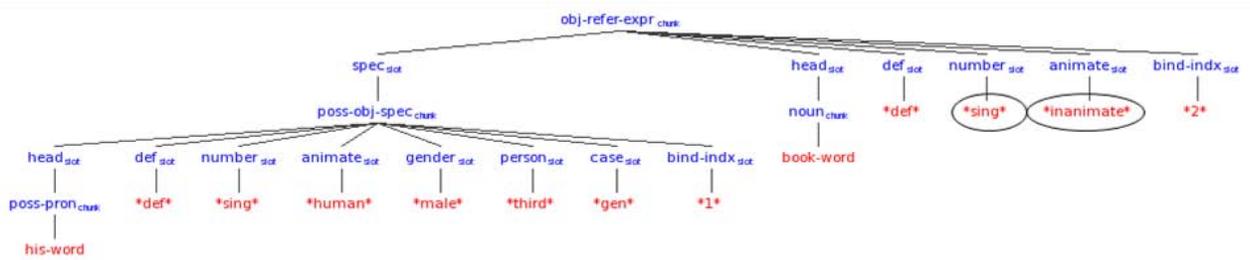
The word “restrictions” is identified as a noun. To accommodate “restrictions” the object head that was projected in parallel by “altitude” replaces “altitude” as the head of the object referring expression. In addition, “altitude” is shifted into the pre-head modifier slot of the object head (abbreviated “mod”) to allow “restrictions” to function as the head. Finally, the *plural* number feature of “restrictions” overrides the *singular* number feature of “altitude”. Note that at the end of processing it appears that “altitude” was treated as a modifier all along. The context accommodation mechanism gives the appearance of parallel processing without the computational expense of building and carrying forward multiple representations in parallel, although a limited amount of parallelism is supported. Context accommodation also minimizes the amount of structure building.

Whereas context accommodation can handle mundane examples like those discussed above, such examples differ from the disruptive garden-path examples which are typically used in psycholinguistic studies of reanalysis (e.g., the famous “the horse raced past the barn fell” from Bever, 1970). Context accommodation is not capable of handling such disruptive inputs.

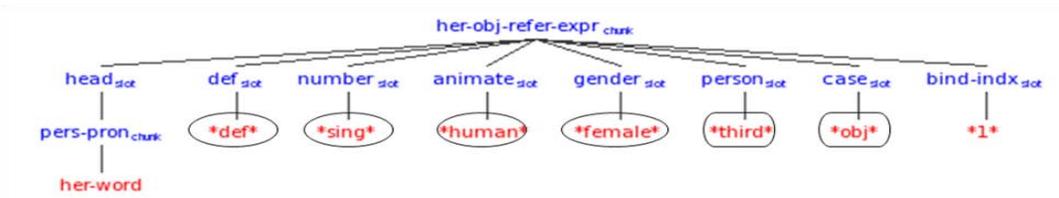
“his” →



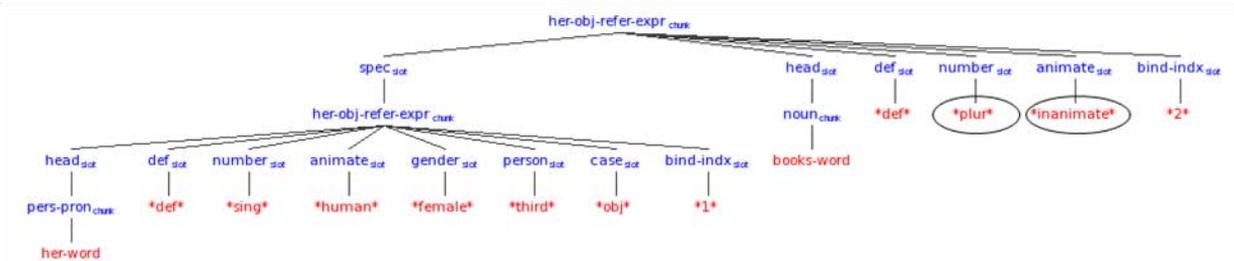
“his book” →



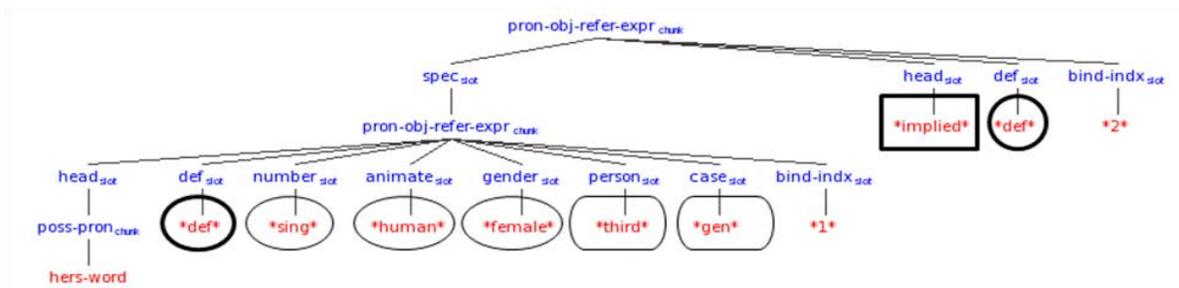
“her” →



“her books” →



“hers” →



## 6. Summary

This paper has focused on describing aspects of the cognitive processing theory and computational implementation of grammatical feature processing in nominals within a larger model of language comprehension implemented in the ACT-R cognitive architecture. A serial, pseudo-deterministic processing mechanism grounded in ACT-R's production system, combines with a parallel, probabilistic mechanism grounded in an interaction between ACT-R's DM and production system. The pseudo-deterministic mechanism functions to build representations of the linguistic input, whereas the parallel, probabilistic mechanism functions to select between DM alternatives. A context accommodation mechanism for handling feature overriding and blocking supports modest adjustment of the evolving representation.

## 7. References

- Abney, S. P. (1987). *The English noun phrase in its sentential aspect* (Doctoral dissertation). Massachusetts Institute of Technology, Cambridge, MA.
- Altmann, G. (1998). Ambiguity in sentence processing. *Trends in Cognitive Sciences*, 2(4), 146-152.
- Altmann, G. & Mirkovic, J. (2009). Incrementality and Prediction in Human Sentence Processing. *Cognitive Science*, 222, 583-609.
- Altmann, G., & Steedman, M. (1988). Interaction with context during human sentence processing. *Cognition*, 30, 191-238.
- Anderson, J. (2007). *How Can the Human Mind Occur in the Physical Universe?* NY: Oxford University Press.
- Ball, J. (2003). Beginnings of a Language Comprehension Module in ACT-R 5.0. *Proceedings of the Fifth International Conference on Cognitive Modeling*. Edited by F. Detje, D. Doerner and H. Schaub. Universitaets-Verlag Bamberg. ISBN 3-933463-15-7.
- Ball, J. (2007a). A Bi-Polar Theory of Nominal and Clause Structure and Function. *Annual Review of Cognitive Linguistics*, 5, 27-54.
- Ball, J. (2007b). Construction-Driven Language Processing. *Proceedings of the 2<sup>nd</sup> European Cognitive Science Conference*, 722-727. Edited by S. Vosniadou, D. Kayser & A. Protopapas. NY: LEA.
- Ball, J. (submitted). Projecting Grammatical Features in Nominals: Cognitive Theory and Computational Model.
- Ball, J., Freiman, M., Rodgers, S. & Myers, C. (submitted). Toward a Functional Model of Human Language Processing.
- Ball, J., Heiberg, A. & Silber, R. (2007). Toward a Large-Scale Model of Language Comprehension in ACT-R 6. In R. Lewis, T. Polk & J. Laird (Eds.) *Proceedings of the 8<sup>th</sup> International Conference on Cognitive Modeling* (pp. 173-179). NY: Psychology Press.
- Ball, J., Myers, C. W., Heiberg, A., Cooke, N. J., Matessa, M., & Freiman, M. (2009). The Synthetic Teammate Project. *Proceedings of the 18<sup>th</sup> Annual Conference on Behavior Representation in Modeling and Simulation*. Sundance, UT.
- Bever, T. (1970). The cognitive basis for linguistic structures. In J. Hayes (Ed.), *Cognition and the development of language* (pp. 279-362). New York: Wiley.
- Boden, M. (2006). *Mind as machine: A history of cognitive science*. New York: Oxford University Press.
- Chomsky, N. (1970). Remarks on Nominalization. In Jacobs & Rosebaum, (Eds.), *Readings in English Transformational Grammar*. Waltham, MA: Ginn.
- Crocker, M. (2005). Rational models of comprehension: addressing the performance paradox. In A. Culter (Ed), *Twenty-First Century Psycholinguistics: Four Cornerstones*. Hillsdale: LEA.
- Culicover, P. & Jackendoff, R. (2005). *Simpler Syntax*. Oxford: Oxford University Press.
- Gibson, E., & Pearlmutter, N. (1998). Constraints on sentence comprehension. *Trends in Cognitive Sciences*, 2(7), 262-268.
- Heiberg, A., Harris, J., & Ball, J. (2007). Dynamic Visualization of ACT-R Declarative Memory Structure. . In R. Lewis, T. Polk & J. Laird (Eds.) *Proceedings of the 8<sup>th</sup> International Conference on Cognitive Modeling* (pp. 233-234). NY: Psychology Press.
- Jackendoff, R. (1977). *X-Bar Syntax: A Study of Phrase Structure*. Cambridge, MA: The MIT Press.
- Langacker, R. (1987, 1991). *Foundations of Cognitive Grammar*, Volumes 1 & 2. Stanford, CA: Stanford University Press.
- Lewis, R. L. (1998). Leaping off the garden path: Reanalysis and limited repair parsing. In J. D. Fodor, & F. Ferreira (Eds.), *Reanalysis in Sentence Processing*. Boston: Kluwer Academic.
- Marcus, M. (1980). *A Theory of Syntactic Recognition for Natural Language*. Cambridge, MA: The MIT Press.
- McClelland, J., & Rumelhart, D. (1981). An interactive activation model of context effects in letter perception: I. An account of basic findings. *Psychological Review*, 88(5), 375-407.
- Paap, K., Newsome, S., McDonald, J., & Schvaneveldt, R. (1982). An Activation-Verification Model of Letter and Word Recognition: The Word-Superiority Effect. *Psychological Review*, 89, 573-594.
- Tanenhaus, M., Spivey-Knowlton, M., Eberhard, K., & Sedivy, J. (1995). Integration of visual and linguistic information in spoken language comprehension. *Science*, 268(5217), 1632-1634.
- Taylor, J. (2000). *Possessives in English: An exploration in cognitive grammar*. Oxford: Oxford University Press.

## Author Biography

JERRY BALL is a Senior Research Psychologist in the Human Effectiveness Directorate, 711<sup>th</sup> Human Performance Wing, Air Force Research Laboratory.

## Modeling a Visual Search Task with a Secondary Task in IMPRINT

*Carolyn J. Buck-Gengler, William D. Raymond, Alice F. Healy, and Lyle E. Bourne, Jr.*  
Center for Research on Training, University of Colorado, Boulder, Colorado 80309-0345  
buckc@colorado.edu, raymondw@colorado.edu, healy@colorado.edu, lbourne@colorado.edu

Keywords: IMPRINT, visual search, secondary task, resource competition, modeling cognitive processes

### 1. Introduction

IMPRINT is an Army modeling tool used to simulate complex, long-term activities involving personnel and equipment. Recently, it was used to model a simple psychomotor task, digit data entry (Buck-Gengler, Raymond, Healy, & Bourne, 2007). In parallel with ACT-R modeling efforts (Best, Gonzalez, Young, Healy, & Bourne, 2007), the work reported here involves IMPRINT modeling of a visual search task (RADAR) coupled with an auditory secondary task. The ACT-R and IMPRINT models are part of a larger research program aimed at understanding the effects of training on performance. The RADAR model implements the effects on performance, during training and delayed test, of several training manipulations, allowing investigation of the consequences of varying training parameters through simulation.

### 2. Experimental basis of the model

The RADAR task was developed by Gonzalez and Thomas (2008). In the experiment modeled here (Young, Healy, Gonzalez, & Bourne, 2007), subjects searched for symbol targets in 4 squares moving from the 4 corners to the center of a radar-like display in 2.062 s. Different sets of symbols were shown in each of 7 frames comprising a trial. Squares did not always contain a symbol. Subjects were to respond only if a target appeared, and were scored on response speed and accuracy.

The experiment contained both consistent mapping (CM) and variable mapping (VM) trials. In CM targets and foils came from different symbol types (letters, digits), so could be distinguished by set membership alone; in VM both targets and foils were from the same set, requiring specific memory for target items. Processing load was manipulated by varying memory load and search difficulty. In low processing load trials (LP) the target set consisted of a single symbol and only 1 square contained a symbol, with the rest being blank. In high processing load trials (HP) the target set consisted of 4 symbols and all 4 squares contained a symbol, although only at most 1 symbol was from the target set.

Trials were grouped in blocks of 20, with 8 blocks in each of 2 sessions. Session 1 (training) occurred 1 week before Session 2 (test). A random 15 of the 20 trials in each

block contained a target. All trials in a block had the same mapping type and processing load, and the block type varied systematically across the 8 blocks in the following order: CM1, CM4, VM1, VM4, VM4, VM1, CM4, CM1 (where 1 indicates LP and 4 indicates HP).

The effects on the main task of a concurrent secondary task, namely, counting and reporting the number of tones heard during a trial that deviated from a standard (base) tone, were also examined. In tone-counting conditions tones were played throughout the experiment, 500-1500 ms apart. About 15% of the tones deviated obviously from the base tone. There were 48 subjects; half trained with tone counting and target detection and half performed target detection in silence. At test, half the subjects in each tone condition stayed in the same condition and half switched to the other tone condition.

For the primary task of target detection, correct response times (RTs) were faster overall for CM than for VM, and also for LP than for HP. The disadvantage for HP was larger overall for VM than for CM; this interaction was evident at both training and test. Accuracy in terms of hit rate (HR) also showed an interaction; HR was lowest for the VM4 trials. The results for false alarm rate (FAR) were more complex and demonstrated improvement across trials as well as effects of mapping type and processing load.

Tone counting negatively impacted all measures in both sessions. Furthermore, counter-intuitively, training with tone resulted in reduced speed and accuracy in both tone conditions at test.

### 3. Model

The cognitive model of the visual search task simulated in IMPRINT consists of three processing subtasks: (1) eye movement to a square containing a symbol, (2) decision as to whether that square contains a target, and (3) manual response when a target is detected. Subtasks are repeated until the target is found, all squares have been searched, or the trial times out.

Implementation details of the eye movement subtasks differed depending on processing load; details of decision subtasks differed depending on mapping type and training

condition. Eye movements in the LP conditions were to the square containing a symbol; in the HP conditions any square could be moved to first, resulting in shorter movement time, with equivalent times for subsequent movements. In CM, whether the square with a symbol contains a target can be decided simply by comparing the target's symbol type to the symbol type of a square's content. In VM, target decisions require comparison of the square's content to the target set in memory. In VM1, the decision is a comparison of the single target with the square's content, with decision time equivalent to that for CM. In VM4, 4 possible targets must be compared against each square examined, resulting in longer decision times. In all trials, if a target is detected, a response is made and the trial ends; otherwise, the condition-appropriate subtasks repeat until a target has been detected or all 7 trial frames have been presented.

The IMPRINT model was implemented as two parallel networks: one network represented the computer presenting the visual stimuli (and tones, in those conditions); a separate network represented the subject processing stimuli as they were presented.

Hits were modeled stochastically for frames with targets. HR was lower for VM4 trials than other trial types. False alarms were also modeled stochastically for frames without targets. The FAR declines were implemented with exponential functions across trials, with exponents determined by block type. Initial rates in a block were based on the FAR at the end of the previous block and the type of change in difficulty from the previous block to the current block.

RTs for frames with hits were the sum of eye movement, decision, and response times. Eye movement and response times were based on IMPRINT micromodels for eye movement and key pressing. CM and VM1 decision times were modeled stochastically. Greater VM4 decision times were multiples of VM1 times to model search of the memory set. RTs were increased and HRs were decreased to simulate the additional load of the secondary task and the impairment at test from training with tone counting.

#### 4. Results and conclusion

The empirical data were used informally to derive reasonable parameter values, but it was not practical to optimize all values. The final model was used to simulate the experimental data twice, with two different seeds to produce different statistical subject populations. For each simulation the model was executed with 48 statistical subjects, 12 in each tone counting  $\times$  session condition. The model's goodness of fit was evaluated by computing  $r^2$  and RMSE values on the block means produced by the

two runs of the model and comparing those with each other and with the experimental data from Young et al. (2007) for each measure. The model fit the experimental data well for RT ( $r^2(30) = .975$ ) and HR ( $r^2(30) = .969$ ), but less well for FAR ( $r^2(62) = .461$ ); however, the comparisons for FAR had twice as many data points to fit, and the experimental data were not as regular.

The modeling effort was valuable because it revealed that learning within a session on the RADAR task only occurred for the FAR measure. The critical aspect of this model with respect to broader issues concerning training a complex skill is the ability to reproduce both the immediate effects of a secondary task and the counterintuitive finding that training with a secondary task hurt rather than helped subsequent test performance, even when training and testing conditions matched.

#### 5. References

- Best, B. J., Gonzalez, C., Young, M. D., Healy, A. F., & Bourne, L. E., Jr. (2007). Modeling automaticity and strategy selection in dynamic visual detection, *Proceedings of the Sixteen Conference on Behavior Representation in Modeling and Simulation* (pp. 3-11). Orlando, FL: Simulation Interoperability Standards Organization.
- Buck-Gengler, C. J., Raymond, W. D., Healy, A. F., & Bourne, L. E., Jr. (2007). Modeling data entry in IMPRINT. *Proceedings of the Sixteenth Conference on Behavior Representation in Modeling and Simulation* (pp. 205-206). Orlando, FL: Simulation Interoperability Standards Organization.
- Gonzalez, C., & Thomas, R. P. (2008). Effects of automatic detection on dynamic decision making. *Journal of Cognitive Engineering and Decision Making*, 2, 328-348.
- Young, M. D., Healy, A. F., Gonzalez, C., & Bourne, L. E., Jr. (2007, July). *The effects of training difficulty on RADAR detection*. Poster presented at the joint meeting of the Experimental Psychology Society and the Psychonomic Society, Edinburgh, Scotland.

#### 6. Acknowledgments

This research was supported in part by ARO Grant W911NF-05-1-0153 to the University of Colorado.

#### Author Biographies

**CAROLYN J. BUCK-GENGLER** and **WILLIAM D. RAYMOND** are Research Associates, University of Colorado, Boulder.

**ALICE F. HEALY** and **LYLE E. BOURNE, JR.** are Professors, University of Colorado, Boulder.

## Dynamic Data and Modeling Services Suite

*Tina H. Chau*

*Alexander P. Moore*

*Richard L. Mullikin*

*Janet E. Wedgwood*

Lockheed Martin Corporation

720 Vandenberg Drive

King of Prussia, PA 19406

571-313-6605, 610-354-7590, 917-497-0424, 856-792-9879

tina.h.chau@lmco.com, alexander.p.moore@lmco.com, rick.mullikin@lmco.com, janet.e.wedgwood@lmco.com

### Keywords:

counter-insurgency, data harvesting, data services, data query, innovation, integration, proof-of-concept fusion environment, Services Suite, modeling services, geo-location, social sciences

**ABSTRACT:** *This paper presents the feasibility of a complete services suite for end-to-end systems integration of data and modeling services that is tailored for use by commanders, military advisors and intelligence analysts involved in Counter-insurgency Operations. Through the integration of existing and innovative technologies – including automated harvesting of near real-time data from the cyber domain – the Dynamic Data and Modeling Services Suite will enable astute socio-cultural behavior exploration. The existing proof-of-concept fusion environment feeds its predictive behavior models with comprehensive human terrain data from dynamic sources. Future work will include additional models and sources resulting in a complete services suite for facilitating solid, fact-based decision making for Counter-insurgency Operations.*

### 1. Introduction

Dynamic socio-cultural modeling is essential to the operational performance of coalition forces and their host country partners engaged in Counter-insurgency (COIN) Operations. At its core, COIN is a competition with the insurgent to win the hearts, minds and acquiescence of the population. The more commanders, military advisors and intelligence analysts (hereafter referred to as “Users”) understand about the human terrain (*e.g.* behaviors, causes and motivations, foundational thoughts and beliefs, etc.), the more leverage Users will have in that competition.

However, no region of the world is comprised of identical indigenous populations. Each population has several influencing factors that determine its composition, actions, beliefs and motives. These social dynamics, as well as core social sciences, must be considered at all levels for accurate and effective full-spectrum mission planning. Posing an additional challenge is the harvesting of vast and accurate intelligence, which is required to model dynamic socio-cultural environments. This critical mission task is both challenging and time consuming. Open-source intelligence (OSINT), for example, is an increasingly useful data source owing to the expansive

nature of the Internet. At the same time, the diversified and ever-changing cyber domain – from inputs, to access, to content – renders socio-cultural OSINT difficult to collect, manage and store for operational application.

This paper defines the technical and theoretical methodologies behind data harvesting and behavioral modeling as proposed by the Dynamic Data and Modeling Services Suite (hereafter referred to as “Services Suite” or “Suite”). The existing proof-of-concept fusion environment (hereafter referred to as “Environment”), on which the future Suite will build, is a Lockheed Martin research and development effort that began this year. The overall effort incorporates underlying technologies spanning development efforts over the past five years. The authors of this paper detail the ways in which the existing Environment integrates innovative technologies with legacy platforms in order to capture the precise data Users require. The authors further describe Suite methodologies, which are tailored to future real-world applications by operational Users.

The existing Environment takes the dynamic nature of various social sciences into account while investigating population behaviors. This socio-cultural consideration is achieved through the ingestion, management and storage

of behavioral data from diverse sources, all of which is supported by Service Oriented Architectures – primarily the Internet. Feeding various models with data from its data services repository, the Environment then generates current and predictive representations of dynamic social environments. These practices result in behavioral assessment and forecasting models that are founded on ground truth data, definable metrics, powerful visualizations and operational utility.

The future Services Suite will further address the challenge of collecting OSINT from the dynamic cyber domain by automatically harvesting online socio-cultural data. Near real-time data from the Internet will fuel behavioral and predictive models with timely and accurate intelligence. The complete Suite will thus provide Users with the monitoring and predictive technologies necessary to optimize current courses of

action (COAs) to 1) defeat insurgents and terrorists; and 2) ensure the protection of the most important terrain on the battlefield – the Human Terrain.

## 2. Methodology

It is our assertion that Users desire new applications that capitalize on technological advancements in behavioral modeling and data integration in order to achieve maximum mission success in the irregular warfare environment. The existing Environment leverages these technological advancements to enable Users to ingest, manage, store and model human terrain intelligence that is essential to COIN operations. Future work to form the complete Suite will further increase model accuracy by harvesting and integrating online social networking data. This OSINT data is evolving into a pertinent, though largely untapped, source for near real-time behavioral information.

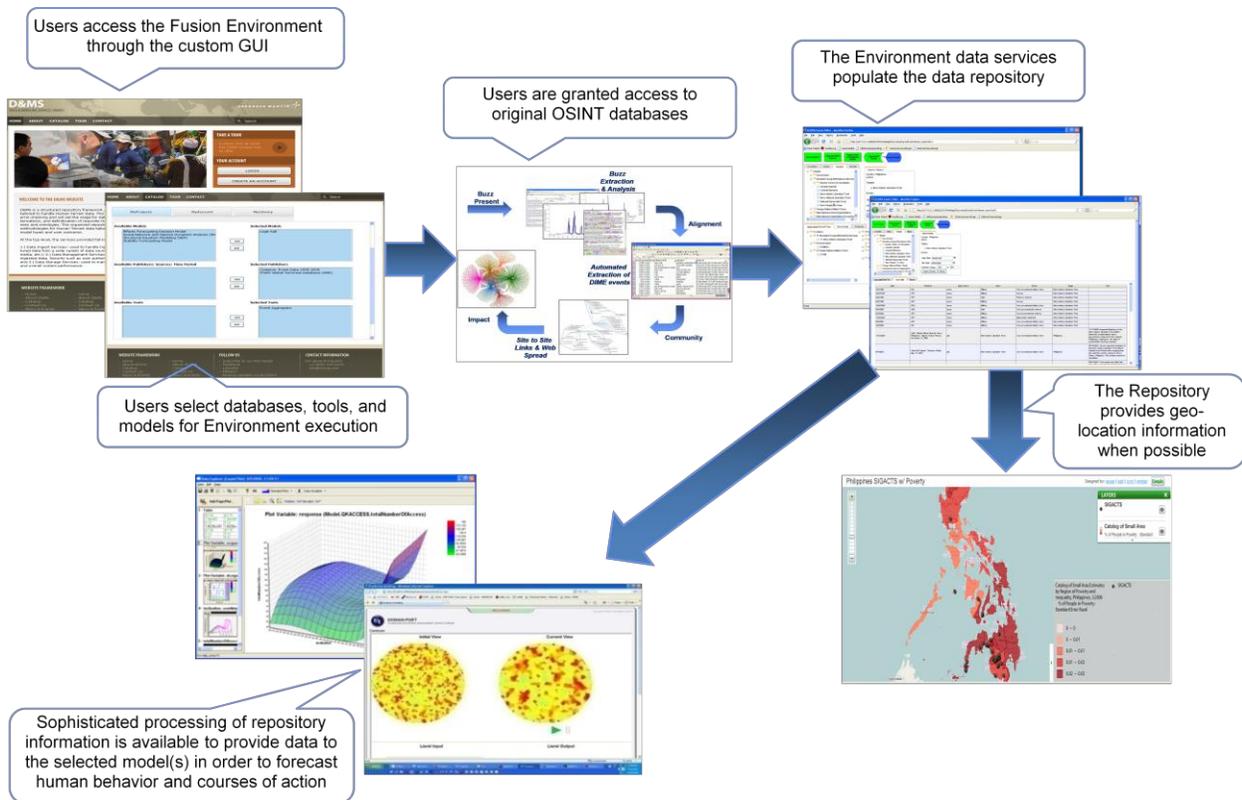


Figure 1. User Flow

## 2.1 Graphical User Interface (GUI)

The current Environment encompasses a custom designed GUI through which the User is able to build a tailored data and modeling services project, configured to specific requirements. The future Suite will further enable the User to 1) grant controlled user access to the custom project based on pre-determined security credentials; 2) view the results of previous model runs and various datasets; and 3) incorporate the use of additional analytical tools, such as visualization capabilities and exploration and optimization engines. This future work will thus expand the overall value of the GUI by enabling Users to access critical human terrain information drawn from dynamic environments.

The following process details the ways in which the current Environment provides enhanced behavioral data and modeling services.

The GUI serves as the key interface to the data services repository (hereafter referred to as “Repository”). The Repository ingests, manages, stores and processes data to create model sets according to a User-customized selection of data and modeling services:

User Actions in GUI	Environment Results
Selection of various databases to query (2.2).	Automated harvesting of datasets targeted by customized parameters.
Model selection from diverse list of options (2.5).	Datasets loaded into models.
Coding and aggregating tool selection from list of options (2.3).	Aggregation of desired datasets and models to form User’s custom services project.

**Table 1. GUI Process**

## 2.2 Human Terrain Databases

The Environment ingests databases from diverse sources to provide full-spectrum coverage of relevant information. For example, data queries currently access two dynamically evolving databases: 1) the Global Terrorism Database (GTD) developed and maintained by the National Consortium for the Study of Terrorism and Responses to Terrorism (START) at the University of Maryland; and 2) an Internal Lockheed Martin database containing thousands of stories related to terrorism and insurgent activity. These datasets are

event coded to support both geo-spatial display and model integration.

As online social networking evolves, OSINT will play an increasingly influential role in COA performance assessment and optimization. The future Suite will exploit this evolution by generating and integrating original databases comprised of online social networking data, as well as standard OSINT sources (*e.g.* newspaper feeds, structured databases, etc.). Future work will integrate innovative algorithms, which have been developed this year under Lockheed Martin research and development, to generate these original databases.

These algorithms currently govern existing technologies (*e.g.* crawling, tagging, agents, visualizations, etc.) to provide near real-time monitoring of the cyber domain via automated content targeting, harvesting and visualization. In 2009 experiments, the algorithms enabled successful, near real-time collection of online content that was released by active populations within the [cyber] human terrain. Metrics work validated that this harvesting method not only retrieves maximum relevant data while avoiding noise, which reduces the burden of information overload, but also keeps pace with the dynamic cyber environment. Metrics work further confirmed that the resultant algorithm-based visualizations, including trending analyses and social network mapping, are pertinent to intelligence analysts and information operations planners.

## 2.3 Data Services Repository

The next piece of the Environment is the data services repository, or Repository. Following database selection in the GUI, the Repository enables the User to target and organize datasets. Dataset selection is based on the following User-defined queries and parameters:

- Date ranges: Selected by the User.
- Groups of interest: Defined by the User according to geographic location, individual and group actors, targets and events. Geographic locations are entered by country but may be narrowed through geo-spatial display and advanced filtering (2.4). Actors may include both enemy (*e.g.* insurgents, terrorists, etc.) and friendly forces on whom the

User is interested in gathering information. Targets include groups, people, institutions and physical targets like infrastructure. Events may be defined as any geopolitical events, including physical attacks, elections, etc.

The advanced service oriented architectures (SOAs) are tailored to identify and harvest only those datasets that are targeted by the User-defined parameters.

The Repository aggregates and categorizes the datasets as event data. More specifically, the event data is organized in an aggregated event tree, through which the datasets are further categorized according to events, actors and targets. This unique format provides the User with 1) a list of the organized datasets; 2) query logic leading to Repository harvesting; 3) links between the coded events and the raw data from which they were derived; and 4) geo-spatial location of events via latitude and longitude. The Environment is primed for the addition of new services, including additional data sources and ingestion, processing and modeling tools.

This framework flexibility will expedite future work on the Services Suite.

## 2.4 Geospatial Display

Geo-locations for each dataset are triangulated within the Environment via a combination of GeoIQ, the geospatial engine from FortiusOne, and Repository coding. The Repository integrates original coding and event data with GeoIQ to generate the following information: date of the story, publisher, data source, city, actor, event and target. This integration enables movement from metadata to a listing of all datasets, accompanied by event coding for a high level view of each piece of information. The resulting data storage allows the User to manipulate the datasets for modeling and geospatial display. GeoIQ further enables graphical and census overlay displays of the datasets on pre-constructed maps, which supports examination of the event data in the context of other geospatial information (*e.g.* income by region, population, ethnicity, etc.). This geo-spatial coding aspect enables users to test on-the-fly hypotheses in order to initiate actions as required.

Figure 2. Database Flow

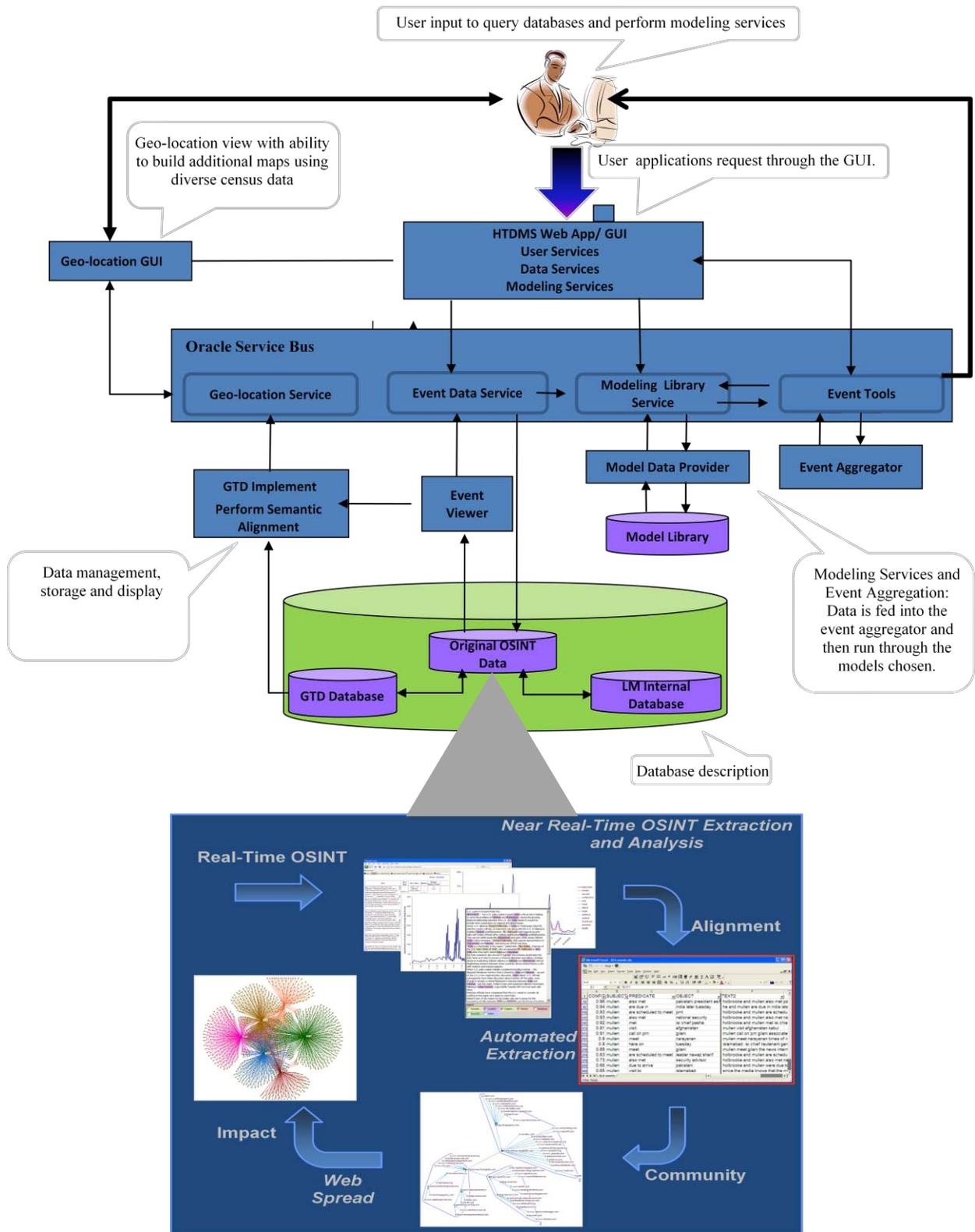


Figure 2.1. Suite Original OSINT Databases (above): The flow chart represents the innovative algorithms' capture of dynamic online social networking data and integration into Environment via data modeling and services.

## 2.5 Modeling Services

The Environment acts as an end-to-end integrator of data and modeling services by incorporating a range of models to meet User requirements within the human terrain spectrum. The existing GUI enables the User to select the models that are pertinent to the event(s) of interest (EOI). For example, the Environment currently incorporates numerous models to forecast enemy actions and population behaviors, as well as to assess User inputs. This combination of models supports course of action evaluation.

More specifically, forecasting of indigenous population responses and reactions to government and insurgent actions (*i.e.* targets and actors) can be tested from reactions to former events. Development work in 2009 resulted in the successful integration of such a model, which is able to relate data from previous interactions between targets and actors, in order to forecast future actions by various groups. Accuracy of this innovative model is achieved by increasing time periods, which narrows the forecasting gap. Coupled with custom datasets generated by the GUI and Repository, this model thus enables Users to forecast future personnel actions and refine decision making to counter negative audience reactions and enhance positive actions.

Moreover, the existing Environment is capable of supporting additional model types. The following models have been generated and/or modified through Lockheed Martin development work for future integration into the complete Services Suite:

- Statistical and agent-based models: The Social Network and Opinion Dynamics Analysis (SNODA) Model forecasts opinion propagation through social networks in response to an action plan. Forecasts of various groups' reactions are based on key leaders, social networks and previous actions undertaken by User-identified actors of interest. SNODA agents represent individuals within a population, each linking to a number of neighbor agents at varying distances. One set of controls is indirectly available to the User through specification of an action plan. Another set of controls is available to the modeler. The modeler controls allow flexibility in link structure and agent behavior. This flexibility enables tailoring according to varying

social structures in regions of interest. Moreover, each agent has an opinion, an uncertainty about one's opinion (*i.e.* the ability to change one's opinion and to accept a new opinion) and influencing factors that originate from one's opponents. Updates to an agent's opinion may be further affected by the opinions of neighbors, the current popular opinion, and/or a smaller network of key influential actors or leaders. A combination of math, physics and social science disciplines further enhances behavior model accuracy.

- Decision models: Lockheed Martin's original decision model supports action plan development aimed at influencing selected audiences. The model framework relates stakeholders' strategic intent, desired effects, influencing actions and additional inputs to arrive at quantitative evaluations of proposed alternatives. The resultant value models thus provide a rationale for identifying preferred plans and/or quantitative prioritizations of actions.
- Linear regression and structural equation models (SEM): Lockheed Martin's unique SEM takes the form of a linear regression equation, in which the variables are latent or unobservable. Underlying constructs include knowledge, beliefs and attitudes that motivate actions. The SEM consists of an explanatory or predictive set of equations to estimate measures of effect on a receiving audience (*i.e.* the population or intended group) in response to an action plan that is tailored to a precipitating event. The model is thus able to forecast general population trends and human actions.

As future work is conducted to transform the existing Environment into a complete Services Suite, the aforementioned models will be integrated to support accurate representations of dynamic human terrain scenarios, in diverse regions of interest and at different levels (*i.e.* strategic, operational and tactical) of conventional and irregular warfare. The underlying framework of the Environment is agnostic to the modeling paradigm and model execution framework. Sophisticated data processing architecture enables the

repository data to be pre-processed in nearly infinite ways in order to support the various models in the overall Environment and future Suite.

### 3. Future Work

Future Suite work will build on existing data services to incorporate additional data sources – both external and original – and to improve capabilities to ingest, manage, store and process the data. This refinement will include expansions of, and improvements to, the data query and data source filter parameters. Future work will likewise enhance modeling services, with a focus on improved data access flexibility, processing and model data formatting. Lockheed Martin's innovative models (*e.g.* SEM and SNODA) will be further refined and incorporated into the current Environment. These model additions, coupled with the exploitation of additional data sources and processing methods, will greatly improve and enhance the existing Environment. Future work will continue to take strong consideration of social sciences and behavioral reasoning, leading to a powerful and astute Services Suite.

### 4. Conclusion

Our Lockheed Martin Services Suite will lead full-spectrum data services and behavioral modeling. The current Environment's GUI and Repository expand data services through precise entity extraction and metadata filtering. Moreover, that behavioral data is accurately modeled with innovative processes and end-to-end integration of math, physics and social science based models. Collectively, the Environment ingests, manages, stores and models precise behavioral characteristics of selected audiences and indigenous populations.

The future Suite will further integrate original Lockheed Martin algorithms and models to track, harvest and represent near real-time online communities of interest. The complete Services Suite will thus continue to incorporate social sciences into its modeling piece by moving beyond standard computational models. Similar to its data services, its modeling will continue to take into consideration relationships, cultures and history to accurately reflect human dynamics.

### 5. References

Barber, Daniel and Nicholson, Denise, "Intelligent Resource Operational Network (IRON) for Cultural Modeling." BRIMS Conference. March 2009.

Global Terrorism Database, START. Accessed on October 2009.

Hayden, Michael, "Director's Remarks at the DNI Open Source Conference 2008." DNI Open Source Conference. Washington D.C. 2008.

Kilculen, David, "Fundamentals of Company Level Counter-insurgency." Washington D.C. 2006.

Pallaris, Chris, "Open Source Intelligence: A Strategic Enabler of National Security." Center for Security Studies. 2008.

### 6. Related Work

Gavrilis, Maj. James A., "Army Must Address Irregular Warfare Needs." National Defense Magazine. March 2006.

Hu, D., Kaza S., and Chen, H., "Identifying Significant Facilitators of Dark Network Evolution." Journal of the American Society for Information Science and Technology 60.4 (2009): 655-665.

Hung, Victor C. and Gonzalez, Avelino J., "Towards A Human Behavior Model Based On Instincts." BRIMS Conference. University of Central Florida, School of Electrical Engineering and Computer Science. 2007.

Office of the Under Secretary of Defense on Acquisition, Technology and Logistics. "Understanding Human Dynamics: Report of the Defense Science Task Board on Understanding Human Dynamics." March 2009.

Petraeus, LTG David and Mattis, LTG James, "Counterinsurgency Field Manual 3-24." USMC. December 2006.

Wu, Wenshang, Yu, Clement, Doan, AnHai and Meng, Weiyi, "An Interactive Clustering-based Approach to Integrating Source Query Interfaces on the Deep Web." Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data. 2004.

## 7. Author Biographies

TINA H. CHAU is an Intelligence Analyst at Lockheed Martin. She serves as an analytical representative and jihadist subject matter expert for the Enterprise Integration Group's Human Terrain Team. She holds an M.A. in International Relations, with a concentration in Security Studies, from Boston University and was awarded the Graduate Prize for her academic achievements and thesis: *Safeguarding U.S. Security Against Unauthorized Disclosures of Classified Intelligence*.

ALEXANDER P. MOORE is currently a Systems Engineer at Lockheed Martin working on all Human Terrain IRAD's for the company's Enterprise Integration Group. Alex is also a Captain in the U.S. Army Reserve, and serves as the Brigade Assistant S-2 for the 304<sup>th</sup> Civil Affairs BDE. Before coming to Lockheed Alex served as an Armor Officer in the Army on Active Duty. He has served two combat tours in Iraq, in the role of mechanized infantry and tank platoon leader in 2004, and as a combat advisor in 2007-8. His awards include the Bronze Star, Purple Heart, Army Commendation Medal, and the Order of Saint George. Alex holds a B.S. in Systems Engineering from the United States Military Academy.

RICK MULLIKIN is currently with Lockheed Martin working several R&D projects related to behavioral modeling. Dr. Mullikin holds a PhD in Information Science with a focus on Artificial Intelligence from the Claremont Graduate School, an MBA in International Marketing from Loyola Marymount, and a BS in Electrical Engineering from the University of Maryland.

JANET E. WEDGWOOD is currently with Lockheed Martin providing decision support to systems integrating multi-paradigm models using the DIAS framework. Under her leadership in Human Social and Cultural Behavior investigations, the core architecture for the proposed Data and Modeling Services Suite has evolved into a highly modular experimentation. Ms. Wedgwood earned her BSEE from Rensselaer Polytechnic Institute and her MSEE from Stanford University.

# Dynamic Data and Modeling Services Suite

Tina H. Chau

Alexander P. Moore

Richard L. Mullikin

Janet E. Wedgwood

Lockheed Martin Corporation

720 Vandenberg Drive

King of Prussia, PA 19406

571-313-6605, 610-354-7590, 917-497-0424, 856-792-9879

tina.h.chau@lmco.com, alexander.p.moore@lmco.com, rick.mullikin@lmco.com, janet.e.wedgwood@lmco.com

## Keywords:

counter-insurgency, data harvesting, data services, data query, innovation, integration, proof-of-concept fusion environment, services suite, modeling services, geo-location, social sciences

## 1. Research Objective

The objective of our research is to translate our proof-of-concept fusion environment – currently feeding its predictive models with comprehensive human terrain data from dynamic sources – into a complete Dynamic Data and Modeling Services Suite that is tailored for use by Counter-insurgency (COIN) Operations commanders, military advisors and intelligence analysts.

At the core of COIN Operations is the mission to win the hearts and minds of the population. Full-spectrum mission planning thus requires an actionable consideration of social dynamics and core social sciences, collectively referred to as the human terrain (*i.e.* indigenous populations’ behaviors, motives, foundational thoughts and beliefs, etc.). This requirement raises two primary technical challenges:

1. Modeling Dynamic Behavioral Environments: COIN modeling services must support varied behavioral and predictive models to accommodate for differences in population compositions, actions, beliefs and motives.
2. Operationalizing Data Services: Dynamic socio-cultural models require vast and timely intelligence harvests, which is both challenging and time consuming. The evolving nature of the cyber domain renders online content, while of increasing value for near real-time behavioral data, difficult to collect, manage and store for operational use.

This project leverages ongoing research and development – including the integration of existing technologies and

innovative coding, algorithms, modeling and theoretical methodologies – to form a data and modeling services solution to the aforementioned challenges. This abstract outlines presentation material on the existing proof-of-concept fusion environment (hereafter referred to as “Environment), as well as the future work that will form the complete services suite.

## 2. Fusion Environment and Services Suite

The Dynamic Data and Modeling Services Suite will build on the proof-of-concept fusion environment for end-to-end systems integration of human terrain datasets and modeling services. The Environment’s GUI serves as the key interface to the dynamic data and modeling services.

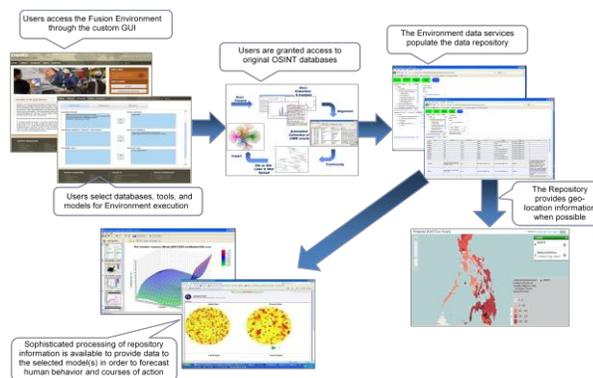
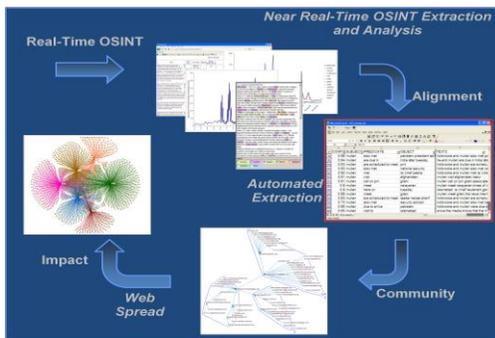


Figure 1. User Flow

The data services repository ingests, manages, stores and processes data to create User-customized model sets. Databases are ingested from diverse and dynamic human terrain sources, (*e.g.* Global Terrorism Database at the University of Maryland, Lockheed Martin internal

database, etc.) to provide full-spectrum coverage. The repository, supported by advanced service oriented architectures, aggregates and organizes datasets according to User-defined queries (*e.g.* date range, location, actors, targets, events, etc.). The User is furnished with organized dataset lists, links between coded events and raw data, query logic, and geo-spatial event locations. Geo-locations for each dataset are triangulated via a combination of GeoIQ, the geospatial engine from FortiusOne, and original coding for graphical and census overlay displays of the datasets on preconstructed maps.

The Environment is primed for additional data services. The future Suite will exploit the evolution of the Internet by generating original databases comprised of online social networking data, standard news feeds, structured databases, etc. Automated harvesting of near real-time behavioral data will be achieved by integrating innovative algorithms, which have been successfully developed and tested under Lockheed Martin, into the Services Suite.



**Figure 2. Suite Original OSINT Databases:** algorithms' capture of dynamic online social networking data.

The GUI also enables model selection that is pertinent to event(s) of interest. The Environment supports numerous models – including innovative and existing statistical, agent-based, decision, linear regression and structural equation models – to forecast enemy actions and population behaviors, as well as to assess User inputs.

As future work is conducted to transform the existing Environment into a complete Services Suite, existing and new models will be fully integrated into the underlying framework, which is agnostic to the modeling paradigm and model execution framework. Sophisticated data processing architecture enables the repository data to be pre-processed in nearly infinite ways to support these modeling service additions.

It is our assertion that data and modeling service additions to our proof-of-concept fusion environment will lead to a powerful and astute Services Suite that is tailored to address the challenges facing COIN operators.

### 3. References

Barber, Daniel and Nicholson, Denise, “Intelligent Resource Operational Network (IRON) for Cultural Modeling.” BRIMS Conference. March 2009.

Global Terrorism Database, START. Accessed on October 2009.

Hayden, Michael, “Director’s Remarks at the DNI Open Source Conference 2008.” DNI Open Source Conference. Washington D.C. 2008.

Hu, D., Kaza S., and Chen, H., "Identifying Significant Facilitators of Dark Network Evolution." Journal of the American Society for Information Science and Technology 60.4 (2009): 655-665.

Hung, Victor C. and Gonzalez, Avelino J., “Towards A Human Behavior Model Based On Instincts.” BRIMS Conference. University of Central Florida, School of Electrical Engineering and Computer Science. March 2007.

Petraeus, LTG David and Mattis, LTG James, “Counterinsurgency Field Manual 3-24.” USMC. December 2006.

### 4. Author Biographies

TINA H. CHAU is an Intelligence Analyst at Lockheed Martin, serving as an analytical representative and jihadist subject matter expert for Human Terrain efforts.

ALEXANDER P. MOORE is a Systems Engineer at Lockheed Martin working on Human Terrain. He is also a Captain in the U.S. Army Reserve, and serves as the Brigade Assistant S-2 for the 304<sup>th</sup> Civil Affairs BDE.

RICK MULLIKIN is currently with Lockheed Martin working several behavioral modeling projects and holds a PhD in Information Science, with a focus on Artificial Intelligence from the Claremont Graduate School.

JANET E. WEDGWOOD is currently with Lockheed Martin and is a leader in the development of the core architecture for the proposed Dynamic Data and Modeling Services Suite.

# Tactical Behavior Composition

*Evan C. Clark*

*Joel C. Eichelberger*

Physical Science Laboratory

New Mexico State University

Las Cruces, NM 88003

575-496-9915

research@evanclark.net, eichel@psl.nmsu.edu

*Jeffrey A. Smith*

U.S. Army Research Laboratory

Survivability/Lethality Analysis Directorate

White Sands Missile Range, NM 88002-5513

575-678-1332

jeffrey.a.smith@us.army.mil

Keywords:

tactics description language, commander agent, tactical decision making

**ABSTRACT:** *Behavior composition for computer generated forces is a technique that facilitates the creation and validation of agent behavior. It refers to the practice of creating reusable primitives that can be combined to construct new complex agent behaviors. Research in behavior composition has often focused on the use of procedural primitives. This paper discusses a framework for commander agent behavior composition that includes not only procedural primitives, but also those representing tactical concepts such as spatial relationships, subordinate coordination, terrain analysis, firepower and mobility. These primitives give the domain expert the ability to influence the manner in which tactical decisions are made. These primitives are elements of a tactics description language called Tesla. Using the Tesla language, a tactical behavior expert composes tactic templates which can later be used by commander agents in course of action development and to solve tactical problems.*

## 1. Introduction

Both military modeling and simulation and commercial gaming require software agents that can solve tactical problems. For both industries, realism and immersion are enhanced when commander agents can dynamically adapt to tactical challenges in a reasonable way. However, because the current level of artificial intelligence technology does not permit a software agent to derive its tactical behavior from first principles, some medium is required to facilitate the transferral of tactical expertise from domain experts to software agents.

One technique that has been developed to facilitate this transferral of domain expertise is behavior composition. This technique has been used to allow a domain expert to directly configure the actions an agent will undertake.

This paper describes an approach to agent behavior configuration that extends the number of things a domain expert can specify, giving him or her a greater influence not only on what actions an agent performs but also on how it performs them.

Section 2 motivates this approach by discussing the advantages behavior composition systems already enjoy. Section 3 gives a general overview of the Tesla language and its use in agent configuration. Section 4 provides an example of using this approach. Section 5 describes Tesla's composition primitives. Section 6 discusses the implications of this approach on testing and validation.

## 2. Background

In the context of commander agent configuration, behavior composition refers to the practice of combining reusable primitives to construct new complex agent behaviors. What constitutes a primitive may vary by echelon and from system to system, but in all cases, a primitive refers to functionality implemented in source code and packaged up so as to be available to an editor application or scripting engine.

Behavior composition is used as an alternative to specifying all agent behavior in code, providing more productive roles for software engineers and domain experts alike. In such an arrangement, software engineers develop behavior primitives rather than *ad*

*hoc* complex behaviors. It is the nature of these primitives to be modular, encapsulated and reusable (Fu, 2003) (Reece, 2004). Modular and encapsulated code is easier to develop and verify, while code reuse engenders an overall increase in productivity. Engineer productivity is also increased when the time spent soliciting requirements from domain experts is limited to a finite set of primitives rather than a larger set of more complex behaviors.

Domain expert productivity is also benefited by behavior composition, which allows them to use a language directly relevant to their domain. Further, when equipped with an appropriate tool set, the reliance on software developers is dramatically reduced (Summers, 2004). This has the added benefit of increasing the overall productivity of teams that are limited by software engineer availability.

Perhaps the strongest argument in favor of composition systems is that they facilitate model verification and validation. They do this not only because access is extended to those who lack training in software development, but because when behaviors are implemented in code the domain knowledge so represented is mingled with and obscured by code that fulfills other roles.

Behavior composition systems generally fall into one of two broad categories. The first category, knowledge-based systems (also called rule-based systems or embedded expert systems), is characterized by the use of some form of finite state machine (FSM). Examples of this approach can be found in: Obst (2001), Gilgenbach (2006), Fu (2003), Reece (2004), and Kosecka (1997). States in the FSM represent different things in different systems. They can correspond to activities, goals, or behaviors, but in each case, they devolve into actions taken by the unit the agent commands. Typically, only one state may be active at a time. Transitions between states are governed by Boolean expressions whose fluents reflect some bit of the agent's knowledge or some environmental condition. Figure 1 shows an example of FSM-based behavior composition for tactical reasoning.

In order to be used in tactical decision making, there must be a place for tactical concepts in any given knowledge-based system. Some of these concepts, such as time and the ordering of events and actions, are expressed naturally by the arrangement of primitives in an FSM. But other tactical concepts, such as spatial relationships, subunit coordination, cover and concealment, positional analysis and attrition, must be captured in source code in either the actions associated with states or in the fluents' evaluation functions.

Goal-based systems are another broad category into which many behavior composition systems fall. In these systems, a goal condition or optimization function is specified external to the agent. The agent performs a search of some kind to discover a sequence of actions that meets its assigned objective. This search occurs at execution time and gives the agent the ability to dynamically adapt to its particular circumstances. In goal-based systems, domain experts ensure that plan inputs such as atomic actions and their pre- and post-conditions are appropriate to the domain rather than directly specifying action sequences or flow charts. In this sense, the act of composition is shared between the domain expert and an automated planner. Zhang (2001) and Pittman (2008) are examples of this approach.

As with knowledge-based systems, goal-based systems also have the ability to aid in tactical reasoning. But as with knowledge-based systems, apart from temporal relationships and the ordering of events and actions, tactical reasoning must be done in source code.

Both knowledge- and goal-based systems may be termed procedural composition systems, because they focus on agent actions and the manner in which sequences of actions are chosen.

It is the purpose of this paper to assert that non-procedural primitives can also be used in behavior composition and that the gains in accessibility and productivity made possible by procedural composition systems can be extended by increasing the number and kinds of primitives made available to domain experts.

### 3. Overview

This approach utilizes both procedural and non-procedural composition. To do so, it uses a tactics description language called Tesla to capture tactical concepts and convey them from a human expert to a software agent in a format that is accessible to both.

As depicted in Figure 2, the domain expert uses an editor to create a tactic template. In this template is encoded enough of a tactic's underlying concepts that an agent can later use it to apply the tactic to its particular situation.

Figure 3 shows a simple tactic template displayed in the Tesla editor. In this tactic, the commander agent directs a single subordinate unit to move to a destination while avoiding observation by all known enemies.

The Tesla language is part graphical and part textual. The graphical part is the sketch view which

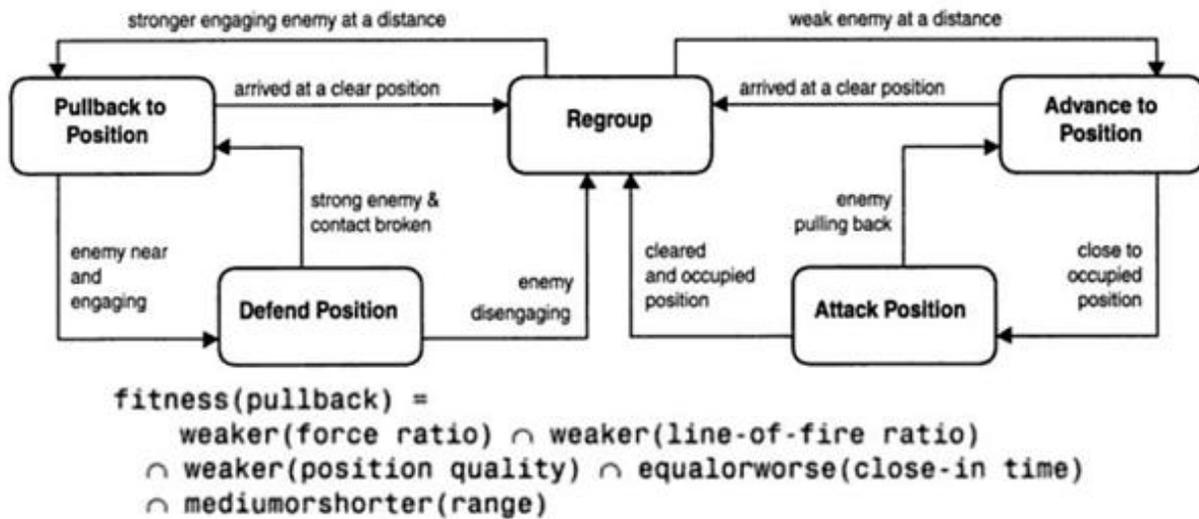


Figure 1: Tactical behavior in a knowledge-based composition system (Gilgenbach, 2006)

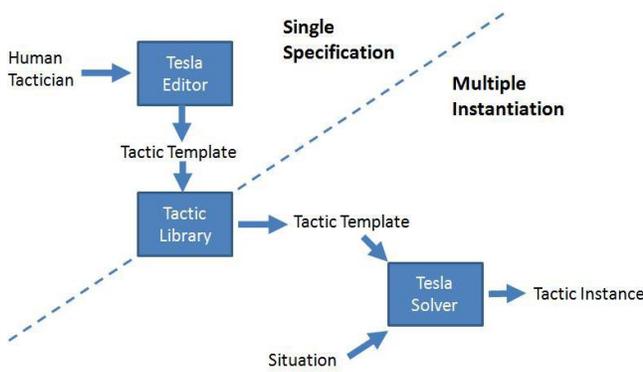


Figure 2: Tesla use case

corresponds roughly to a course of action sketch. Found in the sketch view are 1) all entities (including relevant control measures) that take part in the tactic and 2) the constraints that define how entities and control measures may be converted from abstract concepts into instances of a particular situation.

The textual part of a template is the execution matrix. As with the sketch view, its semantics and syntax are borrowed from military course of action development (FM 3-90, 2001). Both parts of the language are described in more detail below.

### 3.1 Nominals

One of the principal elements of the Tesla language is the *nominal*. In grammar, a nominal is a noun phrase. In the Tesla language, a nominal is a unit, location or object on the battlefield.

The example in figure 3 contains four nominals. Starting on the left and proceeding in a clockwise manner, they are: a subunit (A), a generic direction of attack (DA1), a checkpoint (CP1) and an enemy unit (ENY1).

Nominal icons come mainly from US military symbology (FM 1-02, 2004). Note that the subunit and enemy unit symbols do not have echelon designators, because in a template they can refer to any echelon.

### 3.2 Constraints

In the Tesla language, constraints modify nominals. In this respect, they serve as adjective phrases indicating what kind of object the nominal should be. Above the sketch view in figure 3 is the constraint glyph bar.

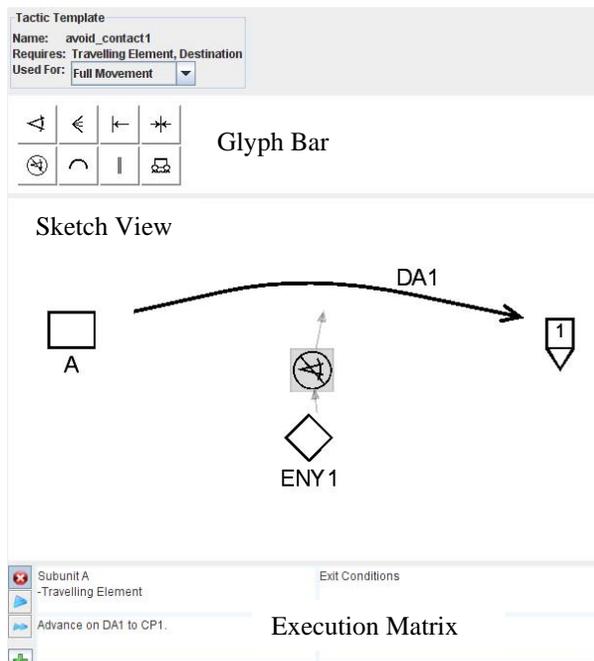


Figure 3: Simple tactic template

Constraints are chosen from this glyph bar, configured and added to the nominals they modify.

The template in figure 3 contains a single constraint. This constraint points from ENY1 to DA1. It is read to mean, "Constrain DA1 such that it is concealed from all enemies identified as belonging to ENY1."

The natural language expression of a constraint can sometimes be ambiguous. To remove this ambiguity, each constraint has one or more associated *location metrics*. A location metric contains the algorithmic interpretation of the constraint that the domain expert wants to use in the tactic. The concealment constraint from figure 3, for example, can be alternately interpreted as meaning the absence of optical line of sight or as referring to an estimated probability of detection being below some threshold. Each interpretation has a corresponding location metric that can be chosen for the constraint. Other interpretations would also be possible.

### 3.3 Execution matrix

The Tesla execution matrix is conceptually similar to the execution matrices used in military course of action development. It contains the procedural parts of the tactic template. In it, each subunit has a column, and each phase in the course of action has a row. Every cell in the execution matrix contains instructions for that column's subunit. Cells in a row are executed simultaneously. In the Tesla language, instructions are composed of a task word and some number of modifying phrases. These modifying phrases are task word specific and generally relate to one or more nominals from the sketch view.

The execution matrix from figure 3 has a single subunit and a single phase. Its instruction has the task word, *Advance*, with the modifying phrases, *on DA1* and *to CP1*.

### 3.4 Resolution

Template resolution is the process by which a template is applied to the agent's particular situation. It consists of mapping each nominal to an appropriate counterpart in the agent's environment. In the template from figure 3, for example, subunit A would be mapped to one or more of the agent's subordinates; DA1 would be mapped to a concealed route; CP1 would be mapped to a location; and ENY1 would be mapped to a group of known or suspected hostile units.

In order to ensure that a proper mapping is found, the domain expert assigns and configures a so-called *nominal resolver* to each nominal in the template.

Each type of nominal has one or more nominal resolvers to choose from, and each nominal resolver is responsible for making sure that a mapping is found that obeys each of the constraints placed on the nominal.

Once each nominal has been resolved, the instructions in the execution matrix refer to concrete locations and objects rather than abstractions. At this stage, these instructions can be used to generate maneuver and fire orders for subordinates.

## 4. Example Tactic

To illustrate how a tactic template works, this section examines an implementation of the fix-flank tactic. In this tactic, a force is divided into fixing and flanking elements. The fixing element engages the enemy unit and seeks to pin it in place. The flanking element takes a concealed route to a position of advantage from which it can surprise and flank the enemy. Parts of this template are shown in figures 4 and 5.

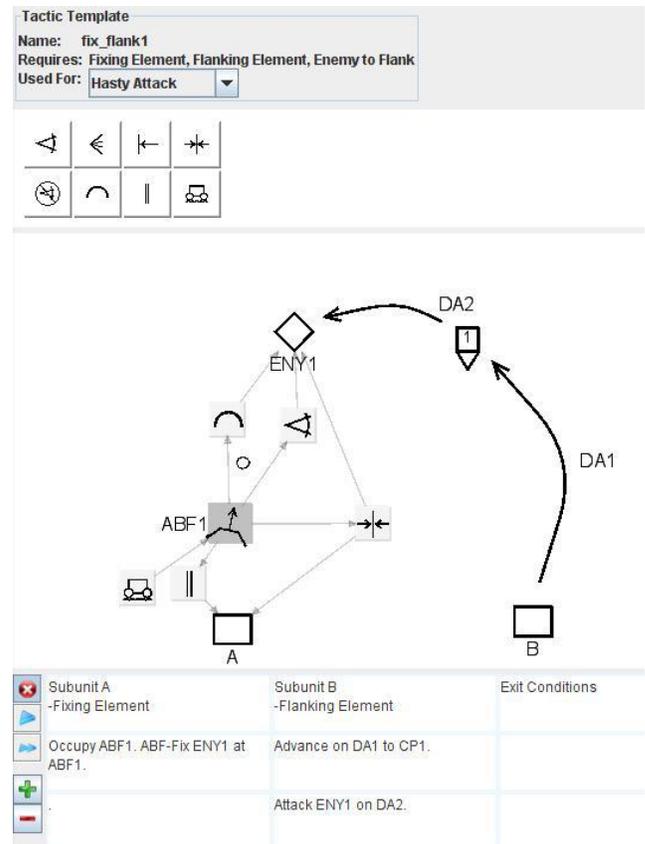


Figure 4: Fix-Flank tactic template

In the fix-flank template, subunit A is the fixing element. It moves to ABF1, an attack by fire position, from which it can engage ENY1. In order for the solver to select a suitable location for ABF1, five constraints are supplied that indicate the properties that

ABF1 must have in order to play its role as a fixing position in this tactic. In the Tesla editor, when a nominal is selected, its constraints become visible. Figure 4 shows the fix-flank template with ABF1 selected. Starting above ABF1 and proceeding in a clockwise direction, its constraints are interpreted as meaning:

- A unit at ABF1 should have cover from ENY1.
- A unit occupying ABF1 should be able to see ENY1.
- ABF1 should be roughly between subunit A's starting position and ENY1.
- ABF1 should be somewhat near subunit A's starting position.
- ABF1 should be on trafficable terrain.

The other nominals from this template also have constraints specified in a similar manner.

Figure 5 shows the user interface for the nominal resolver that was chosen for ABF1. This type of nominal resolver is called a *location scorer resolver* because it uses the constraints' location metrics to score and rank candidate locations. In the location scorer resolver, the domain expert chooses whether to use constraints as a basis for excluding locations as candidates or to use them as contributing to a location's score. As seen in the first two rows of figure 5, only locations with line of sight to all of ENY1 and at least some cover from ENY1 are considered as candidates.

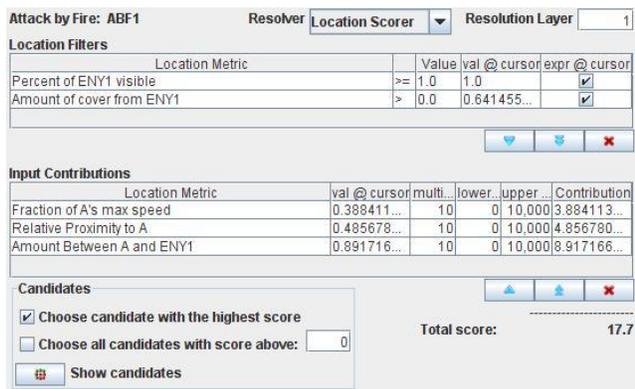


Figure 5: Location scorer resolver configuring ABF1

Location metrics create values that range from zero to one, making them suitable for nominal resolvers that use fuzzy logic. This property also makes it easy to visualize how location metrics operate. Figure 7 shows heat maps for the five location metrics used by the ABF1 nominal resolver.

To apply the template to a situation, the Tesla solver iterates over each nominal and invokes its nominal resolver. The order of resolution matters, since the outcome of one mapping can be used as an input into a

subsequent nominal resolver's location metric. In the fix-flank example, A, B and ENY1 are template inputs, meaning that in order to use the template, the agent must supply mappings for these three nominals. The other nominals, ABF1, DA1, CP1 and DA2 are all resolved using constraints, location metrics and nominal resolvers as configured by the template developer.

Figure 6 shows the fix-flank template resolved in two different situations. The top situation is the same as the one from figure 7.

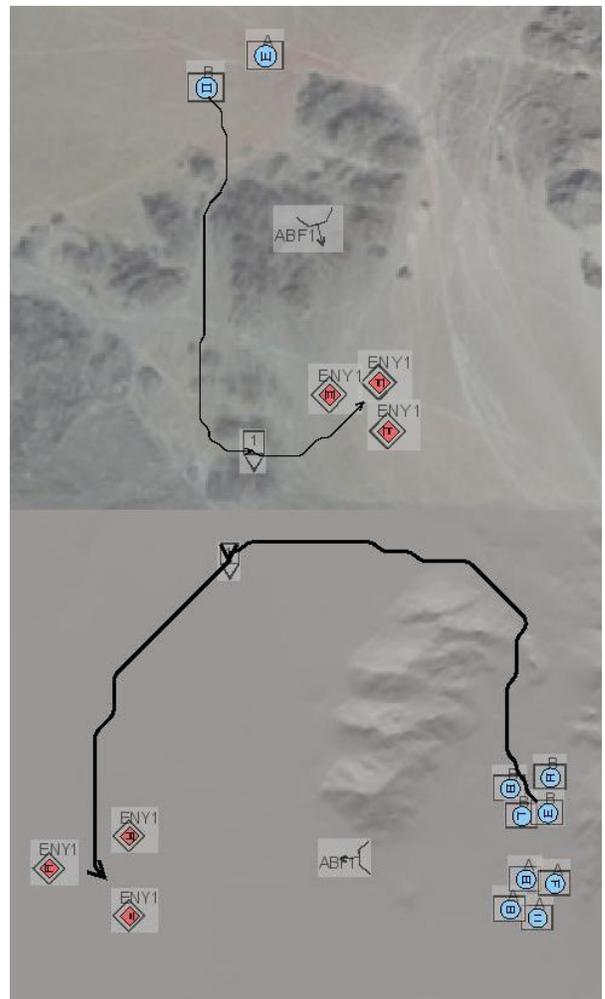


Figure 6: Two resolutions of the fix-flank tactic

## 5. Tesla Composition Primitives

Each type of behavior primitive in a composition system represents a kind of functionality available to the domain expert for manipulation and validation. The behavior primitive types available indicate the points where the system is easily extensible.

This section discusses some of the composition primitives available to a domain expert in Tesla.

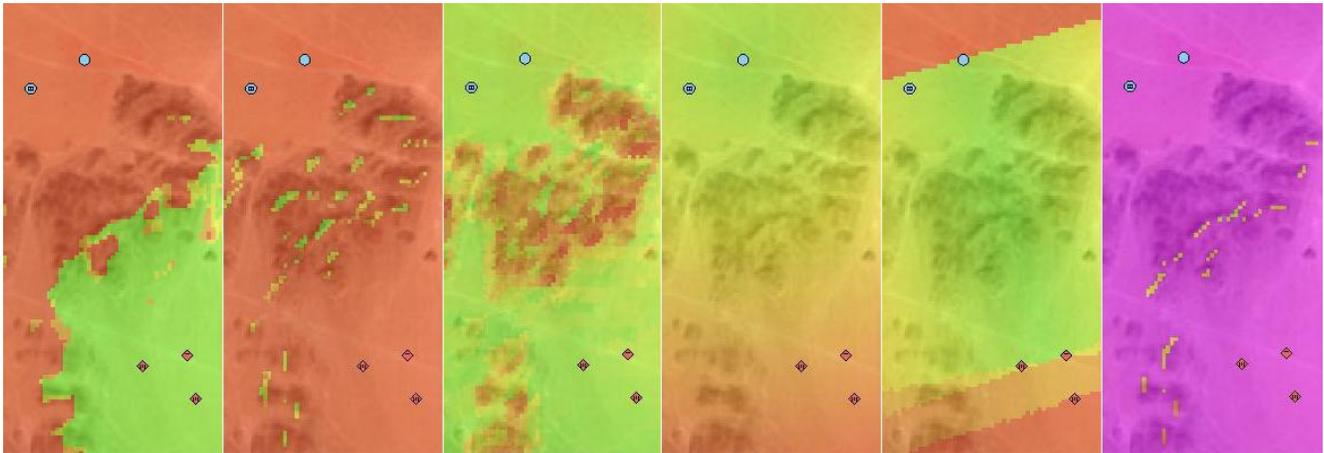


Figure 7: Location metrics used in the fix-flank example. From left to right, they are: Percent visible, Amount cover, Fraction of max speed, Relative proximity and Amount between. The last panel shows the composite scores as calculated by the location scorer resolver. In each panel, green indicates a metric value of one, while red indicates a metric value of zero. In the last panel, magenta indicates a location that has been filtered out and not considered as a candidate.

### 5.1 Nominals

The number of kinds of battlefield objects that can be represented by the Tesla language is increased by adding more nominals. Nominal types currently supported in the language are:

- subunits i.e. a subordinate of the commander agent
- enemy units
- locations - e.g. point target, support by fire position, point of interest
- line segments - e.g. linear target, lane
- segmented lines - e.g. unit border, phase line
- routes - e.g. avenue of approach, direction of attack
- areas - e.g. objective, free fire zone

### 5.2 Constraints and location metrics

Constraints and location metrics represent the most basic tactical concepts that can be expressed in the Tesla language. They provide the building blocks for terrain and positional analysis and reasoning over firepower, mobility, communications and sensing. As domain experts develop templates for which existing constraints and locations metrics do not suffice, new ones can be requested of and implemented by a software engineering team.

### 5.3 Nominal resolvers

The algorithms found in nominal resolvers are themselves behavior primitives. Nominal resolvers currently exist for location selection, enemy classification, route planning and template input handling. More can be built and added to the framework as necessary.

### 5.4 Verbs and verb modifiers

Similar to other systems, these procedural primitives map to actions that must be individually implemented in source code. But these actions should be much simpler to implement because they are for individual subordinates and not for the unit as a whole. Subunit coordination is done in the template editor rather than by a software engineer.

### 5.5 Expressivity

The Tesla language allows for the representation of sophisticated tactical concepts. Its primitives can be used to design coordinated attacks, plan ambushes, identify kill sacks and areas of overlapping fire, trace infiltration routes, find overwatch positions, plan defensive positions and so forth.

A reverse slope defense is one that keeps the defender concealed from the attacker until the attacker has approached to close range (such as by defending the reverse side of a hill). This allows the defender to neutralize any weapon range overmatch the attacker might have by forcing the engagement to occur at close range. This concept can be included in a tactic by using and giving proper weights to direct fire constraints. Conversely, an agent can be configured to capitalize on a weapon range overmatch by applying different weights to those same constraints.

Some tactical concepts have fine distinctions that can be difficult for a software agent to make. For example, three different tasks, attack, suppress and fix, all involve seeking advantageous terrain and engaging the enemy. All three are successful if the enemy is destroyed, but the manner in which the tasks are

executed is sometimes different. For attack, the desired effect is the destruction of the enemy. For suppress, the desired effect is to make enemy fires less effective. For fix, the desired effect is to prevent enemy movement. Because fix and suppress tasks have more relaxed goals, troops are permitted a more defensive posture when executing these tasks. These distinctions between the attack, suppress and fix tasks can be realized through judicious use of direct fire and line of sight constraints on ABF and SBF nominals.

The expressivity of the Tesla language gives commander agents the ability to reason over sophisticated tactical concepts. This gives an agent the ability to interpret changes to its tactical situation and dynamically adapt when necessary. This adaptability increases model realism. It also makes scenario development less time consuming, because it decreases the number of eventualities that have to be explicitly scripted for.

## 6. Iterative Refinement and Behavior Validation

Figure 8 shows the Tesla editor application. It is divided into a template editor and a situation editor. The template editor allows the user to create and view tactic templates. The situation editor is where the template is tested. It allows the user to create a number of situations against which to test the template.

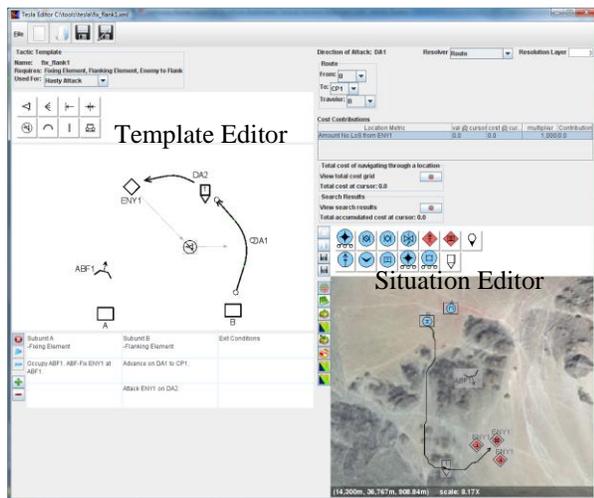


Figure 8: Tesla Editor

The ability to quickly test a template has a number of significant implications. First, it allows template development to be a process of iterative refinement. The domain expert creates a template and a situation and then invokes the solver to see how it interprets the template. If there are unexpected results, debugging is facilitated by overlays showing the contributions of individual parts of the template. These overlays, such

as the heat maps from figure 7, are displayed in the situation editor. As problems are worked out, the domain expert creates more situations and tests the template against them as well. The process continues until the user is confident that the template is flexible enough to be applicable in many situations.

This same functionality is useful in behavior validation. Rather than waiting to validate a template until the agent can use it in a fully configured simulation, the validating authority can see how a tactic is used in a number of situations. If applicable, the template can be checked for validity at different echelons as well. These situations are saved with the template library and can be invoked again later, allowing the template library to be separately validated at any time

The easy and full access to this aspect of agent behavior is a significant aid to the validation process.

## 7. Conclusion

Although the Tesla language shares similarities with other composition systems, it is qualitatively different from many of them. In the military context, the decisions of commanders are more often manifest through communication and the actions of their subordinates than through their own shooting, moving and sensing. For a commander agent to develop a course of action for its subordinates requires it to reason about what it knows about friendly and enemy force positions, composition and capability. As a tool for commander agent configuration, Tesla encodes formulae for the deployment of maneuver forces rather than encoding procedures for equipment operation.

The Tesla language, editor and solver constitute part of a kind of knowledge-based system. It does not compete with automated planners or systems that use FSMs, since they solve different kinds of problems. Procedural composition systems are primarily concerned with determining *what* to do, whereas this approach seeks to identify *how* something should be done. Rather than competing with procedural composition systems, this approach should be viewed as complementary. When equipped with the appropriate metadata, these templates can serve as robust primitives in a higher-level composition system. In particular, they can provide a mechanism for managing subordinate coordination, which can be problematic for a purely procedural system.

The approach described in this paper aids in the specification of commander agent behavior. It is offered as a way to extend the benefits of composition systems to more functionality than is exposed in purely

procedural systems. Doing so facilitates validation and verification by giving domain experts more direct access to agent behavior, enables a more cost effective division of labor between domain experts and software engineers and provides a highly extensible framework for configuring tactical agent behavior.

## 8. References

- Reece, D., McCormack, R., and Zhang, J., (2004). A Case-Based Reasoning Tool for Composing Behaviors for Computer Generated Forces. *In Proceedings of Behavior Representation in Modeling and Simulation.*
- Fu, D., Houlette, R., and Jensen, R., (2003). A Visual Environment for Rapid Behavior Definition. *In Proceedings of Behavior Representation in Modeling and Simulation.*
- Gilgenbach, M., McIntosh, T., "A Flexible AI System through Behavior Compositing," *AI Game Programming Wisdom 3*, pp 251-264, Charles River Media, 2006.
- Obst, O., (2001). Specifying Rational Agents with Statecharts and Utility Functions.
- Howden, N., Curmi, J., Heinze, C., Goss, S., and Murphey, G., (2003). Operational Knowledge Representation: Behavior Capture Modelling and Verification. *In Proceedings of the Eighth International Conference on Simulation Technology and Training.*
- Löttsch, M., Bach, J., Burkhard H., Jüngel, M., (2004). Designing Agent Behavior with the Extensible Agent Behavior Specification Language XABSL. *In Proceedings 7th International Workshop on RoboCup*, 114-124.
- Kosecka, J., Christensen, H., (1997). Experiments in Behavior Composition. *Journal of Robotics and Autonomous Systems*, vol. 19,287-298.
- Zhang, Y., Biggers, K., Sheetal, R., Sepulvado, D., Yen, J., Loerger, T., (2001). A distributed intelligent agent architecture for simulating aggregate-level behavior and interactions on the battlefield. *In Proceedings 5th Multi-Conference on Systemics, Cybematics, and Informatics*, 58-63.
- Pittman, D., "Command Hierarchies Using Goal-Oriented Action Planning," *AI Game Programming Wisdom 4*, pp 383-391, Charles River Media, 2008.
- Summers, J., McLaren, M., Aha, D., (2004). Towards Applying Case-Based Reasoning to Composable Behavior Modeling. *In Proceedings of Behavior Representation in Modeling and Simulation.*
- Department of the Army. (2001). *Field Manual 3-90, Tactics*. Washington, D.C..
- Department of the Army. (2004). *Field Manual 1-02, Operational Terms and Graphics*. Washington, D.C..

## Author Biographies

**EVAN CLARK** is a software engineer at the Physical Science Laboratory of New Mexico State University. He works as the simulation subject matter expert for the System of Systems Survivability Simulation (S4). He graduated from Brigham Young University with a B.S. in Electrical Engineering in 1997. He received a M.S. in Computer Science from New Mexico State University in 2006 and a Ph.D. in 2009. His research interests include agent modeling, decision theory, military modeling & simulation, human vision and audition, computational geometry and visual languages.

**JEFFREY SMITH** is an Electronics Engineer for the Survivability/Lethality Analysis Directorate (SLAD) of the Army Research Laboratory. He is the lead engineer for developing and fielding a Systems of Systems Analysis capability for SLAD and a provider of survivability, lethality and vulnerability analysis expertise to the System of Systems Survivability Simulation (S4) agent based model, a core component of this capability. He graduated from New Mexico State University with a B.S. (1984) degree in Electrical Engineering. He has an M.S. (1998) and a Ph.D. (2004) in Industrial Engineering (Operations Research/Stochastic Systems) and a minor in Mathematics (Statistics/Probability Theory). He entered Federal Service in 1978 as a Co-op engineer, and continued as an Electronics Engineer with the completion of his B.S.E.E. 1984. He has worked his entire career with the U.S. Army in the area of close combat weapons, assessing the effectiveness and hardness of various weapons systems, and the survivability of numerous developmental and fielded combat platforms. He has a lifelong interest in military history and combat simulations.

**JOEL EICHELBERGER** is the communications subject matter expert and lead communications model developer at the Physical Science Laboratory of New Mexico State University. In 2001, he earned a B.A. in Computer Science and Business from Northwood University, where he was the University's Network Administrator. In 2009, while working at NMSU, he earned his M.B.A.. He has interests in military communications systems and the impact of information on the battlefield, with a focus on the modeling and simulation of said systems.

# CogTool: Predictive Human Performance Modeling by Demonstration

Bonnie E. John  
Human-Computer Interaction institute  
Carnegie Mellon University  
5000 Forbes Avenue  
Pittsburgh, PA 15213  
412-268-7182  
bej@cs.cmu.edu

Keywords:  
Human performance modeling, Cognitive modeling, Keystroke-Level Model, KLM, ACT-R

## 1. CogTool

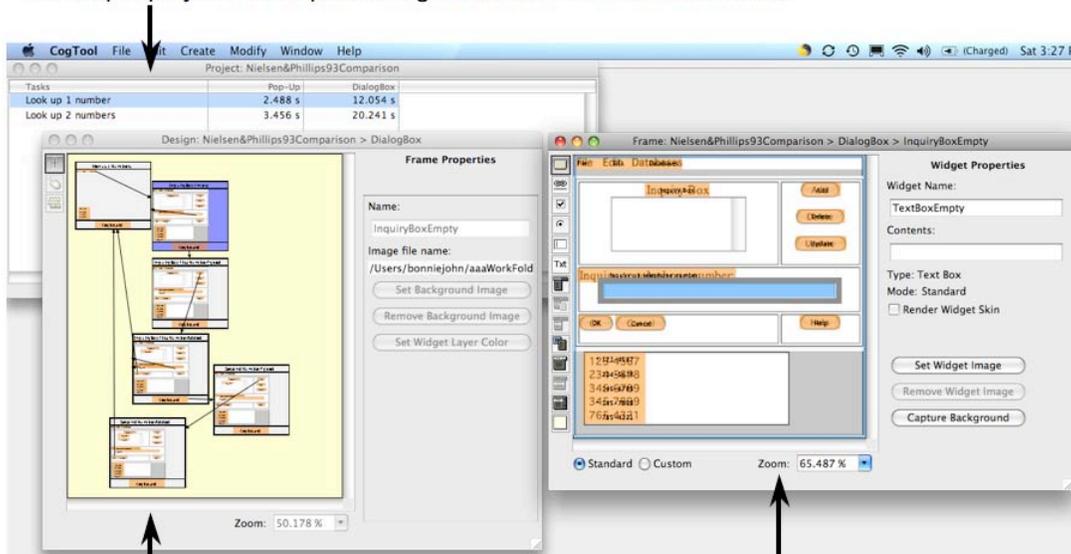
CogTool is a general purpose UI prototyping tool with a difference - it automatically evaluates a design using a predictive human performance model (a "cognitive crash dummy") (John, et. al, 2004)

To use Cogtool, simply create a storyboard of your design idea with sketches, images or on a canvas with CogTool's widgets, demonstrate tasks on that storyboard, then press a button to produce a valid cognitive model (implemented in ACT-R, Anderson,

et. al., 2004) predicting how long it will take a skilled user to complete those tasks (John, 2009). CogTool can be used today to baseline your current interface, or compare competitors' interfaces, and predict how much better your new designs will be.

Looking toward tomorrow, ongoing research is creating and validating new models to predict other metrics of interest to UI designers, for example, the exploration paths of new users (including the errors they are likely to make) (Teo & John, 2008).

1. Set up a project to compare design alternatives on a suite of tasks

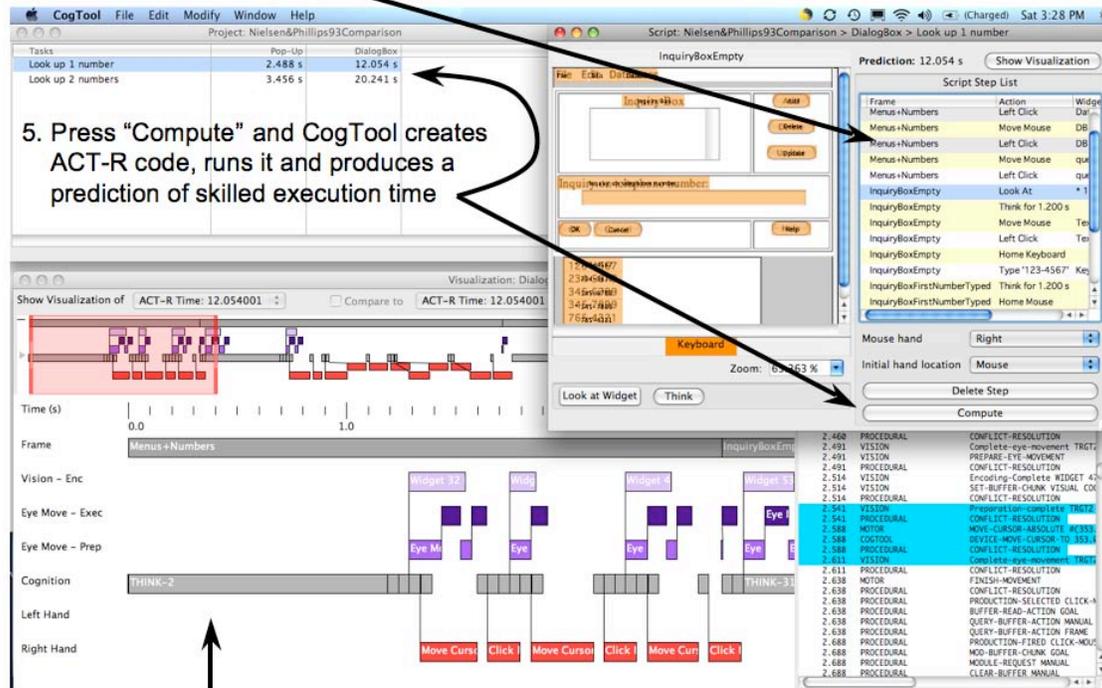


2. Lay out a storyboard of frames (what the user will see) and transitions between them (what the user will do)

3. Detail each frame with the interactive widgets available to the user

Figure 1. CogTool's Project window where projects are set up and results are tabulated (upper left), Design Window where a storyboard is displayed and transitions are defined (lower left), and Frame Window where widgets are placed to mock-up the display and controls presented to users (right).

4. Demonstrate the tasks, CogTool creates a valid cognitive model of a skilled user.



5. Press “Compute” and CogTool creates ACT-R code, runs it and produces a prediction of skilled execution time

6. Examine what the ACT-R model did to produce the prediction in an interactive timeline visualization.

Figure 2. CogTool’s Project window (upper left), Script Window where tasks are demonstrated and computation launched (upper right), and Visualization Window where timelines can be interactively inspected to see what ACT-R did to produce the predictions (bottom).

## 2. The Interactive Demonstration

The interactive demonstration will include CogTool analyses at different stages of completion, much like a cooking show, which will allow the demonstrator to focus on aspects of the tool requested by the audience. Depending on the size and engagement of the audience, this can be a linear presentation or it can move in many different directions, as varied as the audience’s interests. There will be examples from desktop applications, web-based services, parallel programming environments, cell phones, among others.

## 3. References

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review*, *111*(4), 1036-1060.

John, B. E. (2009) *CogTool user guide version 1.1*. Carnegie Mellon University, Pittsburgh PA, September 17, 2009.

John, B. E., Prevas, K., Salvucci, D. D., & Koedinger, K. (2004). Predictive human performance modeling made easy. In *Proceedings of the SIGCHI Conference on Human Factors in*

*Computing Systems. CHI '04* (pp. 455-462). New York: ACM.

Teo, L. & John, B. E. (2008) Towards predicting user interaction with CogTool-Explorer. *Proceedings of the Human Factors and Ergonomics Society 52nd Annual Meeting* (New York, NY, Sept 22-26, 2008).

## Author Biographies

**BONNIE E. JOHN** (B. Eng. 1977, The Cooper Union; MS 1978, Stanford; PhD, 1988 Carnegie Mellon University), is a Professor, founding member of Carnegie Mellon University’s Human-Computer Interaction (HCI) Institute, and a member of the ACM SIGCHI Academy. She has been researching human behavior modeling and using it to guide HCI design since 1983. As the Director of the Masters program in HCI, Dr. John has researched and taught many HCI design and evaluation techniques. She has brought these experiences together, through human-centered design and automating substantial portions of the modeling process, to create modeling tools that are easier to use, one of which (CogTool) will be demonstrated at this session.

## DETAILED DESCRIPTION OF TUTORIAL

Twenty-five years ago, Card, Moran and Newell introduced the concept of engineering models that could make a priori, quantitative predictions of human behavior with computer interfaces (Card, Moran & Newell, 1983a, b). In principle, these models could help design by quickly evaluating many alternative ideas before empirical data could be collected on running systems or prototypes. Research in this area has continued and over one hundred research papers have been published about GOMS and the Keystroke-Level Model (KLM) (see the GOMS bibliography, <http://www.gomsmodel.org/gomsbib.html>). Applications in the real world have been reported, but adoption into industrial practice has been slower than the success of the research might warrant. One hypothesis has been that there were no reliable, freely available, easy to use tools that made modeling easy for practitioners with little psychology training. In the past few years, several groups have been building user-centered tools for modeling (sponsored by the Office of Naval Research and other organizations) and it is now possible to accelerate adoption of modeling in industry through short courses.

Interest in this area is evident from the number of papers at CHI2007 that included modeling as one of the techniques that brought value to a project (e.g., see papers by Google, NASA, the Carlsbad Police, Drexel, Fraunhofer IASI, the UK's Transport Research Laboratory, among others) and by the attendance of practitioners from many companies at tutorials at BRIMS 2007 & 2009, HCI International 2009 and HFES 2008 & 2009. No one suggests that modeling is the only tool necessary, but it is a tool that is ready for more HCI professionals to feel comfortable using, and the BRIMS Conference is an appropriate place for them to attain these skills.

The day will begin with a short lecture on the history and state of the art of predictive human performance modeling, leading directly into a hands-on modeling session before the first break. The example task will be web-based collaborative shopping, with the collaboration supported by gmail, Google notebook, or a wiki. Comparing these three interfaces and analyzing what the models say for the design of a new collaboration system will be the focus of the first morning session.

There are two ways to use the tool that will be taught. The first way is to use screenshots from an existing system to baseline skilled performance on that system. This will be the topic of the first hands-on exercise. However, if the tool could only baseline existing systems, it would not be any more useful in design than conducting empirical tests! The second way to use the tool is to rapidly build new designs and predict skilled performance on many design ideas. This will be the focus of the second hands-on exercise. The participants will redo the storyboards and models of the collaborative shopping task in this more powerful way. We will reuse the same task so they participants already have an understanding of it and how a baseline model is built. Given this basis, they will be able to appreciate the different modeling approaches provided by the tool. This activity will finish before lunch.

After getting comfortable with using the tool on these simple examples, the participants will spend most of the rest of the day using the tool to model their own projects from their own work, or, if they do not have a work project to use, the instructor will provide several more complex projects. They will get one-on-one assistance from the instructor.

The tutorial will end with a short lecture on a variety of applications of this modeling technique and current research that will be available in the tool in the future. This will include being able to predict exploratory behavior, emergent strategies, and learning time as well as skilled execution time.

Time allotted	Topic or Event
10 min	Instructor introduction & course objectives; Survey of the audience background and interests
20 min	State of the art of predictive human performance modeling
15 min	Introduction to the software, set-up & begin first hands-on exercise
45 min	Hands-on exercise continues, with instructor and graduate student aids helping the participants. When questions or issues of general interest arise, the instructor will discuss them with the class as a whole. Participants who finish early will move on to a second exercise, which can either be supplied by the instructor or can be of their own systems.
30 min	Break
30 min	Q&A about the first hands-on exercise session. Peer discussion of modeling options and trade-offs discovered during the first session.
1 hour	Second hands-on session where the participants re-do the model from the first exercise in a more powerful way.
1 hour	LUNCH
30 min	Q&A about the second hands-on exercise session. Peer discussion of modeling options, trade-offs, and approaches to design exploration discovered during the second session.
1 hour	Third hands-on session where the participants model a more difficult interface, either from the instructor's materials or their own system.
30 min	Break
1 hour	Presentation of designs and models of volunteer participants from the third hands-on session. Peer discussion of modeling options, trade-offs, and approaches to design exploration.
30 min	Wrap up of what has been explored today and the future of predictive human performance modeling and tools to support it.

#### WHO WOULD BENEFIT FROM THIS TUTORIAL

The target audience includes human factors professionals and system developers who want to evaluate alternative designs before building running prototypes. No prior knowledge of perceptual, cognitive, or motor psychology, or predictive human performance modeling is required.

Participants in previous BRIMS, HCI International and HFES tutorials were from industry and government, (with a few from academia interested in learning to teach human performance modeling) from organizations such as Boeing, BAE, Lockheed-Martin, Toyota, Nissan, Department Of Veterans Affairs (Health Data And Informatics), and all branches of the US armed forces. Comments on the feedback forms from the Sept 2008 HFES tutorial (which HFES calls a "workshop") included:

"This tool will be very useful to me as an HF practitioner. Often we are asked how "much" better one design is compared to another and it is difficult to obtain our target users to participate in a test like this. Modeling is a much easier effort to get the answers we need."

"The workshop has excellent application to product design in industry! This was something I can take back and use immediately in HCI."

"Well taught, organized, with examples that are applied and therefore very interesting to HSI [Human System Integration] practitioners."

"Groundbreaking theories being applied to real-world designs to accurately and easily predict user performance."

"Wonderful. I can clearly see how, as a practitioner in industry, I can apply this to the numerous projects I work on."

## Cultural Geography Model Validation

*Lisa Jean Bair*

*Eric W. Weisel*

WernerAnderson, Inc.

6609 Main Street

Gloucester, VA 23061

(804) 694-3173

[lbair@werneranderson.com](mailto:lbair@werneranderson.com), [eweisel@werneranderson.com](mailto:eweisel@werneranderson.com)

*Richard F. Brown*

Captain, Field Artillery

Operations Research Analyst

US Army TRADOC Analysis Center – Monterey

(831) 656-7586 (DSN 756)

[richard.f.brown1@us.army.mil](mailto:richard.f.brown1@us.army.mil)

### Keywords:

cultural geography, human terrain, human behavior modeling, HSCB, validation, social network analysis

**ABSTRACT:** *In the current warfighting environment, the military needs robust modeling and simulation (M&S) to support Irregular Warfare (IW) analysis across the range of tactical, operational, and strategic levels of warfare to help inform decisions concerning operations within the IW environment. In support of this need, the military requires a responsive family of Models, Methods, and Tools (MMT) able to credibly represent US and Coalition ground forces conducting operations in a Joint and Combined IW environment, from the tactical to strategic levels. As a first step in this direction, TRAC Monterey (TRAC-MTRY) is developing a prototype capability that credibly represents ground forces conducting IW operations and focusing on the relevant relationships and interactions within the population. This paper describes work being performed on behalf of TRAC-MTRY to develop a measurable, repeatable method for assessing, understanding, and describing the risk of using an M&S for analysis, to enhance the ability of decision makers to assess the risk in using an IW M&S, and add to the core body of knowledge in Validation Best Practices.*

### 1. Introduction

In the current warfighting environment, the military needs robust modeling and simulation (M&S) to support Irregular Warfare (IW) analysis across the range of tactical, operational, and strategic levels of warfare to help inform decisions concerning operations within the IW environment. Violent extremist networks, which are tactful, complex adaptive systems with the outward appearing ability to act without direction are implicit within IW. Appropriate and meaningful responses to these violent extremist networks require understanding of the underlying population, its dynamics, and its driving forces. In support of this need, the military requires a responsive family of Models, Methods, and Tools (MMT) able to credibly represent US and Coalition ground forces conducting operations in a Joint and Combined IW environment, at the tactical to strategic levels. As a first step in this direction, TRAC Monterey (TRAC-MTRY) is developing a prototype capability that credibly represents ground forces conducting IW operations and focusing on

the relevant relationships and interactions within the population. To this end, TRAC-MTRY has developed the Cultural Geography Model (CGM), a government owned, open source multi-agent system utilizing Bayesian networks, queuing systems, the Theory of Planned Behavior, and Fischer's Narrative Paradigm, as a first step in the development of a family of models to support the defense analyst in answering questions relevant to IW such as "Is security adequate?", "Will the outcome of upcoming elections be legitimate?" or "Will the presence of troops increase civilian violence?" with responses similar to polling data (Alt et al 2009 – JDMS pre-pub copy). Effective validation of models within this context requires progress in the theory of validation. This paper reports on the necessary background required to support work being performed on behalf of TRAC-MTRY to develop a measurable, repeatable method for assessing, understanding, and describing the risk of using an M&S for analysis, to enhance the ability of decision makers to assess the risk in using an IW M&S, and add to the core body of knowledge in Validation Best Practices.

## 2. Modeling IW

The M&S of IW requires the development of new M&S methods. The social science on which this development hinges is in its infancy. In particular, the social science is often biased by western perspectives in many areas; includes multiple theories to describe the same phenomena, often uncorrelated and sometimes contradictory; and lacks empirical data and underlying computable, mathematical structures to inform and validate modeling efforts. In fact, the data that is available is often qualitative vice quantitative and the relationships between available quantitative data and its effects on the social systems of interest are unknown (e.g., the human engagement that occurs between military units and the population, and its mutual relationship with DIME/PMESII at higher levels over time). Even in well understood, homogeneous populations, population modeling is difficult because of the complexity of human cognition. Heterogeneous, unfamiliar populations only exacerbate this problem. A method is needed to assess the available data, social science, and the developed M&S in a measurable, repeatable way for assessing, understanding, and describing the risk of using an M&S for analysis. Development of this risk assessment method is a key element in Validation Best Practices.

### 2.1 Validating IW models

The DoD guidance for accomplishing VV&A is well known and documented. While results validation and face validation are often used methods for the validation of models, the difficulties with this approach for simulations having sensitivity to initial conditions, chaotic, or emergent effects, and the difficulties with validating human based representation models is well known (Harmon et al. 2002, Defense Modeling and Simulation Office 2006, Akst 2006, Moya et al. 2007). The validation literature consists mainly of validation approaches, paradigms, and techniques as well as specific validation applications and assessments. There is no mechanism guiding the appropriate selection of approach and techniques in a given M&S application. Progress is required that will lead to effective validation, supporting the need for developing “fundamental new approaches of conducting VV&A ... [and] ... developing new VV&A methods and techniques ... [with] practical value” (Sargent et al. 2000).

To address this need, the Marine Corps Combat Development Center (MCCDC) Operations Analysis Division (OAD) commissioned an Agent Based Simulation (ABS) Verification, Validation, & Accreditation (VV&A) Framework Study in 2008 to

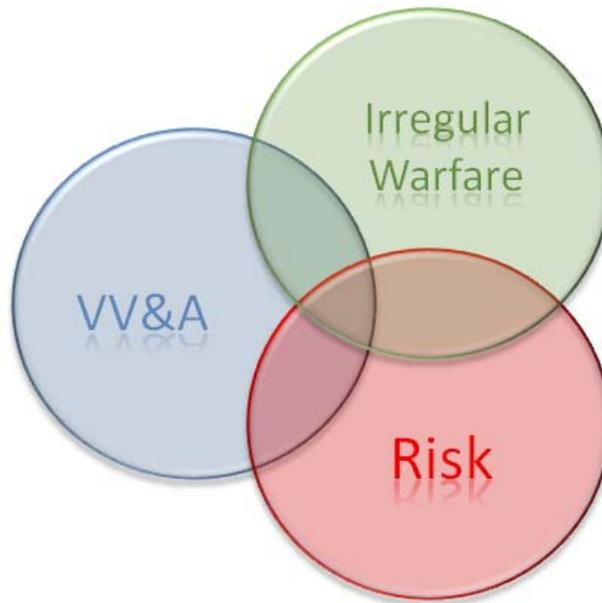
develop general, institutionally acceptable processes and criteria for assessing the validity of agent-based simulations used as part of DoD analyses with a focus to IW analyses. At its onset, this study focused on the concept of validity, viewing the verification process for simulation as the same as for software verification and accreditation as an agreement between analysts and the study sponsor that a particular model is useful for a particular analysis problem. It addressed the verification and accreditation processes with respect to their interdependencies with the validation process.

The MCCDC OAD effort focused on the validation of the non-physics based aspects of the validation problem with the goal to maintain the analytic rigor of the traditional VV&A process, while expanding it to cover non-traditional topics (e.g., population dynamics and cultural shifts). The effort demonstrated the validation process of ABS in two applications to guide the development of a framework that would provide a means for assessing the reliability, applicability and feasibility of the ABS for its intended use, preferably in a quantifiable way for future validation efforts. A key finding of this work is that the validation of an M&S for analysis cannot be decoupled from that analysis. The effort for TRAC-MTRY will leverage and expand on the MCCDC OAD effort in an applied way.

### 2.2 CGM validation project

The DoD requires robust IW modeling in the current environment. TRAC-MTRY is developing capabilities to help determine the potential impact of culture and the actions of the civilian population on current operations. As part of this larger effort, it is essential to have a validated conceptual model underlying the CGM reflective of the selected social science underpinnings. This project will develop a measurable, repeatable method for assessing, understanding, and describing the risk of using an M&S for IW analysis as well as develop validation methodologies for assessing the CGM conceptual model and implementation (Figure 2.1). It has the objective to assess the operational utility of the CGM with suggestions for its analytical use that make the operational utility accessible and mitigate any issues within the uses of interest. It supports Key Tenets of the TRAC IW Campaign plan by enabling an incremental development cycle, with interim proof-of-principle and prototype applications (“build-use-learn-fix” approach) and fits within the MMT line of effort by supporting the development of a Validation and Verification (V&V) methodology that helps achieve useable capabilities as fast as acceptable risk and resourcing permit.

**Figure 2.1. Problem Context**



### 3. Validating Human Behavior Models

The validation of IW M&S for analysis lies within the intersection between the spheres of VV&A, IW, and Risk as shown in Figure 2.1. Developing core knowledge of the IW is the purview of our military specialists. The question of how VV&A may be applied within the IW sphere has been asked (reference to be added). Questions arising from the intersection of the VV&A and risk spheres are more often well-understood for physics-based or engineering models but less frequently so for M&S techniques such as agent-based simulation. The intersection of the risk and IW spheres is the domain of the art of warfare and out of scope for the technical discussion. The addition of risk to the analysis allows a more formal discussion of the usefulness and limitations of M&S derived information. Our focus is on the innermost intersection where these questions may be answered in a real way for the IW problem.

#### 3.1 Validation importance

Acceptability and usability get at the key points for why validation is important: to establish the credibility of a simulation for a specified intended use (Modeling and Simulation Coordination Office 2004b). This includes determining that the simulation is correct and meets requirements through software engineering and other processes but is not limited to that. It also includes providing users with sufficient information to determine if the simulation can meet their needs as well as determining the simulation's capabilities, limitations, and performance relative to the real-world objects it simulates. User

participation throughout the development process facilitates this confidence.

The DoD guidance for accomplishing VV&A is well known and documented. While results validation and face validation are often used methods for the validation of models, the difficulties with this approach for simulations having sensitivity to initial conditions, chaotic, or emergent effects, and the difficulties with validating human based representation (HBR) models is well known (Harmon et al 2002, Modeling and Simulation Coordination Office 2004b, Akst 2006, Moya et al 2008).

Understanding the validity of the M&S of physics based and engineering systems for a given use is well understood. Further, physics-based combat models have a long history of use. However, the M&S of IW requires the development of new M&S methods. Further, the social science on which this development hinges is in its infancy. In particular, the social science is often biased by western perspectives in many areas; includes multiple theories to describe the same phenomena, often uncorrelated and sometimes contradictory; and lacks empirical data and underlying computable, mathematical structures to inform and validate modeling efforts.

#### 3.2 Necessary elements for HBR validation

The robust documentation of the conceptual model; testing; and the theoretical support, traceability and justification for assumptions facilitate user confidence. Using a well-defined, documented validation process

supports model credibility. Using strong validation methodologies ensure that models are built on a solid framework of standardized organization, process, products, and techniques; and that they simulate accurate, consistent, and reproducible results. Without strong, documented methodologies, valid simulations may be rejected, invalid outcomes may be accepted, or simulations may be used improperly (e.g., outside of intended use or in opposition to embedded assumptions). Formal methods allow for the precise description of a simulation's capabilities. Further, the ability to make general statements about individual, general, and federated models facilitates use and re-use of those models.

Any effective validation methodology needs to have the following characteristics (Weisel and Moya 2007):

- 1) Transparent – to provide an understanding of the assumptions, decisions, and activities that went into V&V (I know what I have)
- 2) Traceable – to ensure the flow of activities and actions is logical and that appropriate referents for those activities can be located and consulted (I know where I got it)
- 3) Reproducible – to provide for the event that the same model/data/users will be applied to a similar effort in the future (Another researcher can get the same)
- 4) Communicable – to produce sufficient, understandable documentation so the effort can be independently duplicated, and so the consumer can make an informed, and perhaps qualified, decision (It is understandable to those who care)

Other objectives include the ability of the process to do the following:

- 1) Describe the bounds of use for the specified purpose
- 2) Communicate the risk of use for the specified purpose

The necessary information when communicating the results of validation activities includes, but is not limited to, data sources; referent sources and descriptions; designs of experiments; data and metadata for the model; initial conditions; boundary conditions; parameters; assumptions; analyses performed and methodologies followed; and appropriate uses of results.

The primary purpose, and importance, of conducting validation activities is to assess the risk of using an M&S for a specific application of use. The validation process culminates in the communication of that risk to model and

simulation users and the recipients of their data. This includes determining that the simulation is correct and meets requirements through software engineering and other processes but is not limited to that. It also includes providing users with sufficient information to determine if the simulation can meet their needs as well as determining the simulation's capabilities, limitations, and performance relative to the real-world objects it simulates.

### 3.3 The validation of HBR models

The validation literature consists mainly of validation approaches, paradigms, and techniques as well as specific validation applications and assessments. There is no mechanism guiding the appropriate selection of approach and techniques in a given M&S application. Further, in the physical sciences the concept of valid models is well-understood; this is not the case in HBR modeling. In particular, these models have inherent validation difficulties due to the characteristics of these models (referents that have poor computational underpinnings, complexity, chaotic effects, etc.) and to their desired uses (e.g., Course of Action (COA) Analysis). Techniques for validation will require methods grounded in the larger validation, computational sciences, and experimental design literature and apply them to the growing field of HBR model validation. Any technique applied in this domain will require an assessment of the chosen conceptual model, its implementation in codes, and the subsequent simulation results once used.

### 3.4 Conceptual model validation

The conceptual model is the representation of the content and concept for the model that includes the logic, algorithms, assumptions, and limitations (Department of Defense 1998). Verification ensures that the code correctly captures this conceptualization. In validation, the conceptual model is compared against the specified referent. In particular, the conceptual model must be true to within the limits of acceptability criteria in terms of the true statements within the referent. While there may be things that are true in the referent that are not true in the conceptual model, the obverse should not occur. That is, *not true* in the conceptual model does not necessarily imply *not true* in the real system that the referent represents. However, there may be things that are true in the real system and in the referent for that system that are not true in the conceptual model because those items purposely were neglected or abstracted out.

While initial assessments may find the conceptual model to be valid, the simulation may produce invalid results nevertheless. This may result from elements initially deemed not important in the model development, incorrect relationships between elements, inappropriate

abstraction for the intended use, or poor assumptions. This may especially be true in systems where the conceptual model reflects a referent based in underlying theories of the system without a strong mathematical, analytical, or logical description that translates itself more easily into code. This is partly because programmers can only code those relationships they understand and in part due to the fact that there are many ways to describe desired relationships computationally. For instance, just as there are many possible rule sets for describing a single agent system, there are multiple ways to model the relationship  $y$  increases with  $x$ . Results validation may uncover needed changes in the specification of the conceptual model thereby uncovering an invalid conceptual heretofore thought of as valid.

The testing of assumptions made in the model may also uncover previously undiscovered defects in the M&S. These assumptions could include seemingly inconsequential assumptions made during coding efforts such as the precision used for  $\pi$  or the simulation time step or more obviously important assumptions like whether the earth is flat or spherical or the selected social theory. Documentation for every assumption used in developing and coding a model is rarely complete. However, assumptions' testing does not require the explicit identification of every assumption. Only those assumptions potentially affecting the use of the M&S need assessment for their impact. Part of the art in devising the validation analysis assessing a model's assumptions is in recognizing the types of assumptions that might be significant on its use given a description of the model and the context of its specific use and devising tests to assess the impact of the assumptions made. Tests might include sensitivity analyses about the assumptions, accuracy assessments to ensure that the chosen precision is sufficient, or any other appropriate test. Thus, one cannot decouple the results validation from validation of the conceptual model.

### 3.5 Results validation

Results validation is only meaningful in the context of specific identification of what constitute valid results. This is stressed both in the *VV&A RPG* and by Harmon and Youngblood in the importance of stating the acceptability and validation criteria up front; i.e., the necessary elements for using and trusting the M&S. That is, stating up front the necessary elements for using the M&S. This is equivalent in the validation theory of describing the natural system or referent trajectories against which M&S trajectories will be compared and the validity relation that will be used to make the comparison. It could include statistical comparisons of simulation output to assess the real world match. Often this is an accuracy specification required to support the intended

use of the M&S. Engineering models (e.g., for system design and development or for test and evaluation) require predictive accuracy most likely assessed using a metric relation. On the other hand, campaign models may only require sufficient accuracy to enable relative comparisons between alternative outcomes based on changes to tactics, forces, or equipment. Necessary to this assessment is the determination of the simulation results to be measured, the material in the referent against which these results are compared, the mechanism of comparison, and the requirements of the results' acceptability. Results validation could run the gambit from a state-by-state match to observed or empirical data or with some theoretical or posited expectation to an assessment that the overall trends occurring in the model match the theory. In the absence of this specification, the validator, users, and subject matter experts will make their own implicit assumptions of what is required.

Comparing simulation results to empirical or observed data is preferable. While a metric relation could be used to assess accuracy (i.e., the delta between values), other accuracy measurements are possible (e.g., comparisons of direction, slope, or relative magnitude). When this kind of data is not explicitly available, the validator still needs to assess whether the simulation output meets the needs of the intended use (e.g., can help answer the analytical questions). In this case, results validation relies on robust test cases and specification of expected results within the referent determined either from theory or SME opinion.

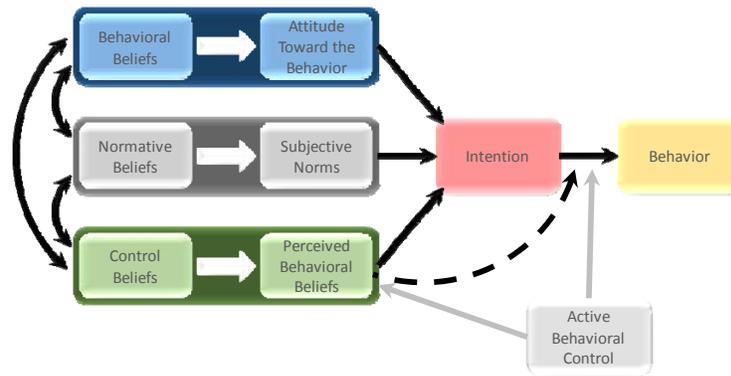
## 4. CGM Overview

The CGM is a government-owned, open source, data driven multi-agent social simulation. Actors, rules, and laws within the model are built upon social and behavioral science theories. A modular framework is used to allow the incorporation of other social theories or the use of different applications as the CGM grows in maturity. The current implementation of the model uses the narrative paradigm, theory of planned behavior, and Implementation of Entity Cognition with Bayesian Belief Networks (BBN) to determine entity states.

### 4.1 Narrative paradigm

The use of the CGM requires understanding of the culture in which the scenarios of interest take place. Within the model, cultural beliefs of the entities drive reactions to events occurring within the scenario along with social interactions between entities. To provide a basis for the connection between cultural factors, entity beliefs, and activities, narrative theory plays a critical role in the development of data in the model. In narrative theory, people are storytellers and view the world through a

**Figure 3.1: Theory of Planned Behavior, By: Icek Aizen, 2006 (adapted)**



narrative lens, thus irrational actions may actually be rational given their history and culture. Its selection was based on Fisher’s argument (Fisher, 1988) as follows:

- 1) people are essentially storytellers;
- 2) reasons for decisions include history, culture, and perceptions about the status and character of the other people involved (all of which may be subjective and incompletely understood);
- 3) narrative rationality is based on the probability, coherence and fidelity of the stories that underpin the immediate decisions to be made; and
- 4) the world is a set of stories from which each individual chooses the ones that match his or her values and beliefs.

Selection of stories for use in data development follow Fisher’s proposal of evaluating stories based on whether the narrative’s coherence, probability, and fidelity. Narrative coherence means the story should make sense structurally, have detail and characters, and should be free of surprise. Narrative probability concerns the belief of listeners in the truthfulness of the story irrespective of the story’s actual truthfulness. Narrative fidelity addresses the truthfulness of a story with respect to cultural values that include embedded values, relevance between the story and the values espoused, consequences, consistency, and transcendence.

## 4.2 Theory of planned behavior

The theory of planned behavior provides the underlying basis for the development of data for entity intention, action, choice, and selection within the CGM. In the

theory of planned behavior (Figure 3.1)<sup>1</sup>, entities form behavioral intentions based on attitudes, perception of group norms, and perceived level of control.

## 4.3 CGM Conceptual Model

To Be Added in final paper – Provide a description of the CGM mathematical and logical implementation guiding the direction for the validation effort.

## 5. Challenges

The problems we face in the current warfare environment make the development of HBR models sufficient to address the problems of interest and their validation importance. Having useful, credible, robust information is critical for the support of sound decision-making. However, limitations in the current state of the art create challenges. First, the systems of interest are complex. One of the reasons for developing the models is to develop an understanding of the systems’ behavior in response to various scenarios that might occur. That is, we want to understand the system of interest. However, the social science that forms the underpinning of these models often has multiple, conflicting theories for behavior, complicated by variances in responses by culture and stressor. This creates difficulty in model development and acceptability. That is, our understanding of the system is limited.

<sup>1</sup> Copyright Notice: The theory of planned behavior is in the public domain. No permission is needed to use the theory in research, to construct a TpB questionnaire, or to include an original drawing of the model in a thesis, dissertation, presentation, poster, article, or book. However, if you would like to reproduce a published drawing of the model, you need to get permission from the publisher who holds the copyright. You may use the drawing on this website for non-commercial purposes so long as you retain the copyright notice. – To Be Redrawn

Second, the systems of interest are dynamic. The development and testing of models requires data to support them. Further, these models also require data related to the relationships between elements or entities within the model. This includes influence relationships between elements as well as cause-effect relationships. Not only is obtaining this data difficult, especially for the problems of interest, the data developed is often qualitative vice quantitative and has an unknown valid lifetime. In particular, it is unknown whether the data valid lifetime exceeds the initial stressor events of interest.

The third challenge is a direct result of the first two. Since these M&S exist in a computer, necessary to the model development is a computational representation of the social theories, interactions, and behaviors of interest. While there are some accepted representations such as Bayesian networks, this is far different from the general acceptance found in the computational representations found in the physical sciences. To create valid models, both conceptual model and results validation is required. The validation of either requires progress in both the social sciences to develop accepted computational representations as well as measureable system responses to events or inputs to the system.

## 6. Next Steps

The objective of this project is a repeatable approach for validating cultural behavior models, particularly the conceptual model, including risk measures and criteria for assessing risk using the CGM as a vehicle for the method's implementation. While there are many challenges in HBR modeling, making progress in techniques for the M&S of HBR and in developing methods validating those M&S is necessary. The next steps in this project are to continue evaluation of the CGM conceptual model. Critical to the effective use of M&S is the understanding of the risk in that use for a specific problem of interest. This is the key goal for validation. The understanding of the risk in using a simulation for a specified use is a core area of research for this work.

There are two components of risk in general (Defense Acquisition University 2003):

1. The probability or likelihood of achieving (not achieving) a given outcome
2. The consequences of achieving (not achieving) a given outcome

There is higher risk with a higher likelihood or with significant consequences. Risk assessment includes both the identification of risk (determination of outcomes) and

the analysis of risk (determination of probability and consequence of an outcome). It is in this latter aspect that M&S often plays a role. That is, the intended use for an M&S is to identify and help to mitigate risk, identified as part of some specified objective. However, the use of M&S in this analysis poses an inherent source of risk. The sources of risk could lie in the development of the model, development risk, or in the running of the simulation, operational risk (Modeling and Simulation Coordination Office 2004b). Development risk is that the model does not meet the requirements for its intended use. Operational risk is that the M&S exhibits insufficient accuracy to provided needed information. The V&V process addresses both these risk areas. When considering intended use, risk can be described generally using the three familiar error types:

1. Type I Error: Reject correct information; the information provided by the M&S is not used in solving the problem even though the information provided is correct.
2. Type II Error: Accept incorrect information; the information provided by the M&S is used in solving the problem, however, the information provided is incorrect.
3. Type III Error: Solve the wrong problem; the information provided by the M&S is irrelevant to the actual problem to be solved.

Validation primarily assesses the Type II error. When assessing the consequences of using incorrect data in a decision, considerations include who is affected, the severity of the effect, and the visibility of the consequences. Development risk assesses the effect of not meeting requirements, the likelihood of a deficiency, and the probability that a deficiency will cause the M&S not to meet requirements. These assessments drive toward the fundamental assessment of whether the M&S support the intended use. Operational risk assesses the probability of making an incorrect decision, the effect and visibility of making an incorrect decision, and specific user considerations.

## 7. References

- Akst, G. (2006). Musings on Verification, Validation, and Accreditation (VV&A) of Analytical Combat Simulations. *Phalanx. Volume 39. Number 3.* Pages 9-11.
- Defense Acquisition University. (2003). Risk management guide for DoD acquisition. Ft. Belvoir, VA. Defense Acquisition University Press.
- Fisher, Walter R. (1987). Human Communication as Narration: Toward a Philosophy of Reason, Value,

and Action. Columbia: University of South Carolina Press. ISBN 0-87249-500-0

- Fisher, Walter R. (1988). The Narrative Paradigm and the Assessment of Historical Texts. *Argumentation and Advocacy*. 25. Fall. 49-53.
- Fisher, Walter R. (1989). Clarifying the Narrative Paradigm. *Communication Monographs*. 56. 55-58.
- Fisher, Walter R. (1994). Narrative Rationality and the Logic of Scientific Discourse. *Argumentation*. 8. 21-32.
- Fisher, Walter R. (1995). Narration, Knowledge, and the Possibility of Wisdom. in *Rethinking Knowledge: Reflections Across the Disciplines*. Suny Series in the Philosophy of the Social Sciences. (Fisher & Robert F. Goodman Eds). New York: State University of New York Press.
- Guarro, S., Vessely, W. E. (2004). Risk Modeling. (Rainey, L. B. Davis, P. K. Eds.) pp. 369-450. Space Modeling and Simulation. AIAA.
- Harmon, S. Y., Hoffman, C. W. D. Gonzalez, A. J. Knauf, R. Barr, V. B. (2002). Validation of Human Behavior Representations: B3-1-B3-34. In *Foundations for V&V in the 21st Century Workshop (Foundations '02)*. San Diego, CA. The Society for Modeling and Simulation International (SCS).
- Mansoor, P.R.C., and Ulrich, M.S.M. (2007). Linking Doctrine to Action: A new COIN Center-of-Gravity Analysis. *Military Review*.
- Modeling and Simulation Coordination Office. (2006). The Verification, Validation, and Accreditation Recommended Practices Guide, RPG Build 3.0. Special Topics: Human Behavior Representation Validation. Updated 15 August 2001. Last accessed 6 January 2010, <http://vva.msco.mil/>.
- Moya, L. J., McKenzie, F. D. and Nguyen, Q.-A. H. (2008). A visualization technique to support human behavior representation model rule validation. *Simulation & Gaming: An Interdisciplinary Journal of Theory, Practice, and Research*. 39. 101-117.
- Sargent, R. G., et al. (2000). Strategic Directions in Verification, Validation, and Accreditation Research. (J. A. Joines, R. R. Barton, K. Kang, P. A. Fishwick, eds.) In 2000 Winter Simulation Conference. 909-916.

## Acknowledgements

The authors gratefully acknowledge MCDCC OAD for its support during the ABS Verification and Validation (V&V) Framework Study. We further acknowledge the help and support of TRAC-MTRY in the Validation Methods for Assessing Conceptual Models and Risk in Modeling & Simulation (M&S) for Irregular Warfare (IW) Analysis project.

## Author Biographies

**MS LISA JEAN BAIR** (formerly Moya) is Chief Scientist of WernerAnderson and leads its modeling and simulation (M&S) research and development effort. She is co-chair of the M&S Congressional Caucus Standing Committee in support of M&S Professional Development and Education and serves on the Certified Modeling and Simulation Professional (CMSP) Board of Directors. Ms Bair's areas of expertise include Multiple Objective Decision Analysis (MODA), Multi-Attribute Utility Theory (MAUT), Modeling and Simulation, simulation validation, human behavior modeling, and the development and application of analysis. Ms Bair has experience in analyses of alternatives; requirements evaluation; and M&S planning, use, and validation for the department of defense. Ms Bair is a PhD candidate in the Modeling and Simulation program at Old Dominion University and received an M.S. in Operations Research from The College of William and Mary and a B.S. in Applied Mathematics from Old Dominion University. Her research interests include agent based simulation, simulation theory and formalisms, human behavior modeling, and validation.

**DR. ERIC W. WEISEL** is CEO of WernerAnderson, Inc. Dr. Weisel received the Ph.D. in Modeling and Simulation from Old Dominion University in 2004, the M.S. in Operations Research from the Florida Institute of Technology in 1995, and the B.S. in Mathematics from the United States Naval Academy in 1988. Prior to founding the company, he served as a U.S. Navy submarine officer on Los Angeles class attack submarines and various Navy and joint staffs with experience in nuclear engineering; navigation; and submarine, battle group and joint operations. He is active in local government in Virginia. Dr. Weisel serves on the Board of Advisors for the Virginia Modeling, Analysis, and Simulation Center. He has recently served on the Virginia General Assembly's Joint Commission on Technology and Science Modeling & Simulation Advisory Committee and as Chair of the Gloucester County Planning Commission. He is an Adjunct Professor at Old Dominion University teaching courses in Operations Research and Modeling and Simulation. His research interests include human behavior modeling and the theoretical foundations of simulation.

**CPT. RICHARD F. BROWN** serves in the US Army as an Operations Analyst in TRAC-MTRY and is leading the effort described in this paper.

# Reducing the Variability between Novice Modelers: Results of a Tool for Human Performance Modeling Produced through Human-Centered Design

Bonnie E. John

Human-Computer Interaction institute  
Carnegie Mellon University  
5000 Forbes Avenue  
Pittsburgh, PA 15213  
412-268-7182  
bej@cs.cmu.edu

Keywords:

Human performance modeling, Cognitive modeling, Keystroke-Level Model, KLM, HCI, HCD

**ABSTRACT:** *The variation between novice modelers has not been extensively studied, but it is important to organizations wishing to employ predictive human performance models in their system design process. This paper reports on the statistically-significant reduction in variation between novice modelers achieved by CogTool over the previously-established by-hand method of predicting the task execution time of skilled users (Keystroke-Level Model). CogTool was developed using human-centered design techniques specifically to understand and prevent novice errors by transforming the modeling process into an integral part of the system design process and these techniques seem to have worked.*

## 1. Introduction

The variability between modelers as they create human performance models has not been studied extensively. There have been comparisons between models in both AI and cognitive modeling, e.g., Sisyphus (Gaines, 1994), Project Halo (Chaudhri, et. al., 2009), the Ambr Project (Gluck & Pew, 2005) and the Predicting Cognitive Performance in Open-ended Dynamic Tasks Modeling Challenge (Lebiere, et. al., 2009), but each model in these comparisons is created by one person or team using their own modeling approach, and it is unknown whether a different person or team using the same approach would create a similarly-performing model.

The only instance of a comparison between modelers known to this author was a “by product” of a paper comparing different approaches to predicting skilled performance time on different user interfaces (UIs). Nielsen and Phillips (1993) were comparing heuristic estimation techniques to a predictive human performance modeling approach called the Keystroke-Level Model (KLM, Card, Moran & Newell, 1980) and provided data on 19 novice modelers building KLMs for two tasks on two UIs. This author followed up by publishing data from 8 additional novice modelers (John, 1994). In both instances, the coefficient of variance in these data hovered around 20%. This phenomena, called the “evaluator effect” in Human-Computer Interaction (HCI) has been shown for several different HCI techniques (e.g., heuristic evaluation (Nielsen & Molich, 1990) think-aloud usability studies

(Jacobsen, Hertzum & John, 1998), and Cognitive Walkthrough (Hertzum & Jacobsen, 2001)).

The evaluator effect, about 20% for all the techniques yet studied, is particularly troublesome with a predictive human performance modeling technique like KLM, since it claims to have a prediction accuracy of about 20%. Thus, the variation between modelers is on the order of the expected accuracy of the technique itself and should therefore be of special concern to the behavior representation community.

This paper reports on an attempt to reduce the variation between novice modelers by providing tool-support for KLM analyses. Specifically, human-centered design (HCD) techniques were used to create a tool for constructing valid KLMs, called CogTool (<http://cogtool.hcii.cs.cmu.edu/>).

The next section reviews the original by-hand procedure to produce KLMs, what errors novice modelers tended to make using that procedure and how CogTool was built to obviate these errors. Section 3 describes the data assembled to establish the variability in KLMs created by both procedures. Section 4 analyzes the difference in variability between these two sets of models. Section 5 discusses the source of variation that remains in CogTool models and the final section maps future work stemming from these analyses.

## 2. Background

The KLM was introduced by Card, Moran and Newell (1980) as a method for predicting the task execution time of skilled users on UI design ideas before any code had been written to implement those ideas. The procedure for doing a KLM was to list all the overt actions that a user would have to take to accomplish the task: keystrokes on a keyboard or mouse clicks (K), pointing with a mouse (P), moving the hand between the mouse and keyboard (homing, H), and drawing (D, on a very constrained grid in a particular CAD system). The modeler then placed a single type of mental operator (M), to represent all the unobservable operations a user would perform, e.g., eye movements, memory retrievals, decisions, using a set of five heuristics defining where the Ms should appear in the model. These heuristics made distinctions between commands and arguments and depended on ill-defined terms like a “cognitive unit”. Finally, if the system required its user to wait for it to respond, an R operator was included in the model.

Quantitative estimates for the KLM operators were established empirically (except for R, which must be estimated for each system), e.g.,  $K=0.2s$  for an average skilled typist,  $P=1.10s$  for the average display size in 1980 (but could be calculated using Fitts’s Law),  $H=0.4s$ , and  $M=1.35s$ . The modeler then added up these estimates to predict skilled execution time on the entire task. Doing a KLM “by-hand” means following this procedure using a spreadsheet to list the operators and do the addition.

I examined eight novice modelers’ KLMs in detail to discover if systematic errors could be identified (John 1994). Comparing to the 87 operators that comprised a KLM that I created for these four tasks, that examination revealed several common errors.

1. Novice modelers leave out overt steps necessary to do the task. If you were to follow the exact Ks, Ps, and Hs listed in their KLMs, you would not complete the tasks successfully. Of all the overt operators left out by novices 31% were Hs, 31% were Ks, and 22% were Ps. Seven of eight modelers exhibited this error.
2. Conversely, three of eight novice modelers included extra overt operators, Ks and Ps that were not necessary to do the task.
3. Finally, all novice modelers seemed to find it very difficult to apply Card, Moran and Newell’s heuristics for placing M operators. Some novices put in extra Ms in one place and omitted Ms from other places in the models, but all novice KLMs included more Ms than my KLMs for the same tasks.

This last problem has been exacerbated by the arrival of modern UIs. KLM was created in the era of command-line interfaces and command-based text editors, where it was relatively clear when something was a command or an argument. With direct-manipulation UIs, this distinction blurs. For example, when a user double-clicks on a word in a text-editor, is that operating on an argument or issuing a command to highlight the word? Card, Moran and Newell’s heuristics are still applicable, but it takes interpretation and increasingly more experience to apply them to UIs as they evolve further from command-line operations.

In the early 2000s, under the support of ONR’s Affordable Human Behavior Modeling Program, the above error analysis was one of several human-centered design (HCD) techniques used to design CogTool. The aim of the CogTool Project is to create a tool that allows UI designers to use predictive human performance modeling to evaluate their design ideas quantitatively before investing resources in programming those ideas. We used the aforementioned error analysis to guide the design of CogTool so that it would eliminate the identified errors as much as possible. We used Contextual Inquiry (Beyer and Holzblatt, 1998) to understand the pain points of cognitive modelers and how such a tool would fit into the workflow and culture of UI designers. We used competitive analysis to understand what had already been tried in this regard (Baumeister et. al, 2000), and a series of usability analyses (Cognitive Walkthrough (Polson., et. al. (1992), think-aloud usability studies, and, yes, KLM with an early version of CogTool itself). All results from these analyses were fed into the design of CogTool, and continue to be, so that CogTool is now being used in real-world design and evaluation processes and taught to hundreds of HCI, UI design, and Human Factors students and professionals each year.

To do a KLM with CogTool, a modeler follows a very different procedure from doing a KLM by hand. Instead of listing overt operators in a spreadsheet divorced from a UI design, the modeler expresses the UI design in a graphical storyboard by placing pre-established widgets (e.g., buttons, check boxes, text fields) in frames that represent what users would see as they progress through a task. The modeler then connects those frames by drawing a transition from a widget to another frame, which represents the user’s action that would cause the screen display to change (e.g., clicking on a button, typing on the keyboard). Finally, the modeler demonstrates a particular task on the storyboard, which creates a KLM by demonstration. CogTool creates ACT-R code (Anderson, et. al., 2004) from this demonstration and runs it to get the prediction of skilled execution time.

CogTool automatically places Ms consistently and in the correct position as suggested by Card, Moran and Newell's heuristics applied to modern UI widgets. Thus, CogTool has transformed the modeling process to a design process, where modelers decide what type of widget to use in their design rather than decide where a user might have to stop and think, addressing error (3) mentioned before. Errors (1) and (2) were addressed by the "modeling by demonstration" on the storyboard, as we surmised that modelers would be less likely to leave out or insert Ks and Ps if they were looking at a picture of the actual interface. Likewise, "bookkeeping errors" like forgetting to home the hand between devices should be eliminated because CogTool keeps track of where the simulated hand must be and automatically places H operators.

CogTool is now at a point where we can examine if it has met any of its aims. John et. al. (2004) demonstrated that a novice modeler could produce model estimates as well as an expert modeler. This paper examines whether CogTool has reduced the variability in novice modelers' models.

### 3. Data

I assembled data from previously-published papers that reported the results of groups of novice modelers creating KLMs on the same interfaces and tasks (Groups1&2), and from unpublished exercises in university classes (Groups3&4), to establish the variability of predicting skilled execution time with the original formulation of the KLM. I then acquired new data on 100 novice modelers using CogTool to investigate the variability of prediction with that modern tool.

#### 3.1 Previously-collected data: Performing KLMs by hand

##### 3.1.1 The interfaces and tasks

The groups who created KLMs by-hand were predicting the performance of skilled users of two telephone-number look-up systems described by Nielsen & Phillips (1993). The first interface, Design A Dialog Box, used menu selection, then a dialog box in which a telephone number was typed into a text field, and then a series of mouse clicks on on-screen buttons to submit a query. The second interface, Design B Pop-Up Menu, submitted the query through context menus accessed by clicking on displayed telephone numbers. Each modeler created four KLMs, looking up one telephone number and looking up two telephone numbers, on each of two interfaces. The predicted task

execution times for these four tasks range from 5s (PopUp-1) to 22s (DialogBox-2).<sup>1</sup>

Ideally, designers of the system who know the screen layout and procedures for accomplishing tasks well are the people who create predictive human performance models for UI evaluation and design. To simulate this familiarity, the modelers were given step-by-step instructions showing what would be on the screen and what actions to take at each point in the tasks. We are looking for variability in the models they produce, not variability in how well they understand the interfaces, so this level of direction is appropriate and was used in all groups analyzed here.

##### 3.1.2 ByHand-Group1

The data for ByHand-Group1 was published by Nielsen and Phillips (1993). The modelers were described as "19 upper-division undergraduate students in a human-computer interaction class as their second assignment using GOMS." Actually, the Keystroke-Level Model [1] was performed, not a full GOMS model (Erik Nilsen, private communication, 6 Sept 1993). Although no information was published about the instructional sessions or materials given to these students, it is likely that they were given one of the two publications about KLM by Card, Moran and Newell (1980 or 1983), as they were the readily available. Nielsen and Phillips reported means and standard deviations for each of the four models for these 19 novice modelers. Because the magnitudes of the task execution times vary, the coefficients of variance (CV = standard deviation/mean) is calculated and appear on the first line of data in Table 1.

##### 3.1.3 ByHand-Group2

The data for ByHand-Group2 was published by this author (John,1994). The modelers were "eight Carnegie Mellon undergraduate students at the end of their first HC1 class." The class was an elective offered in the computer science department, although students from other disciplines attended. These student had one lecture on KLM, one prior homework assignment on KLM, and Card, Moran and Newell 1980 was a required reading in the class. I "reproduced the Nielsen and Phillips interfaces from their descriptions" to create the materials given to the modelers. The means

<sup>1</sup> The purpose of KLMs is to predict skilled execution time and Nielsen and Phillips (1993) provided empirical data against which to compare those predictions. ByHand-Goup1 had an average absolute percent error of about 30%, whereas ByHand-Goup2, 3 & 4 had about 15%. No user data is available for the tasks and interfaces modeled by the CogTool-Group, regrettable, but not necessary to study variability.

and standard deviations for each of the four models for these 8 novice modelers were converted to CVs and appear on the second line of data in Table 1.

### 3.1.4 ByHand-Group3 & ByHand-Group4

The data for ByHand-Group3 was supplied by Wayne D. Gray (personal communication, November 28, 2009) from classes he taught in 1996 and 2002 using the same materials given to ByHand-Group2. The class, "Cognitive Task Analysis" was a core course in a masters program in Human Factors and Applied Cognition at George Mason University. These students had five weeks of other task analysis lectures but only one lecture on specifically how to do KLM and this was their first assignment using it. They were assigned Chapter 8 of Card, Moran and Newell (1983), which is essentially the same as Card, Moran and Newell, 1980. Twelve modelers were in the 1996 class and nine in the 2002 class. The means and standard deviations for each of the four models for these 21 novice modelers were converted to CVs and appear on the third and fourth line of data in Table 1.

### 3.2 New data: Performing KLMs with CogTool

The data labeled "CogTool" in Table 1 was recently generated in the "HCI Methods" class at Carnegie Mellon University (Fall 2009), which is a required class for the bachelors and masters programs in HCI and about 3/4 of the students class are in those programs. All students in the class are in an undergraduate major other than HCI (the bachelors in HCI is a 2<sup>nd</sup>-major) or already hold a bachelors degree in another major, with about half from a technical background, 1/4 from the behavioral sciences and 1/4 from design in a school of fine arts. The class included 101 students, all of whom completed the assignment. One student had worked as a programmer on the CogTool Project the previous year and was removed from analysis because he had considerably more knowledge of the tool than the other modelers, resulting in an N of 100.

These students had one 1.5-hour lecture on predictive human performance modeling, about 20 minutes of which was a demonstration of CogTool.

John (1995) was required reading and the students were encouraged to download the CogTool User Guide (<http://cogtool.hcii.cs.cmu.edu/using-today/documentation-and-other-support>). There was a 3-hour session where this author, a graduate student, and a programmer were available to answer questions about the mechanics of using CogTool (e.g., "I closed my Project window, how do I get it back?" and "I still have Tiger on my Mac and CogTool's not working, what do I do?"), but not about decisions that would effect predictions. About 2/3 of the students attended this session.

#### 3.2.1 The interfaces and tasks

The interfaces and tasks modeled by this group were considerably more modern than those modeled by the other groups. Pragmatically, a teacher cannot continue using the same assignment for 15 years; students can get the answers from previous classes and they become so dated that they are irrelevant to the students' lives. Therefore, these novice modelers compared two web-based interfaces on three tasks, for a total of six models apiece.

The interfaces were real-world web services for cataloging books and sharing collections on-line: Booktagger (<http://www.booktagger.com/>) (Figure 1) and LibraryThing (<http://www.librarything.com/>). The tasks were (1) sign-in and add a book to your collection, (2) tag the book you just added, and (3) rate that book and sign-out. The task execution times for five of these tasks were on the order of those for the telephone look-up tasks (ranging from 5s to 48s). This



Figure 1. Booktagger page showing information about a book. Interactive widgets on this page include a textbox, buttons, links, checkboxes, and a star rating widgets.

assignment mimics what a designer would do in the real world to benchmark competitors' services before designing a new book-sharing service or the next release of an existing one.

As with the interfaces in the ByHand groups, these modelers were given step-by-step instructions of how to do each task on each interface with pictures of the screens that a user would encounter while doing the tasks. Again, we are looking for variation in predictions due to the modeling process, not due to a modeler's misunderstanding of the interfaces or task procedures, so providing this detailed information is justified.

**3.2.1 The data**

Each modeler produced a quantitative prediction of task execution time and the bottom line of Table 1 shows the CVs for each of these six predictions. In addition, each modeler turned in a CogTool file, which contains all the information relevant to coming up with that prediction. These 100 files were analyzed to understand the source of variance, e.g., the decisions the modelers made that led to different numeric predictions, and will be discussed in Section 5.

**4. Analysis of Difference in Variability**

To determine whether the predictions produced by novice modelers creating a KLM by-hand are more variable than those produced by novice modelers using CogTool, we follow Dow (1976). Dow explored the statistical tests used by ornithologists to study geographical variation in birds from previous

publications reporting only the N, mean and standard deviation (SD) of their observations. The N of these studies is often as small as 5 and, in Dow's exploration, not more than 65, their means often differ in magnitude, and SD is sometimes correlated with the mean and sometimes not. Thus, ornithologists face a situation similar to the data sets I have been able to assemble. Dow explains both a t-test procedure and an F-test procedure, finding each more conservative under different characteristics of the data and concludes that the t-test is marginally better for comparing variability when studies have both small (<22) and large N (>=22) as is the case for the studies compared here.

As the data were reported as individual models for each task on an interface, Table 1 shows 4 models (2 tasks x 2 interfaces) per study, for a total of 16 instances (N-mean-SD triples) in the ByHand condition. In the CogTool condition, Table 1 shows 6 instances (3 tasks x 2 interfaces) of N-mean-SD triples. I calculated the average CV, weighted by N for ByHand (CV=22%), and CogTool (CV=7%), and used Dow's equation to calculate the t value for the comparison:

$$t = (CV_1 - CV_2) / \sqrt{SE_1^2 + SE_2^2}$$

where  $SE = CV / \sqrt{N}$   
 $N_{ByHand} = \sum N_{Group} = 48$   
 $N_{CogTool} = 100$

The resulting difference in CV is highly significant using a 2-tailed t-test as recommended by Dow (t=6.3, df=146 p<0.0001). Thus, we can conclude that the models produced using CogTool are less variable than those produced by-hand.

Table 1. Coefficients of variance (CVs) for each prediction of a task on an interface for four groups using KLM by hand and one group using CogTool.

Group	N	Interface/Task Coefficient of Variance (CV)					
		DialogBox 1 number	DialogBox 2 numbers	PopUp 1 number	PopUp 2 numbers		
ByHand-G1 (Neilsen & Phillips,1993)	19	0.22	0.24	0.14	0.17		
ByHand-G2 (John, 1994)	8	0.22	0.21	0.22	0.21		
ByHand-G3 (1996) <sup>2</sup>	12	0.19	0.13	0.35	0.33		
ByHand-G4 (2002) <sup>2</sup>	9	0.23	0.25	0.21	0.25		
		Booktagger Add book	Booktagger Tag book	Booktagger Rate book	LibraryThing Add book	LibraryThing Tag book	LibraryThing Rate book
CogTool (2009)	100	0.03	0.04	0.12	0.03	0.09	0.13

<sup>2</sup> Data supplied by Wayne D. Gray, personal communication, November 28, 2009

## 5. Discussion of Sources of Variability that Remains in CogTool Models

In addition to the numeric predictions, data exist on exactly what was in every CogTool model and can be analyzed to determine the source of the remaining variability. Unlike the analysis done of the eight by-hand KLMs (John 1994), it is intractable to visually inspect 600 CogTool models. As this had to be done to grade the students' assignments and give them appropriate feedback on their models, we devised a more automated way to focus our attention on deviations from an acceptable model.

CogTool files can be exported to several formats that help with this analysis. First, the demonstrations can be exported to a csv format appropriate for importing into Microsoft Excel. I created an acceptable CogTool file that contained models of all six tasks, exported their demonstrations to csv, and then imported them into Microsoft Excel. I inspected each line in these demonstrations and inserted one line for each possible deviations from the canonical solution. That is, if a line said "Left Click on the Sign-in Button", I inserted four "error lines" for (1) missing the step entirely, (2) using a transition other than a left-click. (3) using a widget other than a button, and (4) inserting an inappropriate system response time. If an interface object could be reasonably construed as more than one widget, e.g., it is often difficult to decide whether some object on a web page is a button or a link, and the decision between these two would not influence the numeric outcome of the models (see the CogTool User Guide, Appendix C, for a description of the equivalent widgets), then I annotated the error-line to allow multiple answers. For example, error-line (3), above, would be changed to "using a widget other than a button or a link." This resulted in 28 steps that could be influenced by modeler decisions, for a total of 164 error-lines, i.e., opportunities to differ from an acceptable solution.

Again, visually inspecting 100 files for 164 possible errors, is intractable. However, a CogTool file also can be exported to an XML representation that preserves all the components of all the models in the file (the frames, widgets, transitions, and demonstrations). I exported my CogTool file to XML and scripts were used to compare this XML to each novice modeler's XML, highlighting those sections that differed in ways important to the results of a model (e.g., when using a different type of widget, but not when giving a widget a different name). With this highlighted file, four teaching assistants then visually inspected the difference between the novice's XML and the canonical XML and entered a "1" in the appropriate error-line in the Excel file for that particular difference.

This resulted in a Excel chart with 164 rows of possible errors, 100 columns of novice modelers and a matrix of 1s and blanks representing the correct decisions (blanks) errors (1s) each novice modelers made. This matrix was manipulated to find the following sources of variability in the CogTool models.

Recall that in the error analysis of by-hand KLM (John 1994), all eight novice modelers deviated from the canonical model. "The student with the least deviation left out only 1 operator and added only 1 extra operator to the instructor's 87 operators. The student with the most deviations left out 25 operators and added 8 operators to the instructor's 87. (John, 1994, p. 286). With the CogTool models, 3 of 100 students did not differ at all from an acceptable model despite 164 opportunities to do so; 26 differed 1-4 times; 17 differed 5-8 times. Therefore almost half the novice modelers (46) made only 5% of the errors that were possible in this exercise. About one quarter (27) made 5-10% of the possible errors and the remaining quarter (27) made 10-20% of the possible errors, with the average being 7% and the median being 6%.

Recall that forgetting an H operator (homing) was very common in by-hand KLMs. CogTool automatically keeps track of the hand and inserts Hs if the hand must move between the mouse and the keyboard to complete the steps, so these types of errors should not occur in CogTool models. There were 11 H operators across the 6 tasks, for a total 1100 H operators possible in the combined novice's models and 128 Hs were missing, the most common type of error in the models. This occurs because, although CogTool keeps track of the hand as it goes through the task, the modeler must tell CogTool where the hand starts at the beginning of each task. The current default is for the hand to start on the keyboard, but all 6 tasks had the hand starting on the mouse (which was told to the modelers in the written assignment). 14 modelers did not set the hand's starting position to the mouse in all 6 tasks; 32 modelers did not set the position at least once. The starting position is set with a pulldown menu in CogTool's interface and may have been accidentally overlooked. We will investigate changing that interaction to a more salient one in future releases of CogTool.

Forgetting other operators (keystrokes, Ks, and pointing, Ps) was also prevalent in KLMs done by hand. However, of the 3500 decisions to insert such a step, only 1% (42) were forgotten by the novice modelers using CogTool. The vast majority of these, 30, were forgetting to click in a text box before typing into it. Both interfaces required this action at some point in the tasks (though not consistently), and (as with the by-hand KLMs) the modelers were told about these steps, so why they forgot them is as inexplicable

in the CogTool case as it was in the by-hand case. Perhaps novice modelers are still overwhelmed with the modeling activities, even with CogTool that if the tool does not enforce every step, novices will “just forget.” If this were a running system rather than a storyboard mock-up, the system would prevent the task from progressing if it indeed worked required a click before typing. However, programming a running system defeats the purpose of predictive human performance modeling. I know of no way to solve this problem at this time, but at least it is reduced to less than 1% of the steps with CogTool. (The other 12 forgotten actions were evenly spread across other steps in the tasks with no apparent pattern.)

Recall that placing M operators was difficult for modelers doing KLM by-hand. CogTool modelers do not place Ms at all; the Ms are placed automatically by CogTool depending on the widget choices. There were 21 widgets necessary to do all 6 tasks. Two of the 21 were more difficult and will be discussed next, but of the 1900 relatively straightforward choices, only 5% (100) contained errors. Of these, 58 were choosing some widget other than a link in 4 different frames. In many modern websites, links don’t follow old visual conventions (e.g., underlined text), so the distinction between links and buttons is murky. However, this choice does not influence the outcome of the CogTool predictions, so this common “error” may be considered more of style than substance.

The next most common error in widget choice was 25 choices of something other than a text box widget in three frames. CogTool distinguishes between text boxes and the text inside them. This distinction does make a difference to the predictions (see Appendix C of the CogTool User Guide) and is a known difficulty for novice modelers. There are several sections written in the CogTool User Guide about the difference between these widgets, when to use each one, and how to use them in concert to mock-up editing text, but this prevalent error indicates that either novice modelers do not read the User Guide or do not understand its information as written. Further investigation is necessary to understand how to eliminate this source of variance through redesign of CogTool itself or the documentation and training associated with it.

Two of the widgets in the models were quite difficult because they did not map directly to widgets supplied by CogTool. Both systems had a rating feature where a user clicks one of 5 stars to rate the book. Is each star a button widget? Is the set of stars equivalent to a set of radio button widgets? The novice modelers were asked to choose a widget and justify that choice. The scoring judged the justification, not the actual choice of widget. Thus, the 58 errors (29 modelers making the same error in both systems) were more for the modeler’s ability to

articulate their decision as opposed to actually making the right decision (which is to represent them as a set of radio button widgets). As new interaction styles are designed, modelers will encounter this problem of mapping CogTool’s widgets to those interaction styles. How to best do so, or grow CogTool’s widget set to accommodate innovative design, is an area for further research.

## 6. Conclusion

The evidence seems clear; CogTool has achieved its aim to reduce the variability in models created by novice modelers. In fact, with an average CV of 7%, it is the least variable of any usability evaluation technique studied to date. We attribute this success to using HCD methods (Contextual Inquiry, error analysis, usability evaluation, etc.) in the development of the CogTool. Modelers are simply another type of user and HCD methods (despite their variability), when used in concert and when they provide converging design advice, simply work.

## 7. Acknowledgements

The author thanks Joanna Bresee Brian Lim, Min Kyung Lee, Bryan Pendelton, (the teaching assistants in HCI Methods F09) for their careful identification of errors in the CogTool models, Wayne Gray for archiving and contributing data, and to Nicholas Yee for his help in the statistical analysis. This research was supported in part by funds from ONR (N00014-03-1-0086), NASA, NEC, PARC, and IBM. The views and conclusions in this paper are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of ONR, NASA, NEC, PARC, IBM, or the U.S. Government.

## 8. References

- Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review*, *111*(4), 1036-1060.
- Baumeister, L., John, B. E. & Byrne, M. (2000) A Comparison of Tools for Building GOMS Models. *Proceedings of CHI, 2000* (The Hague, The Netherlands, April 1-6, 2000) ACM, New York. pp. 502-509.
- Beyer, H., & Holtzblatt, K. (1998). *Contextual design: Defining customer-centered systems*. San Francisco, CA: Morgan Kaufmann Publishers.
- Card, S. K., Moran, T. P., & Newell, A. (1980). The keystroke-level model for user performance time with interactive systems. *Communications of the ACM*. *23* (7), 396-410.

- Card, S. K., Moran, T. P., & Newell, A. (1983) *The Psychology of Human-Computer Interaction*. Mahwah, NJ: Lawrence Erlbaum Associates.
- Chaudhri, V. K., Clark, P. E., Mishra S., Pacheco, J., Spaulding, A., Tien, J. (2009) *AURA: Capturing Knowledge and Answering Questions on Science Textbooks*. Technical Report SRI International, <http://www.ai.sri.com/pubs/files/1768.pdf>
- Dow, DD (1976). The use and misuse of the coefficient of variation in analysing geographical variation in birds. *Emu* 76, 25–29.
- Gluck , K., A., and Pew, R. W., (2005) *Modeling human behavior with integrated cognitive architectures: comparison*. Mahwah, NJ: Lawrence Erlbaum Associates.
- Gaines, B. R., (1994). Problem statement for Sisyphus: Models of problem solving. *International Journal of Human-Computer Studies*, 40, 187-192.
- Jacobsen, N. E., Hertzum, M. & John, B. E. (1998) The Evaluator Effect in Usability Studies: Problem Detection and Severity Judgments. *Proceedings of the Human Factors and Ergonomics Society 42nd Annual Meeting* (Chicago, October 5-9, 1998), pp. 1336-1340. HFES, Santa Monica, CA.
- Hertzum, M. and Jacobsen, N.E. (2001), The evaluator effect: A chilling fact about usability evaluation methods, *International Journal of Human-Computer Interaction*, 13(4), 421-443.
- John, B. E. (1995) Why GOMS? *interactions*, 2(4). pp. 80-89.
- John, B. E. (2009) *CogTool user guide version 1.1*. Carnegie Mellon University, Pittsburgh PA, September 17, 2009.
- John, B. E., Prevas, K., Salvucci, D. D., & Koedinger, K. (2004). Predictive human performance modeling made easy. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. CHI '04* (pp. 455-462). New York: ACM.
- Lebiere, C., Gonzalez, C., Dutt, V., and Warwick W. (2009) Predicting cognitive performance in open-ended dynamic tasks a modeling comparison challenge. In A. Howes, D. Peebles, R. Cooper (Eds.), 9th International Conference on Cognitive Modeling – ICCM2009, Manchester, UK.
- Nielsen, J. and Molich, R. 1990. Heuristic evaluation of user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Empowering People* (Seattle, Washington, United States, April 01 - 05, 1990). J. C. Chew and J. Whiteside, Eds. CHI '90. ACM, New York, NY, 249-256.
- Polson, P.G., Lewis, C., Rieman, J., and Wharton, C. (1992). Cognitive walkthroughs: A method for theory- based evaluation of user interfaces. *International Journal of Man-Machine Studies* 36, 741-773.

## Author Biography

**BONNIE E. JOHN** (B. Eng. 1977, The Cooper Union; MS 1978, Stanford; PhD, 1988 Carnegie Mellon University), is a Professor, founding member of Carnegie Mellon University's Human-Computer Interaction (HCI) Institute, and a member of the ACM SIGCHI Academy. She has been researching human behavior modeling and using it to guide HCI design since 1983. As the Director of the Masters program in HCI, Dr. John has researched and taught many HCI design and evaluation techniques. She has brought these experiences together, through human-centered design and automating substantial portions of the modeling process, to create modeling tools that are easier to use, one of which (CogTool) is described in this paper.

# Levy Distributed Search Behaviors for Mobile Target Locating and Tracking

*William Lenagh, Prithviraj Dasgupta*

Computer Science Department  
 University of Nebraska, Omaha, NE 68182.  
 402-554-2380

{wlenagh,pdasgupta}@mail.unomaha.edu

Keywords: Mobile target tracking, Levy flight, autonomous robots

**ABSTRACT:** *We consider the problem of tracking visually identifiable mobile targets using a distributed system of mobile robots. We propose a behavior-based approach where mobile robots with limited sensory range use a search pattern observed in nature - the Levy distributed search, to locate a mobile target. The Levy search pattern is inspired by the foraging pattern exhibited by social insects such as honeybees, albatrosses, etc. We consider two Levy-distributed search patterns - a Levy timed search and a Levy looped search, and determine their performance in locating and tracking mobile as well as stationary targets. Our results show that for locating stationary targets, the Levy length for a search leg is strongly correlated with the distance of the target from the location where the search starts. For locating and tracking mobile targets, we find that the search performance improves as the p.d.f. of the Levy distribution is made flatter. The Levy looped search also performs better than the Levy timed search in tracking mobile targets because its looping property helps in relocating targets that have been observed previously.*

## 1 Introduction

Over the past few years, autonomous robots have been used extensively for unmanned search and reconnaissance related operations in different domains such as unmanned search and rescue, exploration and mapping of unmaneuverable regions, surveillance and patrolling of high-security regions to restrict access, etc. Visually tracking the movement of mobile targets within an area of interest (AOI) is an essential operation during search and reconnaissance. Recently, there have been several efforts to perform search and reconnaissance using multiple mini-robots or mini-UAVs (unmanned aerial vehicles) that operate as a cohesive unit such as a swarm or a fleet. The evident advantage of using a swarm of mini-robots is the considerable reduction in the costs of fielding a large system of mini-robots as compared to operating larger robots. Swarms of robots are also robust because they do not have a single point of failure where the system can be compromised. However, mini-robots typically have limited capabilities such as limited sensor range and accuracy, limited on-board memory and limited computation capabilities. Because of these limited capabilities, it becomes very challenging to perform complex operations such as visually track-

ing mobile targets using mini-robots. To address this challenge, several systems have been proposed that use emergent, swarm-based techniques with simplistic behavior patterns on each robotic swarm unit and allow more complex behaviors to *emerge* from the local interactions of the swarm units. Such behavior-based systems are particularly attractive because the inherent operation of each swarm unit or robot is simple and it is easy to implement and modify such behaviors.

In this paper, we consider a behavior-based system where robots use a nature-inspired search pattern called the Levy-distributed search that is observed in many social insects and animals, to visually (re)acquire and track mobile targets. We compare two types of Levy search patterns - the timed Levy search and the looped Levy search and determine their relative performance in locating and tracking stationary and mobile targets. Our experimental results with simulated mini-robots within the Webots simulator show that the two types of Levy search patterns perform comparably in locating targets, both stationary and mobile. However, the Levy looped search performs better in tracking mobile targets because its looping property helps in relocating targets that have been observed previously.

## 2 Related Work

One of the earliest techniques to track mobile targets using a distributed multi-robot system was described in [3] using the CMOMMT (Cooperative Multi-robot Observation of Multiple Moving Targets) approach. In CMOMMT, robots experience attractive forces towards targets and repulsive forces between each other. Robot motion strategies using both unweighted and weighted force vectors are reported to perform significantly better than random robot movement in simulation as well as on real robots. [1] describes and implements a technique for mobile target tracking that disperses robots based on robots' density within a region and robots' visibility of targets. Each robot is provided with *a priori* knowledge of the environment in the form of a topological map. All the above mentioned approaches rely primarily on the ability of robots' sensors such as sonar or camera, to identify and track mobile targets efficiently. In contrast, we consider robots that have limited sensory range and noisy sensors, and rely on the emergent behavior of the system to locate targets. A pursuit-evasion game(PEG) is another approach that has been used to solve a problem similar to mobile target tracking. In a PEG, the mobile targets are called evaders while the robots tracking the mobile targets are called pursuers. The objective of a PEG is to maximize the probability of locating the pursuers by the evaders. Several techniques for solving pursuit evasion games have been proposed which range from control theory[6], to probabilistic analysis [7], computational geometry[2], and algorithmic analysis[5]. PEGs involve considerable computation either on-board robots or at a centralized location where the information obtained by the robots from the environment is uploaded. In contrast, we consider lightweight robots with limited computation capabilities that might not be amenable to implement complex calculation.

## 3 Levy-Distributed Search

A Levy search is essentially a random walk pattern comprising of several short segments interspersed with turns at random angles. The lengths of the straight line segments are sampled from a stable probability distribution called the Levy distribution given by:

$$L(x, c, \mu) = \sqrt{\frac{c}{2\pi}} \times \frac{\exp\left(\frac{-c}{2(x-\mu)^{\frac{3}{2}}}\right)}{(x-\mu)^{\frac{3}{2}}} \quad (1)$$

where  $c$  is the scale parameter that controls the height of the curve and  $\mu$  is the shift parameter that shifts the mean value of the curve. A sample Levy distribution is shown in Figure 3.

The Levy distribution is particularly attractive from a behavioral perspective because certain species of animals have been shown to exhibit the Levy search as an optimal search strategy for locating a mobile resource such as a food source, or a specific location of interest such as their nest. Levy distribution-based techniques have also been successfully applied to other disciplines where stochastic processes are of great interest, such as geology, finance, cryptography and signal analysis. The specific scenario used in this paper is inspired by the search behavior observed in honeybees [4]. In this scenario, honeybees start out from their nest with *a priori* knowledge of the location of an object of interest such as a flower bed, and move towards its location. However, upon arriving at the location they are unable to locate the object of interest and infer that it has either moved or been depleted. The bees then execute a search pattern, that has been empirically shown to follow a Levy distribution, to reacquire the resource or discover a similar resource nearby. The Levy distribution itself has several properties that make it especially of interest. First, it is a stable distribution which has expressible probability density functions that describe the probability as a continuous function of independent variables. Levy distributions are also *scale-free* which means that their statistical properties remain the same regardless of what scale they are being observed from. The Levy distribution also has a heavy tail which means that the probability of the independent variable drops off slowly as it expands away from the mean, making these values more likely to occur than in other distributions such as the normal distribution. There are two types of Levy-distributed searches that are observed in nature:

- **Levy timed search:** In the Levy timed search, each swarm unit moves in a straight line segment for a random distance that is sampled from the Levy distribution. At the end of each segment, each swarm unit selects a random heading from  $U[0, 2\pi]$  and the next segment starts off from the location where the previous segment ended. The swarm units performing a Levy timed search exhibits a random walk pattern consisting of a series of straight line segments.
- **Levy looped search:** The Levy looped search is essentially similar to the Levy timed search with the exception that at the end of each segment, each swarm unit reverts to the location from which the previous segment started. The swarm units performing a Levy looped search therefore exhibit a loop-like pattern where the length of each loop is sampled from the Levy distribution and the angle at which each loop starts is sampled

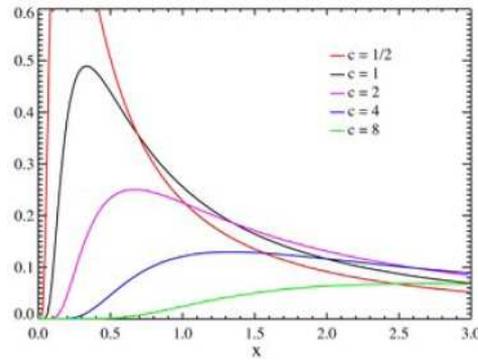


Figure 1: Levy distributions for different values of the scale parameter  $c$

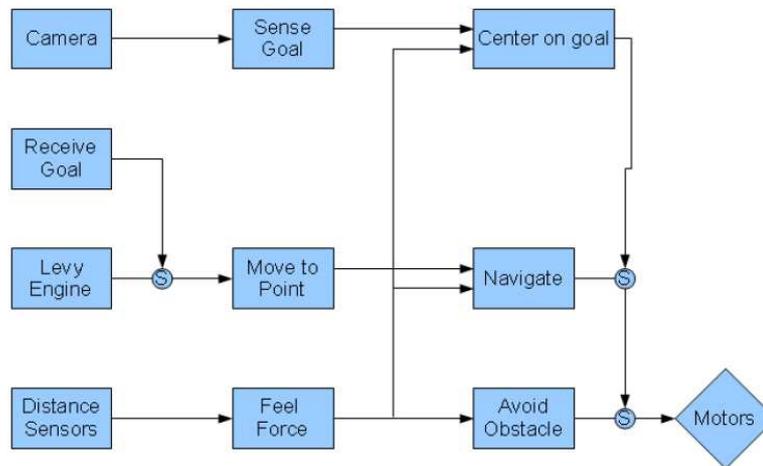


Figure 2: The subsumption architecture based controller of a robot performing a Levy loop search

from the uniform distribution  $U[0, 2\pi]$ .

## 4 Levy Flight Controllers

The controller program of a robot using the Levy search is implemented using a subsumption reactive architecture, as shown in Figure 3. The most primitive behavior, and lowest on the subsumption diagram, is an obstacle avoidance system. Reading the values reported by the distance sensors, this system computes the force of any nearby object using a Braitenberg controller and outputs a resulting speed value based on these computations. Above this level is a more sophisticated *Navigate* behavior which subsumes the output from the *Avoid obstacle* behavior, if present. This behavior takes the input from a Braitenberg controller that calculates the virtual forces on the robot from obstacles based on the distance sensors' readings. The output from the Braitenberg controller is then combined with another

input, *Move to point*, that is driven by either the Levy engine or a goal coordinate received by a transmission from another robot. The *Navigate* behavior directs the motion of the robot while taking into account any obstacles that may be present. The highest level behavior is the *Center on goal* and incorporates both obstacle avoidance and a goal sensing algorithm driven by the image rendered by the robot's camera. The output from this behavior subsumes the output from *Navigate*, which in effect overrides all other behaviors. When active the robot will ignore any goal point and attempt to follow and identify the stimulus which activated the behavior. If it loses contact it will resume navigating, as this output will no longer be subsumed.

**Levy Engine.** The Levy engine implements the Levy flight behavior. A flowchart showing the operation of the Levy engine is shown in Figure 4. The Levy engine can operate either in the loop search mode or in the timed search mode to implement the two types of Levy search patterns. In the loop search mode, the

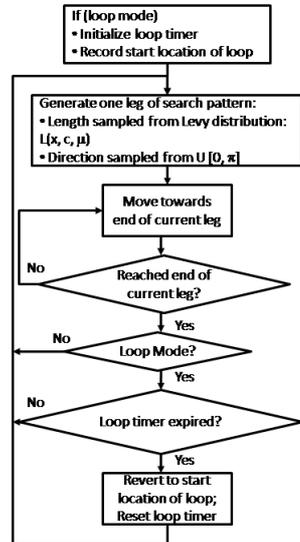


Figure 3: Flowchart showing the operation of the Levy engine.

engine first initializes a loop timer and records the start location of the loop so that the robot can revert to this location after the loop timer expires. It then generates one leg of the Levy search which consists of the distance that the robot will travel (generated from the Levy distribution given in Equation 1), and the heading that the robot will take (drawn from  $\frac{\pi}{2} + \phi$  where  $\phi \in U(0, \pi)$ ). The new heading is offset by  $\frac{\pi}{2}$  because a change in orientation is defined to occur only at angles greater than  $\frac{\pi}{2}$  from the current heading[4].

## 5 Experimental Results

We have tested our Levy search based mobile target following algorithm within the Webots 6.1 simulator. The main objective of our experimental results is to determine how the locating time of targets is affected for different parameters of the Levy search, The two parameters that control the behavior of the Levy search are the scale parameter  $c$  and the shift parameter  $\mu$ . For all our settings we use five robots to locate and track targets and one target that can be either stationary or mobile. The robots are situated with a  $10 \times 10$  m<sup>2</sup> square environment. Each robot is simulated as a mini-robot that has the following sensors: (1) Camera: a color VGA camera with a maximal resolution of  $640 \times 480$ . (2) Eight infra-red distance sensors measuring ambient light and proximity of obstacles in a range of 4 cm. (3) Two wheels controlling speed and

direction by the rotation of stepper motors, and, (4) A Bluetooth-enabled transmitter and receiver for sending and receiving messages between robots. To localize each robot, we have added a GPS node on each simulated robot. (In a system with real robots, localization can be realized using an overhead camera-based localization system.) Mobile targets are simulated as colored cylindrical robots, which can either remain stationary or move in the environment at a certain speed. The robots simulating the mobile targets have two forward looking IR distance sensors to avoid obstacles. When the tracking robot's camera encounters a colored object of interest<sup>1</sup>, it informs other robots that converge on the last observed location of the target and perform a Levy search to locate it.

For our first set of experiments we considered a target that moves from an initial location to a final location and remains stationary after that. The distance between the initial and final locations of the target has an average value of 4.5m. The robots are only aware of the initial location of the target and have to discover the final location of the target using a Levy search starting from the target's initial location. Figure 5 shows the effect of different values of the shift and scale parameters on the time required to locate the target at its final location. The scale parameter  $c$  was set at either 0.5 or 1, while the shift parameter,  $\mu$ , was varied from 0.5 to 1, 2, and 4. With the Levy looped search, we observe that as the length of a leg of the Levy search,

<sup>1</sup>To determine the color of an object perceived on the camera, a tracking robot calculates the average of the R-G-B pixel values for all the camera pixels and determines the object's color as the pixel-color with the highest average value.

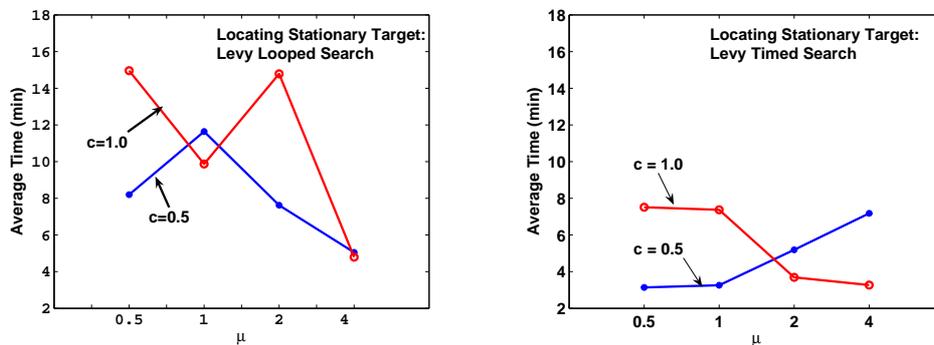


Figure 4: Average time required locate a target that moves from an initial location to a final location and remains stationary thereafter using Levy looped search (left) and Levy timed search (right).

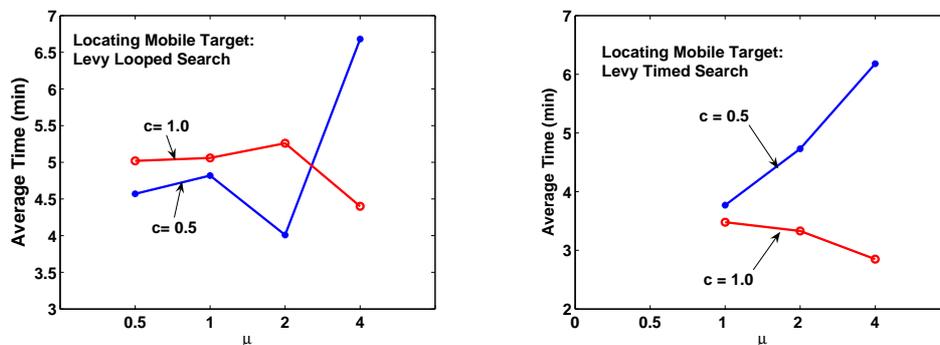


Figure 5: Average time required to locate a mobile target that moves at half the speed of the tracking robots, using Levy looped search (left) and Levy timed search (right).

determined by the shift parameter  $\mu$ , approaches the mean distance of the target’s initial and final locations, the search times successively improve. The best search time occurs when  $\mu$  is set to 4 which is closest to the average distance between the initial and final locations of the target (which is 4.5 m). A similar behavior of the search performance is observed for the Levy timed search when the scale parameter  $c = 1.0$ . However, the performance of the Levy timed search deteriorates for increasing values of  $\mu$  when  $c = 0.5$ . This can be attributed to the fact that when  $c = 0.5$ , the search legs that are closer the value of  $\mu$  are selected with higher probability. As  $\mu$  increases, the search legs are longer and unsuccessful searches tend to persist longer resulting in lower search performance. This behavior is not observed with the Levy looped search as the robots “loop back” to their start location after a certain time and are able to explore different directions around the start location more effectively.

For our next set of experiments, we analyzed the performance of the Levy search on locating and track-

ing a mobile target. All other parameters for the experiment are retained from the previous experiment. The target moves at half the speed of the tracking robots. We used the same combination of Levy parameters as was used for the previous experiment. Figure 5 shows the effect of different values of the shift and scale parameters of the Levy distribution on the time required to locate the target. As before we observe that searches with  $c = 0.5$  result in lower performance because higher persistence for longer search legs (with higher values of  $\mu$ ) can misguide the search in directions where the target is not present.

Figure 5 shows the effect of varying the parameters of the Levy distribution on the time for which the target is observed (tracked) by at least one robot. For the Levy looped search we observe that changing the scale parameter  $c$  from .5 to 1 has the effect of improving the ability to track the target for lower values of  $\mu$ . Similarly, the tracking capability decreases as  $\mu$  increases. On the other hand, when  $c$  is .5, the tracking time increase as  $\mu$  increases. This seems to indicate that lower

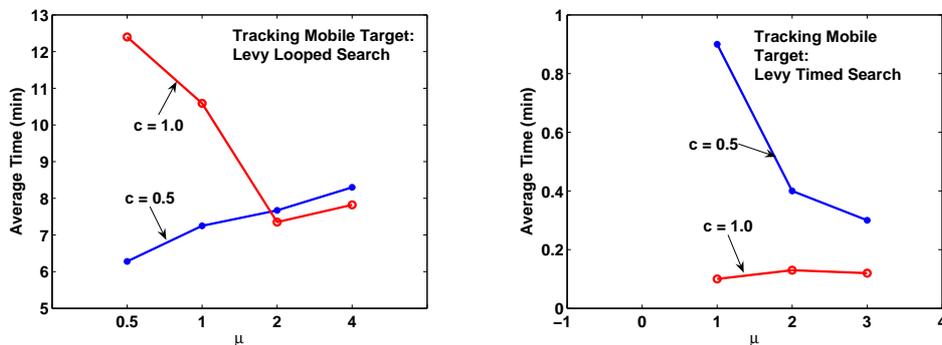


Figure 6: Average time for which a mobile target is tracked by at least one tracking robot using Levy looped search (left) and Levy timed search (right). The mobile target moves at half the speed of the tracking robot.

values of  $\mu$  improve the target tracking times due to the flatter Levy distribution curve resulting when  $c$  is set to 1. The Levy timed search performs very poorly as compared to the Levy looped search for tracking a mobile target. This indicates that looping back to the location where the target was last observed helps in relocating the target and improves the performance of the Levy search. Based on the experimental results reported here, we can infer that a lower value of the scale parameter of the Levy distribution ( $c = 0.5$ ) results in more persistent searches which can result in searches going down the wrong path for longer durations and adversely affect the performance of the search. Also, the closer the shift parameter  $\mu$  of the Levy distribution is to the distance between the start location of the search and the location of the target, the better is the search performance. Finally, between the Levy looped search and the Levy timed search, we observe that their performance is comparable in locating targets (stationary or mobile), but the Levy looped search outperforms the Levy timed search in relocating and tracking mobile targets because of its looping property.

## 6 Conclusion and Future Directions

This work represents our first step in using Levy search for mobile target tracking. Our results show that the parameters of the Levy search can be adjusted appropriately to fine tune the performance of mobile target locating and tracking using mobile robots. In the future, we plan to investigate improved search strategies that dynamically adjust the parameters of the Levy distribution based on the search performance, and mechanisms for tighter coordination between robots after a target is located by one robot. We envisage that with

appropriate techniques along the lines described in this paper, mobile target following with aerial mini-robots will emerge as an important direction for multi-robot systems.

## References

- [1] Jung, B., Sukhatme, G. (2005), "Tracking targets using multiple robots: the effect of environment occlusion," *Auton. Robots*, vol. 13, no. 3, 191-205.
- [2] Isler, V., Kannan, S., Khanna, S. (2006) "Randomized Pursuit Evasion with Lical Visibility," *SIAM J. Discrete Mathematics*, vol. 1, no. 20, 26-41.
- [3] Parker, L. (2002) "Distributed algorithms for multi-robot observation of multiple moving targets," *Auton. Robots*, vol. 12, no. 3, 231-255.
- [4] Reynolds, A., Smith, A., Reynolds, D., Carreck, N., Osbourne, J. (2007) "Honeybees perform optimal scale-free searching flights when attempting to locate a food source," *Journal of Experimental Biology*, vol. 210, 3763-3770.
- [5] Tovar, B., LaValle, S., "Visibility-based pursuit-evasion with bounded speed," (2006) *Proc. Workshop on Algorithmic Foundations of Robotics*.
- [6] Vercaturen, T., Guo D., and Wang, X. (2004) "Joint multiple target tracking and classification in collaborative sensor networks," *IEEE Journal on Selected Areas in Comm.*, vol. 23, no. 4, pp. 714-723, 2004.
- [7] Vidal R., Shakernia, O., Kim, H., Shim, H. and Sastry, S. (2002) "Multi-agent probabilistic pursuit evasion games with unmanned ground and aerial vehicles," *IEEE Trans. Rob. and Auton.*, vol. 18, no. 5, 662-669.

## Author Biographies

**WILLIAM LENAGH** is a Master's student in the Computer Science Department at the University of Nebraska at Omaha. His interests are in the field of artificial intelligence, multi-agent systems and swarming.

**PRITHVIRAJ DASGUPTA** is an associate professor with the Computer Science Department at the

University of Nebraska at Omaha. His research interests are in the area of multiagent and multi-robot adaptive systems, swarm robots, and game theory and computational economics. His research is actively funded by federal agencies including the U.S. Department of Defense and NASA. He has published over 40 papers in leading conferences and journals in the area of multi-agent and multi-robot systems.

# Human Capacity Development Through Simulations: Constructive Simulations as a Basis for Understanding Competency Requirements in Initiative Based Tactics

*Bruno Emond*

National Research Council Canada  
Institute for Information Technology  
1200 Montreal road,  
Ottawa, ON. K1A 0R6  
1-613-991-5471  
bruno.emond@nrc-cnrc.gc.ca

Keywords: Agent-based modeling and simulation, Cognitive modeling, Initiative-based tactics

**ABSTRACT:** This paper presents the state of development of a constructive simulation to better understand competency requirements in initiative based tactics in order to support training scenario design in a virtual training environment. The simulations of interest are cognitive models. The first section situates the development functions of understanding, training, and assisting human capabilities, in relationship to the traditional distinction of live, virtual and constructive simulations. The human development and their associated simulation types can also be laid out on a continuum of agent embedment in physical settings. The second section presents relevant cognitive modeling and simulated environment elements required by initiative based tactics; as well as some initial requirements for training scenario design. A conclusion summarizes the paper and indicates some future work possibilities.

## 1. Introduction

Agent-based modeling and simulation (Macal & North, 2007) is an important element for the development of the next generation of simulators. In particular, training simulations requiring human communication and interaction demand high cognitive fidelity, which must be measured not only by the avatars' physical appearance but also by their psychological and cognitive realisms from a trainee's point of view (Liu, Macchiarella, & Vincenzi, 2009), including natural language processing capabilities (Gluck, Ball, Gunzelmann, Krusmark, & Lyon, 2005).

There are many definitions of what an agent is but the following characteristics seem to describe adequately what being an agent means (Macal & North, 2007). An agent is an identifiable, discrete individual. It is autonomous and self-directed (goal driven); it is situated, living in an environment with which it interacts with other agents (having perceptual, motor, and communication capacities); and it is flexible, having the ability to learn and adapt its behaviors based on experience. Agent-based modeling is divided in two communities, one focused on large numbers of relatively simple and highly-interactive agents; and the other one focused on a

smaller number of agents with more complex internal structures (Guerin, 2004). The current research falls into the second category, and uses the ACT-R cognitive architecture as a means to develop agents (Anderson, 2007; Anderson, et al., 2004).

This paper presents the state of progress of an agent-based modeling and simulation research and development activity as part of a larger project to build a virtual training environment for initiative-based tactics. This virtual training environment, the Immersive Reflexive Engagement Trainer (IRET), is developed as a collaborative research effort between the Canadian Department of National Defence and the National Research Council Canada (Institute for Information Technology). The purpose of IRET is to blend a number of existing technologies to allow soldiers to train simultaneously within virtual and real environments. The primary use of the system is to train personnel in the rapid application of judgment to include the application of rules of engagement and the use of force. The system will provide interactive enemy forces that react to the soldiers' actions and movements, challenging the soldiers' skills and judgment. A secondary purpose of the system is to allow

personnel to practice engagement skills with primary and secondary weapons.

The agent-based modeling and simulation research activity within the IRET project has two principal objectives: a) develop high-fidelity cognitive models to be embedded as game agents in a room-size virtual environment; and b) develop detailed performance and learning models of the learners to support instructions. Both objectives are closely related, as realistic agents should have similar behavior to a range of novice to skilled soldiers. These objectives also require technological advancements in large-display interactive devices (Lapointe & Godin, 2005), speech processing, and the measurement of human performance in virtual environments. The cognitive modeling activity will contribute to the goal of applying realistic behavior representations to application environments (Dimperio, Gunzelmann, & Harris, 2008).

Through out the paper, cognitive models and agents will be considered synonymous. However, because the modeling approach is based on the ACT-R cognitive architecture (Anderson, 2007; Anderson, et al., 2004), when a reference is made to a cognitive model, the internal structure of the model is the point of interest, such as the perceptual and motor modules, or the declarative and procedural memory modules. On the other hand, when the point of interest is not the internal but the individual and discrete nature of an entity, then the term agent will be used.

The paper also focuses on the role of cognitive modeling and simulations can play in human capability development. The first section presents a conceptual framework to place this role in relationships to: 1) live, virtual, and constructive simulations; 2) human development functions such as understanding, training, and assisting human capacity; and 3) agent embedment in physical settings, from low-embedment in simulated environments, to medium-embedment in virtual environment, to high-embedment in field operations. Ideally, cognitive models could initially be developed as constructive simulations, then carried out and refined during development and deployment in virtual simulations, and eventually deployed as assistive agents to be part of the soldier's system.

The second section of the paper gives an overview of a constructive simulation composed of agents, and the simulated environment they live in. Finally, a conclusion summarizes the paper and indicates some future work possibilities.

## 2. Human Capability Development Through Simulations

The distinction between constructive, virtual and live simulations is sometimes a useful one even though the boundaries are often blurred, unique category assignment is not possible, and real systems controlled by artificial agents are not considered in the classification (Department\_of\_Defense, January 1998). The distinction is essentially based on the presence of real or simulated equipment with real or simulated human operators as outlined in Table 1.

Table 1. Simulation classifications and human capacity development functions.

	<b>Real Human</b>	<b>Simulated Human</b>
<b>Real Equipment</b>	Live Simulations	Autonomous Agents <i>[Assisting]</i>
<b>Simulated Equipment</b>	Virtual Simulations <i>[Training]</i>	Constructive Simulations <i>[Understanding]</i>

Live simulations are essential and key to many training operations, tactical exercises without troops within a local community (Burton, 2006), however a lot of attention is given to computer simulations as a means of reducing equipment and training cost, but mostly to save lives by providing efficient and progressive training (Hayward, 2006; Roman & Brown, 2007). When the focus is placed on information technology in simulations, the three relevant simulation types are constructive, virtual and autonomous. From the perspective of human capacity development, other categories also emerge to classify simulations such as simulations for understanding, training, and assisting. Table 1 associates these categories respectively to constructive simulations, virtual simulations and autonomous agents.

Understanding human capabilities is an important aspect of constructive simulations. Research and simulations using Integrated

Performance Modeling Environment (IPME) models (Armstrong, Belyavin, Cain, Gauthier, & Wang, 2007) as well as modeling human-computer interactions using cognitive architectures such as ACT-R (M. D. Byrne, 2001; Emond & West, 2004; Ritter & Young, 2001) are good examples of applications of modeling for understanding human capabilities. The purpose of the modeling effort in a constructive simulation context is to obtain accurate models of perceptual, motor, cognitive, and social skills. The main research trend in this respect consists of ensuring that cognitive models are validated against empirical data collected on human performance and that one can select amongst alternative models (Gluck, Bello, & Busemeyer, 2008). A constructive simulation environment might include not only the computational resources to build cognitive models but also resources to model the environment and collect data on human performance. A typical constructive simulation would have an application that either a human operator or a cognitive model can control. Data collected during human operations can then be modeled, or reproduced by the cognitive model.

Table 2. Cognitive model objects of perception and action, and human-in-the-loop by agent embedment levels

	<b>Objects of perception and action</b>	<b>Human-in-the-loop</b>
<b>Constructive</b> <i>Understanding cognitive processing</i> Low embedment	- Agents - Simulated environment	- Cognitive Modelers
<b>Virtual</b> <i>Training personnel</i> Medium embedment	- Agents - Trainees - Virtual environment	- Trainers - Trainees - Cognitive Modelers
<b>Operational</b> <i>Assisting personnel in the field</i> High embedment (soldier's system)	- Agents - Humans-in-the-field - Physical environment	- Humans-in-the-field

The evolution of models from understanding to assistance is also characterized by more cognitive model embodiment into human operations (Table 2). At the constructive (understanding) level, the objects of perception are restricted to other simulated agents and the simulated environment; the human-in-the-loop is

essentially a cognitive modeler. At the virtual (training) level, the objects of perception and action are other simulated agents, trainees, and a virtual environment; humans-in-the-loop are people involved in training as well as cognitive modelers. A virtual environment is distinguished from a simulated environment because the main purpose of a virtual environment is to be perceived and acted upon by humans, while a simulated environment need only to be perceived and acted upon by cognitive models. Finally, at the operational level (assisting), objects of perception a actions are other simulated agents, humans-in-the-field and the physical environment; humans-in-the-loop are humans-in-the-field.

### 3. Understanding Competency Requirements for an Initiative Based Tactics Training Simulator

Simulators provide many advantages for training. One of the key features is their high fidelity to real-world operating environments. The main argument being that the closer the training environment is to the real world, the better will be the transfer of skills and knowledge acquired during training. However, it is now recognized that a simulator's fidelity must be measured not only by the physical appearance but also by its psychological and cognitive realisms from the trainee's perspective (Liu, et al., 2009). Simulators also offer instructors the capacity to select specific training conditions, as well as detailed recordings of a trainee's performance for the purpose of performance comparison, diagnostic, and evaluation (Moroney & Lilienthal, 2009). Another important aspect of simulators, when applied to skill acquisition, is the capability of going repetitively through a simulation scenario without the cost associated to live simulations. The availability of simulators is crucial to maintain readiness and avoid performance degradation (Gorman, 1990; Proctor & Gubler, 1998).

Constructive simulations are key elements in the development of training simulators. They can be used to help in the acquisition process (National Research Council, 2002), as a foundation for the development of synthetic adversaries (Wray, Laird, Nuxoll, Stokes, & Kerfoot, 2005), as a mean to detail the skills to be acquired in a training simulator, or even to

study the transfer of agent skills (Gorski & Laird, 2007). A broader access to game engines as well as the emergence of new or improved cognitive architectures (M.D. Byrne & Anderson, 2001; Laird, 2008) has allowed the development of many simulation systems of military operations on urban terrain (Best & Lebiere, 2003a; Choi, Konik, Nejati, Park, & Langley, 2007; Cox & Fu, 2005; Evertsz, Ritter, Russell, & Shepherdson, 2007; Ting & Zhou, 2009; Wray & Chong, 2007; Youngblood, Nolen, Ross, & Holder, 2006).

There are very few empirical studies evaluating the knowledge transfer from game playing to effective room clearing operations. However, some results indicate (Proctor & Woodman, 2007) that games could be suitable for the transfer of planning, evaluation, and selection of small-unit tactical operations, but somewhat limited in supporting skill transfer to execution of well-honed techniques involving physical interaction with other people as well as the environment (Proctor & Woodman, 2007). Virtual training room environment have more potential in this respect, but they but be designed using scenario-based training, cognitive task analysis, adequate human-computer interaction strategies, training management systems, and intelligent tutoring systems (Schmorrow, et al., 2009).

Initiative based tactics are driven by the actions and initiative of the individual soldiers. Proper actions must conform to the doctrine and fundamentals of close quarter battle (CQB), but the actions success is highly dependent on the application of skills directed by the challenges of the immediate and specific conditions of a CQB situation. Communication and coordination with teammates, efficient body movements, as well as rapid threats assessment from environmental cues important building blocks of initiative-based tactics skills.

The following paragraphs aim at specifying the competencies to be learnt and the environment affordances to support the acquisition of initiative-based tactics skills in a room-size training simulator. The specification of the perceptual and motor skills as well as the environment affordances will take the form of a constructive simulation based on the ACT-R cognitive architecture.

As the Figure 1 suggests, a constructive simulation needs to identify the high-level primitive perceptual and motor representations essential for a cognitive model to interact with a simulated environment. These primitives constitute the first set of modeling requirements.

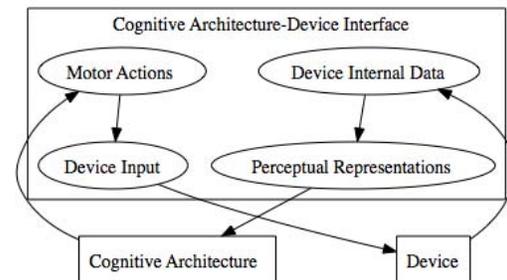


Figure 1. Information flow between a device and a cognitive architecture

The intermediate layer (Best & Lebiere, 2009; Dawes & Hall, 2005) between a cognitive architecture and devices, such as a desktop application or a game engine, can be described by functions transforming internal device data into high-level perceptual constructs feeding in the cognitive model perceptual modules. In the same manner, motor actions get executed in the external device by translating high-level action representations in the cognitive model into device input.

Prior research in CQB tasks analysis and cognitive modeling applications (Best & Lebiere, 2003b; Templeman, Sibert, Page, & Denbrook, 2007; Wray, et al., 2005) provide an initial identification of key perceptual and motor primitives. Table 3 summarizes some of these primitives. The table is divided perceptual and motor modalities. Most of the categories and labels should be relatively easy to understand, such as location and end-points (defined in an egocentric spatial coordinate system), volume, and type. The people category however identifies environmental affordances that are crucial to the assessment of a threat level. Acquired-visual-object and weapon-target for example are the respective projections of the line of sight and weapon pointing direction onto agents in the room. Weapon readiness and potency are also other perceptual factors in threat assessment. A person can also exhibit composition of course and heading variations produce different kinds of body motion such as steering (aligned course and heading); canted (fix alignment offset between course and heading), oblique (constant heading

position), and scanning (free heading movement from the course) (Templeman, et al., 2007).

Table 3. Perceptual and motor cognitive constructs required to operate in a CQB situation. <sup>A</sup> (Best & Lebiere, 2003b); <sup>B</sup> (Wray, et al., 2005); <sup>C</sup> (Templeman, et al., 2007).

Perception Audition	
Verbal messages	Location; Volume; Sender <sup>A</sup> ; Content <sup>A</sup>
Weapon fire	Location <sup>A</sup> ; Volume <sup>A</sup> ; Type <sup>A</sup>
Ricochets	Location <sup>A</sup> ; Volume <sup>A</sup> ; Type <sup>A</sup>
Flash bang	Location; Volume
Footsteps	Location <sup>A</sup> ; Volume <sup>A</sup> ; Direction
Perception Visual	
Non-verbal messages	Sender; Content
Walls	End-points <sup>A</sup> ;
Corners	Location <sup>A</sup> ;
Pathways	End-points <sup>A</sup> ;
Doors	End-points; Hinges-location; Open-state;
Weapons	Location <sup>A</sup> ; Type <sup>A</sup>
Objects	Location <sup>A</sup> ; Type <sup>A</sup>
People	Location <sup>A</sup> ; Type <sup>A</sup> ; Speed <sup>A,C</sup> ; Course <sup>A,C</sup> ; Heading <sup>C</sup> ; Acquired-visual-object; With-weapon; Weapon-potency; Weapon-orientation; Weapon-readiness; Weapon-target
Motor Communication	
Speech	Receiver; Content; Volume
Non-verbal messages	Receiver; Content
Motor Body	
Weapon handling	Type; Trigger-arm&hand; Readiness <sup>B</sup> ; Orientation; Pull-Trigger; Throw <sup>B</sup> ;
Body displacement	Course <sup>C</sup> ; Heading <sup>C</sup> Speed <sup>C</sup> ; Modality
Body rotation	Heading <sup>C</sup> ; Speed <sup>C</sup>

Screen shots of the current implementation of the constructive simulation are given in Figure 2. As Table 3 indicates, most properties can be mapped directly onto a 2D agent visualization representation, however the representation of the agents' prior knowledge and rules is not explicitly represented by the 2D model and could require more advanced visualization techniques (Guerin, 2004; Urbas, Nekrasova, & Leuchter,

2005). Both Figure 2a and 2b contain views of a scene perceived by one agent ACT-R (bottom yellow circle). All other circles are also ACT-R agents. Figure 2a shows what the agent sees, objects and other agents that are in the field of view and not hidden by other objects. Figure 2b shows the full scene, including hidden objects and spatial properties such as corners, end of walls, and pathways between walls (ex. doors). An agent encodes all objects in a scene as an egocentric set of parameters that support threats assessment, and plan execution. The user interface of Figure 2 is also used to drag agents around as initial physical. Initial agent knowledge and plans will also be accessed from the simulated environment user interface.

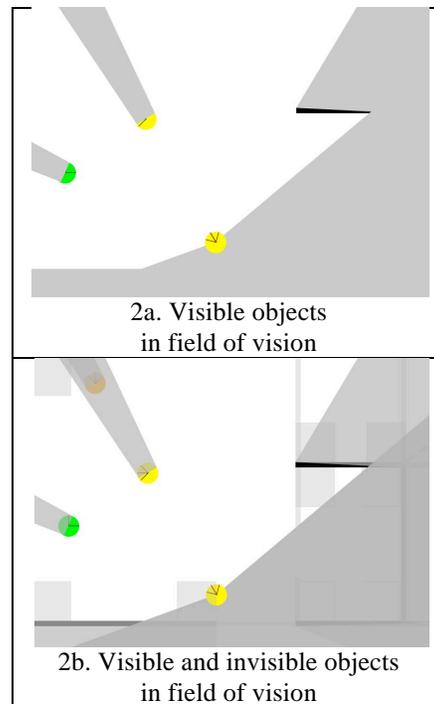


Figure 2. Agent's field of vision in a room with more than 4 walls

#### 4. Conclusion

This paper presented the state of development of a constructive simulation to better understand competency requirements in initiative based tactics in order to support training scenario design in a virtual training environment. This cognitive modeling research activity is part of a larger project to build a virtual training environment, the Immersive Reflexive Engagement Trainer, a collaborative research effort between the Canadian Department of National Defence and the National Research

Council Canada (Institute for Information Technology).

The initial section of the paper presented a conceptual framework where constructive models can be carried out and refined through out development and deployment in virtual simulations and eventually as assistive agents to be part of the soldier's system. The framework situates the development functions of understanding, training, and assisting human capabilities, in relationship to the traditional distinction of live, virtual and constructive simulations. The human development and their associated simulation types can also be laid out on a continuum of agent embedment in physical settings.

The second section presented some primitive perceptual and motor elements as a set of requirements for a constructive simulation of initiative based tactics in close quarter battle. Cognitive models using these primitives in a simulated environment are currently under development.

There is a significant increase in technology complexity from a constructive to a virtual simulation. The main distinctive feature is the intention of the constructive simulation to represent all relevant cognitive and environment features at a high level of abstraction, focusing on requirements, with no immediate concern with providing a high-fidelity training environment. A virtual environment on the other hand aims at presenting objects of perceptual, motor and communication interaction as close as possible to the reality it represents. In this respect, a desktop application fails to provide the proper training environment, which requires trainees to move in space, handle real weapons, and toss flash bangs in a room size space. The coupling between perception and action must be as close as possible to its intended application context (Sanford & Hopper, 2009), using exertion interfaces (Pasch, Bianchi-Berthouze, van Dijk, & Nijholt, 2009), focused on physically moving around the real world and aiming freely at virtual and tangible objects (Zhou, Tedjokusumo, Winkler, & Ni, 2007).

Adversaries will also have to exhibit dynamic behavior with adaptive threats consistent with those increasingly encountered by the military (Jensen, Ludwig, Proctor, Patrick, & Wong, 2008), and ideally, adequate to the level of

trainees' performance. Adversaries can be designed on the basis of the existing teammate model but most than likely adversaries are asymmetric. The training challenge is to present the trainees opponents that have unpredictable tactics, and alternative forms of behavior. These asymmetric and adaptive features are current limitation of virtual training environments (Jensen, et al., 2008).

Observation and analysis of close quarter battle live simulations is currently underway to identify cognitive modeling as well as training requirements. Future work will include cognitive model validation as part of an evaluation of the usability of the IRET system; and separate modeling of opponents' behavior.

## 5. References

- Anderson, J. R. (2007). *How can the human mind exist in the physical universe?* New York, NY: Oxford University Press.
- Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychologica Review*, 11(4), 1036-1060.
- Armstrong, J., Belyavin, A., Cain, B., Gauthier, M., & Wang, W. (2007). *Validation of Workload Algorithms within the Integrated Performance Modeling Environment*. Paper presented at the Conference on Behavior Representation in Modeling and Simulation.
- Best, B. J., & Lebiere, C. (2003a). *Spatial plans, communication, and teamwork in synthetic MOUT agents*. Paper presented at the 12th Conference on Behavior Representation in Modeling and Simulation.
- Best, B. J., & Lebiere, C. (2003b). Teamwork, Communication, and Planning in ACT-R Agents Engaging in Urban Combat in Virtual Environments *Proceedings of the 2003 IJCAI Workshop on Cognitive Modeling of Agents and Multi-Agent Interactions* (pp. 64-72).
- Best, B. J., & Lebiere, C. (2009). Software integration for Computational Cognitive Models in Virtual Environments: The CASEMIL Middleware Infrastructure *Proceedings of the 18th Conference on Behavior Representation in Modeling and Simulation* (pp. 105-112). Sundace, UT.
- Burton, M. G. (2006). An Urban Operations Training capability for the Canadian Army. *Canadian Army Journal*, 9(1).
- Byrne, M. D. (2001). ACT-R/PM and menu selection: Applying a cognitive architecture

- to HCI. *International Journal of Human-Computer Studies*, 55, 41-84.
- Byrne, M. D., & Anderson, J. R. (2001). Serial modules in parallel: The psychological refractory period and perfect time-sharing. *Psychological Review*, 108, 847-869.
- Choi, D., Konik, T., Nejati, N., Park, C., & Langley, P. (2007). A believable agent for first-person shooter games *Proceedings of the Third Annual Artificial Intelligence and Interactive Digital Entertainment Conference* (pp. 71-73).
- Cox, C., & Fu, D. (2005). AI for Automated Combatants in a Training Application *Proceedings of the second Australasian conference on Interactive entertainment* (pp. 57 - 64). Sydney, Australia.
- Dawes, M., & Hall, R. (2005). Towards using first-person shooter computer games as an artificial intelligence testbed, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 3681 LNAI, pp. 276-282).
- Department\_of\_Defense (January 1998). *DoD Modeling and Simulation (M&S) Glossary*.
- Dimperio, E., Gunzelmann, G., & Harris, J. (2008). An Initial Evaluation of a Cognitive Model of UAV Reconnaissance. In J. Hansberger (Ed.), *Proceedings of the Seventeenth Conference on Behavior Representation in Modeling and Simulation* (pp. 165-173). Orlando, FL: Simulation Interoperability Standards Organization.
- Emond, B., & West, R. L. (2004). Using cognitive modelling simulations for user interface design decisions. In B. Orchard, C. Yang & M. Ali (Eds.), *Innovations in Applied Artificial Intelligence: Proceedings of the 17th International Conference on Industrial & Engineering Applications of Artificial Intelligence & Expert Systems* (pp. 305-314). Berlin: Springer-Verlag.
- Evertsz, R., Ritter, F. E., Russell, S., & Shepherdson, D. (2007). Modeling rules of engagement in computer generated forces *Proceedings of the 16th Conference on Behavior Representation in Modeling and Simulation* (pp. 80-89). Orlando, FL: U. of Central Florida.
- Gluck, K. A., Ball, J. T., Gunzelmann, G., Krusmark, M. A., & Lyon, D. R. (2005). A prospective look at a synthetic teammate for UAV applications *Proceedings of the American Institute of Aeronautics and Astronautics Infotech@Aerospace Conference* (pp. 1-13). Reston, VA: American Institute of Aeronautics and Astronautics.
- Gluck, K. A., Bello, P., & Busemeyer, J. (2008). Introduction to the Special Issue. *Cognitive Science: A Multidisciplinary Journal*, 32(8), 1245 - 1247.
- Gorman, P. (1990). *The Military Value of Training* (Vol. Paper P-2515). Alexandria, VA: Institute for Defense Analysis.
- Gorski, N. A., & Laird, J. E. (2007). *Investigating Transfer Learning in the Urban Combat Testbed*. Ann Arbor, MI: Center for Cognitive Architecture, University of Michigan.
- Guerin, S. M. (2004). Peeking into the black box: some art and science to visualizing agent-based models. In R. G. Ingalls, R. M.D., J. S. Smith & B. A. Peters (Eds.), *Proceedings of the 2004 Winter Simulation Conference* (pp. 749-754).
- Hayward, F. (2006). The use of simulation to support training in a resource restrictive environment. *Canadian Army Journal*, 9(2), 142-146.
- Jensen, R., Ludwig, J., Proctor, M. D., Patrick, J., & Wong, W. (2008). Adaptive Behavior Models for Asymmetric Adversaries *Proceedings of the Interservice/Industry Training, Simulation, and Education Conference* (pp. 1-11). Orlando, FL.
- Laird, J. E. (2008). *Extending the Soar cognitive architecture*. Paper presented at the Proceedings of the Artificial General Intelligence Conference, Memphis, TN.
- Lapointe, J. F., & Godin, G. (2005). On-Screen Laser Spot Detection for Large Display Interaction *Proceedings of the IEEE International Workshop on Haptic Audio Environments and their Applications (HAVE'2005)* (pp. 72-76). Ottawa, Ontario, Canada.
- Liu, D., Macchiarella, N. D., & Vincenzi, D. A. (2009). Simulation fidelity. In D. A. Vincenzi, J. A. Wise, A. Mouloua & P. A. Hancock (Eds.), *Human factors in simulation and training* (pp. 61-73). Boca Raton, FL: Taylor & Francis Group.
- Macal, C. M., & North, M. J. (2007). Agent-based modeling and simulation: desktop ABMS. In S. G. Henderson, B. Biller, M.-H. Hsieh, J. Shortle, J. D. Tew & R. R. Barton (Eds.), *Proceedings of the 2007 Winter Simulation Conference* (pp. 95-106).
- Moroney, W. F., & Lilienthal, M. G. (2009). Human factors in simulation and training: an

- overview. In D. A. Vincenzi, J. A. Wise, A. Mouloua & P. A. Hancock (Eds.), *Human factors in simulation and training* (pp. 3-38). Boca Raton, FL: Taylor & Francis Group.
- National\_Research\_Council (2002). *Modeling and Simulation in Defense Acquisition Modeling and Simulation in Manufacturing and Defense Acquisition: Pathways to Success* (pp. 29-46). Washington, DC: National Academy Press.
- Pasch, M., Bianchi-Berthouze, N., van Dijk, E. M. A. G., & Nijholt, A. (2009). *Immersion in Movement-Based Interaction*. Paper presented at the 3rd International Conference on Intelligent Technologies for Interactive Entertainment (INTETAIN 09).
- Proctor, M. D., & Gubler, J. C. (1998). Military simulation worlds and organizational learning. In D. J. Medeiros, E. F. Watson, J. S. Carson & M. S. Manivannan (Eds.), *Proceedings of the 1998 Winter Simulation Conference* (pp. 773-779).
- Proctor, M. D., & Woodman, M. D. (2007). Training "Shoot House" Tactics Using a Game. *Journal of Defense Modeling and Simulation: Applications, Methodology, Technology*, 4(1), 55-63.
- Ritter, F. E., & Young, R. M. (2001). Embodied models as simulated users: Introduction to this special issue on using cognitive models to improve interface design. *International Journal of Human-Computer Studies*, 55, 1-14.
- Roman, P. A., & Brown, D. (2007). Note to file - Constructive simulation versus serious games for the army: a Canadian case study. *Canadian Army Journal*, 10(3), 80-88.
- Sanford, K., & Hopper, T. (2009). Videogames and Complexity Theory: Learning through Game Play. *Loading...* 3(4).
- Schmorrow, D., Nicholson, D., Lackey, S. J., Allen, R. C., Norman, K., & Cohen, J. (2009). Virtual reality in the training environment. In D. A. Vincenzi, J. A. Wise, A. Mouloua & P. A. Hancock (Eds.), *Human factors in simulation and training* (pp. 201-230). Boca Raton, FL: Taylor & Francis Group.
- Templeman, J. N., Sibert, L. E., Page, R. C., & Denbrook, P. S. (2007). Pointman - A New Control for Simulating Tactical Infantry Movements *IEEE Virtual Reality Conference* (pp. 285-286).
- Ting, S.-P., & Zhou, S. (2009). Dealing with dynamic changes in time critical decision-making for MOUT simulations. *Computer Animation and Virtual Worlds*, 20, 427-436.
- Urbas, L., Nekrasova, L., & Leuchter, S. (2005). State chart visualization of the control flow within an ACT-R/PM user model *Proceedings of the 4th international workshop on Task models and diagrams* (pp. 43 - 48). New York, NY: ACM.
- Wray, R. E., & Chong, R. S. (2007). Comparing cognitive models and human behavior models: Computational tools for expressing human behavior. *Journal of Aerospace Computing, Information and Communication*, 4(5), 836-852.
- Wray, R. E., Laird, J. E., Nuxoll, A., Stokes, D., & Kerfoot, A. (2005). Synthetic adversaries for urban combat training. *AI Magazine*, 26(3), 82-92.
- Youngblood, G. M., Nolen, B., Ross, M., & Holder, L. (2006). Building testbeds for AI with the Q3 Mod Base *Proceedings of the Second International Conference on Artificial Intelligence and Interactive Digital Entertainment* (pp. 153-154).
- Zhou, Z., Tedjokusumo, J., Winkler, S., & Ni, B. (2007). User studies of a multiplayer first person shooting game with tangible and physical interaction *Proceedings of the 2nd International Conference on Virtual Reality at HCI International* (Vol. 4555, pp. 738-747). Beijing.

## Author Biography

**BRUNO EMOND** Research officer at the National Research Council with interests in the application of cognitive modeling technology in training simulators, as well as learning and performance in multimedia and broadband e-learning environments.

# Cognitive Flexibility through Learning from Constraint Violations

Dongkyu Choi  
Stellan Ohlsson

Department of Psychology  
University of Illinois at Chicago  
1007 W Harrison Street (M/C 285)  
Chicago, IL 60607  
312-355-0486, 312-996-6643  
dongkyuc@uic.edu, stellan@uic.edu

## Keywords:

cognitive architecture, constraints, constraint violations, error, learning from error, skill acquisition

**ABSTRACT:** *Cognitive flexibility is an important goal in the computational modeling of higher cognition. An agent operating in the world that changes over time should adapt to the changes and update its knowledge according to them. In this paper, we report on the implementation of a constraint-based mechanism for learning from negative outcomes in well-established cognitive architecture, ICARUS. We discuss the challenges encountered during the implementation, describe how we solved them and provide an example of the integrated system's operation.*

## 1. Background and Rationale

An important goal in the computational modeling of higher cognition is to invent techniques that enable computer programs to mimic the broad human functionality that we call *adaptability*, *flexibility*, or *intelligence*. Cognitive flexibility is a multi-dimensional construct. In this paper, we focus specifically on the ability of humans to act effectively and purposefully even when a familiar task environment is changing, thus rendering previously learned skills and strategies less effective or even obsolete.

When the environment changes, the execution of previously acquired skills is likely to generate actions that are inappropriate, incorrect or unhelpful vis-à-vis the agent's goal. A key component of flexible adaptation to the changing circumstances is therefore the ability to recover from and unlearn unsuccessful actions in the service of more effective future behavior (Ohlsson, 2010). This problem differs from the standard view of skill acquisition in two principled ways. First, instead of learning a new skill from scratch, the learning agent in this scenario needs to revise an existing skill or strategy. Second, whereas most work in computational modeling of skill acquisition has focused on how to make use of positive outcomes, the adaptation scenario requires mechanisms for learning from errors, mistakes and other types of negative feedback (Ohlsson, 2008).

In past work, we developed a mechanism for learning from negative outcomes that is called *constraint-based*

*specialization* (Ohlsson, 1993, 1996, 2007). This mechanism assumes that the agent has access to declarative knowledge in the form of constraints, where a constraint consists of an ordered pair with a relevance criterion and a satisfaction criterion,  $\langle R, S \rangle$ . Unlike propositions, constraints do not encode truths, but norms and prescriptions, e.g., traffic laws. A speed limit does not describe how fast drivers are going, but specifies the range within which their speeds ought to fall. Constraints support evaluation and judgment rather than deduction or explanation. In a constraint-based system, the architecture matches the relevance criteria of all constraints against the current state of its world in each cycle of operation. For constraints with matching relevance conditions, the satisfaction conditions are matched also. Satisfied constraints require no response, but constraint violations signal a failed expectation (due to a change in the world or to incomplete or erroneous knowledge); this is a learning opportunity. The purpose of the change triggered by a constraint violation is to revise the current skill or strategy in such a way as to avoid violating the same constraint in the future. The computational problem involved in unlearning an error is to specify exactly how to revise the relevant skill when an error is detected. The constraint-based specialization algorithm is a general solution to this problem (Ohlsson & Rees, 1991).

The constraint-based specialization mechanism was previously implemented in HS, a production system architecture (Ohlsson, 1996). The HS system was limited along several dimensions. First, HS did not explicitly represent or take into account the hierarchical

organization of skill knowledge. Second, HS did not explicitly distinguish between search in a mental problem space and search through the environment via overt actions. The implementation of the constraint-based learning mechanism operated with a simple credit/blame assignment rule: Assume that the last production rule to fire before the discovery of an error is the faulty rule. Finally, the HS model only learned from its errors. It is more plausible that human-level flexibility is achieved through the interactions among a set of learning mechanisms, different mechanisms making use of different types of sources of information (Ohlsson, 2008). At the very least, a powerful learning agent should be able to make use of positive as well as negative outcomes.

In this paper, we report preliminary progress in implementing the constraint-based specialization mechanism for learning from error in ICARUS, a cognitive architecture with hierarchical skill knowledge that interleaves thinking and action and that already has a well-developed capability of learning from positive outcomes (Langley & Choi, 2006a). We first describe the relevant features of the ICARUS architecture. We then describe the challenges encountered in implementing constraint-based specialization within ICARUS, with particular attention to the credit assignment problem. Finally, we report an illustrative example of the extended ICARUS, discuss related approaches and outline future work.

## 2. The ICARUS Architecture

Cognitive architectures aim for a general framework for cognition. An architecture implements as a set of cognitive hypotheses, covering representation, inference, execution, learning and other aspects of cognition. Soar (Laird et al., 1986) and ACT-R (Anderson, 1993) are the most well-known cognitive architectures. ICARUS exhibits some similarities to them, but some differences as well (Langley & Choi, 2006b). Both Soar and ACT-R are rule-based systems, but ICARUS represents skill knowledge differently. Also, ICARUS incorporates a highly developed semantic memory that forces all conceptual knowledge to be grounded in perceptual primitives. In this section, we review the fundamental aspects of the architecture.

### 2.1 Representation and memories

ICARUS distinguishes conceptual and procedural knowledge. *Concepts* are used to describe the environment around ICARUS, and to infer beliefs about the current state of the world. *Skills*, on the other hand, consist of *procedures* that are known to achieve certain goals. The architecture also distinguishes long-term (abstract) knowledge and short-term (instantiated) structures. Long-term concepts and skills are general

descriptions of situations and procedures, and the system needs to instantiate them to apply them to a particular situation. Instantiated concepts and skills are short-term structures, in that they are applicable only at a specific moment in time. ICARUS has four separate memories to support these distinctions; see Figure 1.

	Long-term Knowledge	Short-term Structures
Conceptual Contents	Long-term Conceptual Memory	Short-term Conceptual Memory
Procedural Contents	Long-term Skill Memory	Short-term Skill Memory

Figure 1: ICARUS' four-way classification of memory structures.

All concepts are introduced via definitions. Concept definitions are similar to horn clauses, and consist of a head and a body that includes perceptual matching conditions and reference to other concepts. Definitions that do not refer to other concepts define *primitive concepts*. Table 1 shows four ICARUS concept definitions. (Question marks indicate variables.) The first and second concepts have only perceptual matching conditions in their :percepts and :tests fields, so they are primitive. The third and fourth concepts, however, are non-primitive, because they have references to other concepts in their :relations fields. Percepts and tests access ICARUS' environment directly, so their implementation depends on whether ICARUS operates in a simulated or real environment.

Table 1: Sample ICARUS concepts in a Blocks World.

---

```

((holding ?block)
:percepts ((hand ?hand status ?block)
           (block ?block)))

((hand-empty)
:percepts ((hand ?hand status ?status))
:tests ((eq ?status 'empty)))

((clear ?block)
:percepts ((block ?block))
:relations ((not (on ?other ?block))))

((stackable ?block ?to)
:percepts ((block ?block)(block ?to))
:relations ((clear ?to)(holding ?block)))

```

---

ICARUS' skills resemble STRIPS operators. The head of each skill is the predicate it is known to achieve. Its body consists of perceptual matching conditions, non-primitive preconditions, and references to either subgoals or direct actions to the world. *Primitive skills* are actions that the agent can execute in the world, whereas non-primitive actions operate on ICARUS' subgoals. The hierarchical organization provides multiple layers of abstraction in the specification of complex procedures. In Table 2, the first skill, which achieves the goal to *hold a block*, has two perceptual preconditions, one of the being that there is a block within reach, one non-primitive precondition, *pickupable*, and two primitive actions, *\*grasp* and *\*vertical-move*. The second skill also has perceptual and non-perceptual preconditions but poses two subgoals, *clear* and *holding*, which, in turn, evoke other skills. Because procedures refer to other procedures, the entire set of procedures in long-term skill memory form a hierarchical organization.

Table 2: Sample skills for ICARUS in Blocks World

```
((holding ?block)
:percepts ((block ?block)
           (table ?from height ?height))
:start ((pickupable ?block ?from))
:actions ((*grasp ?block)
         (*vertical-move ?block (+ ?height 50))))

((stackable ?block ?to)
:percepts ((block ?block)
           (block ?to))
:start ((hand-empty))
:subgoals ((clear ?to)
           (holding ?block)))
```

During performance time, the architecture instantiates these long-term knowledge structures based on the current situation. The bottom-up application of concept definitions creates beliefs in the form of instantiated conceptual predicates and stores them in the short-term conceptual memory (a.k.a. the system's *belief state*). During execution, ICARUS finds executable skills to achieve its goals, and stores the instantiations of those skills in its short-term skill memory. For this reason, procedural short-term memory is sometimes referred to as ICARUS' *goal memory*. In the next section, we explain the system's processes in more detail.

### 2.2 Inference and execution

The ICARUS architecture operates in cycles. On each cycle, the system creates its current belief state by applying its concept definitions, and decides what to do next by finding a path through its skill hierarchy from

its top goal and down to some executable action. When it finds such a path, the system executes the actions proposed by the primitive skill instance that is the leaf node of the path. Figure 2 shows the overall process. The rectangular shapes represent memories in ICARUS with the exception of the one to the far right, which represents the environment. The oval shapes stand for processes that process the information in the memories, while the arrows show the flow of information.

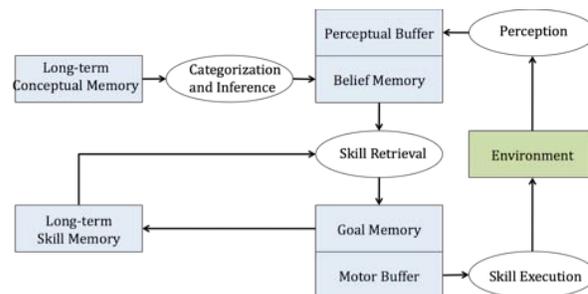


Figure 2: ICARUS' operation in cycles.

The inference process starts with the perceptual buffer, which contains information about the environment. The system attempts to match its concept definitions to the perceptual information. When there is a match, the system instantiates the head of the definitions to compute the current belief state. ICARUS deposits the instantiated concepts in its short-term conceptual memory (a.k.a. its *belief memory*), and it uses those beliefs during thinking and decision making.

The skill retrieval makes use of several different sources of information. First of all, the process uses the top-level goals specified in the goal memory to guide the retrieval process. It also accesses the contents of the long-term skill memory as well as the current belief state. The system finds relevant long-term skills for its goals, based on the current belief state. Once it finds an executable path through its skill hierarchy from goal to primitive actions, ICARUS performs those actions and thereby changes the environment. The system then starts another cycle, once again beginning by re-computing its current belief state.

### 3. Learning From Errors

When ICARUS cannot find a skill path from its current goal to an executable action, it invokes a means-ends problem solving capacity that has been described in prior publications (Langley & Choi, 2006a). If it can solve its problem, it captures the solution in the form of new skills that are added to the long-term procedural memory. In this way, ICARUS' stock of skills grows over time.

However, the means-ends based problem solving and learning capability does not enable ICARUS to recover when the environment changes and some of the previously learned skills become incorrect or obsolete.

We extended the ICARUS architecture by incorporating the constraint-based specialization mechanism originally developed for rule-based systems. This required adding a new representation to allow explicit descriptions of constraints and processes that apply constraints to the current belief state. As a consequence, the system can now detect its failures as constraint violations. We then implemented the constraint-based specialization algorithm that allows ICARUS to revise its skills based on its constraint violations.

### 3.1 Representation of constraints

The architecture stores each constraint as a pair of relevance and satisfaction conditions, following Ohlsson and Rees (1991). Both relevance and satisfaction conditions are conjunctions of predicates. ICARUS keeps a list of such pairs in a separate *constraint memory*, which users define in advance. Table 3 shows some examples of constraints that we imposed on the Blocks World domain. The first constraint, *color*, has a single relevance condition, (*on ?a ?b*), and a satisfaction condition, (*same-color ?a ?b*). It says that two blocks should have the same color if one is stacked on the other; that is, all the blocks in a tower should the same color. The second constraint, *max-tower*, has a high-level relevance condition and a single satisfaction condition. This constraint restricts the maximum height of towers to three blocks. In constraint language: *A tower should not be higher than three blocks*. Similarly, the third constraint decrees that there should be no other block on top of a particular block designated as a *top-block*, while the fourth says that a block that is stacked on top of another block should be smaller in size than the one it rests on. In constraint language: *Blocks should be stacked in the order of decreasing size*. The predicates used to define the constraints are, like all predicates in ICARUS, defined in terms of other predicates and/or perceptual primitives.

Table 3: Four constraints from the Blocks World.

---

(color :relevance ((on ?a ?b))
:satisfaction ((same-color ?a ?b)))
(max-tower :relevance ((three-tower ?a ?b ?c ?t))
:satisfaction ((clear ?a)))
(top-block :relevance ((top-block ?b))
:satisfaction ((clear ?b)))
(width :relevance ((on ?a ?b))
:satisfaction ((smaller-than ?a ?b)))

---

### 3.2 Detection of constraint violations

ICARUS creates its belief state anew on each cycle. It then goes on to retrieve, instantiate and execute one or more skill paths based on the computed beliefs. To learn from errors, the system performs an additional step between inference and execution: It checks if the belief state satisfies all the constraints. It first attempts to match the relevance conditions of its constraints against the current state, and, if a match is found, verifies that the satisfaction conditions also hold.

We distinguish two different types of constraint violations. In the first type, a constraint becomes relevant but not satisfied. For instance, when an agent stacks a red block, A, on top of a blue block, B, it achieves (*on A B*), so the corresponding instance of the *color* constraint in Table 3 matches and the constraint becomes relevant. But its satisfaction condition, (*same-color A B*), is not met in this instance, because one of the blocks is red and the other is blue. We refer to violations like this as *type A* violations.

Another type of violations, which we call *type B* violations, involves a constraint that is relevant and satisfied, but becomes unsatisfied as the result of an action or an environmental event. An example of this type occurs in our constrained Blocks World when an agent stacks a block C on top of a block TB that is designated as a top block. In this case, the *top-block* constraint stays relevant during the stacking action, since the predicate, (*top-block TB*) continues to hold. But the satisfaction condition, (*clear TB*) becomes false as a consequence of the action, so the constraint is violated.

When the architecture finds one or more violated constraints of either type, it invokes the skill revision process to constrain the skill that it just used. The details of the revision process differ between the two types of constraint violations, and we cover both in the following section.

### 3.3 Skill revisions

Once the system detects constraint violations, it attempts to make revisions to the skill just used. The revision process we use shares its basic steps with those used in previous research with production systems (Ohlsson, 1993, 1996, 2007; Ohlsson & Rees, 1991). The goal of the revision process is to constrain the application of the skills to situations in which it will not violate the constraints. This is done by adding preconditions. The key question is which conditions to add.

The architecture randomly chooses one of the detected violations and attempts to make two revisions by

adding preconditions computed based on the type of the violation. For a type A violation, in which a constraint becomes relevant but violated, one of the revisions forces the constraint to stay irrelevant, and the other ensures that it is both relevant and satisfied. On the other hand, a type B violation, in which a constraint stays relevant but becomes violated, invokes one revision that makes the constraint irrelevant, and another that ensures that the constraint stays satisfied. Table 4 shows how the system computes the new preconditions for the two types of violations.

Table 4: New preconditions created in response to constraint violations.  $C_r$  and  $C_s$  represent the relevance and satisfaction conditions.  $O_a$  and  $O_d$  are the *add* and *delete* lists of the executed primitive skill. The rationale for these computations has been developed in detail in prior publications (Ohlsson, 1993, 1996; Ohlsson & Rees, 1991).

Type \ Revision	1	2
A	not ( $C_r - O_a$ )	$(C_r - O_a) \cup (C_s - O_a)$
B	not $C_r$	$C_r \cup \text{not } (C_s \cap O_d)$

### 3.4 Challenges for re-implementation

The differences between the ICARUS architecture and production system architectures force some important changes in the revision process. These pertain to the hierarchical organization of skill knowledge, the definitions of actions and the use of disjunctive definitions.

(a) Hierarchical representation. ICARUS' hierarchical organization of skill knowledge poses one of the most significant changes, in relation to the assignment of credit/blame: which skill should be revised upon detecting a constraint violation? Production systems are flat structures, and it is frequently the case that the last executed rule caused a violation. But in ICARUS, execution involves a skill path, which may include more than one skill instance. Skill instances near the top of the path are more abstract, and those close to the bottom are more specific. Depending on the level of abstraction of the violated constraint, the most reasonable skill to revise might be at the top, at the bottom, or anywhere in between. No simple attribution rule will be sufficient.

In the Blocks World, for example, the system may cause a violation of the *color* constraint by stacking a red block on top of a blue block using the primitive skill, *stacked*. However, the context in which the system executed this particular skill varies based on the situation. Figure 2 shows an example, in which ICARUS did this to achieve its goal, (*color-sorted*). Here, the last action before the violation occurs is

generated by a skill path, (*color-sorted*) – (*one-color-sorted red*) – (*on A B*) – (*stacked A B*). If the system blindly chose the last skill on this path to revise, it would revise *stacked*. This will not prevent similar violations in subsequent runs, since the system decides which blocks to stack further up in the skill path, namely within the skill, *one-color-sorted*. Therefore, the right skill to revise is *one-color-sorted* rather than the primitive skill, *stacked*. This conclusion is obvious to a human observer in this particular case, but the question is how ICARUS can identify the right skill to revise in the general case.

An analysis of multiple examples indicates that the architecture should find the highest level in the skill path in which all the variables involved in the additional preconditions for the revision are bound. The additional preconditions are fully instantiated at this level, and, therefore, it is the highest level in which all the additional preconditions become meaningful, and it is the right level at which to make the corresponding revisions. For instance, the additional preconditions for the case depicted in Figure 2 are null for the first revision and (*same-color A B*) for the second one. Since a null precondition means no revision, the system makes only one revision in this case. The variable bindings involved in this revision are A and B, and the highest level where both of these are instantiated is at the skill, (*one-color-sorted red*), which binds its two variables, *?block1* and *?block2* to A and B, respectively. By making a revision at this level, the system checks if the two objects, A and B satisfy the additional condition (*same-color A B*) as soon as they are bound, and prevents the violation of the *color* constraint before it happens. The results of running ICARUS with this solution in place indicate that it is successful.

(b) Add/delete lists. Another problem occurs during the logical computation of the additional preconditions for skill revisions. Unlike production systems that have explicit and complete *add* and *delete* lists associated with actions, the ICARUS architecture has skills associated with goals. Goals typically do not include any side effects we do not care about, and they do not specify any predicates that should disappear after a successful execution. For this reason, the *add* and *delete* lists are not explicit in the architecture, and we must compute them from other sources.

The use of *add* lists during the revision process is limited to the calculation of logical differences, and we can use goals as if they represent complete *add* lists. This will make the revised skill more restrictive but not in the opposite way, making it safe. However, we should compute the *delete* list explicitly due to its use in the revision computations. We chose to calculate the list by comparing two successive belief states, although

this may include some predicates removed by sources external to the agent. Similarly, this makes the revisions more restrictive, but not more general, keeping the agent safe, because the delete list is negated during the computation of preconditions.

(c) Disjunctive definitions. ICARUS' support for multiple, disjunctive definitions of concepts adds another layer of complexity. During the operations that compute additional preconditions for skill revisions, the system should decompose any non-primitive concepts. Disjunctive concepts create multiple expansions, possibly resulting in more than one set of additional preconditions. We changed the architecture to accept all such expansions and create multiple revisions.

The consequences of this approach are significant. When the system experiences a constraint violation, the situation might involve a particular disjunction of a concept. But the architecture learns multiple revisions from this case, covering all possible disjunctions of the concept. This approach is based on the understanding that there must be a good reason why such definitions have the same head, thereby creating disjunctions, and that the system benefits from learning about all such cases when, in fact, the current situation involves only one of them. In future tasks, the system might confront a situation in which another one of the disjunctions applies, and, due to its prior learning, the system will already know how to avoid making an error in this situation even though it has never encountered it before. In everyday language, we would refer to this as understanding the situation.

#### 4. An Illustrative Example

In this section, we provide an example that illustrates the operation of the extended ICARUS system. We use the Blocks World that has served as an initial test bed during our development and implementation of the system. It supports many constraints with various complexities, and yet it stays relatively simple and easily understandable. We created four different constraints for this world as shown in Table 3.

The system has a skill set that is general in the sense that the skills do not have special preconditions that ensure the satisfaction of the constraints. For example, the system would know how to stack a block on top of another, but does not know if the skill would or would not cause any violations of *color*, *max-tower*, *top-block*, or *width* constraints. We gave the system opportunities to experience several different initial conditions and goals that naturally lead to violations of these constraints, and ICARUS learned revisions based on the violations. The experience eventually resulted in a successful run until completion of its top-level goals.

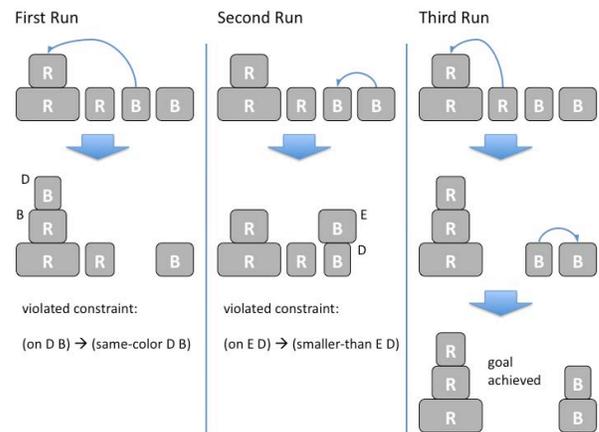
Fig. 3 shows some sample runs where the architecture achieves its goal, (*color-sorted*) in three runs. During the first run, the system stacks a blue block D on top of a red block B. The width of block D is smaller than that of block B, but the colors of them are different, violating the *color* constraint,

$$(on\ D\ B) \rightarrow (same-color\ D\ B)$$

From this error, the architecture learns a revised version of its non-primitive skill, *one-color-sorted*, with an additional precondition, *same-color*. During the second run, the system incurs yet another error and violates the *width* constraint,

$$(on\ E\ D) \rightarrow (smaller-than\ E\ D)$$

Then the system revises another skill with the same head, *one-color-sorted*, to include an additional precondition, *smaller-than*. After that, the system may or may not experience further failures that involve other constraints, but eventually it succeeds in achieving its top-level goal, as shown in the third run. We reset the initial tabletop state between runs, enabling the system to restart from the initial conditions without the need to undo what it has done so far. The puzzle-like characteristics of the Blocks World make this reasonable.



**Fig. 3. Two learning events that lead to a successful run in the Blocks World.**

In short, the solutions to the challenges posed by the architectural characteristics of ICARUS appear to be successful. The hierarchical organization of skill knowledge forces the question of at which level the revisions are to be applied. The principle that they apply at the level at which all relevant variables are bound has so far selected the right level in all simulation runs. Comparing successive belief states

appears to serve instead of explicit *add* and *delete* lists. Finally, ICARUS' support for multiple, disjunctive definitions of concepts poses the problem of which disjunctions to include in a revision. Our solution to a include all of them brings with it a modes form of understanding, because it allows the system to know what to do in situations it has never seen before.

## 5. Related Work

Two types of error correcting mechanisms have been developed in prior work, weakening and discrimination. The idea behind weakening is that when a knowledge structure (rule, skill element, schema, chunk, etc.) contributes to the production of an action, which, in turn, generates a negative outcome, then the strength associated with that knowledge structure is decreased according to some function. Weakening is not a powerful mechanism, because actions are not typically correct or incorrect, or appropriate or inappropriate, in themselves. Instead, actions are appropriate, correct or useful in some situations but not others. The goal of learning from error is thus to distinguish between the class of situations in which a particular type of action will cause errors and the class of situations in which it does not. Weakening does not accomplish this, because lower strength makes an action less likely to be selected in any type of situation.

In the 1980s, Langley (1987) proposed a computational model of discrimination. The key idea behind this contribution was to compare situations with negative and positive outcomes to identify discriminating features. The SAGE system stored every application of every production rule in memory. If an action generated both positive and negative outcomes across multiple situations, the situation features that were true for one type of outcome but not for the other were identified and used to constrain the applicability of the rule. The problems with this computational discrimination mechanism include (a) the lack of criterion for how many instances of either type are needed before a valid inference as to the discriminating features can be drawn; (b) the possible existence of a very large number of potential discriminating features, leading to complex applicability conditions or large numbers of new rules or both; and (c) the inability to identify potential discriminating features with a causal impact from those of accidental correlation.

The production system implementation of constraint-based specialization overcame most of these weaknesses. Unlike weakening, it identifies the specific class of situations in which an action is likely (or unlikely) to cause errors. Unlike Langley-style discrimination, constraint-based specialization does not carry out an uncertain, inductive inference, but

computes a rationally motivated revision to the current skill. These advantages were limited by a simplistic credit/blame attribution algorithm and a lack of learning mechanisms for capitalizing on successful outcome. The implementation of constraint-based specialization within the ICARUS architecture has removed those limitations.

## 6. Future Work

A key problem is to study the interactions among multiple learning mechanisms. People learn in a variety of ways (Ohlsson, 2008) and human-level flexibility is the outcome of the interactions among the multiple mechanisms. Our current understanding of how learning mechanisms interact to produce flexible behavior is limited. We intend to add additional learning mechanisms to ICARUS, including mechanisms for learning from examples and from analogies, and explore the conditions under which multiple mechanisms produce more flexible behavior than single mechanisms. A second key problem is how effectively to interleave thinking – i.e., search in a mental, symbolic problem space – and action – i.e., search in an external, physical environment. The two types of processes differ in a variety of ways, most importantly in that a return to a previous state can be achieved by fiat in the internal search space, but has to be accomplished through physical action in the external environment. We intend to experiment with multiple schemes for controlling the interleaving in multiple task domains.

## 7. Conclusion

An intelligent agent cannot be limited to learning from positive experience. When task environments change, the extrapolation of prior experience to cover future situations inevitably leads to errors, mistakes and unacceptable outcomes. To exhibit human-level flexibility, a computational agent needs learning mechanisms that specify how to change in the face of such negative outcomes. The constraint-based specialization mechanism has been shown to be successful when implemented in a production system architecture. Its implementation with the hierarchical skill representation in the ICARUS architecture posed multiple conceptual problems. The most important of these was the assignment of credit/blame in a hierarchical system. That is, how to locate the right level in a skill path at which to apply the new constraints? The answer is that the constraints apply at the level at which all the relevant variables were bound. Some test runs in the Blocks World support this idea. This solution has the advantages of being easily computable and general across domains. The possibility that it applies to other types of hierarchical systems might deserve attention.

## 8. References

- Anderson, J. R. (1993). *Rules of the mind*. Hillsdale, NJ: Lawrence Erlbaum.
- Anderson, J. R. (2007). *How can the human mind occur in the physical universe?* New York: Oxford University Press.
- Laird, J. E., Rosenbloom, P. S., & Newell, A. (1986). Chunking in Soar: The anatomy of a general learning mechanism, *Machine Learning*, 1, 11-46.
- Langley, P. (1983). Learning search strategies through discrimination. *International Journal of Man-Machine Studies*, 18, 513-541.
- Langley, P. (1985). Learning to search: From weak methods to domain-specific heuristics. *Cognitive Science*, 9, 217-260.
- Langley, P. (1987). A general theory of discrimination learning. In Klahr, D., Langley, P., & Neches, R., (Eds.), (1987). *Production system models of learning and development* (pp. 99-161). Cambridge, MA: MIT Press.
- Langley, P., & Choi, D. (2006a). Learning recursive control programs from problem solving. *Journal of Machine Learning Research*, 7, 493-518.
- Langley, P., & Choi, D. (2006b). A unified cognitive architecture for physical agents. In *Proceedings of the Twenty-First National Conference on Artificial Intelligence*, 1469-1474.
- Ohlsson, S. (1993) The interaction between knowledge and practice in the acquisition of cognitive skills. In A. Meyrowitz and S. Chipman (Eds.), *Foundations of knowledge acquisition: Cognitive models of complex learning* (pp. 147-208). Norwell, MA: Kluwer.
- Ohlsson, S. (1996). Learning from performance errors. *Psychological Review*, 103, 241-262.
- Ohlsson, S. (2007). The effects of order: A constraint-based explanation. In F. E. Ritter, J. Nerb, E. Lehtinen and T. M. O'Shea (Eds.), *In order to learn: How the sequence of topics influences learning* (pp. 151-165). New York, NY: Oxford University Press.
- Ohlsson, S. (2008). Computational models of skill acquisition. In R. Sun (Ed.), *The Cambridge handbook of computational psychology* (pp. 359-395). Cambridge, UK: Cambridge University Press.
- Ohlsson, S. (2010). *Deep learning: How the mind overrides experience*. New York: Cambridge University Press.
- Ohlsson, S., & Rees, E. (1991). Adaptive search through constraint violations. *Journal of Experimental & Theoretical Artificial Intelligence*, 3, 33-42.

## Acknowledgement

The work reported in this paper was supported by Award # N0001-4-09-1025 from the Office of Naval Research (ONR) to the second author. No endorsement should be inferred.

## Author Biographies

**DONGKYU CHOI** is a Visiting Research Specialist in the Department of Psychology at the University of Illinois at Chicago (UIC). He is in the process of completing his Ph.D. degree from the Department of Aeronautics and Astronautics at Stanford University, specializing in cognitive architectures. His work at UIC includes postdoctoral research with Stellan Ohlsson.

**STELLAN OHLSSON** is Professor of Psychology and Adjunct Professor of Computer Science at the University of Illinois at Chicago (UIC). He has held academic positions at the University of Uppsala, Sweden, Carnegie-Mellon University, and the Learning Research and Development Center at the University of Pittsburgh. He has published widely on issues pertaining to skill acquisition, creative insight, the design of intelligent tutoring systems and other cognitive research topics.

# Modeling the Theory of Planned Behavior from Survey Data for Action Choice in Social Simulations

Jonathan K. Alt

Stephen Lieberman

Modeling, Virtual Environments and Simulation (MOVES) Institute

700 Dyer Road

Naval Postgraduate School

Monterey, California 93943

831-656-7576

jkalt@nps.edu, stlieber@nps.edu

Keywords: social simulation, behavior, action choice

**ABSTRACT:** *Current dialogues across a variety of disciplines from the social, behavioral and computer sciences have made clear the need for authentic, repeatable and actionable social simulations. Understanding how the individuals that comprise various populations (and segments of society) might respond to a given set of conditions provides the potential to better inform analysts and decision makers in a wide variety of settings. Here we examine the implications of applying a well-documented behavioral prediction theory, Icek Ajzen's Theory of Planned Behavior (TPB), within a social simulation in the context of public policy decision making. We provide brief overviews of both TPB and the construction of artificial societies, a full description of the TPB implementation within an artificial society, and develop an argument for the benefits of informing action choice models such as TPB from representative survey data.*

## 1. Introduction

Icek Ajzen's Theory of Planned Behavior (TPB) is a predictive paradigm for human behavior that connects attitudes with actions (I. Ajzen, 1991). Specifically, TPB accesses an individual's 1) belief towards a particular behavior, 2) belief about the social norms associated with a particular behavior, and 3) belief regarding the ability to control the outcome of a particular behavior. These are referred to as "behavioral beliefs", "normative beliefs", and "control beliefs", respectively, and together yield the individual's level of intention to carry out a particular action. This "behavioral intention" is assumed to be a direct precursor to actual action, and is empirically well-supported in literature across many behavioral and social domains, including social and cognitive psychology, advertising, marketing, healthcare, and communications (Chang, 1998; Hagger et al., 2007; Mathieson, 1991; Walker, Courneya, & Deng, 2006). TPB was also used as the theoretical basis for examination in over 800 studies in two prominent medically-related scholarly databases between 1985 and 2004 (Francis & Eccles, 2004).

In order to obtain the required information about individual beliefs, TPB surveys are generally used that address specific questions within a particular field of study (Icek Ajzen, 2006). For instance, a healthcare TPB questionnaire would be used to assess individual beliefs related to the use of treadmill exercise for the purposes of weight loss. Once these beliefs are assessed, the model can generate predictions about whether individuals will use treadmills to lose weight. Previous studies have discussed the use of surveys to inform the cognitive state

models (e.g., internal beliefs and interests), and a social structures of multi-agent systems. Here we explore the use of survey data to inform the theory of planned behavior (TPB) as a means of ascertaining and describing an actor's intention to carry out specific behaviors within an artificial society.

## 2. Social Simulations

Social simulations represent large human groups (such as societies) as complex adaptive systems at varying levels of granularity. One of the key goals in the field of social simulation is the representation and analysis of changes in the beliefs, values, and interests (BVIs) of individuals in a population across a range of possible perturbing events (Alt, Jackson, Hudak, & Steven Lieberman, 2010). Data to instantiate these simulation models can be derived from a number of sources, including subject matter expert (SME) input, such as the development of narrative ethnographies, and quantitative survey and polling data, such as the U.S. General Social Survey<sup>1</sup>, and World Values Survey<sup>2</sup>.

Simulated societies provide tools for analysts and researchers from multiple disciplines to conduct experimentation and gain insight into the complex domain described by a society. The endeavor to understand and analyze complex adaptive systems, including societies, has been described as a "wicked problem" (Roberts, 2000). One defining characteristic of these problems is

---

<sup>1</sup> <http://www.norc.org/GSS+Website/>

<sup>2</sup> <http://www.worldvaluessurvey.org/>

that traction is typically only gained through iteration. One cannot experiment with public policies, for instance, without altering the public—namely the target group of the policies. If a trial policy does not have the intended consequences, new policy must be developed not based on the original conditions, but for the newly changed target group. This makes the wicked problems associated with societies ideal candidates for the use of modeling and simulation, where experimentation and “what if” analyses can be performed without changing the target group.

Social simulations must consist of actors, representations of individuals from population subgroups within the real population under study, as well as a representation of the social environment within which these actors interact (National Research Council, 2008). When developing social simulation scenarios, data must be obtained to inform 1) the internal states of each entity on issues relevant within the society, 2) the interaction rules of the social environment (i.e., how entities interact), and 3) the formation of the intention to carry out certain actions (Alt et al., 2010). We demonstrate through case study how TPB can be implemented in one artificial society, the Cultural Geography (CG) model.

### 2.1 Cultural Geography Model

The social simulation used in this paper is the CG model, a government owned, open-source, agent based multi-agent system (MAS), composed of actors, objects and laws, implemented in Java (Ferber, Gutknecht, & Michel, 2004). The CG model is intended to serve as a reusable framework to facilitate analysis of social theories and their interaction in the context of a particular geographic area and time period under study (Alt, Jackson, & Stephen Lieberman, 2009). The model is based on theoretical and empirical work from cognitive psychology, social psychology, and structural sociology. The model emulates a conflict ecosystem and the process of scenario development mirrors Mansoor's counter-insurgency intelligence preparation of the battlefield (IPB) process (F. Mansoor, Zaidi, Wagenhals, & Levis, 2009; P. Mansoor, 2007). The two main components of the model are the cognitive module, which manages the internal states of each agent, and the social structure module, which manages the interaction of agents in the artificial society.

The cognitive module instantiates and controls an entity's stance on a given issue, such as "Are you satisfied with security in your neighborhood?" within the model. Walter Fisher's narrative paradigm theory (Fisher, 1989; Jackson, 2009) describes each human as a collection of stories, gained from first and second person observations, that shape the individual's perception of the world and events. The beliefs, values, and interests (BVI) contained in each population subgroup's unique narrative are implemented within the model in the form of

a Bayesian belief network (BBN). A Bayesian approach to the representation of human decision making is well supported by literature from cognitive psychology (Beppu & T. L Griffiths, n.d.; T. L Griffiths & J. B Tenenbaum, 2001; J Tenenbaum, T Griffiths, & Kemp, 2006), allows for transparency within the model, and ease of subject matter expert input.

Social structure module controls the interaction between entities within the model, which primarily consists of the exchange of information. The likelihood of interaction for every pair of agents in the artificial society corresponds to their similarity across social factors, including socio-economic, socio-demographic, and socio-cultural attributes, as well as BVIs (Blau, 1994; Blau & Schwartz, 1997; M. McPherson, Smith-Lovin, & Cook, 2001; Miller McPherson, Popielarz, & Drobnic, n.d.; Miller McPherson & Ranger-Moore, 1991).

### 2.2 Modeling TPB in Social Simulations

Action choice models provide methods to control the intention to take actions within an artificial society. TPB is one such action choice model that holds that individuals within a group form an intention execute a behavior based on 1) their individual attitude toward the behavior, 2) their perception of group or subjective norms associated with that behavior, and 3) their perceived level of behavioral control (i.e., chances of success) in regard to that behavior. The TPB is widely used in empirical studies for the forecasting of human behavior (I. Ajzen, 1991; Mathieson, 1991; Sparks & Shepherd, 1992; Walker et al., 2006). Accordingly, the empirical data used to drive the majority of these studies is derived from survey or questionnaire data, making TPB attractive for use in social simulations using multi-agent systems where agents are representative of the actual individuals or groups that comprise the society under consideration.

Our goal here is not to gather information on behavioral intentions through a new survey, but rather to model the workings of TPB inside of an artificial society of representative agents. The path to instantiate social simulations with traceable data is tractable given that the area to be modeled can be accessed by survey or polling teams. Each of the three components of the TPB can be calculated via item responses:

The attitude,  $A$ , toward a given behavior,  $B$ , can be expressed as an expected value model where the strength of belief,  $b$ , is expressed as a likelihood and the outcome evaluation,  $e$ , is an evaluation of the value of the potential outcome (Icek Ajzen, 2006; Mathieson, 1991). Thus, if the behavior outcome is beneficial, and this outcome is highly likely, the attitude towards a behavior will be correspondingly favorable. The attitude  $A$  is the sum product of these two terms across the salient observations,  $i$ , out of the possible,  $n$ .

$$A_B = \sum_i^n b_i e_i \quad (1)$$

A similar approach is applied to determine the subjective norms,  $SN$ , associated with the behavior,  $B$ . The components of  $SN$  are similar to those of  $A$ : the normative belief strength,  $nb$ , takes the place of strength of belief,  $b$ , and motivation to comply with the  $nb$ ,  $m$ , takes the place of outcome evaluation,  $e$  (Icek Ajzen, 2006; Mathieson, 1991). In this case however, the terms are summed across the relevant others,  $n$ , opinions are valued by the individual.

$$SN_B = \sum_i^n nb_i m_i \quad (2)$$

Perceived behavioral control,  $PBC$ , also follows a similar pattern. Control beliefs,  $cb$ , serve as the likelihood estimate, while perceived facilitation,  $pf$ , provides the value estimate (Icek Ajzen, 2006; Mathieson, 1991). The summation for  $PBC$  is over each,  $i$ , of the perceived skills, resources or opportunities,  $n$ , associated with the behavior.

$$PBC_B = \sum_i^n cb_i pf_i \quad (3)$$

Finally, the sum of these three components yields a behavioral intention score for each of the behaviors,  $B$ , under study, completing the TPB model.

$$BI_B = A_B + SN_B + PBC_B \quad (4)$$

The TPB survey methodology uses questions (response items) about behavioral beliefs to yield  $A$ , normative beliefs to yield  $SN$ , and control beliefs (or self-efficacy) to yield  $PBC$  (Icek Ajzen, 2006). Through the rest of this article, we discuss techniques to leverage existing social survey data to measure these beliefs, embed intelligent agents with these beliefs, and implement TPB within a full scale social simulation.

### 3. Techniques for Leveraging Survey Data

The identification of relevant existing survey data from populations of interest to construct social simulation models is an ongoing effort across disciplines. In the experience of the authors, there are currently no survey instruments that are executed on a recurring basis in a manner to explicitly inform social simulation development. As such, social simulations seeking to leverage these existing data sources must be flexible in their application and techniques. Previous work has

explored techniques to leverage existing survey data to inform cognitive models regarding issue stance and to construct authentic social structures within simulation societies (Alt et al., 2009). Here we extend this work by exploring techniques to inform representations of the TPB within the model using a relevant social survey.

#### 3.1 General Strengths and Limitations of Survey Use

Since direct observation of a large population's behavior choices over the time scale of interest is not tenable, our model must be informed by either sample observations, or self-report. Even for small populations, where sample observations of very specific behavior choices in precise contexts may be possible (e.g., employees using the treadmill at the company gym), self-report methods are more easily conducted. In general, TPB methodologies use self-reports in the form of TPB questionnaires to inform the behavior choices of populations large and small.

While self-report methods, TPB questionnaires or social surveys, are the plainly preferred technique, it is necessary to clearly state the caveats associated with their use. Self-report prone to direct errors such as memory inaccuracies and misunderstandings of question phrasing that are particularly germane to TPB models. Likewise, they are also susceptible to direct deception on the part of the respondent. Although deception and intentional disinformation can be minimized with appropriate research methodologies that ensure anonymity and confidentiality, the variance in all types of error rates between subjects is difficult to establish. Moreover, ascertaining causal relationships is often difficult with self-report methodologies (Icek Ajzen, 2006) National Research Council, 2008).

#### 3.2 World Value Survey

The World Values Survey (WVS) is an enduring social and behavioral research project that seeks to assess and describe longitudinal and cross-cultural values across 62 different countries with detailed questionnaires of approximately 250 items<sup>3</sup>. Survey items predominantly reflect the current sociocultural, moral, religious, and political views of the respondent. Questionnaires are administered in face-to-face interviews in each country by local (or indigenous) members of the society where local academics can "opt-in" to the decentralized WVS network. The WVS has been repeated in waves (longitudinal slices) from 1981 through 2006, and the

<sup>3</sup> WVS data for all countries from all survey waves, along with a description of WVS methodology and analysis, is freely available at [www.worldvaluessurvey.org](http://www.worldvaluessurvey.org).

number of countries included in the sample has grown from 22 to the current 62 through the iterations.

There are a multitude of freely available longitudinal social surveys that may fit our goals of instantiating an action choice model for an artificial society. The European Social Survey<sup>4</sup> and United States General Social Survey<sup>5</sup> provide notable alternatives. We have chosen to use the WVS because of its unique characteristics of global inclusiveness, indigenous administration, and focus on items that bias extrapolation of actions from personal BVIs.

In the examples that follow, we use the World Values Survey's most recent 2006 wave to illuminate the application of TPB to an artificial society on representative agents from the country of Indonesia. Where appropriate, we have noted the WVS item code (e.g., "V92") to aide follow on work and the docking of models and simulations using a common dataset.

### 3.3 Theory of Planned Behavior Instruments

TPB questionnaire development is straightforward and well-documented (see for instance, (Icek Ajzen, 2006). Given its empirical history, TPB self-reports have addressed issues of sampling methodologies and questionnaire biases across a wide variety of fields. Each behavior is defined by its "target", "action", "context", and "time" elements<sup>6</sup>, where all four items build a complete description of the behavior, and the corresponding intention, *BI*, for that behavior. Given space and scope constraints, the descriptions that follow are necessarily incomplete. The reader is directed to Ajzen, 2006 for a more comprehensive treatment.

Describing the *target* of an *action* is relatively straightforward, for instance in the question, "I will donate 10 dollars (action) to Wikipedia (target)". These types of questions are commonplace in self-reports, and while this may suffice for a basic description of a behavior, it does not supply enough information to generate the predictive *Behavioral Intention* estimator. We also need the context and time elements to fully describe the behavior, such as "I will donate 10 dollars to Wikipedia from my home computer (context) within the next week (time)". Each element can be tightly specified, such as "10 dollars", or highly generalized. The target and context elements can overlap somewhat and, clearly, some context items, such as "from my home computer", may not be necessary to gauge a particular *BI*. In this case, the computer used for the action of donation may be irrelevant, whereas the specific action "donate 10 dollars" and time "within the next week", may be highly relevant.

Once the behavior is described in sufficient scope and language for *BI* estimation, questions using this behavior description must be developed to assess the behavioral, normative, and control beliefs associated with actually carrying out the behavior. Thus, the *latent* variables of theoretical analysis must be associated with *salient*, observable behavioral outcomes. Care must be taken during item development since there is a limited subset of behavioral, normative, and control beliefs that are in fact accessible relative to any well-formed TPB behavior description.

Given these requirements, most TPB questionnaires are developed iteratively, with pilot work dedicated to elucidating what beliefs are genuinely accessible (Ajzen, 2006). One prominent goal is to clarify the *model salient beliefs* (MSBs) associated with each belief category. These MSBs are the most frequently stated beliefs for the population, and may be readily available from existing survey sources for specific types of behaviors. In applying TPB to social simulations using existing survey data, we must postulate that the survey designers have identified the equivalent of MSBs for their populations prior to commencing major investigations. As described in the following section, the researcher must determine MSBs for the salient behavioral, normative, and control beliefs that are relevant to the behavior in question.

## 4. Case Study: Applying TPB to WVS 2005

The application of TPB to an artificial society can be demonstrated using TPB calculations in conjunction with existing data from the 2005 WVS for Indonesia. The applied TPB can then be implemented as a simulation artifact at the instantiation of the simulation. The first step in this process is the selection of a behavior of interest for representation in the simulated society that is feasible to populate from the existing data.

Given that our survey data approach topics in a more generalized fashion, our application of TPB will focus on a more general class of behaviors, rather than an extremely precise behavior. As such, we forgo aspects of exact temporal clarity in favor of wide-ranging applications. It is important to note that, as demonstrated below, many of the survey items in the WVS can be used to temporally-specify TPB results from the broader categorical behavior classes.

There are a number of social and behavioral themes that are well represented in the WVS, and numerous candidates of behavioral classes that are germane to our investigation. We have chosen *participation in organized religious activities*, broadly defined, as the class of behaviors for this case study as we feel it will be of interest to the greatest variety of readers from different fields and subfields within the behavior representation communities. In the examples that follow,

<sup>4</sup> <http://www.europeansocialsurvey.org/>

<sup>5</sup> [www.norc.org/GSS+Website/](http://www.norc.org/GSS+Website/)

<sup>6</sup> These elements are sometimes abbreviated as "TACT".

we have chosen survey items from the WVS that best correspond to Ajzen’s salient observation types (see Ajzen, 2006) to populate the TPB models (equations 1-4).

#### 4.1 Attitude

Recall from equation 1 that an individual's attitude,  $A$ , toward a behavior,  $B$ , is a function of the strength of belief,  $b$ , and the outcome evaluation,  $e$ . In this case we are trying to determine an individual's attitude toward participation in organized religious activities. The TPB process calls for the aggregation of multiple self-report items to specify the variable of interest.

Several candidate items provide access to salient observations germane to our question. One clear item begins: "FOR EACH OF THE FOLLOWING, INDICATE HOW IMPORTANT IT IS IN YOUR LIFE:", where respondents rank "RELIGION" (V9) from "Very important" to "Not at all important" on a four-point scale. Another candidate to inform  $b$  exists in the item: "APART FROM WEDDINGS AND FUNERALS, ABOUT HOW OFTEN DO YOU ATTEND RELIGIOUS SERVICES THESE DAYS?" (V186). Another candidate for correlation of  $b$  is the item: "HOW IMPORTANT IS GOD IN YOUR LIFE?" (V192). V186 is reported on a 7 point Likert anchored with "More than once a week" and "Never, practically never", while V192 utilized a 10 point scale anchored with "Not at all important" and "Very important". A respondent's answer of "4" to V9, "6" to V186, and "10" to V192 thus become  $b_1$ ,  $b_2$ , and  $b_3$ , respectively.

Outcome evaluation  $e$  can be informed by the series of items V188-V191. Each begins with the phrase, "GENERALLY SPEAKING, DO YOU THINK THAT THE [CHURCHES] IN YOUR COUNTRY ARE GIVING ADEQUATE ANSWERS TO:", and concludes with "THE MORAL PROBLEMS AND NEEDS OF THE INDIVIDUAL" (V188), "THE PROBLEMS OF FAMILY LIFE" (V198), "PEOPLE'S SPIRITUAL NEEDS" (V190), and "THE SOCIAL PROBLEMS FACING OUR SOCIETY" (V191). These are each answered simply as "yes" or "no", so we take the sum of the responses from each respondent for the total  $e$ . That is, answering "yes" to all four yields score of 4 for  $e$ . A respondent answering in the affirmative to all  $e$  equates to  $A_B = 80$  as demonstrated below:

$$A_B = \sum_i^n b_i e_i = (4 + 6 + 10)(1 + 1 + 1 + 1) = 80 \quad (5)$$

#### 4.2 Subjective Norm

The subjective norm,  $SN$ , (equation 2) regarding participation in organized religious activities can be determined in a similar manner. Recall  $SN$  is dependent on normative behavior,  $nb$ , and the motivation to comply with the  $nb$ ,  $m$ . Several items on the WVS are germane to

the social norms experienced by the respondent regarding religious activities.

One series of WVS items begins with: "NOW I AM GOING TO READ OFF A LIST OF VOLUNTARY ORGANIZATIONS. FOR EACH ONE, COULD YOU TELL ME WHETHER YOU ARE AN ACTIVE MEMBER, AN INACTIVE MEMBER OR NOT A MEMBER OF THAT TYPE OF ORGANIZATION:" where respondents reply to "CHURCH OR RELIGIOUS ORGANIZATION" (V24) with one of the three response categories. Another WVS item simply asks: "DO YOU BELONG TO A RELIGION OR RELIGIOUS DENOMINATION" (V185). Where respondents reply with either a "no", or a "yes" selection from a list of religious denominations. In this case, we are not concerned about *what* religion a person belongs to, only *if* they identify with a religion. Thus, this item becomes a binary (yes/no) calculation. A respondent's answers of 2 (active member) to V24, and 1 (yes) to V185 thus become  $nb_1$  and  $nb_2$ , respectively.

Illuminating a respondent's motivations to comply with a specific behavior  $m$  is arguably the most elusive variable to draw from surveys such as the WVS. One viable proxy measure for motivation from social norms can be identified in the WVS items that address the respondent's preferences or aversions of different kinds of neighbors, and their relative level of trust for people occupying different social groups. These items make salient important characteristics of in-group versus out-group behavior. In other words, they should reflect to what extent the respondent associates with his or her religious in-group at the expense of maintaining influencing relationships from outside of that group<sup>7</sup>.

The series of items about neighbors begins with "COULD YOU PLEASE MENTION ANY THAT YOU WOULD NOT LIKE TO HAVE AS NEIGHBORS:" where respondents have "mentioned", or "not mentioned" "PEOPLE OF A DIFFERENT RELIGION" (V39). The second salient group measure begins with, "COULD YOU TELL ME FOR EACH WHETHER YOU TRUST PEOPLE FROM THIS GROUP:" where respondents rank "PEOPLE FROM ANOTHER RELIGION" (V129) on a four point scale from "A great deal" to "None at all".

Another series of WVS items proves quite valuable when elucidating  $m$ . This series of items begin with "PLEASE INDICATE FOR EACH DESCRIPTION WHETHER THAT PERSON IS VERY MUCH LIKE YOU" where respondents chose from a six point scale from "Very much like me" to "Not at all like me" to the prompt "TRADITION IS VERY IMPORTANT TO THIS PERSON; TO FOLLOW THE CUSTOMS HANDED DOWN BY ONE'S RELIGION OR FAMILY" (V89). A

<sup>7</sup> For a review of these theories, as well as supporting research, see Blau & Schwartz (1997).

respondent's answers of 1 (mentioned) to V39, 4 (do not trust at all) to V129, and 1 (very much like me) to V89, thus become  $m_1$ ,  $m_2$ , and  $m_3$ , respectively. These values together yield:

$$SN_B = \sum_i^n nb_i m_i = (2+1)(1+4+1) = 18 \quad (6)$$

#### 4.3 Perceived Behavioral Control

The perceived behavioral control,  $PBC$ , (equation 3) in this case refers to the individual's perception of the ability to participate in organized religious activities successfully if they chose to do so and is based on the control belief,  $cb$ , and the perceived facilitation,  $pf$ . The  $cb$  in this case refers to the individual's opportunity to participate in religious services and can be informed by items V185 and V24 as described above (in section 4.2). That is, we ask 1) whether the person belongs to a religion denomination, and 2) whether the person is an active member of that organization. Similarly to above, a respondent's answers of 2 (active member) to V24, and 1 (yes) to V185 thus become  $cb_1$  and  $cb_2$ , respectively.

Correspondingly,  $pf$  can be informed by items V188-V191, which asks respondents: "GENERALLY SPEAKING, DO YOU THINK THAT THE [CHURCHES] IN YOUR COUNTRY ARE GIVING ADEQUATE ANSWERS TO:" "THE MORAL PROBLEMS AND NEEDS OF THE INDIVIDUAL" (V188), "THE PROBLEMS OF FAMILY LIFE" (V189), "PEOPLE'S SPIRITUAL NEEDS" (V190), "THE SOCIAL PROBLEMS FACING OUR SOCIETY" (V191) where these are all binary (yes/no) responses that are aggregated. Confidence also plays a role in the  $pf$  values, and a salient observation can be obtained through the item, "FOR EACH ONE, COULD YOU TELL ME HOW MUCH CONFIDENCE YOU HAVE IN THEM:" where respondents chose from a four point scale from "A great deal" to "None at all" to the prompt "THE CHURCHES" (V131). A respondent's answers of 1 (yes) for V188-V191 and 1 (a great deal) for V131 thus become  $pf_1$  through  $pf_5$ , generating our PBC measure:

$$PBC_B = \sum_i^n cb_i pf_i = (2+1)(1+1+1+1+1) = 15 \quad (7)$$

#### 4.4 Behavioral Intention

Our goal in obtaining the above calculations is the Behavioral intention,  $BI$ , which is the linear sum of  $A$ ,  $SN$ , and  $PBC$ . Following from our example above the BI regarding participation in organized religious activities for

an Indonesia respondent using the method described above is:

$$BI_B = A_B + SN_B + PBC_B = 80 + 18 + 15 = 113 \quad (8)$$

In implementation this raw  $BI$  value can be normalized across the entities within the simulation providing each entity a relative likelihood, as compared to the overall population, of forming the intention to participate in a given behavior.

## 5.0 Discussion and Conclusion

It is important for researchers applying this type of methodology to be keenly aware of the scales used in the self-report items being used. Since the  $BI$  is an aggregate measure of the three belief components ( $A$ ,  $SN$ , and  $PBC$ ), the researcher must make sure that all scales are either ascending or descending values. The calculations used here reflect the most extreme respondent. The  $BI$  value of 113 is the highest possible  $BI$  given the WVS items selected for inclusion.

The measures of subjective norms that are intrinsic to the value of TPB are generally not the domain of social surveys. Here we selected individual WVS items based on our informed interpretation of TPB. Another way to approach questions about subjective norms is to aggregate responses across the population of respondents in the form of expected values. Item V186, used previously to determine the individual's  $b$ , can be used to determine the  $nb$  across relevant others,  $n$ . In this case, the WVS does not provide an explicit match for the TPB and it is necessary to use the surrogate  $nb$  described above with the assumption that the group under study is relevant to the individual by his membership in the group alone.

The mean score across the population subgroup under study can be used. The individual's  $m$  can be obtained from the item: "...PLEASE INDICATE FOR EACH DESCRIPTION WHETHER THAT PERSON IS VERY MUCH LIKE YOU, LIKE YOU, SOMEWHAT LIKE YOU, NOT LIKE YOU, OR NOT AT ALL LIKE YOU?...TRADITION IS IMPORTANT TO THIS PERSON; TO FOLLOW THE CUSTOMS HANDED DOWN BY ONE'S RELIGION OR FAMILY" (WVS:V89). This response is on a six point scale anchored with "Very much like me" and "Not at all like me."

Another potential contributor to  $nb$  is provided in the item "HERE IS A LIST OF QUALITIES THAT CHILDREN CAN BE ENCOURAGED TO LEARN AT HOME. WHICH, IF ANY, DO YOU CONSIDER TO BE ESPECIALLY IMPORTANT:" where respondents have either "Mentioned" or "Not mentioned" "RELIGIOUS FAITH" (V19). It is ultimately up to the researcher, informed of the theory being applied, to select appropriate items for inclusion. Furthermore, automated feature

selection mechanisms, not explored here, can be used to assist the researcher in the clarification and selection of items if there is a well-phrased survey item that can be used as a data mining target. A separate publication by the authors reviews this in greater detail (Alt & Stephen Lieberman, 2010).

The use of well documented theories from the social sciences, such as Icek Ajzen's Theory of Planned Behavior, leverages the existing body of knowledge and data to enhance the representation of human cognition and behavior in artificial societies. Existing data collection instruments, protocols and methodologies from the social and behavioral sciences provide solid theoretical bases to human-centered modeling and simulation across a variety of domains, from traditional research and development, to decision support for policy makers, and training for field analysts. Furthermore, as we have demonstrated, well documented survey and polling procedures, such as the TPB questionnaire process, can provide a reasonable foundation for the development of data to populate action choice models in social simulations.

Here we examined the use of these methods when applied to existing data from the WVS and illustrated one potential means of leveraging this data source while maintaining traceability to the TPB. Future work will propose a survey instrument designed to specifically elicit the information required to instantiate action choice models in an artificial society and provide further discussion of the dynamic implementation of the TPB within simulation.

## 6.0 Reference List

- Ajzen, I. (1991). The theory of planned behavior. *Organizational behavior and human decision processes*, 50(2), 179–211.
- Ajzen, I. (2006). Constructing a TPB Questionnaire: Conceptual and Methodological Considerations.
- Alt, J., Jackson, L., Hudak, D., & Lieberman, S. (2010). The Cultural Geography Model: Evaluating the Impact of Tactical Operational Outcomes on a Civilian Population in an Irregular Warfare Environment. *Journal of Defense Modeling and Simulation*, (Special Issue on Counter-Insurgency).
- Alt, J., Jackson, L., & Lieberman, S. (2009). The Cultural Geography Model: An Agent Based Modeling Framework for Analysis of the Impact of Culture in Irregular Warfare. In *International Conference on Computational Cultural Dynamics*. Presented at the Third International Conference on Computational Cultural Dynamics, Maryland.
- Alt, J., & Lieberman, S. (2010). Developing Cognitive Models for Social Simulation from Survey Data. In *Social Computing, Behavioral Modeling and Prediction*. Presented at the 2010 International Conference on Social Computing, Behavioral Modeling, and Prediction, Maryland: Springer.
- Beppu, A., & Griffiths, T. L. (n.d.). Iterated Learning and the Cultural Ratchet.
- Blau, P. (1994). *Structural Context of Opportunities*. University of Chicago Press.
- Blau, P., & Schwartz, J. (1997). *Crosscutting Social Circles: Testing A Macrostructural Theory of Intergroup Relations*. Transaction Publishers.
- Chang, M. K. (1998). Predicting unethical behavior: a comparison of the theory of reasoned action and the theory of planned behavior. *Journal of Business Ethics*, 17(16), 1825–1834.
- Ferber, J., Gutknecht, O., & Michel, F. (2004). From agents to organizations: an organizational view of multi-agent systems. *Lecture Notes in Computer Science*, 214–230.
- Fisher, W. (1989). Clarifying the Narrative Paradigm. *Communication Monographs*, 56(1), 55-58.
- Francis, J., & Eccles, M. (2004). CONSTRUCTING QUESTIONNAIRES BASED ON THE THEORY OF PLANNED BEHAVIOUR. Retrieved from <http://www.people.umass.edu/aizen/pdf/Francis%20et%20al.TPB%20research%20manual.pdf>
- Griffiths, T. L., & Tenenbaum, J. B. (2001). Randomness and coincidences: Reconciling intuition and probability theory. In *Proceedings of the Twenty-third Annual Conference of the Cognitive Science Society: August 1-4 2001, Human Communication Research Centre, University of Edinburgh, Edinburgh, Scotland* (p. 370).
- Hagger, M. S., Chatzisarantis, N. L., Barkoukis, V., Wang, J. C., Hein, V., Pihu, M., et al. (2007). Cross-cultural generalizability of the theory of planned behavior among young people in a physical activity context. *Journal of Sport and Exercise Psychology*, 29(1), 2.
- Jackson, L. (2009). Narrative Paradigm in Cultural Geography Modeling. In *Human Socio-Cultural Behavioral Focus 2010*. Presented at the Human Socio-Cultural Behavioral Focus 2010.
- Mansoor, F., Zaidi, A. K., Wagenhals, L., & Levis, A. H. (2009). Meta-modeling the Cultural Behavior Using Timed Influence Nets. *Social Computing and Behavioral Modeling*, 146.
- Mansoor, P. (2007). A New Counterinsurgency Center of Gravity Analysis: Linking Doctrine to Action. *Professional Military Review, Journal of the U.S. Army*.
- Mathieson, K. (1991). Predicting user intentions: comparing the technology acceptance model with the theory of planned behavior. *Information systems research*, 2(3), 173–191.
- McPherson, M., Smith-Lovin, L., & Cook, J. M. (2001). Birds of a feather: Homophily in social networks. *Annual review of sociology*, 27(1), 415–444.
- McPherson, M., Popielarz, P., & Drobnic, S. (n.d.). Social Networks and Organizational Dynamics. *American Sociological Review*, 57, 1992.
- McPherson, M., & Ranger-Moore, J. (1991). Evolution on a Dancing Landscape: Organizations and Networks in Dynamic Blau Space. *Social Forces*, 70(1), 19-42.
- National Research Council. (2008). *Behavioral Modeling and Simulation: From Individuals to Societies*. National Academies Press.
- Roberts, N. (2000). Wicked problems and network approaches

to resolution. *International public management review*, 1(1), 1–19.

- Sparks, P., & Shepherd, R. (1992). Self-Identity and the Theory of Planned Behavior: Assessing the Role of Identification with "Green Consumerism". *Social Psychology Quarterly*, 55(4), 388–399.
- Tenenbaum, J., Griffiths, T., & Kemp, C. (2006). Theory-based Bayesian models of inductive learning and reasoning. *Trends in Cognitive Sciences*, 10(7), 309-318. doi: 10.1016/j.tics.2006.05.009
- Walker, G. J., Courneya, K. S., & Deng, J. (2006). Ethnicity, gender, and the theory of planned behavior: the case of playing the lottery. *Journal of Leisure Research*, 38(2), 224.

# Agent Frameworks for Discrete Event Social Simulations

*Jonathan K. Alt*

*Stephen Lieberman*

Modeling, Virtual Environments and Simulation (MOVES) Institute

700 Dyer Road

Naval Postgraduate School

Monterey, California 93943

831-656-7576

jkalt@nps.edu, stlieber@nps.edu

Keywords: social simulation, agent-based modeling, artificial society

**ABSTRACT:** *Discrete event simulation (DES) provides a means of representing abstract concepts in a traceable and rigorous manner that is particularly useful for gaining insights into complex problems associated with human groups. Current problems facing public policy and military decision makers require a greater understanding of societies and their potential responses, both on group and individual actor levels, to a variety of potential policy decisions. Recent work from the military modeling and simulation communities has underscored the need for social simulations that can provide measures designed to inform decision makers of potential futures. Here we describe the application of concepts from DES to the problem of representing societies and provide a framework and overview of core components necessary for the creation and analysis of discrete event social simulations.*

## 1. Introduction

Discrete event simulations (DES) have found extensive use in a variety of applications in operations research and analytic communities across both industry and the government (Henderson et al., n.d.). The DES concept of the event list provides a means of abstracting a variety of concepts and situations into a manageable registry of events that are scheduled and cancelled based on the rules of the simulation (A. Buss, 2001). In social simulations such as the one described herein, this list contains events corresponding to the actions of entities in the model, such as observations, communications, and changes in the internal states (such as belief states) of actors.

Time	Agent ID	Action
1	Blue_1	Observes Political Advertising
2	Blue_1	Changes Political Beliefs
3	Blue_1	Communicates with Blue_2
4	Blue_2	Changes Political Beliefs
5	Blue_2	Communicates with Blue_3
6	Blue_2	Communicates with Blue_4

Table 1: Example of Social Simulation Events List

Crafting an authentic simulated society that is based on real social data, and delineating events such as these, provides a means of gaining insight into the

potential futures of populations and societies that can be applied to a variety of contexts germane to both public policy and military decision makers. DES concepts offer a well understood simulation framework (Schriber & Brunner, 2004) for use in the exploration of the complex behavioral and social systems that comprise a society. With the idea that applying DES to the social and behavioral domains is still under early development, we review DES concepts as applicable to social simulations, provide an overview of a general modeling approach to social simulation that embeds a multi-agent system within a DES framework, and propose several reusable agent patterns for use within these social simulations.

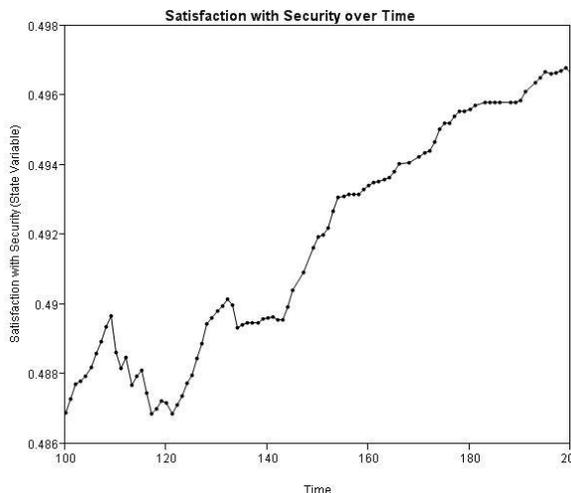
## 2. Discrete Event Social Simulation (DESS) Framework Overview

Discrete event social simulations (DESS) present a simple means of abstracting the complex interactions that exist in societies into model components useful for exploration with simulation experimentation. Below we review concepts from DES, the event graph representation of discrete event simulations, and introduce a specific DESS, the Cultural Geography (CG) model, as a discussion point to explore aspects of this type of framework.

### 2.1 Discrete Event Simulation Overview

DES models are distinguished from time stepped models by the manner in which time is treated in each paradigm. Specifically, in time-stepped models, all simulation events are considered at set intervals as time progresses in the simulation, whereas DES leverages the future events list (FEL) as a means of advancing time in the simulated world (Arnold Buss, 2009). Current events schedule future events to occur at specific times, and update the centrally-maintained FEL accordingly. For example, in Table 1 above, the event the agent’s observation of political advertising at time = 1 schedules the event the corresponding changes to the agent’s political beliefs at time = 2. As events occur, time is advanced in discrete steps from the scheduled execution time of the current event till the scheduled time of the next event on the list, such that the FEL effectively manages the execution of the entire simulation (Arnold Buss, 2009).

The minimum set of elements required for DES models consists of states, events, and scheduling relationships between events (Arnold Buss, 2009). The addition of parameters provides the flexibility to accommodate a broad variety of conceptual models.



**Figure 1.** Entity state transition over model run from CG model, a DESS.

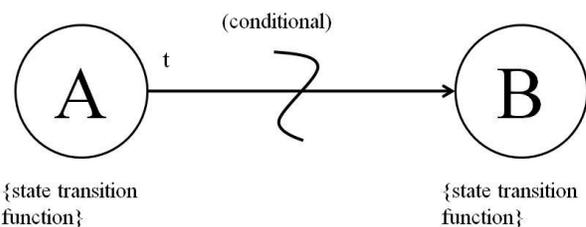
State variables, those DES elements that are able to change at some point during a simulation run, contain the information to provide a complete report on the status of the simulated world at any discrete point in time. State variables are piecewise constant changing instantaneously based on rules described in a state transition function. This approach places the focus on modeling the rules

governing state transitions, but does not restrict the representation of continuous trajectories (Arnold Buss, 2009). Events within DES cause transitions (changes) in state variables. Transitions for all possible cases to be modeled are encapsulated within events that state variables within the simulation. Events may also schedule the occurrence of future events, to include their own. Parameters, by contrast, do not change over the course of a simulation run, but each model instantiation provides a specification of a sequence used during the course of a model run (Arnold Buss, 2009). In the context of social simulations, example state variables include the an entities level of satisfaction on security or other important issues and can be thought of as the results of census polling.

The advance of time relies on the future event list, with time moving forward in non-regular intervals based not on predetermined set time intervals (as in time-stepped simulation), but the time to the occurrence of the next scheduled event on the central event list. All scheduled events are placed on the FEL, maintained, prioritized and canceled based on the rules of the simulation. This centralized management allows for full traceability of model outcomes. For a more complete examination of the implications of time in social simulations, see Alt & Lieberman (2010).

### 2.2 Event Graph Modeling

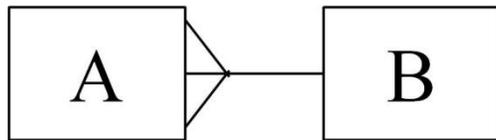
Event graph representations of DES are used to communicate the information described in 2.1 in a more intuitive visual manner. Nodes represent events while edges represent scheduling relationships between events. Conditional relationships can be communicated on the edges and the transition function for each state variable at each node can be fully expressed in associated pseudocode (A. Buss, 2002).



**Figure 2.** Basic event graph, depicting two events (A and B), a conditional scheduling edge, and a delay, t, the scheduling arc.

Event graphs of the specific model components, as shown above, can be combined through the concept of listener patterns in Simkit. This results in a higher level

component mapping described by Buss and Sanchez and referred to as Listener Event Graph Objects (A. H Buss & Sanchez, 2002). Simkit facilitates two listener patterns, the SimEventListener and the PropertyChangeListener. As the names suggest the former listens for the scheduling of events while the latter listens for changes in state variables (A. Buss, 2001). The concept of listeners enables the connection of disparate components maximizing the potential to reuse code objects and event graph components.



**Figure 3.** Graphical depiction of LEGO component model, B listens to events from A.

### 2.3 Cultural Geography Model

The Cultural Geography (CG) Model is an implementation of a DESS that uses an embedded multi-agent system to simulate changes in the beliefs, values, and interests (BVIs) of large social groups (Alt, Jackson, Hudak, & Steven Lieberman, 2010), such as a population<sup>1</sup>. The model, implemented in Simkit<sup>2</sup>, a DES development environment, represents the population in an area of interest as part of a conflict ecosystem (Kilcullen, 2006) that includes conflicting actors (such as government and insurgent forces), and recipients of actions (such as population segments). Scenario development is unique to the area and time period of interest (Alt, Jackson, & Stephen Lieberman, 2009), as well as the population and issues chosen for representation. It closely follows the counter-insurgency intelligence preparation of the battlefield (IPB) framework described by Mansoor (Mansoor, 2007). The key outputs of the model are changes to the BVIs of actors in the population (also called issues stances) on the issues chosen for representation within the simulation. The implementation builds on a conceptual framework grounded in both cognitive psychology and structural sociology (Sanborn, Mansinghka, & T. Griffiths, 2006). Correspondingly, two main modules within the framework are the entity cognition module, which manages the internal states of actors, and the social

structure module, which manages the interactions of agents. Together these modules form the conflict ecosystem within which the agents interact and change their stance on issues of importance.

The theoretical groundwork for the cognitive module relies on Walter Fisher's narrative paradigm (Fisher, 1989) as the premise for the development of issue stances for population sub-groups based on their relevant BVIs. The narrative paradigm proposes that an individual possesses a collection of stories, a unique narrative identity, that encompass their BVIs and shape the way they view the world and interpret events. The narrative identity is implemented as a Bayesian network (Tenenbaum, T Griffiths, & Kemp, 2006).

The social structure module generates theoretically sound and precise patterns of agent interactions based on the internal characteristics of the agent population. A unique social structure exists for every simulated society at each discrete point in time as an expression of the instantaneous distribution of social factors within the society. The well-established idea of homophily, complementary to the narrative paradigm, states that the degree of social factor similarity for every pair of actors corresponds to the pair's likelihood of interaction (McPherson, Smith-Lovin, & Cook, 2001)(Blau & Schwartz, 1997). Social factors are taken to be any attribute that impacts an individual's association, including socio-economic, socio-demographic, and socio-cultural attributes, as well as BVIs. Thus, the more similar a pair in terms of their social factors, the more they interact and influence one another throughout the simulation.

## 3. Event Graph Description of Components for DESS

This section will provide event graph models for generic components used in DESS. These event graphs build on and extend those used in the CG model.

### 3.1 Population Agent

Population agents are modeled as simple reflex agents that interact with the environment, in this case the social network and infrastructure objects, based on a set of conditional statements provided at their instantiation.

Parameter:

- Demographic composition: age, sex, education, occupation.
- Consumption rate of commodities: energy, food.

<sup>1</sup> The CG Model is government-owned, open-source, and available free of charge at [https://soteria.nps.navy.mil/rucgwiki/index.php/Main\\_Page](https://soteria.nps.navy.mil/rucgwiki/index.php/Main_Page)

<sup>2</sup> SimKit is freely available at <http://diana.nps.edu/Simkit/>

- Communication rate.

State variable:

- Issue Stance, {0...1}: satisfaction with security, satisfaction with infrastructure.
- Location, {1...n}: discrete named locations.

Event Graph:

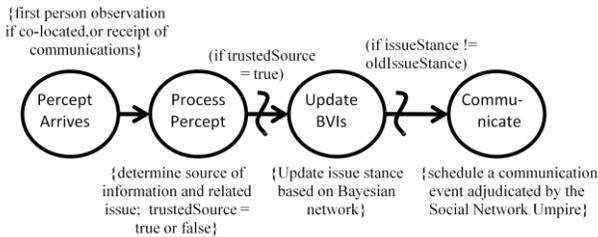


Figure 4. Event graph depicting a civilian entity component.

The state transition function used in the case of civilian entities in the CG model is implemented as a Bayesian belief network (BBN).

### 3.2 Threat Agent

Threat agents, gangs or violent extremist networks (VEN), are currently treated as single reflex agents within the model and not a true network of interacting entities. Work is ongoing to provide add more detail to this portion of the model as traceable data becomes available.

Parameter:

- Demographic composition: age, sex, education, occupation.
- Role: direct action, planner, etc.

State variable:

- Average Population Issue Stance, {0...1}.
- Location, {1...n}: discrete named locations.

Event Graph:

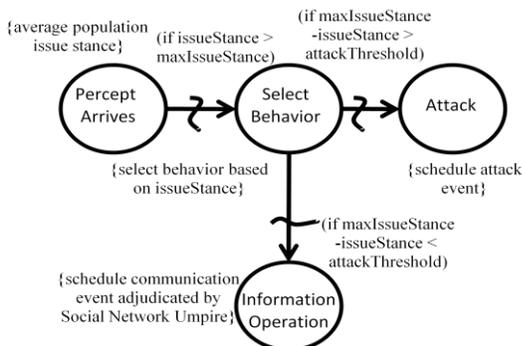


Figure 5. Event graph depicting threat agent component.

The state transition function used in the threat agent in this case based on statistics from the environment that are

accessible by the threat agent. Design decisions describing the level of access to knowledge of other entities aside, the calculation of this is a straightforward calculation of the mean issue stance on a given issue.

### 3.3 Media Agent

Media agents receive information and retransmit information from the simulation environment. They can also send out messages in a semi-autonomous manner, regardless of the incoming information from the simulation environment depending on design decisions made during scenario construction.

Parameter:

- Affiliation: political party, pro/anti government.

State variable:

- Publication rate.
- Location, {1...n}: discrete named locations.

Event Graph:

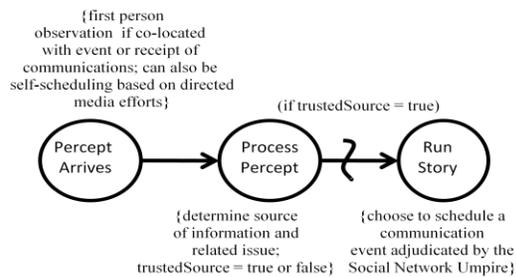


Figure 6. Event graph depicting media entity component.

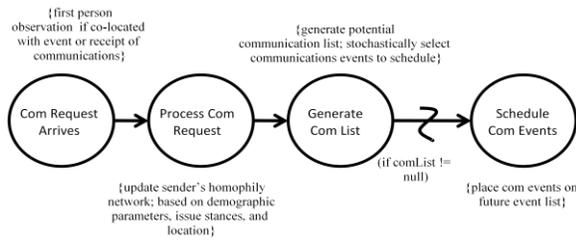
### 3.4 Representing the Social Network through Referees

The central component that allows for and facilitates the interaction of agents is the social network referee. This component adjudicates and schedules communications throughout the artificial society. The entity itself does not contain state variables, but instead a set of rules in the form of parameters are used to determine the recipients of communications that are scheduled by the other entities within the simulation.

Parameter:

- Social distance equation.
- Relationship threshold.
- Communications rate.

**Event Graph:**



**Figure 7.** Event graph depicting social network umpire component.

The social distance equation used in the artificial society is a realization of the concept of homophily as explained above. Each agent occupies a position in multidimensional space based on their internal attributes. This space is a hyperrectangle where the length of each edge is determined by the range of values of the corresponding social attribute. Each dimension of this space represents a social factor, that is, an internal attribute that influences the interactions of the agent. The likelihood that a pair of agents will interact is directly proportional to their distance in this space where more similarity (shorter distance) indicates increased likelihood of interaction. Thus, social distance is calculated simply as the Euclidean distance between any two agents occupying positions in this hyperrectangle.

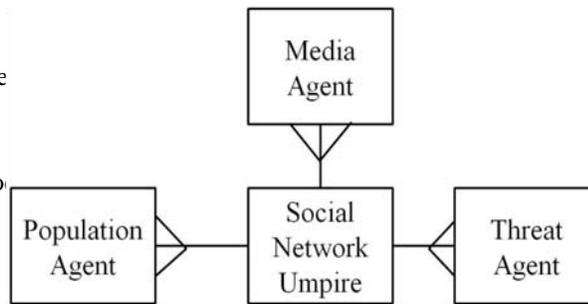
While every agent is connected in the society (i.e., it is possible for all agents to interact), there is a practical bound or threshold on the distance. Since agents are more likely to communicate with those in proximate space, we can understand the social structure of the artificial society by thresholding relationships between agents (i.e., for visualization) where agent-pairs that surpass a certain social distance are understood to *not* be connected with one another.

The social distance directly controls *which* other agents will be targeted for communication by an agent. The communication rate, likewise, specifies the time it takes for that communication to be initiated and completed. Similarly to the intrinsic relationship threshold, there is an inherent limit to the number of communications that an agent can engage in over a set period of time. This parameter is controlled directly for the agent population with a communications rate specification. This controls both the maximum number of other agents engaged, and the maximum number of messages that can be passed, over a certain period of time.

**4. Component Level Architecture**

The use of component level architectures flows naturally from the event graph. A single event graph depiction of even the simple components described in section 3 would rapidly become confusing and unreadable. The use of component level diagrams allow the communication of complex models in an efficient manner and facilitate the rapid re-use of previously developed and functional code.

Each component represents a fully complete instance of the event graph model. In the case of social simulation the components are linked using an event listener pattern. In the diagram below, the SocialNetworkUmpire component listens for the scheduling of communications events and attack events.



**Figure 8.** Component level architecture for discrete event social simulation with SimEventListener pattern.

**5. Conclusions and Future Work**

The use of DES for social simulation presents opportunities to develop emergent societies and behavior in a fully traceable manner. The use of these techniques have implications for the validation of this class of models for use in a variety of settings in support of decision makers. The use of modular frameworks supported by DES facilitates the re-use of code and the implementation of competing theoretical concepts for experimentation.

**6. Reference List**

Alt, J., Jackson, L., Hudak, D., & Lieberman, S. (2010). The Cultural Geography Model: Evaluating the Impact of Tactical Operational Outcomes on a Civilian Population in an Irregular Warfare Environment. *Journal of Defense Modeling and Simulation*, (Special Issue on Counter-Insurgency).

Alt, J., Jackson, L., & Lieberman, S. (2009). The Cultural Geography Model: An Agent Based Modeling Framework for Analysis of the Impact of Culture in

- Irregular Warfare. In *International Conference on Computational Cultural Dynamics*. Presented at the Third International Conference on Computational Cultural Dynamics, Maryland.
- Alt, J., & Lieberman, S. (2010). Exploring the Implications of Time in Discrete Event Social Simulations. *Association for the Advancement of Artificial Intelligence Spring Symposium 2010*.
- Blau, P., & Schwartz, J. (1997). *Crosscutting Social Circles: Testing A Macrostructural Theory of Intergroup Relations*. Transaction Publishers.
- Buss, A. (2001). Discrete event programming with Simkit. *Simulation News Europe*, 32(33), 15–25.
- Buss, A. (2002). Simkit: component based simulation modeling with Simkit. In *Proceedings of the 34th conference on Winter simulation: exploring new frontiers* (pp. 243–249).
- Buss, A. H., & Sanchez, P. J. (2002). Modeling very large scale systems: building complex models with LEGOs (Listener Event Graph Objects). In *Proceedings of the 34th conference on Winter simulation: exploring new frontiers* (pp. 732–737).
- Buss, A. (2009). Discrete Event Simulation Modeling. Naval Postgraduate School.
- Fisher, W. (1989). Clarifying the Narrative Paradigm. *Communication Monographs*, 56(1), 55-58.
- Henderson, S. G., Biller, B., Hsieh, M. H., Shortle, J., Tew, J. D., & Barton, R. R. (n.d.). COMPOSABILITY AND COMPONENT-BASED DISCRETE EVENT SIMULATION.
- Kilcullen, D. (2006). Counter-insurgency Redux. *Survival*, 48(4), 111–130.
- Mansoor, P. (2007). A New Counterinsurgency Center of Gravity Analysis: Linking Doctrine to Action. *Professional Military Review, Journal of the U.S. Army*.
- McPherson, M., Smith-Lovin, L., & Cook, J. M. (2001). Birds of a feather: Homophily in social networks. *Annual review of sociology*, 27(1), 415–444.
- Sanborn, A., Mansinghka, V., & Griffiths, T. (2006). A Bayesian framework for modeling intuitive dynamics.
- Schriber, T. J., & Brunner, D. T. (2004). Inside discrete-event simulation software: how it works and why it matters. In *Proceedings of the 36th conference on Winter simulation* (p. 152).
- Tenenbaum, J., Griffiths, T., & Kemp, C. (2006). Theory-based Bayesian models of inductive learning and reasoning. *Trends in Cognitive Sciences*, 10(7), 309-318. doi: 10.1016/j.tics.2006.05.009

# Modeling the Control of Attention in Complex Visual Displays

Kelly S. Steelman-Allen

Jason S. McCarley

University of Illinois at Urbana-Champaign

2251 Beckman Institute

405 North Mathews Avenue

Urbana, Illinois 61801

ksteelm2@illinois.edu, mccarley@illinois.edu

Keywords:

Modeling, visual attention, display design

**ABSTRACT:** A stochastic model of overt attention within a visual display or workspace is presented. The model integrates elements from several existing models of attention (Bundesen, 1987, 1990; Itti & Koch, 2000; Wolfe, 1994; Wickens et al., 2003) to provide predictions of the allocation of visual attention among discrete display channels and the number of eye movements needed to fixate the onset of a visual signal or event. The model was validated against data from an alert detection experiment (Nikolic, Orr and Sarter, 2004), with results demonstrating that the model can accurately predict the effects of color similarity, eccentricity, and dynamicity on attentional behavior and target detection.

## 1. Introduction

In many operational domains, including aviation, nuclear power, and process control, one of the operator's primary tasks is to monitor for visual warnings or alerts. The detectability of such visual events is modulated by a variety of bottom-up and top-down factors, including the display context, the operator's mental model of system, and task demands. In a study by Nikolic, Orr, & Sarter (2004), for example, subjects monitored a display for the onset of a visual alert while engaged in a game of Tetris. Alert location and contrast, the presence of movement in the display, and the operator's level of attentional load were all varied. The detectability of alerts was found to depend on the interaction of these various factors, suggesting that design criteria that consider any one factor in isolation may not encourage effective display design.

The present paper describes a computational model to predict attentional behavior and target detectability within complex displays, offering designers a tool to test the effectiveness of various alerts in multiple display configurations and under varying task demands. The model incorporates elements from several computational models of basic attentional processes (Bundesen, 1987, 1990; Itti & Koch, 2000; Wolfe, 1994) within the heuristic SEEV framework of Wickens and colleagues (Wickens et al., 2003) to create a model of attentional behavior in dynamic environments.

## 2. The Model

The model assumes a scenario in which an operator monitors a display, comprising an array of discrete information channels, for some amount of time before the onset of a target event in one channel. The model predicts the steady-state distribution of attention among display channels, as measured in percentage of visual dwell time (McCarley & Kramer, 2006), prior to target onset; the likelihood of a scanning transition between any pair

of channels prior to target onset; and the number of eye movements needed to fixate the target channel after the target appears. The model was implemented using Matlab 2008a and the Saliency Toolbox (Walther & Koch, 2006).

The model builds on the framework of Wickens' SEEV model (Wickens et al., 2003), which derives its name from the four forms of attentional influence that it posits: signal saliency, the effort needed for attention to reach the signal, the operator's expectancy of the signal, and the task-relevance or value of the signal. The current model modifies and elaborates on the original SEEV model in multiple ways. First, it distinguishes between two forms of visual saliency: *static saliency*, based on local image-based feature contrast (cf. Itti & Koch, 2000), and *dynamic saliency* (cf. Yantis & Jonides, 1990), based on moment-to-moment changes of static saliency. Second, it distinguishes between two forms of top-down control: *channel prioritization*, based on the operator's estimates of the bandwidth and value of a given channel (cf., Senders, 1983), and *feature prioritization*, based on the operator's attentional set for a given color (cf. Wolfe, 1994). Third, it determines the saliency of each channel computationally using the Itti and Koch (2000) saliency model. Finally, it models the effects of effort on attentional scanning using a Gaussian spatial filter that simulates acuity loss in the peripheral retina and/or attentional tunneling, reducing the probability of long shifts of attention.

### 2.1 Inputs and model assumptions

As input, the model accepts image files of the pre- and post-target displays, a map of the display's information channels or *areas of interest (AOIs)*, and a parameter file specifying the bandwidth and value of each AOI. For simplicity, the model assumes that the pre-defined AOIs are the only locations in the image that can be fixated and that fixations always occur at the center of a given AOI. In its current form, the model also assumes that the target is noticed once it has been fixated, but this assumption could be easily replaced with the assumption of

a probabilistic signal detection judgment.

## 2.2 Operation

The model operates by first producing a set of *base maps* representing various sources of attentional guidance. These maps are assigned *pertinence values* (Bundesen, 1990) based on the operator's task set, and the pertinence-weighted maps are averaged to produce a master map of attentional activation. Finally, a probabilistic choice model (Bundesen, 1987; Luce, 1959) determines the location of the operator's next fixation based on the attentional activation map.

## 2.3 Base Maps

The base maps represent four sources of attentional guidance: static salience, dynamic salience, channel priority, and feature priority.

*Static Salience Map.* The current model estimates the salience of each display channel using the computational model of Itti and Koch (2000). The Itti and Koch model employs center-surround filters to create a set of maps that represent feature contrast within the luminance, chromatic, and orientation dimensions. These within-feature contrast maps are then combined to form an overall saliency map, rendered in 16x16 logical pixels, with possible salience values ranging from 0 to a maximum value of 3. The current model normalizes the overall saliency map with respect to the maximum, to allow comparison of the static salience maps across simulations. For each iteration,  $i$ , of the model, a static salience map is generated based on the current display image. If the target has not yet appeared, then the pre-event input image is used to generate the map. If the change has already occurred, the post-event image is used.

*Dynamic Salience Map.* The dynamic salience map represents moment-to-moment changes in static salience resulting from the onset of the target or other sources of movement or flicker within the display. The model generates the map by calculating the Perceptual Euclidean Distance (PED) between the pre- and post-change images. The PED is similar to the traditional Euclidian distance but weighted to represent perceptual differences in color change detection for red, green and blue (Gijssenij, Gevers, & Lucassen, 2008). Calculating the PED for each pixel in the image produces a grey-scale map of changes in the display. This change map is then passed to the salience model, resulting in the dynamic salience map.

*Feature Priority Map.* The feature priority map is created by assessing the match between each pixel in the image and a set of target colors (e.g., red, green, blue, and amber). To assess the match for each color, the PED is calculated between the target RGB value and each pixel in the image. The color match is represented discretely, with a value of 1 indicating a match and zero otherwise. Pixels that fall within 40 units of the target color are considered a match. Each individual color map is then weighted according to its relevance to the task. For example, if red alerts represent danger and amber alerts represent potential danger, red may be assigned a value of 1 and amber a value of .75. The weighted color maps are then

combined to form the final feature priority map.

*Channel Priority Maps.* The value and expectancy maps are both created heuristically. For each information channel in the display, the modeler provides the value and expectancy levels on a scale from 0-1. Both value and expectancy are assumed to remain constant during the task and are considered to be a function of the operator's mental model of the system and task. Accordingly, the values and expectancies are not considered model parameters that can be changed to better fit a set of data. Appropriate determination of the expectancy and values is thus an important step and requires the modeler to carefully consider both the nature of the display and the knowledge of the assumed operator.

## 2.4 Master Map

The master map of attentional activation values is created by averaging the activation of the base maps, with the input from each base map weighted by a pertinence value (Bundesen, 1990) assigned by the modeler. Pertinence values allow strategic changes in a modeled operator's attentional policy in response to changing task demands. For example, to allow attentional guidance driven entirely by bottom-up salience, the modeler can assign values of 1 to the static and dynamic salience maps and 0 to the other maps. Alternatively, to allow guidance based purely on top-down influences of bandwidth and information value, the modeler can assign a value of 1 to the two channel priority maps and 0 to the remaining three maps. Assigning equal pertinence values to all five base maps ensures that all five contribute equally to attentional guidance.

In order to simulate the effort required to execute a long attention shift (e.g., Ballard, Hayhoe, & Pelz, 1995) and/or the effects of acuity losses in the peripheral retina, a Gaussian spatial filter is applied to the master map at the center of the currently fixated AOI,  $L_i$  (cf., Parkhurst et al., 2002). The size of the filter,  $\sigma_{VL}$ , represents the size of the operator's visual lobe (Chan & Courtney, 1996) and can be adjusted to model individual differences (e.g., Pringle et al., 2001) or the influence of workload or stress (e.g., Atchley & Dressel, 2004) on attentional breadth.

## 2.5 Target selection

Finally, the mean activation level within each AOI is calculated to determine a single activation value,  $A_j$ , for each of the  $j$  AOIs. This value is the *attentional weight* of the AOI. The choice of an AOI for attentional selection is determined probabilistically based on the AOIs' relative attentional weights. More particularly, the probability that a given AOI is selected as the target for the next attention shift is given by a choice model (Bundesen, 1990):

$$P(\text{select } AOI_j) = A_j / \sum A,$$

where  $A_j$  is the attentional weight of AOI $_j$ , and  $\sum A$  is the summed value of the attentional weights for all AOIs. The choice equation effectively implements an independent race between AOIs for attentional selection (Bundesen, 1993)

To discourage consecutive attentional fixations on the same AOI, inhibition of return (IOR) can be applied to the attentional weight for the currently fixated AOI. IOR is a value between 0 and 1. In the case that  $IOR > 0$ , the attentional weight of the currently fixated AOI is multiplied by  $(1-IOR)$  before it is entered into the choice model, reducing the probability of a subsequent fixation in the same AOI. Thus, a value of  $IOR = 1$  ensures that the model will never fixate the same AOI consecutively. Conversely, a value of  $IOR < 1$  allows for consecutive fixations on a single AOI, introducing the possibility of attentional tunneling on channels of high bandwidth, value, or salience (Wickens & Alexander, 2009).

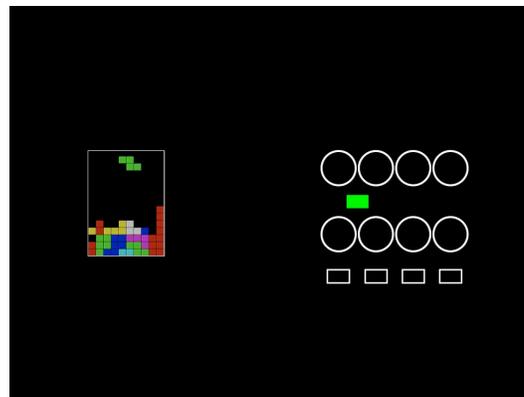
After the new fixation location is selected, a new master attentional activation map is created based on the current fixation location, and the selection process repeats. After the target event onset, the process continues until the model fixates the target AOI.

### 2.6 Model output

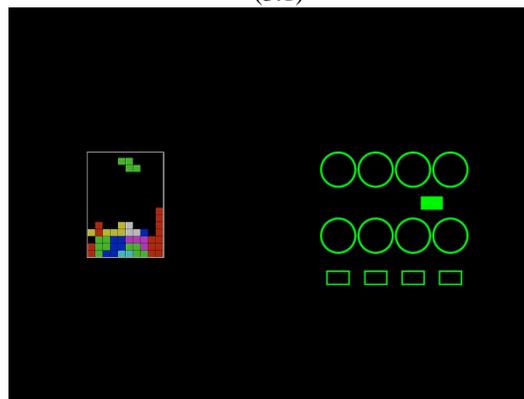
The model can be set to run for any number of fixations prior to the onset of the target, providing a distribution of steady-state scanning behavior within the pre-change display. After the onset of the target, the model continues to run until the changed AOI is fixated. Because the model is stochastic, the number of fixations required to locate the changed AOI varies between runs, producing a distribution of noticing times. This distribution can be used to predict mean cumulative target detection rate as function of time following target onset (Wickens et al., 2009).

### 3. Results

The model was validated against miss rates from an alert detection experiment (Nikolic et al., 2004). In the experiment, participants played a game of Tetris while simultaneously monitoring an adjacent display for the onset of a green alert. Three factors were manipulated in a  $2 \times 2 \times 2$  design: eccentricity of the alert with respect to the Tetris display (35 vs. 45 degrees of visual angle), color similarity between the alert and surrounding display objects, and dynamicity of objects near the alert. Schematic images from each of the eight conditions served as input to the model. Figure 3.1 presents the display for the low color similarity, near target location condition. Figure 3.2 illustrates the display image from the high color similarity, far target location condition. In the dynamic condition, the eight circular gauges contained random movement of the gauge pointer. In the static condition, there was no movement.



(3.1)



(3.2)

*Figures 3.1 and 3.2* Representative displays from the low similarity, near target location condition (left) and the high similarity, far target location condition (right). Each display contained 15 areas of interest: 1 Tetris game, 8 gauges, 2 possible target locations, and 4 text boxes. The target was a green box, located between the two rows of gauges. In the low similarity condition, the objects surrounding the target were white. In the high similarity condition, the objects surrounding the target were green.

Pertinence values were assigned heuristically based on judgments about the relative usefulness of various forms of attention guiding information for detecting the target within each condition. More specifically, a pertinence value of 1 was assigned to each form of information that differentiated the target event from non-target events, and a value of 0 was assigned to all the remaining forms of information. Thus, for example, dynamic salience (due to the onset of the target) was assigned a pertinence of 1 in the static distractor conditions and 0 in the dynamic distractor conditions. Two experimenters independently assigned pertinence values for each condition and were in 100% agreement in all assignments (Table 3.1).

Source	High Similarity/ Dynamic	Low Similarity/ Dynamic	High Similarity/ Static	Low Similarity/ Static
Static Saliency	0	1	0	1
Dynamic Saliency	0	0	1	1
Value	1	1	1	1
Expectancy/Bandwidth	1	1	1	1
Attentional Set (Color)	0	1	0	1

Table 3.1 Pertinence values for each condition.

Note that the same sets of pertinence values were used in the near and far conditions. Distance effects were implemented by a Gaussian Spatial Filter with a standard deviation of 190 pixels, or approximately 15 degrees of visual angle. The IOR parameter was set to zero.

Pre- and post-alert images and the set of model parameters were input to the model. The model was run for 1000 iterations. Each iteration, the initial fixation was on a randomly selected AOI. After 100 fixations, the alert onset occurred, and the model was then allowed to run until the alert was fixated. To calculate a miss rate, the number of fixations-to-detection was first converted into a detection time by assuming a mean fixation duration. As the alert was assumed to remain visible for 10 seconds, if the detection time was greater than 10 seconds, that iteration was considered a miss. Accordingly, miss rates were dependent on the assumed fixation durations, with misses occurring after 10, 20, 30 or 40 fixations depending on whether 1000, 500, 333 or 250 ms fixations durations were assumed (corresponding to 1-4 fixations/second).

For each of the four assumed fixation durations, the Pearson correlation, Spearman’s rank order correlation, and the root mean square error (RMSE) were calculated between predicted and actual miss rates (Table 3.2). Neither the Pearson correlation between the predicted and actual miss rates nor the rank order correlation varied significantly with the assumed number of fixations per second. The RMSE was minimized

with assumed fixation durations of 250 or 333ms.

Fix/sec	r	r <sub>s</sub>	RMSE
1	0.91	0.95	0.28
2	0.94	0.95	0.13
3	0.95	0.95	0.05
4	0.95	0.95	0.06

Table 3.2 Pearson correlation, Spearman’s rank order correlation, and the root mean square error.

Figure 3.3 presents the predicted and observed miss rates for each condition, based on an assumed fixation duration of 250 ms. Figure 3.4 presents the same data collapsed across condition to illustrate the effects of target eccentricity, target color distinctiveness, and dynamic distractor content on predicted and observed miss rates. The model accurately predicted the empirical difference between the dynamic and static conditions, with moving gauges producing higher miss rates. The model also predicted the effects of both the eccentricity and color. As is evident in both figures, predicted miss rates generally underestimated observed miss rates ( $M_{diff}=-.042, SD=.043$ ). Employing an assumed fixation duration of 333 ms helped to correct this effect, with underestimation of the miss rates in only 3 conditions, but overestimation in all others ( $M_{diff}=.025, SD=.432$ ).

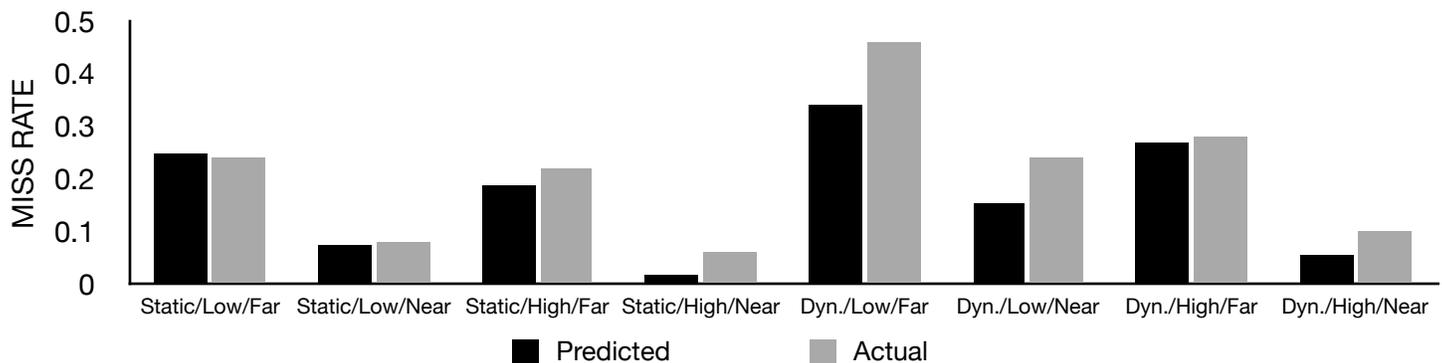


Figure 3.3 Predicted and actual miss rates for all 8 conditions.

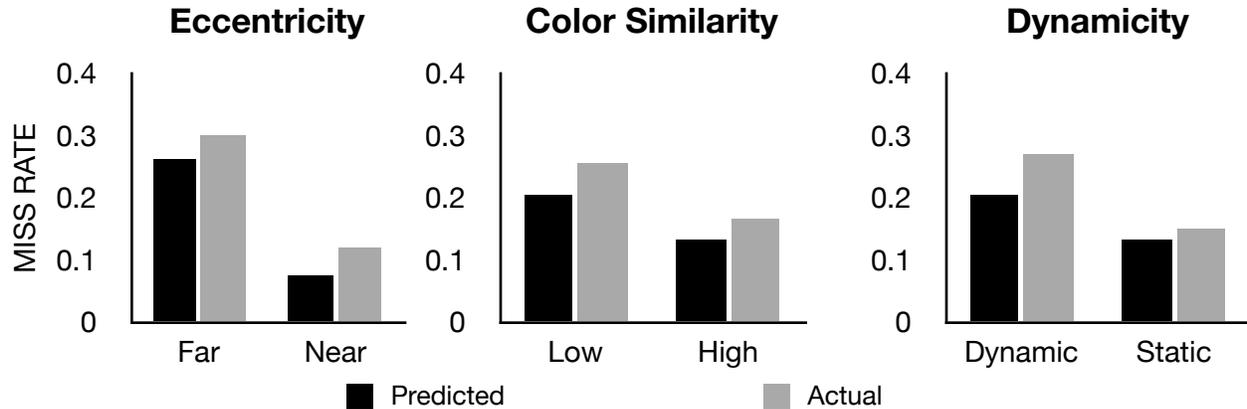


Figure 3.4 Predicted and actual miss rates, collapsed across conditions to illustrate the effects of eccentricity, color similarity, and dynamicity.

## 4. Conclusions

Based on the general framework of SEEV (Wickens et al., 2003), the current model assumes attentional guidance driven by signal salience, expectancy, and value, but distinguishes between static and dynamic visual salience and two manifestations of top-down guidance. The model thus accommodates multiple bottom-up and top-down factors that influence the noticeability of a visual event. It provides predictions of steady-state attentional behavior in a display and the number of eye movements required to fixate a visual event.

The model was validated here against miss rates from Nikolic et al.'s (2004) alert detection experiment. Results suggest that the model can reliably predict noticing behavior and can account for the effects of color similarity, eccentricity, and dynamic noise on target detection rates. Moreover, the validation confirmed that the model can be successfully fit using pertinence values selected through a simple heuristic. Additional validation is underway, focusing on modeling the distribution of oculomotor fixations within a complex workspace. Future efforts will attempt to model individual differences in attentional guidance and noticing, as well as the effects of mental workload on attentional behavior.

## 5. References

- Atchley, P., & Dressel, J. (2004). Conversation Limits the Functional Field of View. *Human Factors*, 46(4), 664-673.
- Ballard, D. H., Hayhoe, M. M., & Pelz, J. B. (1995). Memory Representations in natural tasks. *Cognitive Neuroscience*, 7, 66-80.
- Bundesen, C. (1987). Visual attention: Race models for selection from multielement displays. *Psychological Research*, 49, 113-121.
- Bundesen, C. (1990). A theory of visual attention. *Psychological Review*, 97(4), 523-547.
- Bundesen, C. (1993). The relationship between independent race models and Luce's choice axiom. *Journal of Mathematical Psychology*, 37(3), 446-471.
- Chan, A. H. S., & Courtney, A. J. (1996). Foveal acuity, peripheral acuity, and search performance: A review. *International Journal of Industrial Ergonomics*, 18, 113-119.
- Gijssenij, A., Gevers, T., & Lucassen, M.P. (2008) A perceptual comparison of distance measures for color constancy algorithms. Proceedings of the European Conference on Computer Vision, Volume I (pp. 208-221). Marseille, France: Springer.
- Itti, L. & Koch, C. (2000). A saliency-based search mechanism for overt and covert shifts of visual attention. *Vision Research*, 40(10-12), 1489-1506.
- Luce, R. D. (1959). *Individual choice behavior*. New York: Wiley.
- McCarley, J.S. & Kramer, A.F. (2006). Eye movements as a window on perception and cognition. In M. Rizzo & R. Parasuraman (Eds.), *Neuroergonomics: The Brain at Work*. N.Y.: N.Y. Oxford University Press.
- Nikolic, M. I., Orr, J. M., & Sarter, N. B. (2004). Why Pilots Miss the Green Box: How Display Context Undermines Attention Capture. *The International Journal of Aviation Psychology*, 14(1), 39-52.
- Parkhurst, D., Law, K., & Niebur, E. (2002). Modeling the role of salience in the allocation of overt visual attention. *Vision Research*, 42, 107-123.
- Pringle, H. L., Irwin, D. E., Kramer, A. F., & Atchley, P. (2001). The role of attentional breadth in perceptual change detection. *Psychonomic Bulletin and Review*, 8(1), 89-95.
- Walther, D. & Koch, C. (2006). Modeling attention to salient proto-objects. *Neural Networks*, 19, 1395-1407.
- Wickens, C.D., & Alexander, A.L. (2009). Attentional tunneling and task management in synthetic vision displays. *International Journal of Aviation Psychology*, 19, 1-17.
- Wickens, C., McCarley, J., Sebok, A., Steelman-Allen, K., Sarter, N. & Bzostek, J. (2009). NT-SEEV: A model of attention capture and noticing on the flight deck. Submitted to 2009 Annual meeting of the Human Factors & Ergonomics Society.
- Wickens, C. D., Goh, J., Helleberg, J., Horrey, W., & Talleur, D. A. (2003). Attentional models of multi-task pilot

performance using advanced display technology. *Human Factors*, 45(3), 360-380.

Wolfe, J.M. (1994). Guided search 2.0 A revised model of visual search. *Psychonomic Bulletin & Review* 1(2), 202-238.

Yantis, S., & Jonides, J. (1990). Abrupt visual onsets and selective attention: Voluntary versus automatic allocation. *Journal of Experimental Psychology: Human Perception and Performance*, 16, 121-134.

## Author Biographies

**KELLY S. STEELMAN-ALLEN** is a PhD student in Psychology at the University of Illinois at Urbana-Champaign. She has a M.S. in Human Factors from UIUC and Masters of Mechanical and Aerospace Engineering from Illinois Institute of Technology. Her research interests include basic and applied attention, display design, and instructional design.

**JASON S. McCARLEY** is an assistant professor in the Department of Psychology and the Human Factors Division at the University of Illinois at Urbana-Champaign.

# Collaboration and Modeling Support in CogLaborate

*Reuben Cornel*

*Robert St. Amant*

Department of Computer Science

North Carolina State University

Raleigh, NC 27695

reuben.cornel@gmail.com, stamant@csc.ncsu.edu

*Jeff Shrager*

Stanford University Symbolic Systems Program (consulting)

Palo Alto, CA 94301

jshrager@stanford.edu

**ABSTRACT:** *This paper describes the CogLaborate system, a collaborative, tool-based environment for the ACT-R cognitive modeling community. CogLaborate is based on BioBike, which supports collaboration between biologists and computer scientists. This paper discusses how comparable benefits can be brought to cognitive modelers, and presents the design of CogLaborate, its frame-based representation for models, and a proof of concept in the form of an ACT-R module developed within the environment.*

## 1. Introduction

Research on cognitive modeling has driven the formation of active, thriving communities. With ACT-R, for example, beyond the core group of researchers at Carnegie Mellon University, we have annual workshops, a summer school to introduce new researchers to the framework, a Web site, an active mailing list, and any number of small interdisciplinary groups of collaborators distributed throughout the world. The result has been a continuous stream of refinements and extensions to ACT-R, both the theory and the software architecture, as well as models, experiments, development tools, and the like.

In important ways the ACT-R research community is not unique *as* a community. Consider a vision of online communities that dates back to 1968 [Licklider and Taylor, 1968]:

They will be communities not of common location, but of common interest. In each field, the overall community of interest will be large enough to support a comprehensive system of field-oriented programs and data.

A subfield of human-computer interaction, computer-

supported collaborative work (CSCW), has produced a variety of concepts and tools based on this vision to help support collaboration between people and to foster online communities. The research described in this paper is an attempt to build a collaborative online environment for cognitive modelers, to explore the potential benefits of a CSCW approach to the field. We have developed a system called CogLaborate for this purpose.

In contrast to related research on extending the scope of modeling efforts beyond individual researchers and small teams (e.g. [Gluck et al., 2007]), the focus of CogLaborate is on model development rather than model execution. CogLaborate currently runs in prototype form on the Cyano server at the Carnegie Institution of Washington in Washington DC and client machines at the North Carolina State University. We have built CogLaborate to support the following:

- *Sharing of architecture extensions and running models.* Some extensions to ACT-R are more difficult to set up than others. In CogLaborate, such extensions can be tested and uploaded by modeling researchers to a shared environment for others to use immediately, saving repeated effort. Further, in contrast to a static model repository or a

conventional software configuration management system, CogLaborate can maintain models in a long-running Lisp environment, where they can be ready to execute, paused in their execution, or even executing in the long term.

- *Sharing of software and hardware resources* to support the development and dissemination of models and modeling software. Although CogLaborate does not approach the model execution capabilities of other systems (e.g. [Gluck et al., 2007]), it outmatches the performance of our local machines, even given network communication overhead.
- *Support for model analysis tools.* One important aspect of the CogLaborate project is the potential to support analysis of the structure and content of models. CogLaborate translates ACT-R models into a frame-based representation [Minsky, 1974], to support search and browsing by modelers. This means that procedures for analyzing models (currently under development) need not parse ACT-R code directly; instead they can rely on a slightly more abstract and uniformly structured representation.

CogLaborate is a new system, and we have not yet evaluated how and whether collaboration can benefit cognitive modeling research. Even in its prototype state, however, the promise of CogLaborate can be seen in two ways. First, we believe that a frame-based representation offers significant advantages for sharing and analyzing models, in comparison with their storage as modeling code. Second, we have exercised CogLaborate by building a specialized ACT-R module that relies on an existing extension to ACT-R (WN-Lexical) [Emond, 2006] and a model to test the new module. This experience exposed some of the procedural difficulties in carrying out such a task as well as the benefit that CogLaborate could provide the cognitive modeling community. We believe that our proof of concept—a new model running on an ACT-R extension that requires no more effort to install than logging into a remote server—demonstrates the value of our approach.

## 2. BioBike

CogLaborate is built on the Biobike platform. BioBike is an instantiation of KnowOS [Travers et al., 2005], a refinement of the concept of the operating system. Operating systems provide useful abstractions for users

to work with the elements of a system. Files, for example, abstract away the details of how data is stored on hardware, and an OS provides functions for creating, managing, and manipulating data using this abstraction. The KnowOS vision extends this analogy to the realm of knowledge. An implementation of the KnowOS consists of the following layers [Travers et al., 2005]:

- A knowledge base, in a frame representation.
- An extensible programming language with appropriate abstractions for users to work with the system.
- A interface to the programming language and to other KnowOS services.

BioBike (originally known as BioLingua) provides biologists with the ability to perform computational biology operations on large data sets using a simple language [Massar et al., 2005]. BioBike ties a number of knowledge bases together transparently, using frames to represent organisms. As a KnowOS, it provides features customized for molecular biologists. These include

- A common framework to access genomic, metabolic, and experimental data.
- A general-purpose programming language (Lisp) customized for transparent access to the underlying knowledge bases.
- A highly interactive environment where code can be evaluated and its results displayed immediately.
- A number of general-purpose tools that help in analyzing interactions.
- A wiki through which scientists can collaborate and announce results.

BioBike provides biologists, in principle, with an environment in which they interact with the computer in the same terms as they would interact with their peers; with a uniform framework for accessing knowledge from a number of different knowledge bases; and with a common work area where data and results can be shared and external tools can be integrated. BioBike has been in place over a number years and has demonstrated benefits to collaborating teams of biologists and computer scientists during that time [Massar et al., 2005].

From a CSCW perspective [Rodden, 1991], the type of collaboration BioBike is designed to support is asyn-

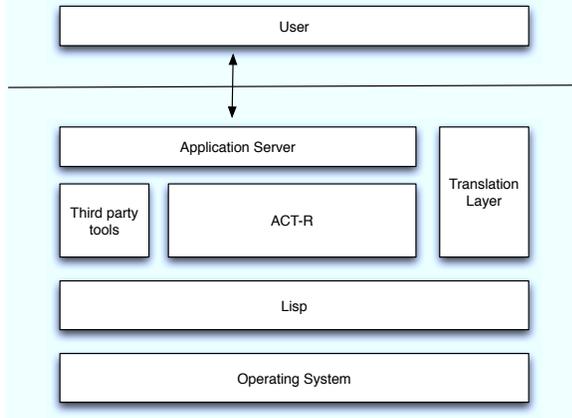


Figure 1: System Design

chronous (not requiring collaborators to interact in real time) and geographically distributed (not requiring collaborators to be co-located). The synchronous/asynchronous and co-located/distributed distinctions do not create hard boundaries between categories of CSCW systems, but they do help us distinguish between message systems, conferencing systems, meeting rooms systems, and co-authoring systems. Of these categories, BioBike can be seen most naturally as an example of the last.

Figure 1 provides a high-level overview of CogLaborate, implemented on the BioBike chassis. Users interact through a Web-based application server with ACT-R and its third party extensions. The translation layer runs side by side with ACT-R, creating frame-based representations of ACT-R models when they are loaded and compiled; the user has access both to ACT-R and to these representations. These components are layered on top of a Lisp environment, which in turn runs on the operating system of the servers. This organization is fleshed out in more detail in Section 4.

The ACT-R component in CogLaborate replaces biology-specific functionality in BioBike; the modular structure of BioBike made this feasible. CogLaborate added only about 1,000 lines of new code to the existing code bases of ACT-R and BioBike.

### 3. Model representation

ACT-R models are essentially Lisp data structures. One plausible representation of models in CogLaborate is simply the Lisp code that defines models at the top level. This approach has a few disadvantages, how-

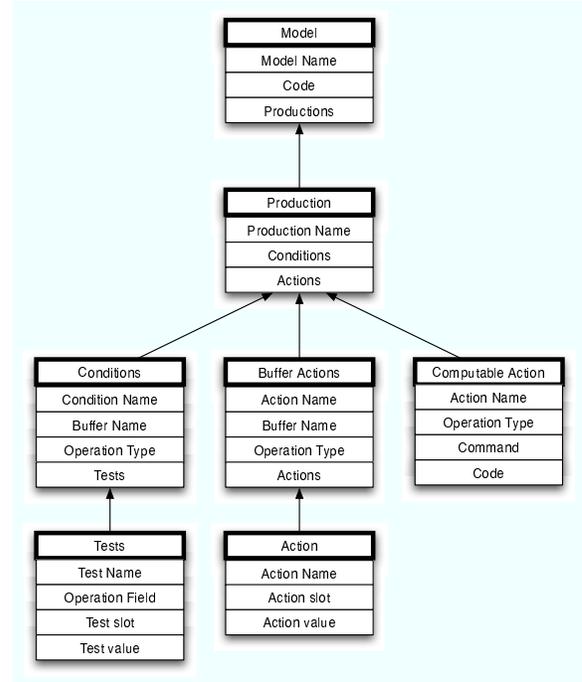


Figure 2: Frame representation for ACT-R models

ever. A direct representation exposes search, browsing, and analysis tools to the syntax and structure of models, in some cases requiring parsing at the textual level. (For example, forms such as `=goal>` and `+goal>` are related—they access the goal buffer—but they are not tokenized as such by the Lisp reader.) Other software engineering issues arise as well in the context of collaboration, such as the difficulty of managing meta-data associated with models and knowledge structures (e.g., for version control).

Instead, CogLaborate adopts a frame representation. Frames were introduced by Marvin Minsky [Minsky, 1974] in a seminal paper on knowledge representation. Frames are structures that can represent objects, situations, and concepts. Frames are arranged in a parent-child hierarchical taxonomy, with child frames representing specializations of their parents [Karp, 1993]. A frame contains slots that define the properties of the object being represented by the frame. Slots can also represent relationships between two frames.

CogLaborate provides translation between the Lisp source code of models and a frame representation, in both directions. Descriptions of the frames for representing models are given below; their structure is shown in Figure 2.

- *The model frame* represents an ACT-R model. It consists of a code slot that holds all the code that is required by the model, including code for initialization of the model, chunk definitions for the model, and miscellaneous utility functions that may be required by the model. It also has a slot for productions.
- *Production frames* contain a conditions slot, which defines the tests that are required for the condition to fire, and an actions slot, which lists all the actions that will be executed if that production is fired.
- *Buffer test frames* capture the tests that are part of conditions in a production. Each buffer test frame represents one such test. A buffer test frame has a slot to represent individual clauses within the test.
- *Conditions frames* represent an individual clause consisting of a test field and a value field for comparison of a buffer slot and a value. The value field can also hold variables, as is common in ACT-R productions.
- *Buffer actions frames* hold actions that can modify, clear, or retrieve a chunk in a buffer.
- *Action frames* represent individual clauses for modifications to a buffer.
- *Computable Action frames* specify actions executed by the ACT-R architecture that have side-effects, such as printing information to the screen.

The AllegroServe Web Application server acts as a front end for interaction with CogLaborate. When a model is evaluated in CogLaborate, it is compiled by ACT-R, running on the server. CogLaborate code is plugged into the ACT-R compiler to allow access to the internal data structures generated as the model is parsed. This model representation is then converted into frames as described above. The frame-based representation thus exists side by side with the source model code (as well as with the running model).

## 4. Using CogLaborate

Briefly, cognitive modelers using CogLaborate for ACT-R development rely on a Lisp listener in a Web browser, where code can be evaluated; a structured representation for models in frames; and mechanisms for sharing and examining models at different levels of detail.

The user interacts with the CogLaborate sys-

tem through a Web interface. On logging in, users are put into the ACT-R package. Models are submitted through the Web interface in their source code representation, with code wrapped in a `with-user-meta-process` form. This macro creates a new meta-process for each user and allows models to be run without conflict with other users of the system, who may be running their own models at the same time. No other changes to model code are required for use in CogLaborate.

Development on CogLaborate up to the present has focused on basic functionality, which means that the Web interface does not provide as rich an environment as the graphical user interface to ACT-R. The workflow of using CogLaborate in its current state means building and testing models and architecture extensions locally before uploading the work to the server. Even though it is possible to build models completely from scratch in CogLaborate, a more efficient workflow for model development must await further work on the front end.

Let's consider a slightly more detailed scenario to illustrate the use of the system. A user creates a model and evaluates it in CogLaborate. This is done by entering a model into the Lisp listener displayed in the Web interface, as shown in Figure 3. The Lisp listener has two text boxes. The larger text area is used to enter complete models; the smaller text box to enter individual commands.

Once a model is entered into CogLaborate, it can be accessed (via its name) by any other user of the system, through a simple search. The model resulting from the search is displayed in its frame representation. The model can be navigated by active links corresponding to the slots of the current frame, whether at the level of models, productions, or lower in the frame hierarchy. To see the source code of the model, users can click the `Frame→Listener` link on the index page of the model. The result is shown in Figure 4.

## 5. A proof of concept

To evaluate the capabilities of CogLaborate we built a simple, medium-scale model. The point of this exercise is twofold. First, it shows that the system is capable of supporting a non-trivial cognitive modeling effort. Second, it demonstrates the level of maturity of the system. This section discusses the problem description, the approach we took to solving the problem,

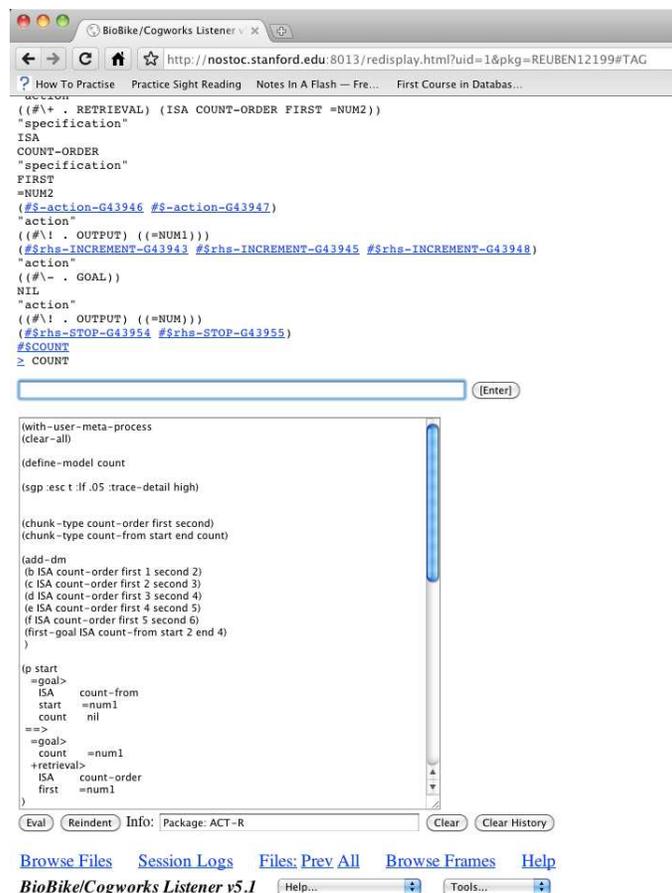


Figure 3: A user creates and evaluates a model

and what we learned through the exercise.

In a crossword puzzle, words or phrases are positioned in an interlocking grid, horizontally and vertically. The words are to be guessed by a set of clues that define the words or phrases. Our proof-of-concept problem is a crossword puzzle where the clues and the solutions are synonyms of each other.

This problem is appropriate for the following reasons: it demonstrates that the system is ready to solve practical problems; it shows that the system can be used to write and test an ACT-R module, with the environment acting as a sandbox; finally, it places considerable demands on the hardware of the computer, in terms of memory and CPU.

The crosswords are generated by a new *Crossword* module for ACT-R. This module relies on information from the WNLexical module [Emond, 2006], which

enables ACT-R to make use of the WordNet lexical database. WordNet is [Miller, 1995] “an online lexical database designed for use under program control. English nouns, verbs, adjectives, and adverbs are organized into sets of synonyms [synsets], each representing a lexicalized concept. Semantic relations link the synonym sets.”

Each clue is represented as a list that consists of the starting co-ordinates of the word, the direction (across or down), the clue string, a location to put in a solution, and the actual solution. These data structures are manipulated by the crossword module, which translates clues into chunks. It can also set words in specific locations, verify that the crossword solution under construction respects the constraints of the puzzle, and return results from queries about the parameters of a specific clue. The module maintains the current state of the crossword solution, with some entries filled in and others empty.

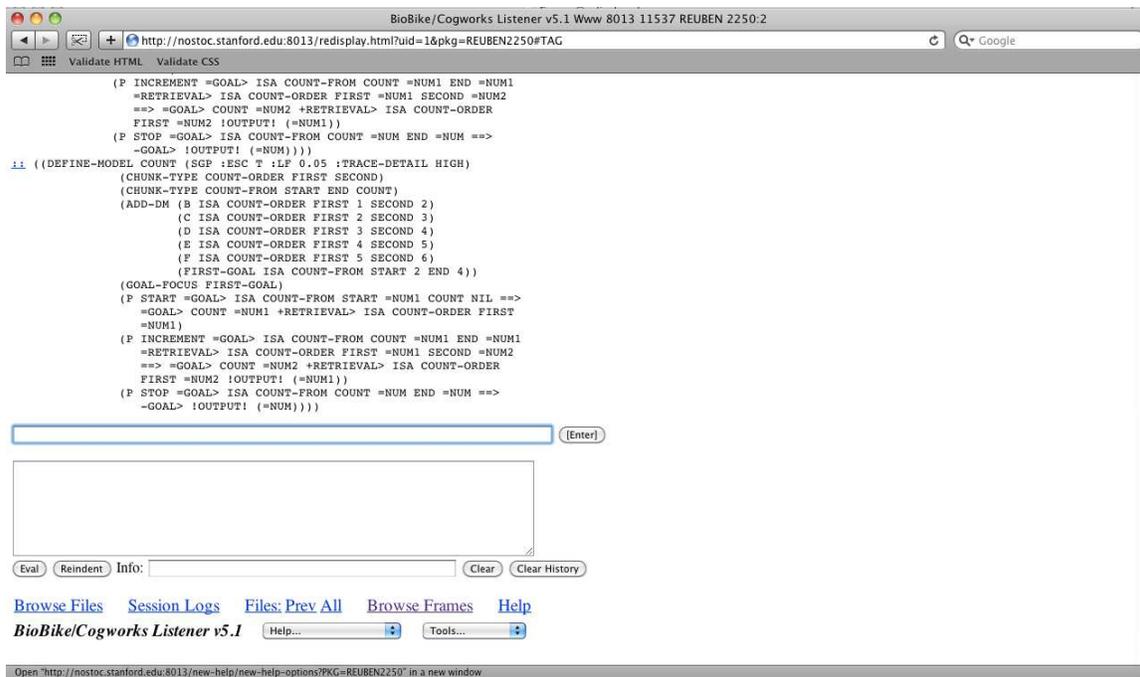


Figure 4: A user displays the source of a model

When the model is run it defines three chunk types, one for clues and two for maintenance of the state of the crossword problem as it is being solved. The basic problem-solving strategy the model follows is to check memory for clues that have not been added to the puzzle representation. If one is found, it is used to retrieve all the synsets of the clue word via the `wn-lexical` buffer. (A single word may have more than one synset.) For every synset found a chunk is created with the `imaginal` buffer. If the word is not found, this results in an error. For each synset chunk, its corresponding words are tested against the constraints of the puzzle by the crossword module, which also marks the clue as being solved. This process repeats until all the clues have been solved or have been marked as being unsolvable.

This is not intended to be a cognitively plausible model of crossword puzzle solving, but rather to exercise CogLaborate. The model consists of sixteen productions with a total of about four hundred and sixty lines of code, which can be fairly described as medium-sized. The source for the model and a sample execution trace, as well as the Crossword module, are publicly available but are not given here due to space limitations [Cornel, 2009].

During the development of the Crossword module, a

difficulty arose when an older version of the `WNLexical` was used; we were not aware that a newer version was available that contained a bug fix we needed. This caused us some wasted time. The conventional lesson learned is that developers should consider such possible sources of problems, but another possibility is that dissemination of modules (along with models and other software to support modeling) might be improved with a centralized resource for modeling such as CogLaborate.

## 6. Discussion

The concept of repositories for cognitive models is not new, and there has been continuing interest in establishing such shared resources.<sup>1</sup> Such resources can have obvious benefits: improved access to computational capabilities, a stable and growing body of explicitly expressed knowledge about a domain, and so forth. Our work on CogLaborate explores a new dimension of potential benefits for cognitive modeling research: collaboration.

On creating a frame-based abstraction for ACT-R

<sup>1</sup>A panel at the Biologically Inspired Cognitive Architectures symposium, at the 2009 AAAAI Fall Symposium Series, was devoted to this topic.

models it quickly became clear that this representation could be used to explore a number of other possibilities beyond our original conception of CogLaborate. As observed by Langley et al. [Langley et al., 2009], an important issue facing cognitive modeling is support for software reuse. This project promotes reuse of models in the sense that the representation allows for models to be represented, analyzed, and distributed in a more transparent fashion than in their current representation as Lisp code. Today, it is impossible to determine the similarity between two ACT-R models except through code inspection and ad hoc judgments. The frame-based representation introduced in this research makes more sophisticated analysis possible: comparison of the use of buffers across productions, for example. Such analyses remain for future work.

Another interesting research direction is to investigate software reuse as provided by object-oriented programming environments. That is, we can develop features such that models can inherit behavior from other more general models. This way we should be able to identify general patterns that emerge from human cognition. A third and obvious possibility is the investigation of user interfaces that allow cognitive scientists to create models without having to learn Lisp; the issue of cognitive modeling languages and ease of modeling is a continuing concern in the field [Ritter et al., 2006].

Some of the core features of CogLaborate are partially supported by other systems. For example, conventional systems for source control provide some of the same benefits as CogLaborate, as do model repositories such as the ACT-R Web site (<http://act-r.psy.cmu.edu/models/>), which even includes a few Web-based simulations. We believe that CogLaborate demonstrates new possibilities. The most interesting for us are the following:

- CogLaborate can be used as a collaborative sandbox for learning and exploration in modeling. Access to a shared environment in which models and even modeling processes can exist for long periods of time provides continuity and a persistent context for the exchange of ideas. We expect this to be most useful for remote collaborations.
- CogLaborate, with its frame representation of models, supports the development of new techniques for development, analysis, and comparison. Does my new model share structure with any existing models already in the environment? How different are two models for the same task, developed for different versions of the ACT-R ar-

chitecture?

We are actively building on these ideas.

## References

- [Cornel, 2009] Cornel, R. (2009). *Coglaborate - an environment for collaborative cognitive modeling*. Master's thesis, Department of Computer Science, North Carolina State University.
- [Emond, 2006] Emond, B. (2006). *WN-LEXICAL: An ACT-R module built from the WordNet lexical database*. In *Proceedings of the Seventh International Conference on Cognitive Modeling*.
- [Gluck et al., 2007] Gluck, K., Scheutz, M., Gunzelmann, G., Harris, J., and Kershner, J. (2007). *Combinatorics meets processing power: Large-scale computational resources for BRIMS*. In *Proceedings of the Sixteenth Conference on Behavior Representation in Modeling and Simulation*, pages 73–83. Citeseer.
- [Karp, 1993] Karp, P. D. (1993). *The design space of frame knowledge representation systems*. Technical Note 520, SRI International, Menlo Park (CA), USA.
- [Langley et al., 2009] Langley, P., Laird, J. E., and Rogers, S. (2009). *Cognitive architectures: Research issues and challenges*. *Cognitive Systems Research*, 10(2):141–160.
- [Licklider and Taylor, 1968] Licklider, J. and Taylor, R. (1968). *The computer as a communication device*. *Science and technology*, 76:21–31.
- [Massar et al., 2005] Massar, J. P., Travers, M., Elhai, J., and Shrager, J. (2005). *Biolingua: a programmable knowledge environment for biologists*. *Bioinformatics*, 21(2):199–207.
- [Miller, 1995] Miller, G. A. (1995). *Wordnet: A lexical database for english*. *Communications of the ACM*, 38(11):39–41.
- [Minsky, 1974] Minsky, M. (1974). *A framework for representing knowledge*. Technical report, Massachusetts Institute of Technology, Artificial Intelligence Laboratory, Cambridge, MA, USA.

[Ritter et al., 2006] Ritter, F. E., Haynes, S. R., Cohen, M., Howes, A., John, B., Best, B., Lebiere, C., Jones, R. M., Crossman, J., Lewis, R. L., St. Amant, R., McBride, S. P., Urbas, L., and Vera, A. (2006). High-level behavior representation languages revisited (group symposium). In Fum, D., del Missier, F., and Stocco, A., editors, *Proceedings of the Fifth International Conference on Cognitive Modeling (ICCM)*, pages 404–407, Trieste, Italy. Edizioni Goliardiche.

[Rodden, 1991] Rodden, T. (1991). A survey of CSCW systems. *Interacting with Computers*, 3(3):319–353.

[Travers et al., 2005] Travers, M., Massar, J. P., and Shrager, J. (2005). KnowOS: The (re)birth of the knowledge operating system. In *The International Lisp Conference*.

## Author Biographies

**REUBEN CORNEL**, M.S., is a recent graduate of the Knowledge Discovery Laboratory at North Carolina State University. His research interests are in intelligent systems and cognitive science.

**ROBERT ST. AMANT**, Ph.D., is an Associate Professor in the Department of Computer Science at North Carolina State University. His research explores models of interaction, drawing on concepts in artificial intelligence, human-computer interaction, and cognitive science.

**JEFF SHRAGER**, Ph.D, is the CTO of CollabRx, and a consulting associate professor in the Symbolic Systems program at Stanford University. As a computational psychologist of science, Dr. Shrager seeks to understand how science works, and to build human-computer networks that facilitate discovery.

# Cognitive Model Exploration and Optimization: A New Challenge for Computational Science

*L. Richard Moore Jr.*

Lockheed Martin Systems Management  
Air Force Research Laboratory  
Warfighter Readiness Research Division  
6030 South Kent Street  
Mesa, Arizona 85212-6061  
[Larry.Moore@mesa.afmc.af.mil](mailto:Larry.Moore@mesa.afmc.af.mil)

Keywords:

adaptive mesh, exploration, searching, parameter space, predictive analytics,  
volunteer computing, high performance computing

**ABSTRACT:** *Parameter space exploration is a common problem tackled on large-scale computational resources. The most common technique, a full combinatorial mesh, is robust but scales poorly to the computational demands of complex models with higher dimensional spaces such as those found in the cognitive and behavioral modeling community. To curtail the computational requirements, I have implemented two parallelized intelligent search and exploration algorithms, both of which are discussed and compared in this paper.*

## 1. Introduction

Research in cognitive science often involves the generation and analysis of computational cognitive models to explain various aspects of cognition. Typically the behavior of these models varies across a continuous parameter space composed of a number of theoretically motivated parameters, but most commonly we are left to our own devices to find the right balance of parsimony and fit within that space.

We are certainly not alone. The modeling community more generally is already well aware of the challenges associated with parameter optimization. Furthermore, there appears to be a growing appreciation of the parameter space itself—a qualitative understanding of the space can provide valuable insights regarding a model's behavior, optimal parameter ranges, the number of optima, and the distance(s) from canonical values. It is this deep understanding of the model's parameter space that allows us to find a balance between parsimony, optimization and generality (Gluck, Stanley, Moore, Reitter & Halbrügge, 2010). However, this is difficult to achieve on the computational scale of a workstation, so we have turned to high performance computing (HPC)

clusters and volunteer computing for large-scale computational resources.

The majority of applications on the Department of Defense HPC clusters focus on solving partial differential equations (Post, 2009). These tend to be lean, fast models with little noise. While we lack specific data regarding typical job sizes and durations, HPC maintenance is regularly scheduled at two-week intervals, so it seems reasonable to assume that most jobs fit within this window.

In contrast to HPC applications, volunteer computing projects tend to involve singularly specific, highly parallelizable tasks crunching vast quantities of data over time spans measured in months and years, such as SETI@home's analysis of interstellar radio signals and Folding@home's studies of protein folding. Both of these examples run on a common software framework called the Berkeley Open Infrastructure for Network Computing (BOINC), which enables volunteers to donate idle time from their computational resources to projects of their choice. The volunteer computing application developed by my colleagues is called MindModeling@Home, and it too runs on the BOINC infrastructure (Harris, Gluck, Mielke & Moore, 2009). Projects that work well with

BOINC tend to be long lasting and can tolerate latencies measured in days, which happen quite commonly when volunteer resources are interrupted or retasked.

Cognitive models fit somewhere between these two extremes. Our models are computationally expensive and produce stochastic results, quite unlike the partial differential equations typically solved on HPC clusters. And unlike most of the BOINC projects, we strive to analyze many different models with vastly differing performance characteristics within a calendar year. Our unique requirements present new methodological challenges for both HPC and volunteer resources. This paper describes some of the methodologies we have explored, the trade space among them, and my latest research efforts to apply HPC and volunteer resources to characterize and search parameter spaces.

## 2. Meshing

In its simplest form, “meshing” involves the construction of an n-dimensional grid by iterating through each parameter range by a fixed interval, and capturing the combinatorics to be used as the basis of model runs. The resulting simple orthogonal grid seems to suffice for most of our cognitive models.

Once the mesh is defined, portions can be distributed amongst computational nodes and executed completely independently. Meshing has been widely used for many years (Chen & Taylor, 1998) and it lends itself well to both HPC and volunteer resources. The complete independence among computational nodes affords the ultimate in “embarrassingly parallel”—a term commonly used to describe computational tasks that can be efficiently executed with little or no serial operations. Parallelizability is the key to realizing the full potential of large-scale computational resources.

Full combinatorial meshes have other advantages, as well. For example, there is little software overhead in computing these meshes (at least for our relatively simple requirements) and the corresponding job files for the HPC schedulers. For volunteer resources, my colleagues have developed a web interface specifically for this purpose with plans to make it available as a community resource (Harris et al, 2009).

Combinatorial meshes are also flexible. No assumptions are made about the structure or even the continuity of the parameter space. The data can be stored in any format convenient for the modeler to analyze. Analysis is straightforward, and the results can be visualized or mined indefinitely, within the limits of precision defined by the original mesh.

Another point to consider about full combinatorial meshes is that counting the results files quickly reveals the success of the jobs; one result should be present for every parameter combination. While we might shrug off a failure on our desktop as a 1 in a million fluke, when running models millions of times this seemingly innocuous failure rate becomes noticeable, and quick methods to detect and recover are desirable—in this case the modeler can simply rerun the specific mesh nodes that failed to produce results files.

How do full combinatorial meshes fare with cognitive models? In one research effort, we have developed a model that performs a Digit Symbol Substitution Task (DSST) (Moore, Gunzelmann & Gluck, 2008). This is a simple task where the model is presented with 9 digit / symbol pairs, and when prompted with a symbol the model responds with the appropriate digit. This fairly typical cognitive model has 7 relevant quantitative parameters and due to stochasticity must be resampled at least 10 times to establish a reliable measure of central tendency. With an average run time of 2 minutes, a mesh with 10 increments per variable would require 271 days to compute if run continuously on 512 cores. A computational challenge of this magnitude would overwhelm any computational resource for quite some time, and as mentioned previously there is some desire to analyze more than one model per calendar year.

There are primarily two issues that drive the computational demands of the DSST. First, the 7 parameters exhibit the “curse of dimensionality”—a phrase used to describe the exponential requirements of additional parameters in a space (Bellman, 1961). After examining the parameter space and understanding the interrelationships, dimensionality can often be reduced, but not until after an initial analysis is completed.

The second primary issue contributing to the computational requirements is the 2-minute run time required for each node in the parameter space. The DSST is a learning model—its behavior changes across sessions as it gains knowledge and experience. Therefore, to properly compare learning characteristics with human subjects, the entire learning curve must be constructed at each parameter combination across all sessions. Considering that, in this case, the model is performing the task across 32 sessions (96 simulative minutes), 2 minute run times seem quite reasonable.

Recognizing that large-scale computational resources can only take us so far, we have turned our attention to intelligent exploration and search strategies that run on both HPC and volunteer resources. Our interests are specifically focused on approaches that allow searching a

parameter space for optimal values, as well as characterizing the overall space in general.

### 3. Adaptive Mesh Refinement

Adaptive mesh refinement (AMR) is an intelligent search strategy that dynamically divides the overall search space into subcubes of varying size, each of which is capable of making predictions about measures in its local area of space to a predefined degree of accuracy (Berger & Olinger, 1984).

My parallelized implementation of AMR is called Quick, and it consists of about 11,000 lines of C++ code. The code has been ported to several HPC clusters, as well as our BOINC-based MindModeling volunteer computing system.

Implementing AMR—or any intelligent algorithm, for that matter—on large-scale computational resources requires a serious engineering investment. The software needs to be robust enough to recover from faults throughout the system—including models under evaluation-- and it needs to be reliable enough to run for hundreds or thousands of hours without memory leaks, crashing, etc.

To initiate an AMR using Quick, the modeler begins by defining the independent variables, their ranges, and the increment for each. The increment is identical to the increment used when constructing a full combinatorial mesh—although hypercubes produced by an AMR may span large portions of space, their boundaries are always constrained to the implicit grid lines defined by the increment. The hypercubes never overlap, and the sum of their volumes equals that of the parameter space overall.

The user also specifies the dependent measures that the model will produce, as well as a threshold value for each. The threshold is an important consideration, because ultimately it will constrain how accurate the results will be.

Once configured, the procedure to execute Quick varies between HPC and MindModeling. Running software on HPC resources is accomplished through “job” submissions. A job is defined through a simple shell script that describes the requested computational resources and the software to run. Jobs are submitted to a dedicated scheduler that executes the software when the requested resources become available. Quick begins with a single job that requests a single computer. As the AMR progresses, Quick will automatically schedule more jobs to run in parallel as aggressively as possible.

On MindModeling things behave quite differently. In this case, Quick is automatically executed on the servers at periodic intervals to determine which points in the parameter space need to be computed for the AMR. As volunteers request work, they are provided with these points to compute, and as they return results and the AMR progresses new points will be generated by Quick. Thus, parallelization is achieved at the level of sample acquisition.

Regardless of the computational context, the AMR methodology is the same. Quick begins by treating the entire parameter space as a single large single hypercube. The process begins by executing the model with parameter values at each of the corner points. AMR assumes that measurements are accurate, so we typically resample the model a fixed number of times and collapse across the dependent measures to remove stochasticity. In any  $n$  dimensional space, there will be  $2^n$  corners to sample.

In addition to the corner points, the center of the cube is measured as well. (As with all nodes considered in the space, the center is constrained to the specified grid, so it may not reflect the precise mathematical center.) In addition to measuring the center, Quick will also make a mathematical prediction of the center, assuming that the model’s behavior changes smoothly across the parameter space, yet accounting for twisting that can occur. If the difference between the measured value and the predicted value is within the specified threshold for each dependent measure, then the hypercube is considered smooth and predictable, and the process is complete. However, if any of the dependent measures exceed the threshold, the hypercube is divided into  $2^n$  subcubes about the center point, and each subcube is analyzed using the same process just described.

When hypercubes split into subcubes, each subcube can be treated as a parameter space in its own right, albeit smaller than the true overall space. This is the key to parallelizing AMR on HPC resources, as the analysis of each subcube can be scheduled as an independent HPC job. Aside from the shape of the parameter space, these new jobs are identical to the original that started the analysis.

AMR can result in substantial computational savings, yet the quantitative quality of the results typically remains high (Best et al, 2009). The quality of the results is consistent across the space, too, so unmeasured points can be interpolated and the resulting grid can be mined just as a full combinatorial mesh. Further, because the space is mathematically defined, off-grid interpolation can also be calculated if desired. There is also something to be said

for the reduction in data that needs to be transferred to the workstation for analysis.

Nevertheless, AMR does have its drawbacks. First, the computational savings with AMR are unpredictable. This is also consistent with the Best et al (2009) work, which showed that AMR efficiency was heavily influenced by threshold and implementation factors that can be difficult to predict a-priori. Furthermore, the structure of the space, (which in turn depends on the parameters and their relationships) and the number of dependent measures can also heavily influence AMR efficiency. In my experience with our models, it is not uncommon for an AMR to compute nearly all the nodes in the space, resulting in little savings.

Recall that AMR must evaluate the corners and center of each hypercube before it can move forward. In a volunteer environment such as MindModeling, this can be problematic because of the large latencies. At any moment, a volunteer might turn off their computer, or use it for something else, and processing is stalled until an explicit timeout is reached, which is usually at least a day. So while volunteer networks provide huge computational power, they are a poor match for the methodological requirements of AMR.

AMR on HPC suffers for different reasons, but with similar effects. In this case, parallelization is not usually the problem, but each parallel analysis requires a new HPC job to be scheduled. HPC schedulers vary in reliability and performance—which is in itself problematic for AMR—but they all share a first-in-first-out paradigm, so new jobs must wait for resources to be made available from jobs scheduled prior. A simple 3-dimensional parameter space with 8 divisions per parameter could potentially result in millions of job submissions, each with its own wait time in the job queue.

To test how many submissions are actually made, and their impact on the overall wall clock time, I ran six adaptive meshes on the Jaws high performance computing cluster in Maui using a model of the Psychomotor Vigilance Task (PVT). The PVT is a simple model that simulates a button press when a visual stimulus is presented at random time intervals (Gunzelmann, Gross, Gluck & Dinges, 2009). Two variants of this model were tested, and each was run using three different values for the threshold that controls the likelihood of searching deeper into the parameter space. All six meshes explored the same three-parameter space.

The mean number of HPC jobs submitted was 577. The average run time for each job was 2 minutes, and the average wait time in the scheduler queue was 5.9 minutes. One must be cautious when interpreting these results due

to the small sample size and large variation in HPC usage, but in this case the mean wait time was nearly 3x longer than the mean run time per job.

Although AMR is more computationally efficient than a full combinatorial mesh on large-scale resources, it can be slower in terms of wall clock time. If you recall, our original motivation for combining intelligent search and exploration with large-scale computational resources was to improve analytical capacity with cognitive models, yet AMR does not consistently deliver.

Despite its shortcomings, AMR has clearly demonstrated that combining intelligent search with HPC and volunteer resources is indeed possible. My most recent research re-imagines optimized search specifically for the context of cognitive models on parallel computational resources.

#### 4. Regression Trees

Recognizing that parallelization is the key to fully leveraging HPC and volunteer resources, I have developed a flexible stochastic search methodology that allows massive parallelization with virtually no interdependencies. Furthermore, recognizing the necessity for qualitatively understanding the parameter space, I have also developed accompanying visualization software that operates in real time as the space is constructed. The visualization software is called Hurricane, while the intelligent search software is called Cell.

Hurricane and Cell are written in Objective C, and at 5300 lines combined they are about half the size of Quick, testifying to their relative simplicity. They were developed on Mac OS X, and Cell specifically has been ported to Linux to support HPC and MindModeling integration. At this time Cell has been successfully ported and tested on four different HPC clusters, with MindModeling integration underway.

As was the case with Quick, Cell and Hurricane begin with a user-specified configuration including independent variables, their ranges and increment, and the dependent measures. In contrast to the AMR configuration for Quick, no threshold is required.

Like all software run on the HPC, Cell is executed through a job submission. However, because Cell is immediately parallelizable any number of job submissions can be made during startup. Typically I limit myself to 128 jobs, mostly to avoid complaints from other HPC users.

On MindModeling, a single instance of Cell runs on the server for the duration of a model run. This “listener” process analyzes incoming data, and upon request,

generates lists of points that are distributed to volunteer resources as they request work. Like Quick, Cell achieves parallelization on MindModeling by distributing model runs to volunteer resources.

Cell can analyze the parameter space in either of two ways: exploration or searching. Both approaches divide the space into a set of hypercubes that are geometrically analogous to AMR. However, rather than sampling just corners and the center, Cell samples stochastically within the hypercube space and calculates the best fitting hyperplane for each dependent measure—an analytical approach sometimes referred to as a regression tree (Alexander & Grimshaw, 1996).

Regardless of whether Cell is searching or exploring, it tries to maintain a consistent sample density among the hypercubes, regardless of size. This means that areas of the space with higher sampling will have greater numbers of hypercube divisions. The minimum number of samples targeted for each hypercube is based on the work of Knofczynski and Mundfrom (2008), which suggests a linear relationship between the number of samples required to make a good regression prediction and the dimensionality of the space. It is not until a hypercube contains 2x this amount does it split along its longest dimension. Within the confines of a single hypercube sampling is uniform, so the split should roughly divide the samples equally between both subcubes.

The key distinction between Cell’s two analytical approaches lies in the way they construct their sampling distribution. The exploration approach performs a characterization of the space—in this case the sampling distribution is positively correlated with the residual variation in each hypercube. Unexplained variation is presumably the result of noise or a poor regression fit, and in either case it is prudent to sample more, and potentially to subdivide more, to resolve the ambiguity. In this mode, the exploration process has no definitive end and runs as long as the modeler desires.

In truth, I rarely use exploration mode because our work typically involves parameter optimization as well as characterization, and search mode provides both. In this case, the user supplies additional configuration information consisting of dependent measure “target goals” to search for. In terms of cognitive modeling, this typically takes the form of human data. When supplied, the sampling distribution is skewed towards hypercubes with the lowest deviation from the human data (or whatever target goals are supplied), and so the space winds up being more intricately constructed in those areas. The search is considered complete when the best fitting hypercube cannot divide any more based on the constraining grid.

With data in hand (or even while it is being obtained in the case of running on local resources), Hurricane can be used to visualize the results, as is shown in Figure 1. Hurricane conducts the same analysis that Cell does, and produces the regression tree in the form of a 3D graph. Any two independent measures can be selected for the x and z-axis, and any dependent measure can be selected for the y-axis (vertical). The remaining independent measures can be manipulated in real time via sliders, which provides a convenient mechanism to grasp an otherwise esoteric hyperdimensional space. Hurricane can also scan the space for optimal parameter values or make predictions, which can then be imported into more generalized analytical tools like R or SPSS.

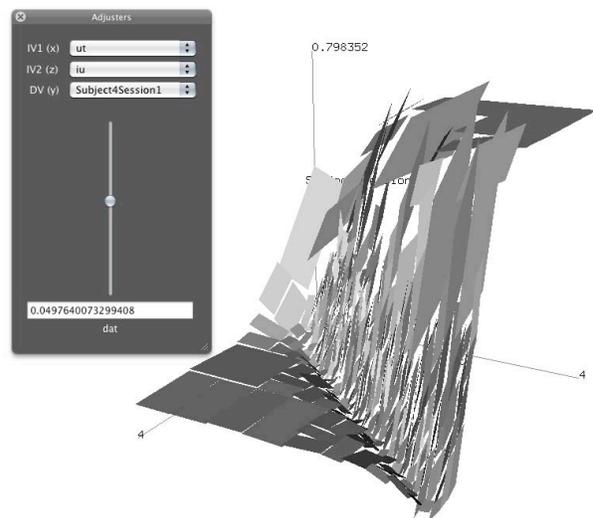


Figure 1. Hurricane visualization of a PVT parameter space. The vertical axis represents RMSD between human measures and the model, while the other two axes represent independent variables. A third independent variable can be manipulated with the slider. Best fitting parameter values are located within the trench area, which received more samples and therefore is more finely subdivided.

Searches conducted with Cell provide large computational advantages over AMR and full combinatorial meshes. This is primarily because vast sections of the space—those areas that are distant from target areas of interest—are only lightly sampled and mostly ignored once deemed suboptimal. As an example, I ran the PVT model through a full combinatorial mesh, an AMR with Quick, and a regression tree analysis with Cell. Identical grid slicing was used for all three, and they were all run on the same Mana HPC cluster in Maui.

Figure 2 shows the number of model runs required to complete an analysis of the parameter space for each methodology. In this example, the AMR—although it was configured with a liberal 5% threshold—wound up sampling most of the space anyway, while the Cell required two orders of magnitude fewer model runs.

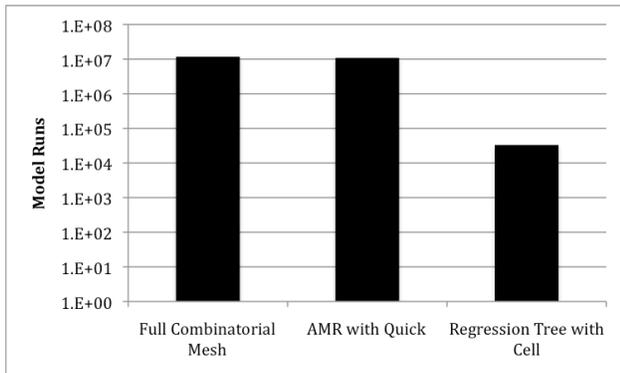


Figure 2. Comparison of computational requirements for each of the three methodologies discussed.

The amount of time required to complete the analyses is shown in Figure 3. Note that the AMR took 4.2 times longer than the full combinatorial mesh, which is almost exactly what would be expected if queue wait times were 3x the run time as discussed above. Because Cell parallelizes immediately upon startup and does not auto-schedule new jobs like Quick, most of the scheduling queue delays are avoided. For more complex searches that fail to complete within the scheduled amount time, I can simply reschedule more jobs, and each Cell instance will read the samples acquired previously from disk, and pick up where the older Cell instances left off.

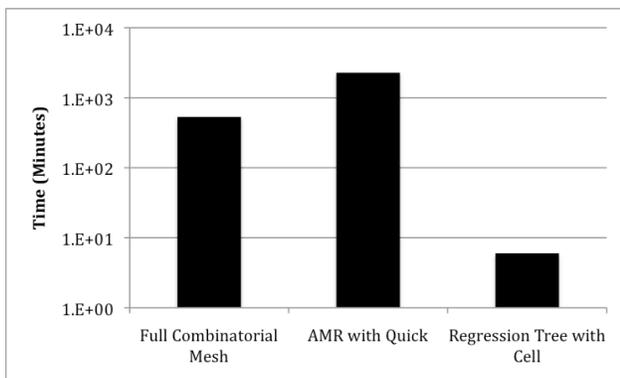


Figure 3. Comparison of wall clock time required to analyze the PVT parameter space using the three methodologies.

Speed and efficiency are important, but they are only useful if the resulting analysis is viable. Figure 4 compares the optimized parameter predictions from each of the three methodologies. To produce this table, I reran

the model at the predicted optimal parameter values and computed an RMSD against the human data for each methodology. The model was run 100x to reduce noise—the same amount used during the AMR and full combinatorial runs. As expected, the full combinatorial mesh produced the best results. It was surprising to see that the regression tree methodology edged out AMR, but this is likely caused by variation in the model’s performance.

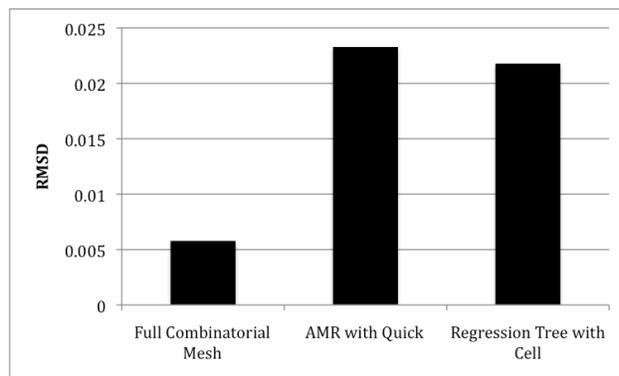


Figure 4. RMSD between best fitting parameter predictions and human data.

Many of the issues challenging parallelized AMR disappear in the context of regression tree exploration and searching. This is because Cell does not base decisions upon the outcome of specific, accurate, grid-constrained samples. Rather, the decisions are based on statistical analysis of a set of distributed, stochastic samples. As a result, any number of Cell instances can be started at once and run in parallel, each making its own decision about how to divide the space and where to sample.

Although the integration remains a work in progress, I expect that Cell will work well with volunteer resources. In this case AMR was stalled waiting for specific points to complete, but Cell, with its semi-random sampling strategy, can always generate work for volunteers. However, we will need to be careful to limit the number of outstanding points being computed at any given time. The end result of too many outstanding samples could be hundreds or thousands more samples in a hypercube than is really necessary to make a search decision. The extra data would still be useful for visualization purposes, but it would reduce the efficiency of searching.

In my ongoing efforts to combine intelligent search and exploration with large-scale computational resources, Hurricane and Cell represent best results to date. Nevertheless, they present their own new challenges. For example, the confidence of predictions based on discontinuous regression planes is inconsistent, and highly dependent upon the distance from the center of the hypercube. Predictions across the boundary of two

discontinuous hyperplanes can be disturbingly disparate compared to neighboring predictions. This not only makes visualization less appealing, but data mining outside of specified search goals can be problematic.

From an implementation perspective, Cell is more computationally intensive than AMR and full combinatorial meshes. Every incoming sample requires a search to determine its encompassing hypercube, and the introduction of new data into the hypercube will require the calculation of new regressions. To maintain pace with the incoming data stream, results must be stored in RAM rather than disk-based storage, which limits scalability. The number of samples that can be maintained in a fixed amount of RAM depends upon the amount of memory required to store a sample, which includes values for the independent measures, dependent measures, and search targets specified.

Even with in-memory data management, however, the number of regressions required can still be computationally challenging. For example, the DSST model mentioned earlier captures 9 measures across 32 sessions, amounting to 218 total independent measures, each maintaining its own regression tree. Hurricane requires about 5 hours to read in the data from this model and reconstruct the regression trees for visualization, which seems excessive, to say the least.

Despite these limitations, the regression trees seem to be another step in the right direction. Using Cell, our cognitive models scale well on HPC resources from both computational and wall clock time perspectives. Some of our faster cognitive models, in fact, can now be analyzed in a few hours on local resources, which avails large-scale computational resources for even more complex models. Additionally, Hurricane's multidimensional visualization capability has become an indispensable part of my normal workflow.

#### 4. Discussion

In a broad sense, the engineering problem being addressed is one of computational performance and efficiency. Large-scale computational resources take us part of the way, and the remaining effort is incumbent upon us, as the resource users.

In the world of software engineering, there is a basic rule to optimization: focus on the innermost loop. In the context of this discussion, we have a parameter exploration / search methodology exercising a cognitive model, and it is the model itself that constitutes the bulk of processing in the innermost loop.

The model and its implementation are the embodiment of a theory, however, and this can severely constrain optimization options. This is certainly the case for my colleagues and I, where our models are based on a publicly available cognitive architecture (ACT-R; Anderson, 2007) that is shared among a relatively large scientific community. In our case, we routinely share models to combine and test different cognitive moderators, and it is important to maintain a consistent architecture across the community.

Therefore, we optimize our inner loop not by changing code, but by reducing the number of model runs as much as possible. AMR does this well and is used successfully in some contexts, but it appears, however, that the full utility of AMR does not necessarily transfer across domains and contexts. As cognitive and behavioral modelers begin to leverage large-scale computational resources, we must also develop suitable parallel search and exploration algorithms for our models.

This paper described our recent efforts using regression tree predictions to drive sampling distributions, and ultimately hypercube division. Like AMR, the technique reduces computational demands through a reduction in model runs, but the nature of the approach seems to be more agreeable to parallelization.

Regression trees, however, are not the only option. The dynamics of Cell are driven by two governing principles: 1) Sample more in areas of interest and 2) subdivide more in areas of higher density. The regression trees are used to determine the areas of interest, but other predictive analytical techniques can be substituted without compromising the fundamental approach. Multivariate adaptive regression splines (MARS) are one interesting possibility (Friedman, 1991).

However, models like the DSST have demonstrated that the computational demands of the analytical technique are becoming a serious consideration. While I predict that MARS will be more efficient than regression trees in terms of reducing the required number of model runs, I also expect that the analytical processing requirements will be significantly more demanding. It seems a trade space is becoming apparent between the computational demands of the model versus the computational demands of the search / exploration algorithm. For us this is not necessarily a bad trade space, because it is much less problematic to optimize a methodology as opposed to a theory, and there remain many opportunities to do so.

#### 5. Acknowledgements

The views expressed in this paper are those of the authors and do not reflect the official policy or position of the Department of Defense, the U.S. Government, or Lockheed Martin Corporation. This research was sponsored by grants 07HE01COR and 10RH04COR from the Air Force Office of Scientific Research.

I would like to thank Kevin Gluck and Glenn Gunzelmann for reviews of earlier drafts of this paper, as well as the Performance and Learning Models team and Adaptive Cognitive Systems for their influences and indulgence in supporting this work.

## 6. References

- Alexander, W. P., & Grimshaw, S. D. (1996). Treed Regression. *Journal of Computational and Graphical Statistics*, 5, 156-175.
- Anderson, J. R. (2007). *How can the human mind occur in the physical universe?* Oxford University Press, Oxford, UK.
- Bellman, R. E. (1961). *Adaptive Control Processes*. Princeton: Princeton University Press.
- Berger, M., & Olinger, J. (1984). Adaptive mesh refinement for hyperbolic partial differential equations. *Journal of Computational Physics*, 53, 484-512.
- Best, B. J., Gerhart, N., Furjanic, C., Fincham, J., Gluck, K. A., Gunzelmann, G., & Krusmark, M. (2009). Adaptive mesh refinement for efficient exploration of cognitive architectures and cognitive models. In *Proceedings of the Ninth International Conference on Cognitive Modeling*, Manchester, UK.
- Chen J., Taylor V. (1998). Mesh Partitioning for Distributed Systems. In *Seventh IEEE International Symposium on High Performance Distributed Computing*, 292-300.
- Friedman, J. (1991). Multivariate adaptive regression splines. *The Annals of Statistics*, 19, 1-141.
- Gluck, K. A., Stanley, C. T., Moore, L. R., Reitter, D., Halbrügge, M. (2010). Exploration for Understanding in Model Comparisons. Under review, *Journal of Artificial General Intelligence*.
- Gunzelmann, G., Gross, J. B., Gluck, K. A., & Dinges, D. F. (2009). Sleep deprivation and sustained attention performance: Integrating mathematical and cognitive modeling. *Cognitive Science*, 33(5), 880-910.
- Harris, J., Gluck, K. A., Mielke, T., & Moore, L. R. (2009). MindModeling@Home ... and Anywhere Else You Have Idle Processors [Abstract]. In A. Howes, D. Peebles, & R. Cooper (Eds.) *Proceedings of the Ninth International Conference on Cognitive Modeling*. Manchester, United Kingdom: University of Manchester.
- Knofczynski, G. T., & Mundfrom, D. (2008). Sample sizes when using multiple linear regression for prediction. *Educational and Psychological Measurement*, 68, 431-442.
- Moore, L. R., Gunzelmann, G., & Gluck, K. A. (2008). Evaluating mechanisms of fatigue using a digit symbol substitution task [Abstract]. In N. Taatgen, H. van Rijn, L. Schomaker, & J. Nerbonne (Eds.), *Proceedings of the Thirty-First Annual Meeting of the Cognitive Science Society*, Austin, TX: Cognitive Science Society.
- Post, D. (2009). The Promise and Challenges for Next Generation of Computers [PowerPoint slides]. Retrieved from <http://www.cc.gatech.edu/~bader/AFRL-GT-Workshop2009/AFRL-GT-Post.pdf>

## Author Biography

**L RICHARD MOORE JR** is a Research Engineer with Lockheed Martin Systems Management at the Air Force Research Laboratory, Warfighter Readiness Research Division in Mesa AZ. He completed his B.S.E. in Electrical Engineering in 1992, with an M.S. in Applied Psychology in 2008, both from Arizona State University.

## Modeling Behavioral Activities Related to IED Perpetration

Lora G. Weiss\*  
Elizabeth Whitaker  
Erica Briscoe  
Ethan Trehwhitt

Georgia Tech Research Institute  
250 14<sup>th</sup> Street  
Atlanta, GA 30332-0822  
\*404-407-7611  
\*Lora.weiss@gtri.gatech.edu

### Keywords:

Recruitment Modeling, Behavior Modeling, Counter-IED

**ABSTRACT:** *This paper presents a computational approach to modeling the behavioral aspects of IED perpetration that enables the exploration of those behaviors by an analyst or planner. The modeling framework presented supports the identification of potential interdiction points in the events leading to an IED detonation with a focus on insurgent recruitment and on the motivation to construct, emplace, and detonate IEDs. In many cases, individuals become terrorists or supporters of terrorism through a slow and gradual process wherein established terrorists use targeted approaches to convert individuals into terrorists through phases. Because of this phased approach, a strategic means of quelling terrorism involves understanding the process and exploiting insights to disrupt the IED process at an early stage. Knowledge engineering is used to extract and capture domain knowledge which is then represented in a system dynamics model to support the exploration and identification of behaviors associated with adversarial activities. Interchangeable submodels are used to capture subtleties or differing opinions and to allow for the analysis of expected results of alternative decisions or courses of action.*

### 1. Introduction

Multiple modeling paradigms can be used to produce models that aid in the understanding of adversarial behavior. Such models are valuable in that they provide a means to analyze and experiment with the impact of potential influences on population behavior (Zacharias, MacMillan, & Hemel, 2008). As subject matter experts are often used to provide an interpretation of social behaviors and applicable psychological theories (e.g., Crenshaw, 2000), a modeler can choose the appropriate modeling approaches to represent a given interpretation. By including behavioral aspects of adversarial activities in computational models, a framework has been developed that supports identifying potentially effective intervention points that may disrupt individuals' behaviors. This paper focuses on modeling terrorist recruitment and their motivations to construct, emplace, and detonate Improvised Explosive Devices (IEDs), where subject matter experts from the United States and the United Kingdom have collaborated to understand these motivations and behaviors.

The approach couples computational and social science research to develop an improved capability to identify and explore the space of likely activities and behaviors

of potential IED developers before they have successfully deployed IEDs. Content expertise from researchers within the UK is combined with computer-based analysis technologies for the prediction of individual or group-related activities. UK domain knowledge is provided by investigators who have been involved in UK event analysis and who are currently researching methods to explain terrorism, bombings, and other IED-related activities. Content was also obtained from numerous open-source publications to prevent too heavy of a dependence on subject matter experts (SMEs). Knowledge-engineering techniques are being exploited to extract and capture this domain knowledge. This information is linked with modeling approaches to provide a framework to support the identification and exploration of behaviors of individuals or groups of individuals involved in IED-related activities, with a focus on recruitment and the motivation to construct, emplace, and detonate IEDs.

In many cases, individuals become terrorists or supporters of terrorism through a slow and gradual process (Horgan, 2007). Established terrorists target individuals, usually young men, and try to convert them in phases into terrorists or supporters of terrorism. A key to interrupting terrorism is to understand the process and disrupt it in its early stages. The modeling

framework of this paper uses a set of representations that is appropriate for modeling this gradual process.

Specific modeling methodologies utilized include:

- Mind maps for preliminary knowledge engineering
- System dynamics models (Sterman, 2000) using stocks and flows (items, materials, people, etc.) to represent the overall system behavior of the IED process
- Influence diagrams to show the causal relationships between different aspects of culture and society that affect the IED process

The resulting modeling framework can be used for analysis of recruitment deterrents and potential intervention points within the IED process.

The approach to creating a modeling framework for exploring counter-IED (cIED) efficacy revolves around addressing several major scientific issues at the intersection of behavioral sciences, information science, computer science, and systems engineering, including:

- Identification of the domain knowledge and issues that apply to human behaviors related to IEDs as well as identification of relevant features of individuals to be used as inputs to influence diagrams.
- Identification of relevant features of groups and social interactions to be used as inputs to influence diagrams and to system dynamics models.
- Development of effective, interactive methods of analysis for domain experts to inject feedback into the system.

Figure 1 presents a summary of the approach, where the material in this paper emphasizes the content in the blue boxes. Specifically, information is acquired from multiple sources, including open literature, SMEs, doctrine, and reported scenarios. This information is captured via knowledge engineering methods and incorporated into various model types, including influence models and system dynamics models.

The knowledge capture and transformation works as follows. Information from SMEs, doctrine, documented scenarios / events, and open literature is represented in the structured construct of mind maps by researchers. This information is tagged based on the content it provides (e.g., object, relation, etc.). The tagged information is translated into structured data representations for inclusion into either influence diagrams or system dynamics models. For example, if a mind map identifies a category of people called Active Terrorists, then this is translated as a stock in a system dynamics model. Similarly, the transition of a

member of the Grey Population to an Active Terrorist is a flow and represented by an equation capturing the transition as a function of time. Finally, mind map concepts such as opinions of the government influencing the likelihood of involvement in terrorism become represented as directional weights in an influence diagram. The conversion of these concepts to data representations enables the disparate model constructs to be transformed into an analysis tool that incorporates time dependencies of the model components in support of dynamic assessments.

These model components are used to create a modeling environment with specific model instantiations, which are then subjected to evaluation by developers and SMEs. These models can be adapted and updated as new uses and information are obtained. A report by Weiss, et al. (Weiss, et al., 2009) describes the modeling cycle, complete with analyses that can be performed using such a modeling construct. This paper describes the front-end information associated with instantiating the modeling aspects in support of modeling recruitment associated with IED perpetration.

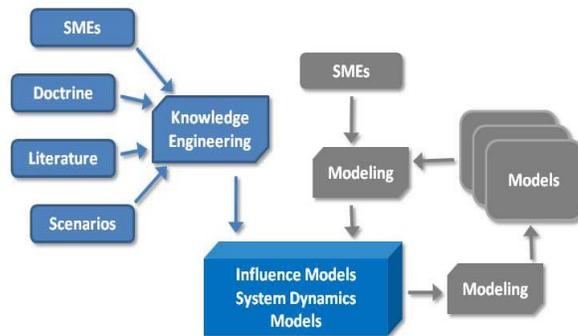


Figure 1. Approach to model construction. The blue boxes are the emphasis of this paper.

## 2. Information Gleaned from SMEs

Several insightful pieces of information were obtained from SMEs that is not evident in the resulting models, a few of which are discussed.

- Effects of Monitoring Groups. In some regimes where terrorists are aware of being watched, they try to operate in a manner to fool their pursuers, so that interactions become more game-like, with one side trying to outsmart the other side. For IED behaviors, the adversaries have less of a game-like attitude, and they put less effort into influencing the monitoring. Instead, they concentrate more on executing their tasks.
- Common End-State vs. Individual Motivations. Motivations within IED ‘teams’ are varied.

Participants are not necessarily focused on the end-state. Rather than having common motivations to achieve common goals and attain common results, individual motivations and goals are manipulated to accommodate an individual's end goals, e.g., one person may be motivated by money while another person is politically motivated, and yet another person is affected by peer influences.

- Experts are conflicted as to whether religion is actually a motivator or just used as a 'clean' explanation.

### 3. Knowledge Engineering Using Mind maps

To create useful models, diligence must be paid to the capture of knowledge from SMEs, literature, and other relevant sources (Burgoon & Varadan, 2006). For SME information capture, a knowledge elicitation document was developed, with details presented in (Weiss, et al., 2009). The document is a structured questionnaire used in interviewing subject matter experts to gather specific information including motivations, purposes, goals, beliefs, perpetrators, supporters, the environment, etc. Figure 2 shows part of this questionnaire's content.

Motivations, Purposes, Goals, Beliefs.....

Perpetrators.....

Supporters.....

Environment.....

Funding.....

Materials.....

Recruitment of participants.....

Resource acquisition.....

Planning.....

Assembly.....

Implantation.....

Location or placement.....

Detonation.....

Characteristics of the IED.....

Triggering Devices.....

Explosive.....

Countermeasures.....

Intelligence to Detect, Predict, Defeat, Assess.....

New approaches and unsolved problems.....

IED Life Cycle.....

Examples of success in preventing an IED attack.....

What are the IED scenarios?.....

Figure 2. Content of Knowledge Engineering Instrument

Mind mapping is a semi-structured technique for initial representation and organization of knowledge. Figure 3 depicts a portion of one mind map showing how related concepts are interconnected via common elements. Mind maps provide a visualization of concept relationships by showing hierarchical connections between textual concepts. For this research, in addition to obtaining information from numerous literature sources, seven SMEs from the US and UK were

interviewed to create a collection of mind maps such as the one in Figure 3.

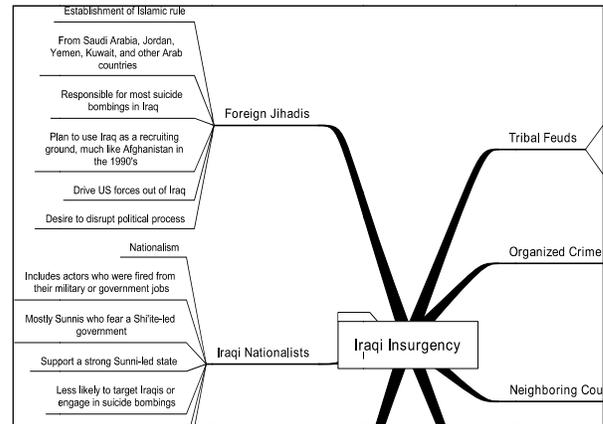


Figure 3. Mind map for preliminary knowledge structuring (portion)

Once domain knowledge has been formally structured, it can be constructed into various representations to support multiple aspects of modeling.

### 4. System Dynamics Models

A system dynamics model is a type of executable model used to represent and understand the dynamic behavior of a complex system over time (Sterman, 2000). This modeling approach uses *stocks* and *flows* to represent system elements and their relationships with each other. Stocks represent an inventory of an accumulated entity (e.g., IEDs, people), while flows represent how entities move between stocks.

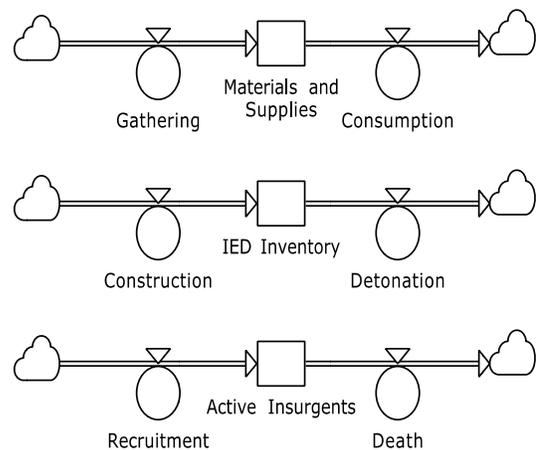


Figure 4 presents a simplified schematic of the core IED perpetration model that has been developed. Stocks are indicated using rectangular boxes. Flows are indicated with double-lined arrows. Clouds, which may take the place of a stock, indicate the world outside the scope of the model where stocks may originate or end.

The core model developed in this research is used as a foundation from which submodels or model expansions are incorporated into the framework, and it encompasses many aspects of the larger IED process. The first graphical line in Figure 4 depicts the process associated with gathering and consuming materials and supplies to develop and emplace IEDs. The second graphical line depicts the process of an IED moving from being constructed through inventory, to being emplaced and potentially detonated. The last graphical line is of particular interest for modeling recruitment in that it reflects many aspects of human behavior associated with IED perpetration.

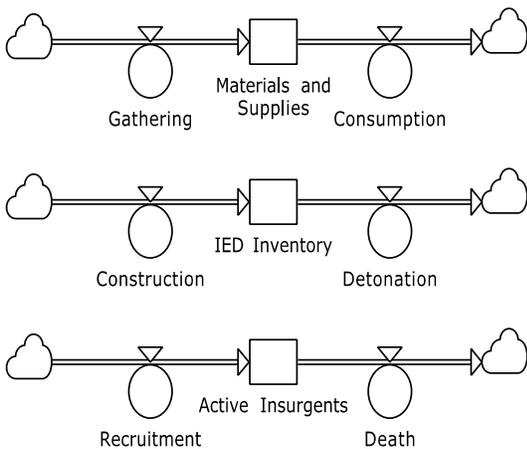


Figure 4. Simplified core model

#### 4.1 The Three Focus Areas

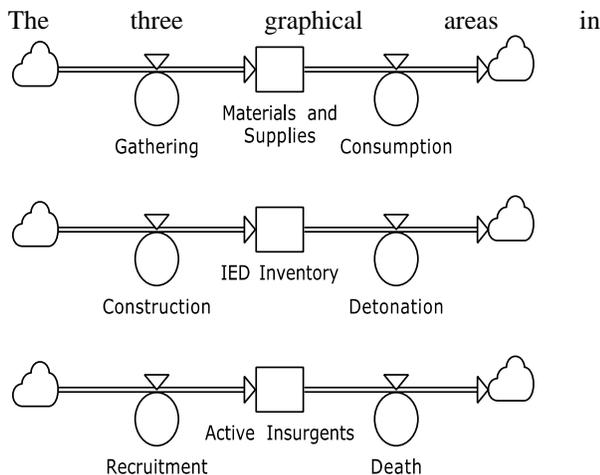


Figure 4 represent three focus areas of the core model, described below.

#### 4.1.1 Materials and Supplies Focus Area

This section of the model is shown in Figure 5. Here, a single stock represents the inventory of generalized materials and supplies available to insurgent groups. Materials are expressed by the generalized unit “item” to represent hypothetical items such as pounds of fertilizer or gallons of fuel. This section of the model contains one stock: *Materials and Supplies*. The input flow, materials *Gathering*, represents actions that cause the accumulation of materials and supplies. The output flow, *Consumption*, represents the use of these materials and supplies in the construction of IEDs. The *Materials and Supplies Gathering* submodel is presented in Section 5.3.

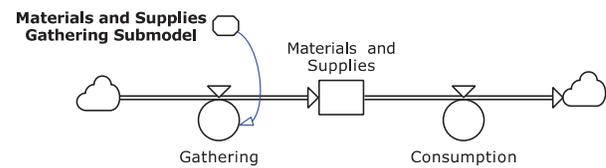


Figure 5. Model components related to IED materials and supplies

#### 4.1.2 IED Process Focus Area

The process of IED deployment is represented in the middle portion of the core model, with five stocks representing actual IEDs, and is shown in **Error! Reference source not found.** In practice, the process is varied and IEDs move through it in different ways, but this model is a generalized representation that the SMEs felt was reflective of the process. Moving through the diagram, a typical IED is constructed either for the purpose of a particular attack or to be stored for future use. Once it is constructed it is moved into inventory, which may be a traditional form of inventory (such as a warehouse), or it may be stored in a less conventional way (e.g., distributed throughout the community). IEDs may also be held by individuals who have little knowledge of the item’s true nature or purpose. Once insurgents have decided to emplace an IED, it is removed from inventory and emplaced in the field or acquired by a suicide carrier. Finally, whenever a target is near, the IED is triggered manually or automatically. Each of these stages is represented in the model by a stock that aggregates the IEDs currently within that stage.

At any point during the process, counter-IED methods may be used to destroy an IED before it is used against a target. This disruption detours the IED and deposits it in the *Disrupted IEDs* stock.

The next step in the modeling is to calculate the flows that represent the transition of stocks from one stage to

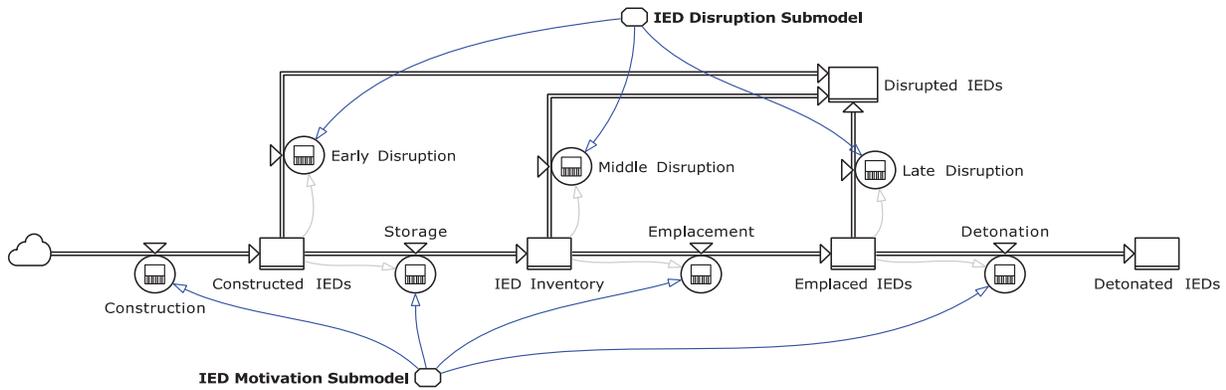


Figure 7. IED Process model components

another. Each flow's value is governed by a corresponding expression derived from other variables in the model that affect it. The movement of an IED between stocks is controlled by a series of flows, which are in turn affected by the number of personnel available within the insurgent groups. The expressions that control how these are related can be identified by an expert or changed by an analyst.

The insurgents' motivations to continue the IED process are represented in the IED Motivation submodel. Research into these motivations is underway and results are included as one of many submodels to allow a series of inputs that drive or reduce motivation.

#### 4.1.3 Personnel Focus Area

Understanding the behavior of people involved in IED activities includes understanding when and where they may be susceptible to being recruited or radicalized. The recruitment process results in several levels of categorization: the *General Population*, the *Grey Population*, and *Active Insurgents*. Each of these groups is represented as a stock within the system dynamics model. See Figure 6.

The stock representing members of the *General Population* shows the transition of a person into a sympathizer (a member of the *Grey Population*

susceptible to further radicalization), then into an *Active* participant within a terrorist group. While the indoctrination and recruitment of insurgents is a nuanced and multi-faceted process (Gerwehr & Daly, 2006), the model initially simplifies this so that the critical aspects can be identified.

The system dynamics model shown in Figure 6 indicates how the flows (radicalization, recruitment, deradicalization, disengagement, and death) are controlled by submodels. The core model sees the final result of each submodel as a single value that influences the stocks and flows.

### 5. Submodel Development Using Influence Diagrams

The use of submodels allows for the development, modification, and reuse of model components as modules within the model. A submodel based on a particular set of assumptions about the environment or about behaviors can then be replaced by a different submodel for analysis or refinement or to incorporate differing views or approaches that SMEs may have. This research leveraged influence diagrams to create the submodel influences on the flows within the system dynamics model.

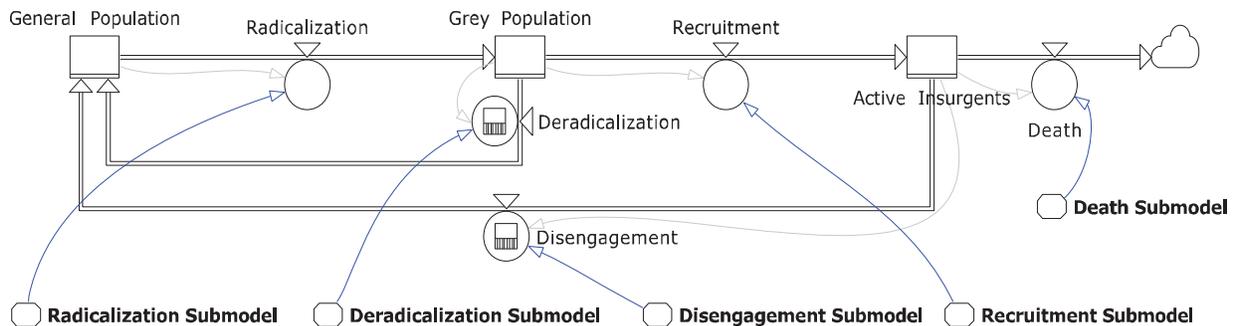


Figure 6. Insurgent personnel model components

An influence diagram is a graphical representation of a group of causal relationships and offers a method to couple the essential elements of a situation, including decisions, uncertainties, and objectives, by describing how they influence each other.

Alternative influence diagrams can be used to explore possible relationships between variables and can be used to provide values to variables that are inputs to stocks and flows. A set of causal relationships that influences these variables can be developed as a submodel for input. In this way submodels can be reused and interchanged to explore the outcomes resulting from different relationships. A model can thus be extended to represent a larger part of a scenario being modeled by attaching multiple, appropriate submodels.

This paper describes three of the submodels that support the core model.

- Radicalization / Deradicalization Submodel
- Recruitment / Disengagement Submodel
- Materials Gathering Submodel

### 5.1 Population Radicalization and Deradicalization Submodel

*Radicalization* represents the transition of a person within the *General Population* into the *Grey Population*. This occurs when a previously neutral person has taken a position of sympathy for insurgent beliefs. Insurgent groups achieve this end through various means, such as spreading broad propaganda supporting their goals, or by using community leadership roles as influence. Whenever a person holds a positive view of the insurgents' goals and tactics, that person is considered vulnerable for recruitment.

*Deradicalization* occurs when the attitudes of an individual are moderated from the radical views of the insurgency to the more mainstream views of the general population.

Figure 8 depicts the submodel showing the variables that affect population radicalization and deradicalization. General factors that affect individuals' behaviors can be grouped into four categories:

- Camus: Moral and religious factors
- Dewey: Social factors
- Smith: Economic factors
- Maslow: Quality of life factors

These factors build on the work of Bartolomei, et al. (Bartolomei, Casebeer, & Thomas, 2004) and were combined using an influence diagram to determine the

value of the flows. Influence diagrams represent influences as directional weights that are combined with other weighted inputs via an equation, and where the output is a rate of change. The outputs of these equations are then inputs to the system dynamics model.

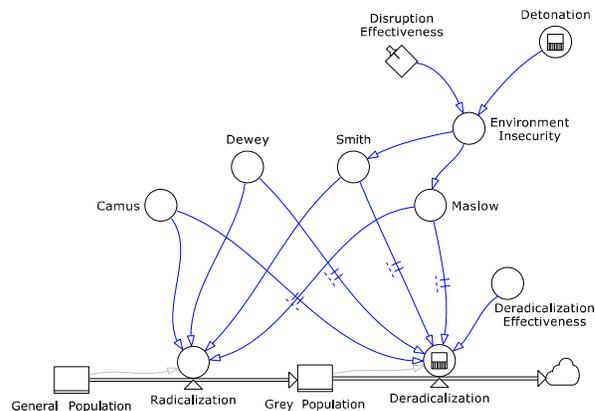


Figure 8. Submodel representing the radicalization and deradicalization of the population

### 5.2 Insurgent Recruitment and Disengagement Submodel

Recruitment and disengagement represent the voluntary or coerced actions of persons joining or leaving the insurgency. As a person becomes an active participant in the IED process, this person is considered recruited and is represented as a *Recruitment* flow. This may be an overt decision by the participant, or it may be a gradual process in which an insurgent group slowly eases a sympathizer into increasingly more severe tasks. The model considers the person to be recruited whenever he or she is actively involved in the process of constructing, storing, emplacing, or detonating IEDs. *Disengagement* occurs when someone has left the group of active insurgents and reduces the number of active insurgents.

The Recruitment and Disengagement submodel is presented in Figure 9. The variables surrounding the *Recruitment* and *Disengagement* flows represent influences that drive those decisions. A feedback loop is visible within the following chain of variables: *Environment Insecurity* → *Resentment* → *Recruitment* → *Detonation* → *Environment Insecurity*. As such, it can be useful to identify potential intervention points in the recruitment process. Within the model, the value of *Disengagement Effectiveness* can be adjusted as part of the system dynamics modeling to assess effectiveness of counter-IED and counter-insurgency efforts.

### 5.3 Materials Gathering Submodel

This submodel, shown in Figure 10, depicts the gathering of materials and supplies by the insurgency. As supplies are consumed, the reduction in the *Materials and Supplies* stock yields an increase in the *Supply Deficit*, which leads to an increase in *Supply Gathering Efforts*. The success of these efforts is hindered by increasing the amount of interference by counter-IED actions, represented as *Supply Gathering Interference*. This submodel can be expanded to include aspects such as the gathering of illicit items that cannot be readily purchased or financial resources that allow for the purchase of base materials (National Research Council, 2008).

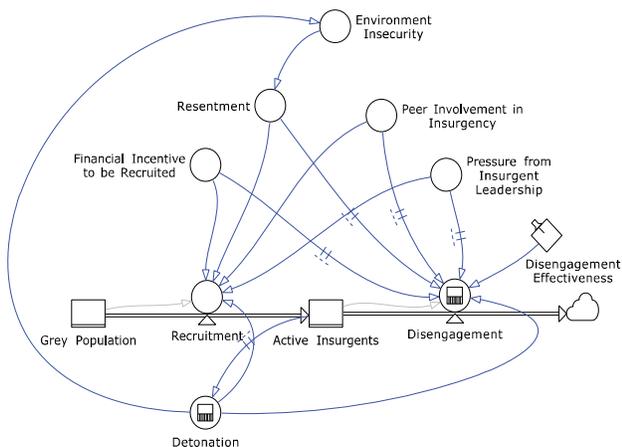


Figure 9. Submodel representing the recruitment and disengagement of insurgents

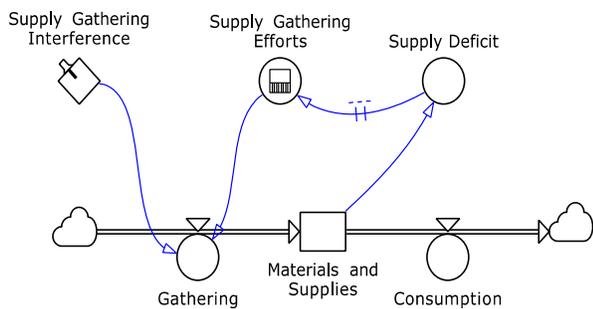


Figure 10. Submodel representing the gathering of material and supplies by insurgents

## 6. Integration of Model Components

When the suite of model and submodel components is integrated into the modeling framework, a modeling environment is created to assess potential intervention options. This paper presents the components that provide the content and the interactions for the models before the analysis process begins. Once those

components are in place, they can be integrated, swapped, modified, and updated to support evaluation of potential intervention options.

The integration framework also supports insertion of different submodels. For example, if two SMEs have differing views on how to disengage insurgents, then Figure 6 can be operated with either submodel feeding the disengagement flow and analyses can be conducted using an integration of these models; preliminary assessments have been conducted (Weiss, et al., 2009), and although there are not immediate plans for a longitudinal study to assess potential interventions, the resulting tool could support such an analysis.

The benefit of such analysis is that, although integrated models will not precisely predict who will become recruited, they can provide insight into two important aspects of the domain:

- (1) The relative importance of factors and influences. For example, it may be suggested that the best intervention is to influence the *General Population* before they are radicalized, but if a large part of the population is inherently radicalized, there may not be much benefit in working with the general population. Often, people are radicalized to some extent through their environment. Therefore, a more effective approach may be to address the flow from the *Grey Population* to the further radicalized stage of *Active Insurgent*.
- (2) Previously unconsidered aspects of the problem become exposed so that insight may be provided on an issue that may otherwise not been considered. It is easier to play-out unrealistic, but potentially eye-opening, scenarios in a modeling environment rather than real-life.

## 7. Conclusions

This paper focuses on an approach to component modeling of behaviors related to terrorist recruitment and the motivation to construct, emplace, and detonate IEDs. The approach combines computational and social science research to develop an improved ability to identify and understand activities and behaviors of potential IED developers in a population. The approach uses various modeling techniques, including mind mapping methods for knowledge engineering, system dynamics models for representing system behavior, and influence diagrams for developing submodels to show causal relationships. When the components are integrated, they provide a framework for analysis of recruitment deterrents and potential intervention points associated with IED perpetration.

## 8. Acknowledgements

This work was supported by a grant from the Office of Naval Research. We would like to acknowledge our colleagues at Penn State (Drs. Kevin Murphy, John Horgan, and Frank Ritter) and the University of Hull (Dr. Caroline Kennedy-Pipe) and thank them for their valuable interactions.

## 9. References

- Bartolomei, J., Casebeer, W., & Thomas, T. (2004). *Modeling Violent Non-State Actors: A Summary of Concepts and Methods*. United States Air Force Academy, Institute for Information Technology Applications, Colorado.
- Burgoon, J. & Varadan, V. (2006). Workshop Report on Detecting and Countering IEDs and Related Threats. *NSF IED Workshop*. National Science Foundation.
- Crenshaw, M. (2000). The Psychology of Terrorism: An Agenda for the 21<sup>st</sup> Century. *Political Psychology*, 21(2), 405-420.
- Gerwehr, S. & Daly, S. (2006). Al-Qaida: Terrorist Selection and Recruitment. In D. Kamien (Ed.) *The McGraw-Hill Homeland Security Handbook*. New York: McGraw-Hill.
- Horgan, J. (2007). *From Profiles to Pathways: The Road to Recruitment*. Retrieved from the web October 11, 2007. <http://usinfo.state.gov/journals/itps/0507/ijpe/horgan.htm>
- National Research Council. (2008). *Disrupting Improvised Explosive Device Terror Campaigns*. Washington, D.C.: National Academies Press.
- Sterman, J. (2000). *Business Dynamics: Systems Thinking and Modeling for a Complex World* (1st ed.). Boston: McGraw-Hill/Irwin.

Weiss, L., Whitaker, E., Briscoe, E., Trehwitt, E., (2009). *Modeling Behavioral Activities Related to Deploying IEDs in Iraq*. Georgia Tech Research Institute, Technical Report #ATAS-D5757-2009-01, Atlanta, GA.

Zacharias, G., MacMillan, J., & Van Hemel, S. (Eds.). (2008). *Behavioral Modeling and Simulation: from Individuals to Societies*. Washington, D.C.: National Research Council.

## Author Biographies

**DR. LORA G. WEISS** is a lab Chief Scientist at the Georgia Tech Research Institute, where she conducts research in all aspects of behavioral systems, from behavioral analyses of individuals and groups to behavior-based unmanned and autonomous systems.

**DR. ELIZABETH WHITAKER** is a Principal Research Engineer at Georgia Tech Research Institute. Her primary research interests are AI reasoning, representation, learning and modeling. Many of her research projects include hybrid representations and reasoning approaches including intelligent agents, case-based learning, planning and reasoning, and system dynamics.

**DR. ERICA BRISCOE** is a Research Scientist at Georgia Tech Research Institute. She works on human social, cultural, behavioral, modeling, with a concentration on cognitive psychology.

**ETHAN TREWHITT** is a Research Engineer at Georgia Tech Research Institute. He works on artificial intelligence and machine learning techniques for building computational models of human behavior and other domains.

# Imperfect Situation Awareness: Representing Error and Uncertainty in Modeling, Simulation & Analysis of Small Unit Military Operations

Victor E. Middleton

V. E. Middleton, Enterprises, LLC.

2356 Whitlock Place  
Kettering OH 45420-1360

(937) 253-1257

middletv@woh.rr.com

## Keywords:

Modeling, simulation & analysis; situation awareness/situation understanding; human behavioral representation; complex adaptive systems; emergence; agent-based models

**ABSTRACT:** *This paper examines the use of agent-based modeling and simulation to represent Situation Awareness/Situation Understanding (SA/SU) and its antithesis, the so-called “fog of war”. “Good SA/SU focuses on support for “the right information, for the right person, at the right time.” As a consequence, measuring improvements in SA/SU will require comparison to baselines of “wrong information, wrong person, wrong time.” Unfortunately, current M&S tools are most generally characterized by model omniscience; individual entities typically “recognize” friend from foe, “know” the precise location, speed and heading of themselves and their targets, and, most importantly, act in accordance with this knowledge. Such omniscience is, of course, at considerable variance with the uncertain, incomplete, inconsistent, and often erroneous data that constitute the “fog of war” in actual operations. The intent of this paper is not to add materially to the theory of SA/SU, but rather to develop an engineering solution to the problem of representing imperfect SA/SU in agent-based simulations of small unit operations.*

## 1 Introduction

In the late 80’s and early 90’s the “soldier as a system concept” (now referred to as the more encompassing “warrior system”)<sup>1</sup> was developed to forge the individual combatant and his equipment into a complex, synergistic, system of systems. This warrior system concept has been widely accepted internationally<sup>2</sup>, and today focuses on integrating the capabilities of new C4ISR<sup>3</sup> technologies to improve individual and unit situation awareness and situation understanding (SA/SU).<sup>4</sup> At present, however,

Modeling & Simulation (M&S) of military operations suffers from inadequate representation of SA/SU and decision-making, and therefore M&S-based analysis lacks the tools to assess potential SA/SU improvements provided by new technologies or proposed systems<sup>5</sup>.

As an example, “good” SA/SU technologies should help to provide “the right information, for the right person, at the right time.” Consequently, measuring SA/SU improvements will require comparison to baselines of “wrong information, wrong person, wrong time.” Unfortunately, current M&S tools do not consider such baselines. These tools are characterized by model omniscience; individual entities “recognize” friend from foe, “know” the precise location, speed and heading of themselves and their targets, and act in accordance with this knowledge. Such omniscience is, of course, at considerable variance with the incomplete, inconsistent, and often erroneous data that constitute the “fog of war” in actual operations.

### 1.1 Objective of the Paper

This paper examines the challenge of representing SA/SU and its antithesis, the so-called “fog of war”,

---

<sup>1</sup> See for example [Middleton & McIntyre 2001]

<sup>2</sup> For example, [Housson 2008] discusses programs by the British: FIST (Future Integrated Soldier Technology), Germans: IdZ (Infanterist der Zukunft), Spanish: COMFUT (COMbatiente FUTuro), French: FELIN (Fantassin à Équipements et Liaisons Intégrés), and Italians: Soldato Futuro. See also [Leuw, 1997], [Hassgård, 2002], [Curtis, 2002], [Hobbs, 2000], [Underhill 2009] for Dutch, Swedish, Australian, and Canadian examples and perspective.

<sup>3</sup> Command, Control, Communications, Computers, Intelligence Surveillance and Reconnaissance

<sup>4</sup> Rather than engage in a discussion as to the differences between SA and SU, I choose to blur them together to a single over-arching concept following the pragmatic definition of [Adam 1993] “knowing what is going on so I can figure out what to do”

---

<sup>5</sup> See for example [Tollefson et.al 2008], [Middleton & Mastroianni 2008], [Pew & Mavor 1998]

through the use of agent-based modeling (ABM) and simulation. My focus is on the warrior system, small unit operations and irregular warfare. My goal is to develop a framework for enhancing ABM SA/SU capabilities. The framework will define agent functions and data structures to: 1) reflect the uncertainty and error in what agents know; 2) represent how they act on that knowledge, and 3) capture metrics that correlate levels of SA/SU with operational outcomes.

I am not looking to develop a new theoretical understanding of SA/SU and decision-making. Rather my goal is an engineering solution to the practical problems faced by decision-makers and the analysts who support them. The solution must support system design requirements and evaluation of the technological approaches that may be proposed to meet those requirements. The solution should facilitate the exploration of tactics, techniques and procedures (TTPs) for the employment of current and proposed new systems. Making the distinction between fidelity and resolution expressed by [Bailey & Kemple 1992], the solution should focus on improving model fidelity, with minimal increases in model resolution, level of detail or complexity.

## 1.2 Problem Statement

Systems analysis of large weapon systems (e.g., manned vehicles and airplanes) is supported by engineering models that describe and predict the operation of these systems. Such models are generally characterized by deterministic, Newtonian physics-based representations of closed systems, i.e. systems whose exchanges of mass and/or energy with their environment are constrained to a relatively few, well-known, factors. These models may incorporate stochastic treatment of systems performance, based on statistical data from measurement of well-defined systems' functions. Their model parameters span the analytically relevant/interesting areas of the problem space, and there is essentially a one-to-one mapping between model features and systems' functions. These features support model verification and validation based on theoretical concepts, and supported by empirical data on operators/systems' performance.

I maintain the problems of warrior systems' analysis begin with the statement, "**we lack an engineering model of the individual soldier.**" The complexities of the warrior system are not amenable to the strict reductionist approach of orthodox systems analysis, which fails to account for the dynamic and highly non-linear interactions of the cognitive and physiological elements that constitute the warrior system. These complexities are exacerbated further by the nature of irregular warfare and asymmetric combat, in which the

interactions between friendly forces, adversaries, and neutrals form a seemingly chaotic dynamic landscape.

Writing for the Military Operations Society's Phalanx in 2002, Vincent Roske<sup>6</sup> spoke of the need for new tools to address the class of "open systems" not accessible using traditional operations research tools. Such systems are characterized by uncertainty and imprecision in both system inputs and system behaviors, which can make their behavior harder to predict. At the same time, embracing the uncertainty and non-linearity of these systems can provide much higher fidelity in describing the performance of systems whose subsystem capabilities can, and often do, lead to the whole being greater than the sum of its constituent parts.

We need to upgrade our concept of "engineering" models. We need engineering models that allow us to explore virtual systems whose behaviors emerge from general rules of operation, that are not limited to functional capabilities that can be reduced to physics-based algorithms. This upgraded concept does not mean eliminating the use of physics or the other "hard" science, it simply means extending the reductionist approach to support a wider variety of systems decompositions. It means, for example, decomposing systems operations into sets of entity or object interactions as in done in agent based models. Ilachinski describes this approach as collectivism:

*Collectivism embodies the belief that in order to properly understand complex systems, such systems must be viewed as coherent wholes whose open-ended evolution is continuously fueled by nonlinear feedback between their macroscopic states and microscopic constituents. It is neither completely reductionist (which seeks only to decompose a system into its primitive components), nor completely synthesist (which seeks to synthesize the system out of its constituent parts but neglects the feedback between emerging levels). [Ilachinski 1996]*

This complex systems approach also suggests the need to measure the "validity" of simulation outcomes as less in terms of their agreement with predictions of real world phenomena, and more in terms of their ability to provide insight and to further our understanding of these phenomena.

## 2 Approach

The above considerations suggest agent-based models that view military operations as complex adaptive

---

<sup>6</sup> Roske was then serving under the Chairman of the Joint Chiefs of Staff as the Deputy Director, J8 (Wargaming, Simulation & Analysis)

systems (CAS)<sup>7</sup> provide a promising approach for analysis of SA/SU issues.

Under this approach, simulated “Intelligent” agents (IA) make decisions and attempt to satisfy mission goals according to their own individual (and probably imperfect) SA/SU. While any simulation maintains its internal “ground truth” knowledge base, each IA will have a “perceived truth” knowledge base – the idiosyncratic view of the combat situation, as seen by that individual IA and obscured by the agent’s local “fog of war”. IA behavior choices are made on “perceived truth” of the agent; the behaviors and their effects, however, take place in the “ground truth” world of the simulation.

Allowing each agent to act on an imperfect worldview supports evaluation of the operational costs of uncertain, incomplete and/or incorrect information. It also supports explicit modeling of leader decision-making processes based on such data, of imperfect command and control, and/or imperfect subordinate receipt of and subsequent execution of orders. Such modeling is critical if we are to estimate the benefits of proposed new or modified systems, and/or adjustments to tactics, techniques and procedures.

This approach supports measures of command and control such as the Objective Information System Assessment (OISA) Paradigm [Davidson, Pogel, and Smith, 2008], which compares the performance of individual decision-makers employing a particular information system, to what that same decision-maker would have produced given an alternative data stream.

## 2.1 ABM as an Engineering Model of the Warrior System

In addition to all of the physics-based phenomena characteristic of military operations, an engineering model of the Warrior System must also address the so called “soft factors” – morale, leadership, training, and the values/beliefs associated with nationality/ethnicity, that are critical to current operations.

The framework for an engineering model of the individual soldier centered on agent-based modeling has already been established through distillation models such as Pythagoras<sup>8</sup> and CROCADILE<sup>9</sup> and more detailed models such as IWARS<sup>10</sup> and Combat XXI<sup>11</sup>. , with intelligent agents that are:

- goal-oriented - able to build courses of action by taking the initiative to change elements of the world state to desired objectives
- perceptive - able to receive data from their environment, including knowledge of their own state and that of other entities of interest to them,
- active - able to perform actions affecting their environment, and
- autonomous - able to use internal logic to make decisions and initiate behavior sequences based on what is appropriate given the perceived environment.

Agents representing combat forces must also generally be:

- mobile - able to move around in their simulated environment,
- insightful - capable of inferring the intentions of others, determining the desires and plans of other agents, and
- social - able to share goals, cooperate with or coerce other agents.

A key distinction between agents that are “intelligent” and those that are merely reactive is the concept of having “knowledge” of the world based on current and historical data from the agent’s sensory input capabilities. Intelligent agents are not omniscient, they do not share the simulation “god’s eye” view of the world, rather they gather and interpret data according to their own capabilities. One can characterize the degree of an agent’s intelligence based on the extent of its historical sensory database, its capability to use inference to supplement incomplete input data, and/or to resolve uncertain or inconsistent data, and its degree of autonomy. Autonomy is of particular importance for simulation-based analysis, because it is gauged by the degree to which behaviors are not pre-scripted by simulation designers. Autonomy is enhanced by increasing both the number of options available to the agent in response to the perceived environment, and the flexibility the agent has in choosing those options.

Autonomy also permits unpredictable (to other entities) behaviors, a key feature of viewing military small unit operations as CAS. Autonomy makes possible the “adaptive” part of complex adaptive systems, providing the potential for emergent behavior through IA co-evolution with a dynamic operational environment and with other systems. Adaptation is a concept taken from the biological view of evolution and implies the operation of a “fitness” function or functions that support “selection” of those characteristics or behaviors of the system that enable it to best “fit” in its environment. In the warrior systems view, fitness

<sup>7</sup> See for example [Ilachinski 2004], [Cioppa, Lucas & Sanchez 2004], [Horne & Leonardi 2001]

<sup>8</sup> See for example [Bitinas et. al 2003]

<sup>9</sup> See for example [Easton & Barlow 2002]

<sup>10</sup> See for example [Bachman et al 2008]

<sup>11</sup> See for example [Kunde & Darken 2006]

functions are derived from satisfying IA goals, and “fitter” systems, e.g. those with improved SA, are those which are better able to achieve mission and unit goals.

The IA must derive data from the environment, through appropriate sensory and communications processes. The IA then interprets these data in the context of the its experience and current knowledge base, achieving some level of comprehension as to what the data means, and finally, develops a set of expectations<sup>12</sup> as to the results of its own or other’s actions and behaviors. Such expectations play a key role in proposing to view the IA as a basis for an engineering model of the warrior system. Following Klein’s (1999 & 2008) concept of Naturalistic Decision-Making (NDM), I see the IA as continually adjusting its behavior based on the degree to which its expectations are or are not met.

## 2.2 Architecture

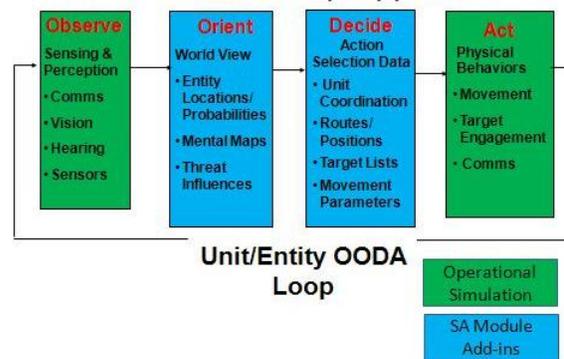
The architecture proposed herein basically conforms to Miller & Shattuck’s (2004) Dynamic Model of Situated Cognition (DMSC). This model represents the perception of ground truth as a function of sensor systems, the capture of those data by command and control systems, and the (possibly imperfect or erroneous) processing of these data into Endsley’s (1995) three levels of SA: perception, comprehension and future projection.

Of course, many of today’s models, such as those listed in section 2.1, already represent aspects of SA/SU and decision-making under uncertainty, incorporating aspects of the DMSC and Endsley’s three levels of SA. The key to augmenting extant representations is the incorporation of model features that further distinguish between an individual’s perceived world view and ground truth. Incorporating these features requires the design, development, and implementation of three inter-related elements:

1. data structures to characterize each entity’s perceived knowledge of the operational environment;
2. algorithms and heuristics to populate, maintain, and update those data structures; and
3. inference schemes employing these data to represent operational decisions.

<sup>12</sup> [Kunde 2005] discusses the role of expectations and mental models in his computational model for mental simulation in a combat simulation environment. He proposes a simulation architecture that incorporates the basic ideas of Recognition Primed Decision-making (RPD) [Klein 1993], and decision-making architecture as a framework for applying mental simulation in a combat simulation environment. [Kunde & Darken 2006]

These features can be incorporated into current simulations in a modular architecture following Boyd’s OODA loop [Boyd 1986] as shown in Figure 1. In this approach, the three elements are encapsulated in modules that constitute the “Orient” and “Decide” components of the OODA Loop, the blue boxes of Figure 1.



**Figure 1 Modular OODA Loop Approach**

This approach provides a controlled interface between new SA/SU capabilities and basic simulation processes. The sense/perception processes native to host simulation entities allow those entities to “observe” their virtual world as before, providing data on the simulation environment and the objects in it. The new “orient” modules interpret those data through (potentially imperfect) filters to populate and update world view data structures unique to each entity. As an example, an entity may observe another entity that it previously would have identified according to its force association and any threat value. New “orient” filters could “translate” entity sightings into levels of evidence for associating that entity with a given force or threat intent. Similarly such filters could add imprecision and/or error to the sighting entity’s perception of the sighted entity’s location. Inference routines could evaluate evidence from multiple sources, resulting in attributes of the sighted entity described as degrees of membership in fuzzy sets as opposed to the generally crisp (e.g., friend or foe, within range, at objective) options currently available.

The “oriented data” is now information that is used by the decision logics of the “Decide” module to choose and direct those entity behaviors deemed most likely to achieve entity/unit goals. The host simulation “Act” capabilities carry out these behaviors and determine effects on other entities and the environment.

## 2.3 Data Structures

There are three main classes of data structures required under this approach: Perception Data Structures (PDS), Inference/Decision Structures (IDS) and Behavior Data Structures (BDS). The role these structures is to

capture the results of filtration and fusion to support inference/decision procedures not necessarily native to the host simulation (PDS), to provide the parameters and inter-object relationships needed by these inference/decision procedures (IDS), and to translate the results of these procedures into directions consistent with host simulation behaviors (BDS).

PDS reflect the operational environment and the entities in it as perceived by a given agent, interpreted and formatted as required by that agent's various inference schemes and decision models. They include:

- cues – environmental data, either direct perception or as a result of shared communications, expressed as object state variables;
- alerts – special cues demanding immediate action;
- thresholds – object state variable values that reflect or initiate a state change in that object or others;
- landmarks – cues that cannot be ambiguously interpreted. Recognition of a landmark either absolutely confirms or refutes elements of an IA's currently held world view associated with that landmark; and
- influence ambits – an area, range or scope over which an object can/does exercise control.

IDS provide the framework and core parameters of the schema used to represent inference<sup>13</sup> and/or decision-making. Examples include:

- patterns for situation assessment and projection heuristics representing mental simulation - needed for recognition-primed decision-making<sup>14</sup>;
- directed acyclic graphs (DAG) and associated conditional probability distributions – needed for Bayesian belief networks<sup>15</sup>;
- causal weighted adjacency matrices- needed for fuzzy cognitive maps<sup>16</sup>;
- a basic probability assignment function (bpa), a Belief function (Bel), and a Plausibility function (Pl) – needed for Dempster-Schaefer theory<sup>17</sup>;
- belief sets and belief states, goal sets and goal states, and plan sets – needed for the Belief, Desire, and Intentions (BDI) paradigm<sup>18</sup>; and

- directed graphs representing input, output and hidden layers of artificial neurons and weighted connections – needed for neural networks.<sup>19</sup>

BDS allow the new orient and decide modules to share data about the problem space with native behaviors. They are the vehicle by which IA decisions are shared with the host simulation. There are three basic forms:

- Course of Action options;
- Behavior parameters - targets and target priority lists, types and rates of fire, shoot/no shoot decision thresholds for engagement; routes and waypoints or direction vectors for movement, speed and movement formations; and
- Communications – Situation reports (SitReps) to other units, especially command units, directives to subordinates, unit coordination, request for fire or other support.

Taken in concert, these structures and the inference/decision schemes they support can address some significant shortfalls in current simulation capabilities. For example, current models generally require an acquired sight picture of a target entity as a prerequisite to firing a weapon. There is little capability for behaviors such as firing at sound cues, "leading" a moving target, suppressive fire at locations with no visible targets, or more rapid acquisition of a target based on previous detection history.

## 2.4 Inference and Decision-Making

Decision-making is frequently looked at as a discrete event, with alternatives considered, a choice made, and that choice acted on. In the world of discrete event simulation this view is certainly justified at some level, even continuous processes are broken into atomic chunks of activity, and the scheduling of the next event represents a decision of some sort. I believe, however, that it is useful to consider decisions as falling into three broad, albeit overlapping, categories:

- prescriptive plans, e.g., course of action selection, scheduling and coordination of entity/unit tasks, macro-level movement parameters (route selection in terms of general destination, waypoints, avenues of advance, etc.)
- reaction to unanticipated events, e.g., correction of meso-level movement (adjust next waypoint to detour around obstacle/threat, modify formation), engage an adversary/ choose engagement tactics, call for fire or request other kinds of support; and

<sup>13</sup> Following the general lead of [Davis, Shrobe & Szolovits 1993] I am using inference in the generic sense as a way to get new information from old, rather than as limited to sound logical inference.

<sup>14</sup> See for example [Klein 1993] or [Warwick et.al. 2001]

<sup>15</sup> See for example [Russell & Norvig 2003]

<sup>16</sup> See for example [Kosko, 1986]

<sup>17</sup> See for example [Sentz & Ferson 2002]

<sup>18</sup> See for example [Kinny, Georgeff & Rao 1996]

<sup>19</sup> See for example Rao & Rao 1993]

- repeated and/or continuous modification of combatant behavior parameters, e.g., micro-level movement (how fast to move, in what direction, fine-tune selection of cover or firing positions), choose which targets to engage when, adjust aim points and rates of fire.

There are a wide variety of approaches to representing and/or facilitating decision-making, some of which are illustrated in Figure 2, and are supported by the IDS examples listed above.

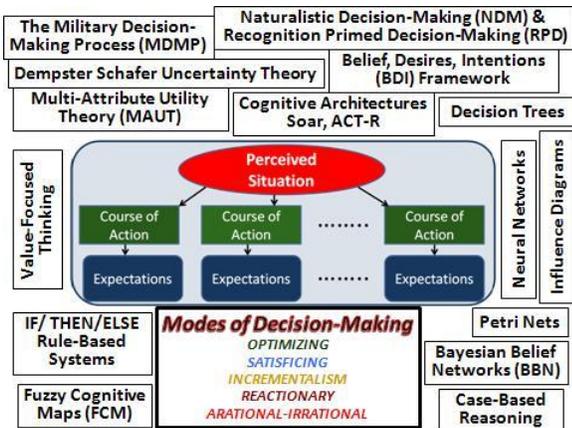


Figure 2 Decision Approaches

Central to all of these paradigms/architectures/methodologies is the view of a decision as the selection of “doing something”- a course of action, based on an understanding of the current situation – an individual’s perceived SA/SU, and with projected outcomes – expectations, associated with each potential course of action.

Also common to these approaches is the concept of rational action, that the decision-maker will attempt to find the “best” course of action to achieve his/her goals. Figure 2, however, also lists the modes of decision-making from [Zim, 1999], which correlates the effects of time pressure and stress to the quality of decision-making. Zim describes a number of problems observed in decision- makers under stress, including:

- changing from deliberative to reactionary modes;
- relying on only a limited fraction of available information with a bias towards that which is familiar and corresponds to earlier perceptions over that which is relevant and/or unexpected;
- making more mistakes but being less likely to acknowledge them; and
- increasing micro-management of subordinates.

Representing these tendencies towards “imperfect” decision-making is critical to providing a robust simulation test bed for SA/SU technologies.

Integrating more of the above “rational” decision approaches (or combinations thereof) into current simulations, is a necessary, but not sufficient condition for robustness, such integration must be accompanied by realistic representation of error and imperfection.

Error and uncertainty can be introduced, for example, by following Miller and Shattuck’s concept of multiple lenses for acquisition and understanding of ground truth data with those lenses dynamically warped as appropriate to degrees of stress and time constraints.

Error and uncertainty also play a big part in the feedback loop between expectations and decisions to adjust or change behaviors. Expectations may not be met because of failure to understand the decision context (flawed SA/SU), because of unpredictable random variations in physical processes, and/or systemic error in the decision process itself (invalid logic, erroneous antecedent/consequent connection or other incorrect schema elements).

## 2.5 Implementation Issues

For many current simulations initial integration/ implementation of new SA/SU features can begin without needing “new” data. By using the data structures described above, and using data filtration, data fusion and inference simulation entities can explicitly recognize information already implicit in the simulation environment. For example, consider a scenario in which a small unit has detected and attempted to engage a number of adversaries who are taking advantage of local terrain for cover and concealment. By adding new data structures to record a shared unit history of detections and positions, an inference scheme such as a Bayesian Belief Network could conclude that the size of the adversary unit was too great for the engaging unit and develop a “call for fire” message, assuming that the host simulation supports indirect fire missions. Alternatively, if the adversary force is more manageable a BDS could post artificial “targets” on a target priority list. These targets, when engaged with host simulation firing behaviors, would have the effect of suppressive fire, enabling the engaging unit to close with and defeat the adversary force.

Augmenting the host simulation with additional scenario data and/or new behaviors would further expand the utility of this approach. For example, the addition of terrain characteristics with semantic content, i.e. operationally relevant meaning, can enhance the representation of engagement behaviors such as those described above. For example if doors or windows are understood to be objects where entities can enter or leave buildings, they become candidates for suppressive fire. Explicit inclusion of soft factors

such as morale, unit cohesion, and training could also play an important role in the representation of suppression and other reactive behaviors.

These features do not come without cost; clearly keeping each entity's unique world view will increase simulation memory requirements. Furthermore, capturing the dynamic nature of complex systems relationships will require maintaining a history of entity perceptions and other state variables that will further increase memory requirements. The persistence of these data, expressed as decreasing validity and/or credibility as a function of time, is not as yet well understood.

Supporting data for defining fuzzy set membership relations or other measures of uncertainty are scarce, and will probably have to be drawn from subject matter experts (SME). Similarly construction of inference schema to supplement incomplete input data, and/or to resolve uncertain or inconsistent data will be supported less by hard data, and rest instead on analysts' judgments and SME estimations.

Adding semantic content to terrain significantly increases the effort required for scenario development. For example, giving goal-driven IA the ability to interpret, and to make better use of, terrain features of military interest, requires the introduction of a complex set of terrain attributes. These attributes would capture such features as: Observation and fields of fire, Avenues of approach, Key and decisive terrain, Obstacles, and Cover and concealment<sup>20</sup>. Linking observed features of terrain with known enemy tactics and tendencies would further allow intelligent exploitation of terrain, with dynamic definition of areas of immediate importance, danger areas, choke points, and so forth.

Implementation of new features would best be approached in a modular fashion through incremental development. In such development, increasingly more robust versions of each element are implemented through a series of integrated cycles. This approach leads to analytical flexibility and accommodates application requirements for varying degrees of resolution and fidelity. It also supports hierarchal layers of inference and decision-making capabilities to address issues of information sharing among multiple potentially heterogeneous problem sets, as for example when an agent may need alternatives to support a "fight or flight" response. Different and possibly competing inference schemes can suggest potential targets and routes for retreat as the overall problem is parsed into

independent parts. The end-result is a flexible data-directed process that allows problem solutions to compete based on different criteria dependent on the situation, the current state of the agent and its active goals.

The incremental approach also helps address issues of model validity. By its very nature, any representation of the human dimensions of error, uncertainty and imprecision lacks the first principles models of cause and effect that are the foundation of "validated", physics-based models and simulations. Such representations can still fall under the purview of scientific rigor, but there is a need to extend that concept to incorporate a "soft", incremental focus, where parametric analysis bounds regions of factor effects and the extent/significance of functional relationships, and where increasing levels of correlation correspond to increased acceptance of predictive validity.

The bottom line is that the actions of an intelligent agent are taken in accordance with that agent's unique SA/SU and in expectation of fulfilling one or more goals. Using the data structures and inference procedures described above, an agent should be able to compare expectations to observable aspects of the environment. Agent behaviors are then seen as a cycle of updating/correcting SA/SU, followed by modification of behaviors as that new SA/SU suggests, until goals are achieved or a recognized failure point occurs.

### Author Biography

**VICTOR E MIDDLETON** is a Senior Operations Research Analyst with over 30 years experience developing, implementing, and applying mathematical models and simulations for a wide variety of military and civilian studies and analyses. He is one of the principal authors of the US Army's Integrated Unit Simulation System (IUSS), developed by Simulation Technologies, Inc, under contract to the US Army Research Development and Engineering Command (RDECOM) Natick Soldier Center and was the Principal Scientist for the evolution of the IUSS into a new model, the Infantry Warrior Simulation (IWARS), to meet the joint needs of NSC and the US Army Materiel Systems Analysis Activity (AMSAA).

### References

Adam, E. C. (1993). Fighter cockpits of the future. The 1993 IEEE/AIAA Digital Avionics Systems Conference. Proceedings of the 12th DASC: (pp. 318-323).

---

<sup>20</sup> Characterized by the mnemonic OAKOC as for example in US Army Field Manual 3.0

- Bachman, J. T., L. R. Hester, et al. (2008). Analyzing the Marine Corps Infantry Squad Using the Infantry Warrior Simulation (IWARS) 2008 Spring Simulation Interoperability Workshop. Westin Hotel - Providence, Rhode Island.
- Bailey, M. and W. Kemple (1992). The scientific method of choosing model fidelity. 24 th Winter simulation Conference, Arlington VA, ACM New York, NY, USA.
- Bitinas, E., Z. Henscheid, et al. (2003). "Pythagoras: A new agent-based simulation system." *Technology Review*: 45.
- Boyd, J. (1986). "Patterns of conflict." unpublished briefing, Defence and the National Interest, available at: [www.dni.net/second\\_level/boyd\\_military.htm](http://www.dni.net/second_level/boyd_military.htm) (accessed 14 November 2004).
- Cioppa, T., T. Lucas, et al. (2004). Military Applications of Agent-Based Simulations. 2004 Winter Simulation Conference M. D. R. R .G. Ingalls, J. S. Smith, and B. A. Peters.
- Davidson, J., A. Pogel, et al. (2008). THE ROLE OF BATTLE COMMAND IN INFORMATION SYSTEM ASSESSMENTS 13th Annual International Conference on Industrial Engineering Theory, Applications and Practice. Las Vegas, Nevada.
- Davis, R., H. Shrobe, et al. (1993). "What is a knowledge representation?" *AI magazine* 14(1): 17.
- Easton, A. and M. Barlow (2002). CROCADILE: An Agent-based Distillation System Incorporating Aspects of Constructive Simulation. SimTecT 2002, Melbourne Convention Centre, Melbourne, Australia.
- Endsley, M. (1995a). "Toward a theory of situation awareness in dynamic systems." *Human Factors: The Journal of the Human Factors and Ergonomics Society* 37(1): 32-64.
- Horne, G. L., Mary, Ed. (2001). Maneuver Warfare Science 2001 Defense Automated Printing Service Quantico, VA 22134-5007 USA
- Housson, J.P.(2008)."Soldier Modernisation: European Overwatch." <http://www.soldiermod.com/summer-08/features-europe.html>.
- Ilachinski, A. (1996). Land Warfare and Complexity, Part I: Mathematical Background and Technical Sourcebook (U), Center for Naval Analyses.
- Ilachinski, A. (2004). Artificial war: Multiagent-based simulation of combat, World Scientific Pub Co Inc.
- Kinny, D., M. Georgeff, et al. (1996). "A methodology and modelling technique for systems of BDI agents." *Lecture Notes in Computer Science* 1038: 56-71.
- Klein, G. (1993). Naturalistic decision making: Implications for design. . Dayton, OH:, Crew Systems Ergonomics Information Analysis Center, Wright-Patterson Air Force Base.
- Klein, G. (1999). Sources of power: How people make decisions, The MIT Press.
- Klein, G. (2008). "Naturalistic Decision Making." *Human Factors: The Journal of the Human Factors and Ergonomics Society* 50(3): 456-460.
- Kosko, B. (1986). "Fuzzy cognitive maps." *Int. J. Man-Machine Studies* 24: 65-75.
- Kunde, D. (2005). Event Prediction for Modeling Mental Simulation in Naturalistic Decision Making. Monterey CA, Naval Postgraduate School. Dr. O Philosophy in MOVES
- Kunde, D. and C. Darken (2006). A mental simulation-based decision-making architecture applied to ground combat. 2006 Behavior Representation in Modeling & Simulation (BRIMS), Baltimore, MD, SISO.
- Middleton, V. E., McIntyre, R.T.III; (2001). "FROM RIFLEMAN TO WARRIOR SYSTEM The Evolution of the Dismounted Close Combatant". SMi Conference on Dismounted Close Combat. London.
- Middleton, V. E. M. G. (2008). Implications of Human Centric Modeling for Operational Analysis. 2008 Conference on Behavior Representation in Modeling and Simulation (BRIMS). Westin Hotel - Providence, Rhode Island, SISO.
- Miller, N. and L. Shattuck (2004). A process model of situated cognition in military command and control. 2004 Command and Control Research and Technology Symposium
- The Power of Information Age Concepts and Technologies San Diego, CA.
- Pew, R. and A. Mavor (1998). Modeling human and organizational behavior: Application to military simulations, National Academies Press.
- Rao, V. B. R., Hayagriva V. (1993). C++ Neural Networks and Fuzzy Logic. New York, New York, Management information Source, Inc.; A subsidiary of Henry Holt and Co., Inc.
- Roske Jr, V. (2002). "Opening Up Military Analysis: Exploring Beyond The Boundaries." *Phalanx*, June.
- Russell, S. J. N., Peter (2003). Artificial Intelligence A Modern Approach 2cd ed. Upper Saddle River, New Jersey, Pearson Education, Inc.
- Sentz, K. and S. Ferson (2002). Combination of evidence in Dempster-Shafer theory. Albuquerque,

New Mexico 87185 and Livermore, California 94550, Sandia National Laboratories 835.

Tollefson, E. S., J. K. Alt, et al. (2008). Soldier Representation in Modeling and Simulation. Monterey CA, US Army TRADOC Analysis Center – Monterey (TRAC-MTRY).

TRADOC (2008). Field Manual No. 3.0 Operations, Headquarters, U.S. Army Training and Doctrine Command,.

Warwick, W., S. McIlwaine, et al. (2001). Developing computational models of recognition-primed decision making. tenth conference on Computer Generated Forces, Norfolk, VA.

Zimm, A. D. (1999). Modeling Maneuver Warfare: Incorporating Human Factors and Decisionmaking in Combat Modeling. Eighth Annual Conference on Computer Generated Forces and Behavioral Representation. Orlando FL, Simulation Interoperability Standards Organization (SISO).

# Modeling Human Eye-Movements for Military Simulations

*Patrick Jungkunz*  
German Naval Office  
18147 Rostock, Germany  
+49 381 802-5734  
patrick.jungkunz@gmail.com

*Christian J. Darken*  
Naval Postgraduate School  
The MOVES Institute  
Monterey, CA 93941  
831-656-2095  
cjdarken@nps.edu

Keywords: Visual attention, eye-movements, search and target acquisition

**ABSTRACT:** *Models of eye movements of an observer searching for human targets are helpful in developing accurate models of target acquisition times and false positive detections. We develop a new model describing the distribution of gaze positions for an observer which includes both bottom-up (saliency) and top-down (task dependent) factors. We validate the combined model against a bottom-up model from the literature and against the bottom up and top down parts alone using human performance data. The new model is shown to be significantly better. The new model requires a large amount of data about the terrain and target that is obtained directly from the 3D simulation through an automated process.*

## 1. Introduction

The modeling of target acquisition and detection has always been a major concern for military simulations. In the past, the capabilities of systems were the focus of attention; now the capabilities and the performance of humans need attention. As noted by Evangelista et. al. (2010), current simulation models of individual soldiers assume that they search a scene using a fixed pattern, e.g. a sweep from left to right. Anyone who has observed soldiers, especially in an urban environment, surely realizes that this is not an accurate model. Failure to model search accurately results in target acquisition times that are not accurate. Worse, it provides a poor basis for modeling detection phenomena such as false positive detections, i.e. seeing a target where none is present, which can have a significant impact on an operation. Current models of false positive detection can do little better than sprinkle false targets uniformly across the simulated battlefield. If we understood what parts of a scene were challenging for an observer, false targets could be placed in these locations instead.

In order to improve target detection mechanisms in military simulations, this work proposes to model human eye-movement behavior during target search as a basis for future enhancements in overall models of search and target acquisition. We provide a new model of eye movements and show that it is more accurate than the dominant model in the literature. This model

can extract its needed data from a 3D simulation through a process that has been largely automated.

Human visual perception is mainly characterized by the receptive qualities of the retina. The fovea, which is the center of the retina, provides high visual acuity and subtends about 2° of visual angle. This acuity rapidly decreases with higher eccentricity from the center. (Rayner & Pollatsek, 1992). The high acuity of the center is necessary for reliable object recognition. It follows that in order for humans to perceive the whole world around them with high acuity they have to perform eye movements. While the gist of a scene can be determined upon a single glance, eye-movements allow humans to serially fixate objects in the visual field one after the other in order to extract high level details from fixated locations (Henderson, 2003).

This means, a target can only be detected if the eyes are directed towards that target and attention is deployed to this location. Also, false targets can only be generated at locations fixated with the eyes.

Eye-movements and deployment of visual attention are both necessary to perceive objects (Itti & Koch, 2001a) and they are closely tied to each other (Hoffmann & Subramaniam, 1995). According to Itti (2003), there are several factors influencing the deployment of visual attention. These are bottom-up factors, which are visual scene features, for example salient edges or contrasting colors. Visually salient locations in a scene capture attention and the eyes of an observer. In addition to

that, there are top-down, task dependent factors driving attention allocation. Humans can voluntarily direct their eyes to locations they want to examine or they need to look at based on their current task.

Eye-movement and visual attention modeling is not a new endeavor. One of the best known computational models of visual attention has been described by Itti, Koch, and Niebur (1998). This model is based on the idea of a saliency map that highlights the locations of a scene that stand out from their background. It has been shown that such salient locations attract the gaze of human observers and that they contribute to the attention allocation of humans (Itti, 2003).

Unfortunately, the model of Itti et al. (1998), as well as other state of the art models of visual attention and eye-movements, do not take task dependent information into account. Extensions to this model try to capture some top-down aspects. For example Navalpakkam and Itti (2005) add top-down modulation to the basic model. Top-down modulation refers to the fact that humans are faster to find targets in visual search if they know the target features beforehand. However, this is at best a partial way of capturing task-dependent information.

So far, not a lot of research has been conducted as to how semantically relevant locations influence eye movements. In addition, there is not any visual attention or eye movement model incorporating this type of information

However, experiments confirmed that scene elements which have a meaning for the task are actually examined by viewers. This has been observed on a qualitative basis in the experimental data of Wainwright (2008), and subsequent experiments showed that scene locations with semantic content for the task are prioritized over scene locations which stand out from the background due to their visual features (Evangelista et al. 2010).

The model described in the next section describes how semantically relevant scene locations can be captured for the task of finding human targets.

## 2. Modeling

The eye-movement model described in this work needs a 3-dimensional graphical simulation environment with its underlying geometry as input. This kind of environment is similar to the ones used in first person shooter games, but also in applications with military background which use 3D graphical displays, e.g. the Maneuver Battle Lab (MBL) in Fort Benning, Georgia.

The model that is presented in the following is based on the observation that humans searching for a human

enemy target tend to fixate two types of scene locations. First, locations at which a ground soldier could take cover, such as small walls, and vertical edges such as window or door frames. Second, locations at which a target would blend in well with the environment and would therefore be hard to detect.

The model will capture these two types of locations in a map that highlights the locations with semantic relevance for the search task. Hence, the map is called relevance map.

### 2.1 Relevance Maps

In order to capture this type of semantically relevant information from the simulation environment, which is the basis for the relevance maps of the proposed eye movement model, two applications based on the Delta3D game engine are used. These two applications directly operate on a simulation environment which provides the stimuli or scenes for a human observer as well as the input for the eye-movement model. These two applications are the waypoint explorer application and the intervisibility application. The waypoint explorer application (Darken, 2007a) creates a dense hexagonal waypoint mesh which is used in conjunction with the simulation environment by the intervisibility application in order to create the relevance map.

The waypoint explorer creates the waypoint mesh in the following way. Starting from one or more waypoint seeds, the explorer travels through the simulation environment. It is able to reach every location within the environment which could be reached by a human. Every location, the explorer visits is marked with a waypoint. From any location the explorer reaches it tries to step into six different directions by a given step size. The six directions have a regular angular separation of 60 degrees. Thus the resulting waypoint mesh has a hexagonal structure (see Figure 1). The explorer only performs a step if the desired location can be reached by a human. The application stops when all reachable locations of the simulation environments have been explored. The output of the application is a set of waypoints with its interconnecting links. The model described in this work makes use of the waypoints only.

The set of waypoints and the simulation environment are the input for the second application, the intervisibility application. The output of this program is the so-called pixelbank, which is used to derive the relevance map. For a given observer's viewpoint the application renders a scene, which is an image or a frame of a visual simulation. The image in Figure 2 shows the simulation environment from the given viewpoint. A scene is rendered once for each waypoint visible from the current viewpoint. Each time, a target

figure is placed in standing position at a different waypoint before the rendering takes place.

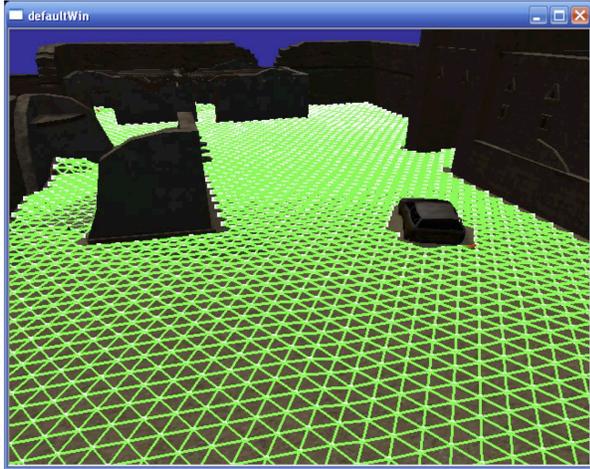


Figure 1: An example of a waypoint mesh laid out in the environment used in this work. The green lines indicate links between waypoints which can be traversed by a person. The waypoints themselves are located at the intersections of the green lines.



Figure 2: A scene of the environment used in this work rendered with the target at one of the waypoints. The waypoints are not displayed.

For this target, visibility information is collected, and for every pixel of the target, an entry is made at the respective pixel coordinate in the pixelbank. The pixelbank is a 3-dimensional data structure where the x- and y-coordinates of the pixelbank are image coordinates, i.e., the horizontal and the vertical position in the rendered image or frame of that scene. The z-coordinate of the pixelbank is a monotonic function of the distance of that portion of the target from the camera.

The visibility information that is computed for each target pixel and stored in the pixelbank includes the fraction of visible pixels (ratio of pixels visible to an observer to the total number of pixels that would be visible if there were no obstructions) and the contrast of the target to its background. The fraction of visible target pixels can be used to determine locations at which a target can hide behind something. If the

fraction of visible pixels is zero, no portion of the target is exposed. If it is one, the target is fully exposed. Any number in between indicates that the target is partially covered. The contrast of the target to its background is a measure of the visibility of a target. High contrasts indicate clearly visible targets and low contrasts indicate targets that blend with the background very well. The contrast computation is performed as defined by Darken (2007b). For each color channel, the target and background ‘intensity’ is computed using the following formulae:

$$R_T = \frac{1}{n_T} \sum_{p \in T} r^2(p)$$

$$G_T = \frac{1}{n_T} \sum_{p \in T} g^2(p)$$

$$B_T = \frac{1}{n_T} \sum_{p \in T} b^2(p)$$

The background ‘intensities’  $R_B$ ,  $G_B$ , and  $B_B$  are computed analogously, where the background comprises all pixels within a rectangle around the target that have a larger scene depth than the target. The rectangle is 5% larger than the smallest rectangle that would include the target completely.

Then, the contrast is computed for each color channel separately:

$$C_R = \frac{|R_T - R_B|}{R_B}$$

$$C_G = \frac{|G_T - G_B|}{G_B}$$

$$C_B = \frac{|B_T - B_B|}{B_B}$$

and the average of the three contrasts is the resulting contrast value:

$$C = \frac{C_R + C_G + C_B}{3}$$

Two maps are computed from the pixelbank. One map, which is based on the fraction of visible pixels, contains the information about hiding locations. The second map, based on the contrast information, indicates locations at which targets blend in well with the environment.

The hiding location map is derived from the pixelbank by taking the minimum fraction of visible pixels from the list at every pixel. This yields a two-dimensional map ranging from 0 to 1. The width and height of this map are the same as the width and the height of the image rendered from the simulation environment. Pixels with small numbers indicate locations at which

at least one target position is occluded and is therefore a likely hiding location. This map is inverted, mapping the range of 0 to 1 to the range of 1 to 0 such that 0 represents a fully exposed target and the numbers close to 1 indicate hiding locations.

Similarly, the contrast map is a two-dimensional map with the same width and height as the hiding location map and the pixelbank. For each  $x$  and  $y$  image position, the minimum contrast is picked from the pixelbank list at this position. The range of pixel values of this map starts at 0 and can be arbitrarily high. In practice, however, the numbers range from 0 to 1 in most cases. Therefore, all values above 1 are set to one and the result is mapped to the range of 1 to 0. Thus, numbers close to 1 represent locations at which the target can blend in well with the environment and numbers close to 0 represent locations at which a target stands out well from the background.

The final relevance map is derived by additively combining the hiding location map and the contrast map. Figure 3 shows an example of a relevance map and Figure 4 illustrates the derivation of the relevance map from the pixelbank.

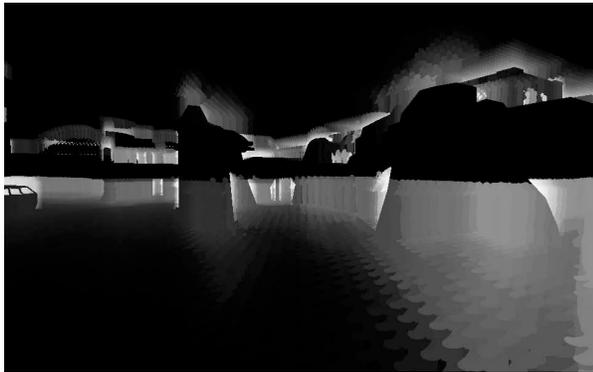


Figure 3: The relevance map for one scene. White pixels indicate the relevant scene locations.

## 2.2 Saliency Map

Since the control of eye-movements does not only depend on task dependent information, but also on visual scene features, the proposed model includes a saliency map in the spirit of Itti et al. (1998) as well. The saliency map used in this work closely follows the implementation of Itti et al. with a few modifications. Similar to the model of Itti et. al. this model considers three basic features: intensity, color and orientation. The details of the saliency map computation have been described in Itti et. al (1998) and therefore only the changes to the saliency map computation will be described here. These changes pertain to the computation of the intensity channel, to the computation of the color center-surround maps and to the normalization scheme used.

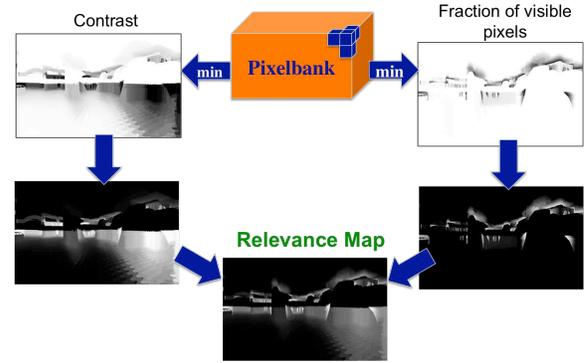


Figure 4: Derivation of the relevance map from the pixelbank.

The computation of the intensity channel uses the ITU-R 601-2 luma transform to convert the RGB-color values of each pixel into one intensity value.

$$I = 0.299 \cdot r + 0.587 \cdot g + 0.114 \cdot b$$

This transform takes the different luminance perception of various colors into account.

The implementation of the saliency map proposed here follows the suggestion of Frintrop (2006). Instead of using two center-surround channels, four color center-surround maps, one for each color, are used. The computation used to create the basic color feature maps is still as defined by Itti et al. (1998).

$$R = r - \frac{g + b}{2}$$

$$G = g - \frac{r + b}{2}$$

$$B = b - \frac{r + g}{2}$$

$$Y = \frac{r + g}{2} - \frac{|r - g|}{2} + b$$

The center surround differences are then computed on six different spatial scales for each color.

$$R(f, c) = |R(f) \ominus R(c)|$$

$$G(f, c) = |G(f) \ominus G(c)|$$

$$B(f, c) = |B(f) \ominus B(c)|$$

$$Y(f, c) = |Y(f) \ominus Y(c)|$$

Where  $f$  refers to the fine scale and  $c = f + \delta$  to the coarse scale and  $f \in \{2, 3, 4\}, \delta \in \{3, 4\}$ . The operator  $\ominus$  denotes the across scale difference as defined by Itti et al. (1998). This means that two maps of a Gaussian pyramid are subtracted from each other. Layer 0 of the pyramid is the original image and the subsequent layers are numbered in ascending order. Before subtraction the coarser map is interpolated to the scale of the finer map.

For every spatial scale, the center surround maps are added up across colors yielding one center surround color map for each spatial scale. These maps are downsampled to scale 4 and added up resulting in the final color conspicuity map. This map is subsequently fused with the intensity and orientation conspicuity maps as defined in Itti et al. (1998).

The original bottom-up salience model uses a normalization scheme which is applied to all center-surround maps before being fused into the conspicuity maps of their respective channel. The same normalization is applied to all conspicuity maps before they are combined into the final salience map (Itti et al., 1998). The motivation for normalization is to account for the different dynamic ranges of different modalities and to avoid having locations which are salient in several maps but nonetheless suppressed due to noise in other maps. Different normalization methods were proposed, but none of them are very convincing (Frintrop, 2006; Itti & Koch, 2001b; Itti et al., 1998). Therefore, an alternate approach is used to take care of the different dynamic ranges. At first, after basic feature extraction, i.e. after creating the intensity map and the four initial color maps, the maps are scaled from 0 to 1 based on the knowledge that the raw color values range from 0 to 255. Then, each time an operation is applied to a map or several maps are fused, the range of the output is determined by considering the possible range of the input maps and the range the resulting maps could have, based on the applied operator. Next, based on this information the intermediate map is scaled to the range of 0 to 1. If, for example, two maps with minimum values of 0 and maximum values of 1 are added to each other, then the values in the resulting map can range from 0 to 2. This resulting map is then scaled to the range of 0 to 1 again by dividing by 2. The scaling does not depend on the actual values in the map, but on the possible minimum and maximum values a map could have based on the operations performed on the input map up to this point. This ensures, that the ranges of all intermediate maps are confined to the range of 0 to 1, and the final salience map will be in the range of 0 to 1 as well. This mechanism not only ensures that all input maps contribute with equal strength, but also that final salience maps can be compared between images. A map with a green dot on a red background, for example, should have a different salience value at the location of the green dot than a red dot on a background with a slightly different shade of red.

### 3. Assessing the Model.

In order to assess the quality of the relevance and salience map they will now be compared to eye-tracking data captured from human observers looking for human enemy targets. The data was collected from

participants viewing realistic scenes containing one to four targets. These scenes were used to derive the relevance maps as well.

The baseline for assessing the quality of the models are the saliency maps of the Visual Attention model of Itti et al. (1998).

#### 3.1 Eye Movement Experiment

In order to derive fixations of human observers looking for a human enemy target an eye-tracking experiment was conducted. The detailed setup of the experiment was described by Evangelista et al. (2010).

The stimuli presented in this experiment were designed as scenes a ground soldier could possibly encounter in an urban environment. The targets in the scenes were enemy soldiers in camouflage uniform hiding in structures, behind walls, or other objects in the scene. Enemy soldiers could also be present in open areas. Each scene contained one to four targets. The targets used were the same as in the previous experiment, but they could appear in four different postures: standing, kneeling, crouching or prone. Sixteen scenes were presented for a maximum of fifteen seconds each. Although a maximum of four targets were present in each scene, participants were told that there could be one to six targets in order to avoid search termination based on the number of targets found. Also, the instructions stressed that it was important to find all targets by pointing out that missed targets could be of continuous danger in future. Each scene was displayed for a maximum of 15 seconds or until the participant announced "next" to indicate that all targets were found.

In order to compare the participant's fixations with the salience and relevance maps, fixations on one scene over all participants are fused into one fixation map per scene. The fixation maps have the same width and height as the stimuli presented: 1920x1200 pixels. The fixation maps are binary maps containing either values of 0 or 1. Each location of the fixation map for which a fixation was recorded is set to 1. All other pixels of the fixation map are set to 0. This means that a 1 in the fixation map indicates a fixated location and a 0 indicates a location which was never fixated.

#### 3.2 Comparison

The fixation maps are compared to the salience and relevance maps using the area under the curve (AUC) of a receiver operating characteristic (ROC) curve following Tatler, Baddeley, and Gilchrist (2005) and Einhäuser, Spain, and Perona (2008). Since the AUC is equivalent to a Wilcoxon rank-sum test, it represents the probability with which positive instances can be distinguished from negative instances (Hanley and

McNeil, 1982). This means that the AUC tells how well the salience and relevance maps correctly distinguish between fixations and non-fixations.

The total number of negative instances for one scene are the number of zeros in the fixation maps, which are all the locations that were not fixated by any participant. Conversely, the total number of positive instances for one scene is the number of ones in the fixation map. These are all the locations that were fixated by at least one participant.

The salience maps and the relevance map are treated as predictors of fixations. All values in the map above a certain threshold are taken to indicate that this location will be fixated. All values below that threshold indicate that these locations will not be fixated. The locations which are above that threshold and are marked as fixations in the fixation map are hits based on that threshold. All locations which are above the threshold and not marked as fixations in the fixation map are false positives. This assumption, however, is very conservative, since in reality a fixation covers more than just one pixel. Pixels with values above the threshold that are not fixated but lie in the immediate vicinity of the fixation location, will be counted as false positives and not as hits. As a result, the values of the metric used will be lower than they should be. However, the proposed comparison metric is still appropriate, since the evaluation of the maps is based on a comparison of the values, not their magnitudes.

In order to account for the eye-tracking error of approximately 1 degree of visual angle, the salience and relevance maps are convolved with a Gaussian kernel.

## 4. Results

A total of four maps are compared to the fixation maps of each scene. This yields one AUC per map and per scene, i.e., 16 AUCs for each map. The ROC curves of all maps are depicted in Figure 5. The assessed maps are the bottom-up salience map of the original implementation of the model described in Itti et al. (1998)<sup>1</sup> (referred to as the Itti map from here on); the re-implemented salience map, which follows the specification of the Itti model with the changes as described in section 2.2, the relevance map and an additive combination of the re-implemented salience map and the relevance map called the combined map. This combined salience/relevance map is computed by adding up the two input maps both weighted with 0.5.

In order to be a useful predictor, the AUC of the maps needs to be larger than 0.5. An area of 0.5 would be achieved by random guessing. The average areas under the curve of the Itti map ( $\mu=0.54$ ,  $\sigma=0.04$ ,  $p=0.0007$ ), the salience map ( $\mu=0.69$ ,  $\sigma=0.05$ ,  $p<0.0001$ ), the relevance map ( $\mu=0.72$ ,  $\sigma=0.07$ ,  $p<0.0001$ ) and the combined map ( $\mu=0.74$ ,  $\sigma=0.03$ ,  $p<0.0001$ ) all statistically significantly exceed 0.5. This means that all of them predict eye fixations better than chance. However, it is apparent that there is a large difference between the average AUCs of the four maps. Therefore, the maps are compared to each other in order to see if they differ in their predictive power.

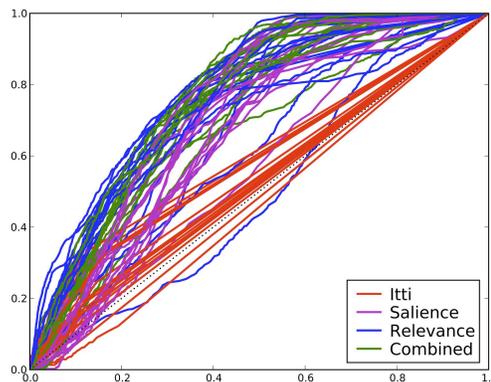


Figure 5: ROC curves of all sixteen scenes and all four predictor maps in one image. It can be clearly seen how the relevance map and the map combining relevance and salience dominate the pure salience maps.

The comparison is performed by counting how often each of the maps has a higher AUC, i.e., the number of scenes in which one map outperforms another. The comparisons are based on a sign test using a significance level of 0.05. Comparing the Itti map with the salience map shows that the Itti map is doing better in no scene, and the salience map is doing better in all 16 scenes. The same result is found for the comparison of the Itti map with the combined relevance and salience map. This difference is statistically significant ( $p<0.0001$ ). As compared to the relevance map, the Itti map is doing better in 1 case and the relevance map in 15 cases. Again, the difference is statistically significant ( $p=0.0003$ ). Clearly, the Itti map is inferior to all other maps. Looking at the salience map, one can see that it predicts eye fixations better than the relevance map on 4 scenes, whereas the relevance map is a better predictor for 12 of the total 16 scenes. A sign test of this ratio shows statistical significance ( $p=0.0262$ ). The salience map is also a worse predictor than the combined relevance and salience map. The proportion here is 1:15, which is significant as well ( $p=0.0003$ ). This means that the salience map performs better than the Itti map only. The other two maps, which both contain information about semantically relevant scene locations, are better predictors of eye

<sup>1</sup> Implementation derived from <http://ilab.usc.edu/toolkit/downloads.shtml>, last accessed 3JAN2010

fixations than the salience map. Finally, the comparison of the relevance map with the combined map shows that each map is doing better than the other for 8 of the 16 scenes. This proportion is obviously not showing a difference of predictive power ( $p=0.5$ ). A summary of these results can be found in Table 1.

	Itti	Salience	Relevance	Combined
Itti		0*	1*	0*
Salience	16*		4*	1*
Relevance	15*	12*		8
Combined	16*	15*	8	

Table 1: Comparison of the prediction performance of all maps with all other maps. Each number indicates the number of scenes in which the AUC was larger for the map of the row as compared to the map of the column. Asterisks indicate statistical significant difference based on a sign test (significance level  $\alpha=0.05$ ).

### 5. Discussion and Conclusions

The most apparent result of the map comparison is that the Itti map, which is the most well-known model of visual attention allocation and eye movements, is outranked by all other maps. This begs the question of whether the stimuli used for this study are special in some way and not representative of actual environments causing the Itti map to do worse than it would on real world stimuli. Previous research of eye movements on real world photographs using the AUC as a metric as well obtained very similar results (Einhäuser et al., 2008). They report that the Itti map predicts fixations above chance ( $AUC > 0.5$ ) in 77 out of 93 scenes, which is 82.8% and an average AUC of  $57.8\% \pm 7.6\%$ . For the scenes in this experiment, the Itti maps predict fixations above chance in 87.5% of all scenes (14 of 16), and the average AUC amounts to  $54.0\% \pm 4.1\%$ . This means that the performance of the Itti maps in the experiment of Einhäuser et al. (2008) is almost exactly the same as the performance observed here.

The most important result of the map comparison is the predictive power the relevance map achieves. The average AUC of the relevance map ( $71.9\% \pm 7.1\%$ ) is larger than the average AUC of the salience map ( $68.9\% \pm 4.8\%$ ), and the relevance map outranks the salience map on a statistically significant number of scenes. This shows very clearly that semantically relevant scene locations are better predictors of eye fixations than visual salience alone. In addition to that, the result shows that the novel approach of using information from the simulation environment to determine the semantically relevant locations is highly effective.

An even better predictor than the relevance map alone is the combined salience and relevance map. This map outperforms the salience map on 15 scenes and reaches an average AUC of  $74.1\% \pm 3.0\%$ . This is the expected result based on the "tier I" experiment described by Evangelista et al. (2010) which showed that both visually salient distractors as well as task-dependent influences affect the eye movements. It is interesting that the combined map does not perform statistically significantly better than the relevance map alone although the average AUC of the combined map is higher than the average AUC of the relevance map.

Looking at the individual scenes more closely reveals that for scenes in which one of the constituent maps has poor performance, the combined map will perform worse than the best constituent map. In cases in which the performance of both maps is rather good, the combined performance increases. Since the salience map is doing worse than the relevance map for most of the scenes, the salience map can reduce the performance of the combined map as compared to the relevance map alone. In contrast, the contribution of the relevance map to the salience map in the combined map improves performance as compared to the salience map alone.

In other words, there are scenes for which the visual scene features are the governing factor. In this case the salience map predicts fixations better than any of the other two maps. Then, there are scenes for which the task influence is the governing factor and the relevance map is the best predictor. Lastly, there are scenes, where both visual features and relevant scene information play a significant role, which yields better performance of the combined map than any of the individual maps. The results indicate that in the minority of the scenes, the bottom-up information is the governing factor. In this experiment, there is only 1 of 16 scenes for which the visual information governs the eye movement. This highlights the importance of the semantically relevant scene location over visually salient locations.

In summary, it becomes evident from this research effort that the most influential factor for the prediction of eye fixations is the set of semantically relevant scene locations. In addition, this model presented in this work employs a novel method which allows the direct extraction of semantically relevant information from a simulation environment. This information is fused into the relevance map, which has very good prediction performance.

### 6. Future Work

The model described here does not include any knowledge about target features. Previously, Pomplun (2006) has shown that image locations that contain

target features receive a higher proportion of eye-fixations than locations which do not. Therefore, it would be interesting to include such a mechanism to see how this changes the prediction performance of the model.

Furthermore, it would be very interesting to explore additional inputs for the creation of the relevance map. At the moment, the relevance map is based on the fraction of visible target pixels and on the contrast of the target to the background. For the contrast input, the size of the target is currently neglected. However, it is not hard to conceive that blending in with the environment is not just a function of contrast, but is also modulated by target size. For example, it would be interesting to explore how a relevance map including the influence 'contrast  $\times$  target size' might be constructed, and how the prediction performance of such a map would compare to the currently used maps.

So far, the model has only been assessed with respect to fixation densities. The next step would be to examine fixation order and its relationship to saliency and relevance maps.

Finally, the model could be extended to not only predict fixations but also to predict target detection probabilities and generate false positives. First of all, it is apparent, that targets which never receive a single fixation will have a detection probability of zero. Furthermore, false positive detections should occur only where a fixation occurred. In addition, the results of the eye-tracking experiment contain false positive predictions. This information can be further analyzed to learn which factors influence false positive generations and detection probabilities.

## 7. References

Darken, C. (2007a). Level Annotation and Test by Autonomous Exploration. *Proceedings of Artificial Intelligence and Interactive Digital Entertainment (AIIDE)*.

Darken, C. (2007b). Computer graphics-based models of target detection: Algorithms, comparison to human performance, and failure modes. *The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology*, 4.

Einhäuser, W., Spain, M., & Perona, P. (2008). Objects predict fixations better than early saliency. *Journal of Vision*, 8, 1-26.

Evangelista, P., Darken, C., & Jungkunz, P. (2010). Modeling and Integration of Situational Awareness and Soldier Target Search. Invited submission to Barchi Prize competition for the 77th MORS, 2010.

Frintrop, S. (2006). *Vocus: A visual attention system for object detection and goal-directed search* (J. G. Carbonell & J. Siekmann, Eds.). Springer.

Hanley, J. A., & McNeil, B. J. (1982). The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*, 143, 29-36.

Henderson, J. (2003). Human gaze control during real-world scene perception. *Trends in Cognitive Sciences*, 7, 498-504.

Itti. (2003). Visual attention. In M. A. Arbib (Ed.), *The handbook of brain theory and neural networks* (p. 1196-1201). MIT Press.

Itti, L., & Koch, C. (2001a). Computational modeling of visual attention. *Nature Reviews Neuroscience*, 2(3), 194-203.

Itti, L., & Koch, C. (2001b). Feature combination strategies for saliency-based visual attention systems. *Journal of Electronic Imaging*, 10, 161-169.

Itti, L., Koch, C., & Niebur, E. (1998). A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11), 1254-1259.

Rayner, K., & Pollatsek, A. (1992). Eye movements and scene perception. *Canadian Journal of Psychology*, 46, 342-376.

Tatler, B. W., Baddeley, R. J., & Glichrist, I. D. (2005). Visual correlates of fixation selection: effects of scale and time. *Vision Research*, 45, 643-659.

Wainwright, R. K. (2008). Look again: an investigation of false positive detections in combat models. *Unpublished master's thesis, Naval Postgraduate School*.

## Acknowledgements

The authors thank Paul Evangelista for lots of fruitful discussions on this work and for his support in conducting the eye-tracking experiment. This research has been partially funded by a grant from the U.S. Army TRADOC Analysis Center Monterey, CA.

## Author Biographies

**PATRICK JUNGKUNZ** is an officer in the CIS Section of the German Naval Office. In his free time he is conducting research on cognitive and behavioral modeling for simulations. He received his Ph.D. from Naval Postgraduate School in 2009.

**CHRISTIAN DARKEN** is an Associate Professor of Computer Science at the NPS, and an academic faculty member of the MOVES Institute. His research focus is on cognitive and behavioral modeling for simulations. Previously he was Project Manager at Siemens Corporate Research. He was also the AI programmer on the team that built Meridian 59, the first 3D massively-multiplayer game. He received his Ph.D. from Yale University in 1993.

# Rapid Development of Intelligent Agents in First/third-person Training Simulations via Behavior-based Control

George Alexander  
Frederick W. P. Heckel  
G. Michael Youngblood  
D. Hunter Hale  
Nikhil S. Ketkar

University of North Carolina at Charlotte  
9201 University City Blvd.  
Charlotte, NC 28223

gralexan@uncc.edu, fheckel@uncc.edu, youngbld@uncc.edu, dhale@uncc.edu, nketkar@uncc.edu

## Keywords:

behavior-based control, intelligent agents, authoring tools

**ABSTRACT:** *First/third-person training simulations in virtual environments have become increasingly used; however, authoring intelligent virtual agents to populate these environments presents a large authorial burden. Our work focuses on building tools to enable rapid creation of intelligent agents for first/third-person game-like environments that enable users with no programming knowledge to develop interactive agents. This is made possible using an intuitive agent architecture known as behavior-based control combined with a user interface employing natural language-like agent specification and an interactive testing during agent development. We present the results of a study indicating that users with no programming experience can successfully design agents using our tool — defined as creating an agent that would carry out at least 80% of role-specific baseline behaviors — after only minimal training in the interface.*

## 1. Introduction

Lately, first/third-person training simulations have played an increasingly important role in facilitating mission rehearsal, environment familiarization, and cultural awareness (e.g., Hill, 2006). The two most important parts of any first/third-person training simulation are the environment (which consists of the terrain, static objects such as buildings, and interactive objects such as vehicles) and the intelligent agents in the environment (for example, bystanders at a car crash or victims in a building on fire). Typically, developing any new scenario requires building a new environment and new intelligent agents. This process is almost always time critical, as the earlier the scenario is developed, the more it can contribute to training. Hence, it is important to develop intuitive tools which facilitate the rapid development of such scenarios. While there has been a lot of progress in developing tools to model environments, the tools to model intelligent agents lag far behind. Often the design of intelligent agents requires programming, which considerably slows down the process of building scenarios. This paper demonstrates how behavior-based control can facilitate the rapid development of intelligent agents without traditional programming.

Behavior-based control has its roots in robotics and is an extension to the subsumption architecture (Brooks, 1986).

At an abstract level, agents created with behavior-based control consist of one or more prioritized layers of behavior, where each layer maps a combination of percepts to a combination of actions. At any instant in the simulation, the agent receives one or more percepts, activating one or more behavior layers, causing the action(s) associated with those layers to be carried out by the agent. We explain this paradigm more concretely with an illustrative example in Section 3, below.

Our recent work has focused on the use of behavior-based control in first/third-person training simulations, initially reported in (Heckel 2009). Our research framework DASSIES (Dynamic Adaptive Super-Scalable Intelligent Entities) incorporates tools to design agents via behavior-based control (BehaviorShop) and a behavior-based simulation engine (BEHAVEngine) which operates on the agent definitions (produced using BehaviorShop) and implements their behavior in any standard first/third-person game engine. In our own work we use the F13RST (First and 3<sup>rd</sup> Person Realtime Simulation Testbed) for experimentation, but are working with VBS2 and RealWorld as well.

We have evaluated the effectiveness of the behavior-based control paradigm in extensive human trials. Participants of the trial (mostly people not from a computer science background) were provided a text

description of an agent and were asked to design an agent using BehaviorShop. The study involved a total of 13 different baseline agent descriptions, divided into five scenarios, and over a hundred participants. Results indicate that at least 80% of the participants were able to build agents with at least 80% behavioral accuracy (based on compatibility with text descriptions of each scenario). This level of performance met our target benchmark, which was set based on initial promising results in a pilot study conducted with a simpler prototype of BehaviorShop.

These results strongly assert the potential of behavior-based control in designing agents for first/third-person training simulations. Behavior-based control and its implementation in BehaviorShop and the BEHAVEngine represent the state of the art in building agents for first/third-person training simulations, and their adoption will greatly enhance all first/third-person training simulations.

### 1.1 Related work

An AI building tool should take into account three major factors: the manipulation of simple atomic decision units into larger wholes (pixels in images, primitive shapes in 3D modeling), immediate feedback as the character is modified, and abstraction and reuse of existing character models.

The existing work in AI builders address, at most, the first of these factors. Tools have been developed for robotics, including the RobotFlow builder from the University of Sherbrooke (Cote et al. 2004). The base-level units of RobotFlow are low-level (higher-level behaviors are built from networks of *nodes*, input/output units roughly equivalent to programming language functions), allowing a great deal of flexibility when creating new systems. Unfortunately, this level of complexity is daunting for non-expert users.

Sony uses Brian Schwab's Situation editor for building characters in sports games (Schwab 2008). This editor has similar goals to our own, but even the author admits that it is difficult to learn, noting that experienced programmers require at least a week of training. The Eki One Configurator from Artificial Technology is a commercial product aimed at games (Artificial Technology 2009). It provides a more polished FSM editor, but does not solve the problem of transition complexity. Xaitment also produces a set of commercial packages for editing FSMs and knowledge bases, but these tools are not appropriate for AI novices (Xaitment 2009).

FSMs (Finite State Machines) are a common choice for the architectures underlying agent builders. The

commercial package SimBionic is designed for building game AI, and provides a HFSM (Hierarchical Finite State Machine) modeling interface, a debugger, and engine (<http://www.simbionic.com/>). SimBionic is an extension of Fu's BrainFrame software (Fu and Houlette 2002). AI.implant is another commercial package for building simulation AI, and is developed by Presagis (Presagis 2001). The AI.implant tool allows the user to model game agents using a variety of methods, most notably FSMs and HFSMs.

Agent Wizard is a specialized interface for building software agents (Tuchinda and Knoblock 2004). It uses a question-based system, which queries the user to specify various facets of the desired agent. This approach is accessible, but this tool is domain-specific for web software agents rather than game/simulation agents.

Each of these builders uses an artificial agent architecture to instantiate the created agents. Many possible architectures exist, but in game AI, FSMs are very commonly used to drive character AI. While they can be used to quickly build AI, and the basic idea is intuitive, the number of transitions between states can grow to an unmanageable level for complex agents. This can be partially overcome through the use of HFSMs or Behavior Trees, which are also commonly used in games (Fu and Houlette 2004). The hierarchical approach can reduce the complexity of the top level FSM, but are still time-consuming to build.

## 2. Designing a Scenario in a First/third-person Training Simulation

The key goal of designing a scenario in a first/third-person training simulation is to facilitate the training of operatives for a particular situation in a test bed that closely resembles the real world. This allows them to develop expertise with respect to the particular situation (for example, training rescue workers to systematically search a building for victims of a fire). Training scenarios can be extremely diverse in nature, but at an abstract level consist of three key elements, namely, the environment, the intelligent agents, and the human agents. A training simulation in a sense emerges from the interaction of these three elements, and designing a particular scenario involves modeling the environment and building the intelligent agents to mimic a real world situation. This process is best described by considering a specific scenario. Consider, for example, a scenario which focuses on training rescue workers to systematically search a building for victims of a fire. In this case, the terrain and the particular building (with static elements such as walls and interactive elements such as doors and elevators) form the environment. The victims of the fire in different

parts of the building are the intelligent agents. Designing this scenario would thus involve modeling the environment and building the agents, after which this scenario would be ready to be used for training rescue workers.

The current state of the art in building such scenarios includes a diverse array of tools for environment design. Intuitive 3D modeling tools, as well as existing libraries of static and interactive objects, can be leveraged to construct a realistic environment. An important fact to note about building environments is that this process is fundamentally a 3D modeling exercise requiring no programming expertise and, given intuitive tools, can be completed relatively easily although, some artistic talent/training is often required. Furthermore, existing libraries of environments and objects can be easily leveraged. For example, objects such as vehicles need to be designed only once and can be reused for multiple scenarios.

When it comes to building intelligent agents, the situation is far more complicated. There are hardly any tools that match the intuitiveness or maturity of 3D modeling tools, and often, agents need to be built by programming or scripting. Achieving the most trivial behaviors takes a significant amount of time, and agent building remains the most time consuming step of scenario design. Furthermore, it is relatively hard to reuse intelligent agents. For example, in the rescue worker scenario described earlier, we cannot use a single agent to model all victims in the building. Typically, we would want a range of behaviors randomly assigned to different agents. The important fact to note about intelligent agents in training simulations is that, in contrast to building the environment, this has fundamentally been a programming exercise requiring a certain amount of expertise in logic/algorithm construction.

Another important point to note is that, unlike environments that can be designed by 3D modelers based on descriptions, building intelligent agents requires a subtle understanding of the scenario and needs to involve domain experts who often lack the programming skills to achieve this task. For instance, building intelligent agents mimicking bystanders in a foreign country would require a nuanced understanding of the culture, which is hard to describe, and should be built by a domain expert, while the buildings and vehicles can be easily described via standard technical specifications. The lack of intuitive tools for building intelligent agents often requires the domain expert to collaborate with a software developer, complicating and delaying the process of scenario development.

It is thus critical to develop a theoretically sound framework for building intelligent agents to serve as a foundation for designing intuitive tools to address the problem of creating interesting agents in first/third-person training simulations. While this overarching objective is clear, achieving it requires incorporating ideas from two seemingly diverse fields, artificial intelligence (AI) and human-computer interfaces (HCI). The field of AI, to a large extent, has focused on building intelligent agents achieving concrete goals in an optimal manner with little regard to the complexity of defining such agents. HCI, on the other hand, studies the design and implementation of intuitive interfaces that allow the human user to achieve the task at hand with relative ease. The problem at hand requires formulating a framework that balances what the agent can achieve with how complex the agent specification is. Behavior-based control achieves this balance, as discussed in the next section.

### 3. Behavior-based Control

Behavior-based control is an extension to the subsumption architecture (Brooks, 1986) and has its roots in robotics. Agents created with behavior-based control consist of one or more prioritized layers of behavior, with each layer mapping a combination of percepts to a combination of actions. At any instant in the simulation, the agent receives one or more percepts, activating one or more layers and causing the action(s) associated with those layers to be carried out by the agent. Behavior-based control is inherently parallel in the sense that multiple percepts can be received at a single instant of time, which can lead to multiple actions also being performed at a single instant of time. There are two key aspects to behavior-based control, the first being the mapping of percepts to actions (or combinations of percepts to combinations of actions) represented using one or more behavior layers, and the second being the prioritization of the layers, which specifies which layers override other layers in the case where multiple percepts are received. We illustrate these key ideas in behavior-based control using a toy example.

Consider an intelligent agent which mimics a simple organism in an environment with the following two predefined percepts: a) perceive food and b) perceive predator. Furthermore, the agent has the following three predefined actions: a) explore new regions, b) consume food, and c) flee from predator. Given these basic percepts, an agent design using behavior-based control is illustrated in Figure 1. The key points to note are as follows. Firstly, note that each of the layers maps percepts to actions. For example, layer L2 maps the percept *food* onto the action *eat*. Furthermore, note that the layers are prioritized. Layer L1 is overridden by layer L2 which in turn is overridden by layer L3. Also note that layer L1

does not have a percept and corresponds to a default action which is performed when no percepts are received by the agent. This simple agent designed using behavior-based control has the following overall behavior. When there are no percepts available, the L1 layer is triggered, and the agent explores new regions. In the case where the agent perceives food, the L1 layer is overridden by the L2 layer, and the agent consumes the food. In the case where the agent perceives a predator, the L3 layer is triggered, all the layers below it (L1 and L2) are overridden, and the agent flees from the predator. Note that the prioritization of the layers is the most important part of the agent definition.

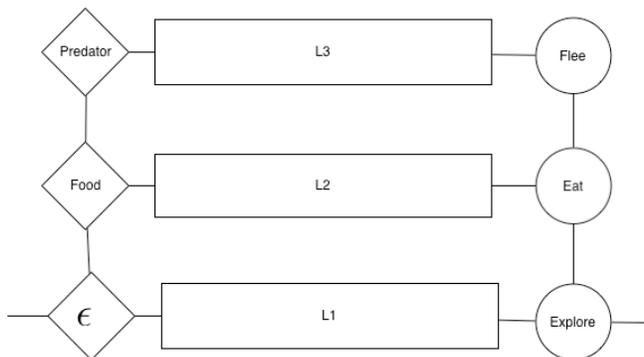


Figure 1. Behavior-based control for a toy agent. The agent has three behavior layers, with associated trigger conditions on the left and resulting actions on the right.

While the toy example discussed here is only for the purposes of illustration, it does demonstrate the key aspects of behavior based control, namely, the mapping of percepts to actions via layers and the prioritization of layers. Given any large set of percepts and actions, any reactive agent of arbitrary complexity can be constructed using behavior-based control. It is now essential to demonstrate how behavior-based control fits in with first/third-person training simulations from a systems perspective, which we discuss in the next section.

Behavior-based control has a number of advantages over the use of other architectures for game agents, such as finite state machines (FSMs), hierarchical finite state machines (HFSMs), and behavior trees. Behavior-based control is inherently parallel, as multiple active behaviors can be run at once by varying the override policy of each layer. The representational complexity of a behavior-based control agent, which is an important consideration for the agent authoring process, is far lower. In the example from Figure 1, the corresponding finite state machine requires more transitions (see Figure 2). If multiple behaviors are allowed to be active at once, the finite state machine becomes increasingly more complex, as each allowable combination of behaviors requires an additional state. HFSMs and behavior trees reduce this

complexity over simple FSMs, but still require more complex models to represent the same agent. The reduced complexity of behavior-based control makes it simpler and faster to create intelligent agents that can embody more complex and expressive intelligence.

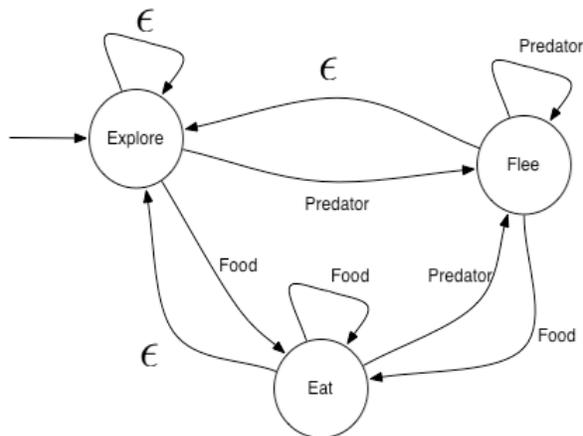


Figure 2. The finite state machine equivalent of our example agent requires only three states but nine transitions, one for each (trigger condition, state) pair.

#### 4. Implementation of Behavior-based Control Using BehaviorShop and BEHAVEngine

DASSIES (Dynamic Adaptive Super-Scalable Intelligent Entities) is our primary research framework, and it includes an industry strength implementation of behavior-based control targeting first/third-person training simulations. The architecture of DASSIES is illustrated in Figure 3.

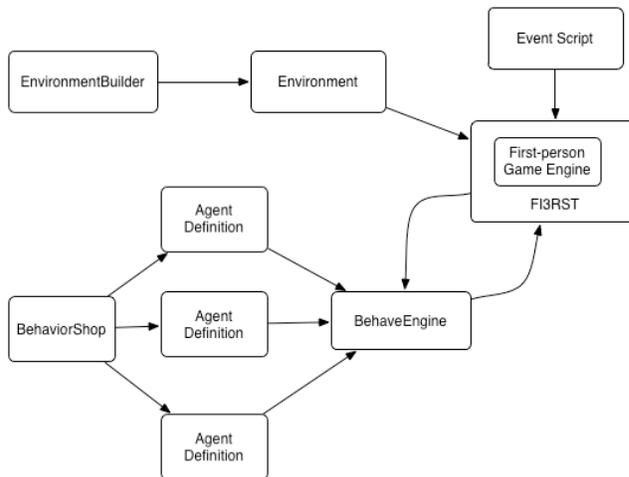


Figure 3. The DASSIES system is composed of four principal components: the agent builder interface (BehaviorShop), the agent engine (BEHAVEngine), a first/third-person simulation environment, and the interface between our agents and the environment (FI3RST).

The key components implementing behavior-based control are BehaviorShop, which is a tool to design intelligent agents, BEHAVEngine, which operates on one or more agent specifications, and any standard game engine to produce a specific simulation. An additional support component is FI3RST (First- and Third-person Realtime Simulation Testbed), which is a wrapper around game engines to provide a standard interface for BEHAVEngine (Currently FI3RST supports the Panda3d ([www.panda3d.org](http://www.panda3d.org)) and Irrlicht ([irrlicht.sourceforge.net](http://irrlicht.sourceforge.net)) game engines, but could be extended to support any standard game engine).

#### 4.1 BehaviorShop

BehaviorShop, the component which allows users to build intelligent agents using behavior-based control, has an intuitive user interface based around using sentence-like constructions to define agents. Screenshots of the startup screen, the layers window (where the user defines the layers), and the trigger-action editor are presented in Figures 4, 7, and 8, respectively.

The layers window, illustrated in Figure 7, can be used to add, delete, and move layers. The layer editing window depicted in Figure 8 can be used to select triggers and behaviors for the layers and define levels of priority. Often, actions performed by the agents require positional parameters. These can be specified by selecting locations on a preloaded map through the map window, illustrated in Figure 5. At any point while designing the agent, the user can test and debug the agent by watching the simulation in the output window, depicted in Figure 6.

Each behavior layer in BehaviorShop is defined by selecting choices to fill out an if-then sentence, possibly with multiple triggers and/or actions. For this reason, the vocabulary presented to the user is very important. The language in our early prototype was based on developer opinion, a practice commonly referred to as armchair design. Of course, the HCI community has long been aware of the difficulties with this approach (Furnas et al. 1987). To bring our interface vocabulary more into line with the vernacular, we conducted a study in which participants were asked to read a brief scenario description and provide free-form text instructions for a selected actor and to watch a short video clip and describe the actions of one of the actors in the scene. From this vocabulary study, we were able to present a more natural syntax in our interface as well as to ensure we included the most commonly used words for describing a scenario.

#### 4.2 BEHAVEngine

Agents defined by BehaviorShop are executed by BEHAVEngine in conjunction with FI3RST and the game

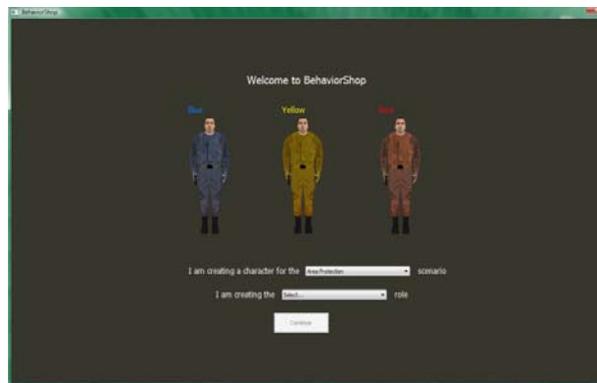


Figure 4. Upon starting BehaviorShop, users are asked to choose a scenario and role to create an agent for.



Figure 5. In order to define triggers and actions that involve location(s), such as guarding or patrolling, users are presented with a 2D top-down view of the simulated world.



Figure 6. Users can interactively test agents in the simulated environment during the agent building process.

engine. BEHAVEngine constantly receives percepts for the agents in the simulation, interprets the agent design, computes the appropriate actions based on the agent design, and passes the action messages on to FI3RST.

These action messages are interpreted by FI3RST and appropriate action animations (for example walking, jumping, and shooting a target) are chosen from a library of basic actions and played in the game engine.

BEHAVEngine is a multi-threaded behavior-based control engine for game agents. In addition to the core intelligence architecture, it integrates navigation using

navigation meshes (McAnils 2008) and a modular perception and action system. Navigation meshes are decompositions of navigable space in the world into convex regions. These enable efficient path planning and information compartmentalization. The perception and action systems make it simple to adapt the engine to different simulation environments.



Figure 7. Behavior layers are constructed individually with the lowest priority layers on the bottom. Layers can be reordered by dragging them to a new location with respect to the other layers.



Figure 8. Each behavior layer is defined by selecting options to fill out an if-then sentence structure, possibly containing multiple trigger conditions (disjunctions or conjunctions) and multiple resulting actions (to occur sequentially or in parallel). By default, each layer overrides every layer below it, but these overrides can be disabled. 192

## 5. Human Trials with Behavior-based Control

The effectiveness of behavior-based control and its implementation in BehaviorShop and BEHAVEngine have been extensively validated using a large-scale human trial. Participants were asked to construct an agent for one of thirteen possible characters based on a written description. Characters ranged in complexity from a simple shopper in a market to a bomb squad technician requiring several complex behaviors. After watching a short instructional video 10 minutes in length, participants built an agent using BehaviorShop. A total of 102 participants submitted an agent they had constructed to be evaluated. These participants were drawn from a random sampling of people, the vast majority of whom had no previous experience creating agents or programming. Agents were evaluated on a ten-point scale by a panel of experts based on how closely they adhered to baseline agents developed for each character. Any borderline agents were loaded into the simulation environment, and their performance was evaluated in the simulation. A score of eight or higher indicated that the agent could successfully perform the assigned task. Based on this scoring metric, 82 successful agents were created (scoring 8 or higher) for a success rate of 80.39% of all agents created. Among the successful agents, the average score was 9.4 out of 10, which indicates a high degree of convergence with one of the baseline agents for a given task.

Feedback from the participants about their experiences with BehaviorShop was recorded using five-point Likert scales, where a rating of one indicates the user strongly disagreed with the statement and a rating of five indicates they strongly agreed. Participants were asked to rate the statement "Creating simulation characters is easy with the DASSIEs Creation Tool"; overall, the users averaged a 3.8, indicating they agreed with the statement and found agents easy to create. Additionally, participants were asked to rate the statement "I understood how to use the tools"; this statement averaged a 3.9 on our Likert scale, which indicates that most of the users did in fact understand BehaviorShop.

## 6. Conclusions

First/third-person simulations are an important part of modern training regimens for complex situations, facilitating mission rehearsal, environment familiarization, and cultural awareness. However, until now, creating complex intelligent agents for these simulations has required similarly complex authoring tools or computer programming knowledge. Behavior-

based control is a new paradigm for modeling these agents in an intuitive manner, without sacrificing the expressive power of more cumbersome formalisms such as finite state machines. Employing BehaviorShop and BEHAVEngine to leverage the power of behavior-based control, users can easily create interesting intelligent agents with complex behaviors, overcoming a major hurdle in the development of first/third-person training simulations.

Our future work includes extending BehaviorShop to incorporate teams of agents to discover whether non-expert users can successfully create teams of cooperative agents in more advanced variations of the scenarios employed in the study discussed here.

## 7. Acknowledgements

This material is based on research sponsored by the US Defense Advanced Research Projects Agency (DARPA). The US Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of DARPA or the US Government.

## 8. References

- Artificial Technology (2009). Eki One. <http://www.eikone.com>.
- Brooks, R. (1986). A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*..2(1), 14-23.
- Cote, C., Letourneau, D., Michaud, F., Valin, J. M., Brosseau, Y., Raievsky, C., Lemay, M., and Tran, V. (2004). Code reusability tools for programming mobile robots. In *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2004)*, vol. 2, 1820-1825.
- Fu, D., and Houlette, R. (2004). The ultimate guide to FSMs in games. In (Ed.), *AI Game Programming Wisdom 2*.pp.283-302. Boston: Charles River Media.
- Fu, D., and Houlette, R. (2002). Putting AI in entertainment: an AI authoring tool for simulation and games. *IEEE Intelligent Systems*, 17(4), 81-84.
- Furnas, G. W., Landauer, T. K., Gomez, L.M., and Dumais, S.T. (1987). The vocabulary problem in human-system communication. *Commun. ACM*, 30(11), 964-971.

- Heckel, F. W. P., Youngblood, G. M., and Hale, D. H. (2009). BehaviorShop: an intuitive interface for interactive character design. In *Proceedings of the Fifth International Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE 2009)*.
- Hill, R. W., Belanich, J., Lane, H. C., Core, M., Dixon, M., Forbell, E., Kim, J., and Hart, J. (2006). Pedagogically structured game-based training: development of the elect bilat simulation. In *Proceedings of the 25<sup>th</sup> Army Science Conference (ASC 2006)*
- McAnils, C. and Stewart, J. (2008). Intrinsic detail in navigation mesh generation. In Rabin, S. (Ed.), *AI Game Programming Wisdom 4*, (pp. 95-112). Boston: Charles River Media.
- Presagis. (2001). Ai.implant.  
<http://www.presagis.com/products/simulation/details/aiimplant/>. November 23, 2008.
- Schwab, B. (2008). Implementation walkthrough of a homegrown “abstract state machine” style system in a commercial sports game. In *Proceedings of the Fourth Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE 2008)*, 145–148.
- Tuchinda, R., and Knoblock, C. A. (2004). Agent wizard: building information agents by answering questions. In *Proceedings of the 9th International Conference on Intelligent User Interfaces (IUI 2004)*, 340-342. New York, NY, USA: ACM.
- Xaitment. 2009. xaitMove and xaitKnow.  
<http://www.xaitment.com>.

## Author Biographies

**GEORGE ALEXANDER** is a doctoral student in the Game Intelligence Group at the University of North Carolina at Charlotte. He received his B.S. in Mathematics and Computer Science, summa cum laude, from UNC Charlotte in 2006. His current work involves improving the BehaviorShop interface to improve usability and reduce user confusion.

**FREDERICK W. P. HECKEL** is a doctoral student in Computer Science at the University of North Carolina at Charlotte. He received his BA in Computer Science & Political Science from Swarthmore College in 2005, and his MS in Computer Science from Washington University in St. Louis in 2008. His current research focus is in reactive control for agents in games, including evaluation of different reactive architectures and methods to reduce agent complexity.

**G. MICHAEL YOUNGBLOOD, Ph.D.** is an Assistant Professor of Computer Science at the University of North Carolina at Charlotte and head of the Game Intelligence Group. He received his Ph.D. in Computer Science and Engineering from the University of Texas at Arlington in 2005. His research interests include interactive AI, game knowledge and information structures, and machine and human learning in games.

**D. HUNTER HALE** is a doctoral student at the University of North Carolina at Charlotte, working in the Game Intelligence Group. He earned his MS in Computer Science from UNC Charlotte in 2008. His research interests include spatial representations for artificial intelligence, computational geometry, graphics and rendering, and artificial intelligence for games and simulation.

**NIKHIL S. KETKAR, Ph.D.** is a Postdoctoral Researcher in the Game Intelligence Group at the University of North Carolina at Charlotte. He received his Ph.D. in Computer Science from Washington State University in 2009. His research interests include machine learning, data mining, and graph theory.

## Shifts of Critical Personnel in Network Centric Organizations

*Craig Schreiber*  
Lenoir-Rhyne University  
Hickory, NC 21580  
828-328-7793  
[Craig.schreiber@lr.edu](mailto:Craig.schreiber@lr.edu)

*Kathleen M. Carley*  
Carnegie Mellon University  
Pittsburgh, PA 15213  
412-268-6016  
[Kathleen.carley@cmu.edu](mailto:Kathleen.carley@cmu.edu)

*Jeffrey T. Hansberger*  
Army Research Laboratory  
Aberdeen, MD 21005  
757-203-3431  
[Jeff.hansberger@us.army.mil](mailto:Jeff.hansberger@us.army.mil)

### Keywords:

Computational modeling, social networks, dynamic network analysis, leadership, network centric organizations

**ABSTRACT:** *Identifying critical personnel has been a problem of interest for sometime as organizations seek to optimize their advantage and disrupt their adversary. This problem has become more difficult with the increasing use of network centric organizations as these organizations have flexible structures that can produce significant shifts of critical personnel. A shift of critical personnel is a change of who is critical within an organization over time. Traditional social network analysis has identified critical personnel using measures applied to static structure. This research adds the process of network change to better understand when shifts of critical personnel may occur. Theory and application are discussed.*

## 1. Introduction

### 1.1 Network Centric Organizations: Organizational Design to Match Change

The world has changed drastically in the last decade. From a military perspective, current operations are characterized by rapidly changing and uncertain conditions. Not only has the nature of warfare changed through the use of advanced weaponry and the tactics of terrorism but the U.S. military is increasingly involved in peacekeeping and humanitarian aid responsibilities. In addition, joint and coalition operations are progressively employed to combat terrorism and to perform the various non-combat responsibilities. These joint and coalition operations provide for interagency cooperation leading to shared intelligence and joint tactical operations – capabilities that are considered essential for quick and effective terrorism response. Military organizations must be highly adaptable in order to quickly and effectively shift between warfighting, peacekeeping and humanitarian requirements.

Military organizations have increasingly employed network forms of organizational design in light of the changing and uncertain operating conditions that have fueled the need for learning, adaptability and resiliency (Powell, 1990; Ronfeldt & Arquilla, 2001). Network centric organizations are characterized by flexibility (Nohria & Eccles, 1992), decentralization (Arquilla & Ronfeldt, 2001), differentiation (Baker, 1992), diversity (Ibarra, 1992), lateral cross-functional ties (Baker, 1992) and redundancy (Ronfeldt & Arquilla, 2001). Thus these organizational forms offer many advantages in high velocity environments. Advantages include communication speed and richness (Powell, 1990), knowledge transfer (Podolny & Page, 1998), reduction of uncertainty (Powell, 1990), cross-functional collaboration (Baker, 1992), greater collective action (Powell, 1990) and quick and effective decision-making (Kanter & Eccles, 1992). As Kanter and Eccles (1992) point out, networks are contexts for action. The actions of a network centric organization lead to a dynamic,

evolutionary structure. The network is flexible, ever-changing and hopefully responsive to the environment.<sup>1</sup>

## 1.2 Identification of Critical Personnel in Network Centric Organizations: Shifts of Criticality

Identifying critical personnel in organizations is a problem that has engendered the interest of practitioners and social network researchers for years. Solutions to the identification problem can be applied both to an organization and its adversary. Internal to an organization, solutions have implications such as sustaining or increasing performance and protecting against risk. Externally, solutions have implications such as destabilizing the enemy and decreasing the adversary's performance.

A shift of critical personnel is a change of who is critical within an organization over time. Shifts of critical personnel are adaptive and resilient responses in the face of change. Such realignment of roles and responsibilities may promote learning within the organization as the internal coordination among members brings together varying expertise and knowledge to deal with the dynamic challenges. Shifts of critical personnel can impact the potential learning, adaptability and resiliency of the organization and it is important to identify who is important when or under what conditions so that opportunities and risks can be managed.

As previously noted, network forms of organizing have been increasingly used in high velocity environments. This is mainly due to other organizational forms, such as hierarchies, struggling to perform in the same environment (Powell, 1990; Ronfeldt & Arquilla, 2001). The usefulness of network centric organizations in highly volatile and uncertain environments – namely the ability to enhance learning, adaptation and resiliency – also creates interesting problems in the identification of critical personnel and in the leadership of such organizations. Particularly, the difficulty lies in the fact that learning, adaptation and resiliency are all dynamic, evolutionary capabilities. With changing environmental conditions and changing organizational structure, critical personnel are now moving targets as shifts may occur more frequently. In other words, the identification of critical personnel in network centric organizations is not a static problem but an evolutionary one. For example, organizational structures in the Cold War Era were more stable and identification of important people or leaders in the Russian hierarchy was a relatively stable phenomena.

---

<sup>1</sup> Although the author recognizes that organizational action also contains feedback to the environment and contributes to changes there as well, it is not the focus this research and lies outside the bounds of this study.

Now, terrorist organizations are a very adaptable, resilient enemy and identifying critical people or leaders is a much trickier, on-going problem. Shifts of critical personnel in a network centric organization is an important evolutionary problem to understand.

## 1.3 Shifts of Critical Personnel: Prior Work and Current Focus

Traditional social network analysis has identified critical personnel through the static examination of organizational structure (Bonacich, 1987; Krackhardt, 1987; Brass, 1984; Blau & Alba, 1982; Freeman, 1979)<sup>2</sup>. Although these studies provide meaningful insight to identifying critical personnel at a particular point in time, the cross-sectional nature of the data precludes any attempt to understand and identify shifts of critical personnel over time, especially as the environmental setting and operational conditions change. This only provides limited insight into the process of network change and the nature of network centric organizations. Therefore, we are interested in how a range of operating conditions affect shifts of critical personnel within an organization.

These shifts, as apparent, are evolutionary and require dynamic, longitudinal methods of analysis. Therefore, process needs to be accounted for in the methodology and added to social network theory (Carley, 2003; Kanter & Eccles, 1992). This work takes a serious view of this need and incorporates process in both methodology and theory. The decision to take this route was not only influenced by the academic need for such but also because leaders have a real need for process in the practical application of network research (Kanter & Eccles, 1992).

## 2. Modeling Shifts of Critical Personnel: Operating Conditions, Stressors and Change

Change and uncertainty create stress on an organization. Stress is something that all organizations face (Perrow, 1999). The variety and strength of stressors induce a range of operating conditions which confront the organization and it is reasonable to conjecture that operating conditions affect shifts of critical personnel. More specifically, low stress operating condition may result in fewer shifts whereas high stress operating

---

<sup>2</sup> There are a few studies that have analyzed networks and critical personnel change over time (Sampson, 1968; Burkhardt & Brass, 1990; Carley, 2003; Johnson, Boster, & Palinkas, 2003). But these and the other studies looking at shifts of critical personnel only study the effect of one factor. The partiality of results makes it difficult to develop an overall theory.

conditions may result in many shifts. Accordingly, it is meaningful to understand the evolution of critical personnel shifts across the range of operating conditions.

Lin and Carley (2003) describe three general types of stress that organizations face: external stress, internal stress and time pressure. External stress originates from the external environment. An environment with rapid change and uncertainty is an example of external stress. Network centric organizations are used in these environments and are considered an advantageous design for dealing with external stress. Internal stress originates from malfunctions in organizational operating conditions. Examples of internal stress are communication barriers, turnover and agent unavailability. This forces sub-optimal conditions for communication and learning within an organization. Time pressure constrains rationality. Under time pressure, organizations may communicate and learn based on limited knowledge. This also forces sub-optimal conditions for communication and learning in organizations. These three stressors can all be simultaneously present in the organization to varying degrees at a given point in time (Lin & Carley, 2003).

Following the work of Lin and Carley, we modeled each type of stress as well as the simultaneity of stressors to represent a range of operating conditions. Stressors were modeled at the organizational level and equally affect each agent concurrently within the virtual experiments. The organizational level is the level of interest for this particular study. Individual differences in reactions to stress would represent stress at the individual level and it is assumed that such individual differences would wash-out at the organizational level<sup>3</sup>.

## 2.1. Construct

Each of the stressors were modeled in Construct. Construct is a multi-agent network model for the co-evolution of the socio-cultural environment (Carley, 1990, 1991, 1999; Schreiber & Carley., 2004a; Schreiber & Carley, 2004b; Schreiber, Singh & Carley, 2004, Hirshman, Carley & Kowalchuck 2007a; Hirshman, Carley & Kowalchuck 2007b; Hirshman, Martin & Carley, 2008). In the model, agents go through an active, adaptive cycle where they choose interaction partners, communicate, learn knowledge, change their beliefs about the world, and adapt their networks based on their updated understanding. Knowledge network data is input into Construct to initialize the model with a real-world representation of an organization. The knowledge network is 'who knows what' in the organization and

knowledge is defined into categories that are relevant to that particular organization. For detailed description of Construct see the above referenced publications.

External stress was modeled as a dynamic task environment whereas the knowledge an organization needs to learn changes at varying rates. In Construct, the external environment represents the task environment of the organization. The agents interacted with the external environment and learned bits of task-related knowledge. The agents then interacted with each other and engaged in task-related communication. Change in the environment occurred by changing the value of the knowledge bits. Agents then had to learn about the change in order to maintain or improve organizational learning. The rate of change in the task environment was probabilistic and occurred at random. For example, when the rate of change was 25% then each knowledge bit had a 25% probability of being changed each timeperiod. A random roll of the dice determined if a particular knowledge bit was changed. The rate of change in the external environment indicated the level of stress. For example, the higher the rate of change the higher the external stress faced by the organization.

Internal stress was modeled as intermittent availability whereas agents are unavailable for interaction and subsequently task-related communications are constrained. The percentage of unavailability indicated the level of stress. For example, the higher the percentage of unavailability the higher the internal stress of the organization. Again, this stressor was modeled at the organizational level and affects each agent concurrently

Time pressure was modeled using an information processing approach based on selective attention. The following reasoning was applied. Stress causes a rise in arousal (Eysenck, 1967) which then causes selective attention of knowledge (Easterbrook, 1959; Matthews, Davies, Westerman, & Stammers, 2000). Selective attention narrows the amount of knowledge that is considered when communicating. Therefore, learning under the influence of time-pressure is cognitively constrained. This approach is consistent with organizational theorists in that individual stress is the enemy of rationality (Simon, 1947) and reduces the search for alternatives (Staw, Sandelands, & Dutton, 1981). In Construct, agents under time pressure only consider a portion of the overall knowledge they possess when communicating. The portion of knowledge was determined by 1 minus the selective attention effect. In other words, if an agent knows 10 bits of knowledge and they have a selective attention of 20% then the agent only considers 80% or 8 bits of their knowledge when selecting a bit to communicate. A random role of the dice determined the knowledge bits which were selected for

<sup>3</sup> Individual level stress could not be modeled even if this were a level of interest because this data was not available to collect from the real-world organization.

consideration. The level of selective attention indicated the level of stress. For example, the higher the level of selective attention the higher the time pressure and cognitive constraint on the knowledge considered for communications.

The model was tested to ensure that the stressors were working correctly. Each organizational stressor decreased organizational learning significantly. Higher levels of stress within each stressor significantly decreased performance as compared to the next lower stress level. And the effects of the stressors were comparable to each other. Confidence interval tests were used to test for significant effects.

### 3. Methodology

#### 3.1 Data

The network centric organization under study was the Battle Command Group. The Battle Command Group is comprised of decentralized, distributed and highly interdependent units performing joint and coalition operations. The particular organization studied consisted of one-hundred and fifty-six people. Data collection occurred during the beginning phases of a wargame exercise and Cross-sectional data was collected on the communication and the task networks of the organization. The task network consisted of fifty-one task nodes and was used as a proxy for the knowledge network in Construct. The task network is an appropriate proxy for the knowledge network because these tasks are actually written products which relay information about the operational environment. Examples of task products include maneuver estimates, intel synchronization plans and support orders. In addition, the task network representation produced initial agent interactions in Construct that were validated against the actual communication network of the organization (Schreiber & Carley, 2007).

#### 3.2 Experimental Design

Table 1 presents the experimental design for the shifts of critical personnel virtual experiment. The network was evolved over 250 timeperiods and each result was obtained using a Monte Carlo technique 25 times.

Variable	Description	Values
Organization	Organizational Model	Battle Command Group
Dynamic Environment	External Stress	No change 25% rate of change 50% rate of change 75% rate of change
Intermittent Availability	Internal Stress	Always available 25% unavailability 50% unavailability 75% unavailability
Selective Attention	Time-pressure	No constraint 25% selective constraint 50% selective constraint 75% selective constraint

Table 1: Experimental Design for Shifts of Critical Personnel

The focus for this virtual experiment was on the outcome of structural change in terms of critical personnel. Agent interaction patterns produced by Construct were averaged over the Monte Carlo runs and analyzed to determine which agents were critical. The agent interaction patterns correspond to organizational communication networks and as noted before, the initial agent interactions in Construct were significantly similar to the real communication networks. Therefore the set of critical agents in Construct at timeperiod 0 represent the initial set of critical personnel in the organization before changes and adaptations occur.

Agent criticality was determined by two factors – social network measures of centrality and measure ranking. Centrality was selected because this family of measures is most commonly used for identifying critical personnel in communication networks. The following centrality measures were calculated: betweenness, eigenvector, information and total degree. It is customary for these measures to be correlated and a correlation analysis verified that this was the case. Therefore, only one measure was used to represent criticality – eigenvector centrality. Eigenvector centrality was selected because it had the highest level of significance among all the correlations but any of the measures would serve the purpose.

The second factor in determining agent criticality was measure ranking. The top five agents in terms of highest centrality value were defined as critical. These five agents make up the critical set for each timeperiod. The decision to use five was basically arbitrary as there is no a-priori basis for determining how many agents within a

measure are considered critical. Five was chosen because it has been commonly used in the applied work I have done within organizations.

Two types of change in criticality are measured and analyzed, total change and unique change. Total change measures the number of changes that occur to the composition of the critical set over time. This measure was calculated as follows. The critical sets for each adjacent comparison timeperiod were contrasted and a change was recorded for each difference between the sets. For instance, if the sets of agents being compared were {1,2,3,4,5} and {3,4,5,6,7} then two changes would be recorded as there are two differences between the sets. The total number of changes across all comparisons equaled the number of total changes. One note - this measure accounts for the situation when an agent was in the critical set, fell out of the critical set, and is now back in the critical set. It counts this as a change.

Unique change measures the number of times a new agent enters into the critical set. A new agent is defined as someone who has not previously been in the critical set. This measure was calculated as follows. The critical sets for each comparison timeperiod were joined to make one union set. The difference between the number of agents that comprise the union set and five (the maximum number of critical agents per timeperiod) equaled the number of unique changes.

Both types of change were measured and analyzed to see if operating conditions affected them differently. For instance, it would be reasonable to presume that many different operating conditions induce high amounts of total change but only a few induce high amounts of unique change. Unique change would be particularly interesting to explore as there are many more agents assuming critical roles and this could have important organizational implications.

Comparative analysis for calculating the total change and unique change measures occurred between timeperiods 0, 50, 100, 150, 200 and 250. The Battle Command Group knowledge network had enough fidelity such that structural changes in Construct needed to evolve over several timeperiods. The above timeperiods were chosen because they allowed enough duration for change to occur between comparisons and because they provided even spacing for calculating change.

The purpose of this study was to build theory about the effects that various operating conditions, as represented by stressors and stress levels, have upon changes in critical personnel. It was previously determined that there were a sufficient number of runs within the virtual experiment to gain significance and obtain a good

estimate of the stressor effects. Therefore, the next step in the analysis was to determine the direction and strength of the relationship between the stressors and structural change. To make this determination, the main effects of the stressors were plotted and multiple regression was performed. The standardized beta coefficients from the multiple regression analysis were used to assess the relative impact of the stressors. These analyses were completed for both total change and unique change.

#### 4. Results and Discussion

The Battle Command Group experiments resulted in a range of 1–9 for total change and a range of 1-6 for unique change. Figure 1 shows the Battle Command Group main interaction plots for both total change and unique change based on data means. Several things are notable. First, the dynamic environment lead to more shifts of critical personnel when there were moderate or high rates of environmental change. Second, intermittent availability increasingly constrained the shifts of critical personnel as the stress level went up. Third, selective attention reduced the shifts of critical personnel but levels of stress beyond 25% had less of an effect. The low average knowledge per agent in the Battle Command Group, which is due to data being collected at the beginning of the exercise and limited scenario training for the participants, explains the plateaus.

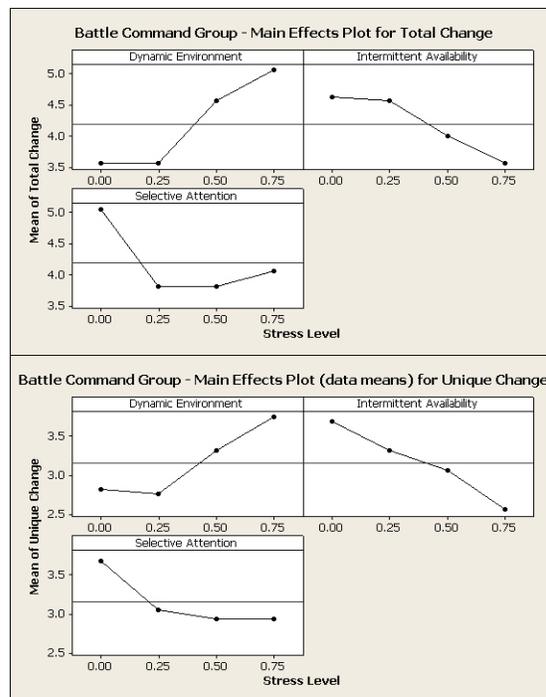


Figure 1: Main Effect Plots for Total Change and Unique Change

For the dynamic environment condition, the 25% rate of environmental change does not increase shifts of critical personnel over the static environment. The low average knowledge in the organization meant that expertise was just forming. As the agents learned and began to gain expertise then considerable shifts of critical personnel occurred, even in the baseline condition. The 25% rate of environmental change was not enough change to induce greater shifts of critical personnel over the baseline. It took higher rates of change to do that.

For the selective attention condition, increased stress levels did not further moderate shifts of critical personnel. The lack of training already resulted in low and constrained overall knowledge. Additional cognitive constraint beyond the 25% stress condition had little effect because of this.

Table 2 presents the results of separate multiple regression analyses for total change and unique change. These results show that intermittent availability had a stronger impact on constraining both types of change as compared to selective attention. These results also show that the dynamic environment again had a stronger impact on total change relative to the other stressors. But this is not the case for unique change as the dynamic environment had a similar strength of impact to that of intermittent availability.

Battle Command Group Total Change		Battle Command Group Unique Change	
Regression Coefficients		Regression Coefficients	
Variable	Standardized Coefficient	Variable	Standardized Coefficient
Dynamic Environment	0.333	Dynamic Environment	0.311
Intermittent Availability	-0.227	Intermittent Availability	-0.334
Selective Attention	-0.182	Selective Attention	-0.219
Model Fit Adj R-Square – 15.6%		Model Fit Adj R-Square – 21.8%	

Table 2: Standardized Coefficients from the Multiple Regression Analyses for Total Change and Unique Change

#### 4.1 Shifts of Critical Personnel – Theory

Theory is proposed about the shifts of critical personnel in network organizations based on the Battle Command Group results. The dynamic environment led to increased shifts of critical personnel as the rate of change in the task intensified. This suggests that re-identification of critical personnel in network organizations should be an on-going activity. A lack of re-identification, especially in volatile conditions, could pose a risk to network organizations. Particularly when strategic decisions such as task assignment, group formation, and personnel retention are made from an offensive perspective or targeting and recruitment are made from a defensive perspective.

The ability of network organizations to exhibit overall structural flexibility in volatile environments is already set in theory. In fact, overall structural flexibility was a key characteristic influencing the use of the network forms by the organization under study. This result builds upon existing theory by proposing that critical personnel substructures also exhibit flexibility during times of change.

*Proposition 1: Shifts of critical personnel are positively related to the rate of environmental change*

*Proposition 2: Shifts of critical personnel can pose a risk to network organizations in dynamic environments when re-identification has not occurred and strategic personnel decisions need to be made*

The results demonstrate a clear negative effect for intermittent availability and selective attention on structural flexibility. (Note: intermittent availability represents communication network constraints and selective attention represents cognitive constraints.) Especially at high levels of stress, these stressors limited the number of shifts that occurred within the critical personnel substructures.

This can pose a significant risk to a network centric organization if such flexibility is an advantage for dealing with change. For example, this could slow the integration of diversity or circumvent resiliency. It could slow the integration of diversity when a situation calls for a variety of expertise that is different than previous conditions and those experts do not step up to enact critical roles. It could circumvent resiliency when current critical experts become unavailable or overtaxed and redundant expertise does not shift into the critical role. Moreover, limitations to the number of agents who can assume critical roles, as in unique change, could pose a risk by restricting the

development of expertise. Fewer agents can assume critical roles that give them valuable experience.

*Proposition 3: Shifts of critical personnel are negatively related to communication network constraints and cognitive constraints.*

*Proposition 4: Communication network constraints and cognitive constraints can pose a risk by modifying the number of flexible responses, in terms of critical personnel shifts, exhibited by a network organization in a dynamic environment. This is a risk only when such flexible responses are advantageous and sufficient to dealing with environmental change.*

To clarify proposition 4, it is recognized that an occurring shift, even when a shift is needed, is not in and of itself sufficient to ensure an effective response. Shifts could occur that are counter to an organization's intended objective. For example, a situation may be misinterpreted and the wrong agent may assume a critical role. In this case, a necessary shift could be insufficient and result in a risk to the organization.

Intermittent availability had a stronger impact on shifts of critical personnel than did selective attention, as evidenced by the standardized beta coefficients from the multiple regressions. This implies that, at the organizational level, communication constraints are a slightly bigger risk to critical personnel shifts than are cognitive constraints.

*Proposition 5: Communication network constraints are a slightly larger risk to shifts of critical personnel in network organizations than are cognitive constraints*

#### 4.2 Normative Implications

The proposed theories on critical personnel risks have several normative implications for the network organization under study. Some normative implications are discussed below.

The Battle Command Group should re-identify critical personnel often. Observations of this organization during the wargame exercise noted rapid changes to the operational scene when the exercise was in full tilt. The theory developed in this thesis suggests that considerable shifts of critical personnel will occur during these times. Re-identification will keep the organization current on who is critical. The organization can then make use of these critical personnel in the present situation and this can provide benefits. For instance, critical personnel may

improve staff decision-making. Critical personnel who are high in betweenness or degree centrality tend to accumulate knowledge which leads to high situational awareness. Integrating these people into the decision loop can provide the staff with a better understanding of the present situation. In other words, current critical personnel can contribute to the observe and orient processes of the OODA loop. They can also contribute to the decision and action processes as well but in any case their inclusion in the loop may serve to improve decisions.

In addition, critical personnel can be used to improve information flow and the rate of learning in the organization. Observations also noted considerable communication network complexity during times of rapid change. Communication network complexity can slow the rate of learning. Central persons in the communication network serve as focal points or conduits for communications. Commanders can send and receive information through these central agents thereby taking advantage of shorter path lengths and possibly decreasing the number of paths. This serves to reduce communication network complexity and also speed the flow of information. This can also serve to more efficiently integrate the information that is flowing through the organization. Of course, critical personnel can shift during times of rapid change and an awareness of current critical personnel is needed for this strategy to be effective. This is another reason why re-identification is important.

#### 5. References

- Arquilla, J., & Ronfeldt, D. (2001). The advent of netwar (revisited). In D. Ronfeldt (Ed.), *Networks and netwars: The future of terror, crime and militancy*, 1-24. Santa Monica, CA: Rand Corporation.
- Baker, W. E. (1992). The network organization in theory and practice. In R. C. Eccles (Ed.), *Networks and organizations: Structure, form and action*, 397-429. Boston, MA: Harvard Business School Press.
- Blau, J., & Alba, R. (1982). Empowering nets of participation. *Administrative Science Quarterly*, 27, 363-379.
- Bonacich, P. (1987). Power and centrality: A family of measures. *American Journal of Sociology*, 92, 1170-1182.
- Brass, D. J. (1984). Being in the right place: A structural analysis of individual influence in an organization. *Administrative Science Quarterly*, 29, 518-539.
- Burkhardt, M. E., & Brass, D. J. (1990). Changing patterns or patterns of change: The effects of a change in technology on social network structure and power. *Administrative Science Quarterly*, 35, 104-127.

- Carley, K. M. (1990). Group stability: A socio-cognitive approach. In H. Walker (Ed.), *Advances in group processes: Theory & research, Vol. VII*. 1-44. Greenwich, CN: JAI Press.
- Carley, K. M. (1991). A theory of group stability. *American Sociological Review*. 56(3). 331-354.
- Carley, K. M. (1999). On the evolution of social and organizational networks. In D. Knoke (Ed.), *Special issue of Research in the sociology of organizations on networks in and around organizations, Vol. 16*. 3-30. Greenwich, CN: JAI Press.
- Carley, K. M. (2003). Dynamic network analysis. In P. Pattison (Ed.), *Dynamic social network analysis: Workshop summary and papers*. 133-145. Washington D.C.: The National Academies Press.
- Easterbrook, J. A. (1959). The effect of emotion on cue utilization and the organization of behavior. *Psychological Review*. 66. 183-201.
- Eysenck, H. J. (1967). *The biological basis of personality*. Springfield, IL: Charles C. Thomas.
- Freeman, L. C. (1979). Centrality in social networks: I. Conceptual clarification. *Social Networks*, 1. 215-223.
- Hirshman, B. R., Martin, M. & Carley, K. M. (2008). Modeling information access in Construct. Carnegie Mellon University, School of Computer Science, Institute for Software Research. *Technical Report, CMU-ISR-08-115*.
- Hirshman, B. R., Carley, K. M. & Kowalchuck, M. J. (2007a). Specifying agents in Construct. Carnegie Mellon University, School of Computer Science, Institute for Software Research. *Technical Report, CMU-ISR-07-107*.
- Hirshman, B. R., Carley, K. M. & Kowalchuck, M. J. (2007b). Loading networks in Construct. Carnegie Mellon University, School of Computer Science, Institute for Software Research. *Technical Report, CMU-ISR-07-116*.
- Ibarra, H. (1992). Structural alignments, individual strategies, and managerial action: Elements toward a network theory of getting things done. In R. G. Eccles (Ed.), *Networks and organizations: Structure, form and action*. 165-188. Boston, MA: Harvard Business School Press.
- Johnson, J. C., Boster, J. S., & Palinkas, L. (2003). The evolution of networks in extreme and isolated environments. In P. Pattison (Ed.), *Dynamic Social Network Modeling and Analysis: Workshop Summary and Papers*. Washington D.C.: National Academies Press.
- Kanter, R. M., & Eccles, R. G. (1992). Making network research relevant to practice. In R. G. Eccles (Ed.), *Networks and organizations: Structure, form and action*. 521-527. Boston, MA: Harvard Business School Press.
- Krackhardt, D. (1987). Cognitive social structures. *Social Networks*, 9. 109-134.
- Lin, Z., & Carley, K. M. (2003). *Designing stress resistant organizations: Computational theorizing and crisis applications*. Boston, MA: Kluwer Academic Publishers.
- Matthews, G., Davies, D. R., Westerman, S. J., & Stammers, R. B. (2000). *Human performance: Cognition, stress and individual differences*. East Sussex, UK: Psychology Press.
- Nohria, N., & Eccles, R. G. (1992). Face-to-face: Making network organizations work. In R. G. Eccles (Ed.), *Networks and organizations: Structure, form and action*. 288-308. Boston, MA: Harvard Business School Press.
- Perrow, C. (1999). *Normal accidents: Living with high-risk technologies*. Princeton, NJ: Princeton University Press.
- Podolny, J. M., & Page, K. L. (1998). Network forms of organization. *Annual Review of Sociology*, 24. 57-76.
- Powell, W. W. (1990). Neither market nor hierarchy: Network forms of organization. In L. L. Cummings (Ed.), *Research in Organizational Behavior, Vol. 12*. 295-336. Greenwich, CT: JAI.
- Ronfeldt, D., & Arquilla, J. (2001). Networks, networks and the fight for the future. *First Monday*, 6(10).
- Sampson, F. (1968). A novitiate in a period of change: An experimental and case study of social relationships. *Unpublished Doctoral dissertation, Cornell University*.
- Schreiber, C., & Carley, K. M. (2004a). Going beyond the data: Empirical validation leading to grounded theory. *Computational and Mathematical Organization Theory*, 10(2). 155-164.
- Schreiber, C., & Carley, K. M. (2004b). *Key Personnel: Identification and assessment of turnover risk*. In *Proceedings of the 2004 NAACSOS Conference*.
- Schreiber, C., Singh, S., & Carley, K. M. (2004). Construct - A multi-agent network model for the co-evolution of agents and socio-cultural environments. Carnegie Mellon University, School of Computer Science, Institute for Software Research, International. *Technical Report, CMU-ISRI-04-109*.
- Schreiber, C., & Carley, K. M., (2007). Agent Interactions in Construct: An Empirical Validation using Calibrated Grounding. In *Proceedings of the 2007 BRIMS Conference*.
- Simon, H. A. (1947). *Administrative behavior*. New York, NY: Free Press.
- Staw, B. M., Sandelands, L. E., & Dutton, J. E. (1981). Threat rigidity effects in organizational behavior: A multilevel analysis. *Administrative Science Quarterly*, 26(4). 501-524.

## Social identity modeling: past work and relevant issues for socio-cultural modeling

Jonathon Kopecky

Nathan Bos

Ariel Greenberg

Johns Hopkins University Applied Physics Laboratory

11100 Johns Hopkins Rd Laurel, MD 20723

{jonathon.kopecky, nathan.bos, ariel.greenberg}@jhuapl.edu

**ABSTRACT:** Many of today's political conflicts are based on social identity differences, and sides are drawn up along ethnic, religious, ideological lines. Socio-cultural modeling efforts need to be able to incorporate realistic social identity dynamics that are based in academic literature and build on prior work. This paper reviews four modeling efforts in this area: Aptima's SCIPR, Salzarulo's Metacontrast model, Lustik's PS-I model, and Johns Hopkins APL's SILAS. Each is analyzed as to its mix of descent (permanent, inherited) and flexible identities, how each handles changing salience using Turner's Accessibility x Fit model, and how each uses data for grounding and validation.

### 1. Modeling Social Identity

"It is increasingly apparent how many of the dangerous conflicts around the world are defined in terms of some variant of 'identity politics'" (Lustick, 2002). Tutsi versus Hutu violence in Rwanda, Sunni versus Shia violence in Baghdad and Serb versus Bosniak violence in Bosnia and Herzegovina are a few recent examples of conflicts in which social identity (in addition to the usual political and economic factors) were critical causes and of conflict. There is a current emphasis on modeling in the human social, cultural, and behavioral area, (HSCB) and the dynamics of social identity should be a prominent part of these models. However, social identity is not the most easily tractable topic area for modeling, with components that are complex, highly contextual, and have important individual differences.

Identity refers to a person's collective identity. All individuals have a sense of belonging to multiple identity groups. Since the 1950's psychologists have used the simple "Twenty statements test" to gauge self-concept (Kuhn & McPartland, 1954) where participants make 20 statements in the form of "I am \_\_\_\_". Responses tend to fall into five groups, one of which is social categorization, or social identities. Social identity responses might be "I am Christian", "I am American", or "I am a Teamster." Individual may have many social identities along dimensions of

ethnicity, religion, politics, economics, and ideology, among others.

Knowing an individual's identity affiliations can be the key to understanding attitudes and opinions, as individuals tend to adopt opinions compatible with their salient identity groups (Haslam & Turner, 1992; Haslam & Turner, 1995; Turner, Hogg, Oakes, Reicher, & Wetherell, 1987). Identity can help explain the day-to-day behavior of individuals when rituals, mores, practices, or more subtle behavior patterns are associated with identity groups. (Abdelal, Herrera, Johnston, & McDermott, December 2006). Understanding the pattern of identities in a population is a key to understanding conflicts, predicting both where conflicts are most likely to occur, and predicting how groups are likely to align in a conflict situation.

Modeling identity is more complex than simply modeling demographic differences, however, which means that modelers have to do more than simply recreate populations with known ethnic, religious, and political statistics. There is an extensive literature in the behavioral sciences dealing with the definitions, implications, and malleability of social identities. It is not the purpose of this paper to comprehensively review this literature. Nor is it probably feasible (or necessary) for socio-cultural models to incorporate every social and psychological nuance of identity. Some issues are more critical than others for modeling, which will be the focus of this paper.

Although identities themselves can change over time (e.g. the ‘Catholic’ identity may become more secular or more religious over a generation), we limit our discussion to models of time scales in which the properties of the actual identity as constant. This paper will focus a subset of issues that are important for models in the HSCB domain about how individuals select a particular identity, which can be used to model political trends, conflict, and related social issues. We will address two overarching modeling issues:

1. Identity permanence. How do modelers differentiate between identities that can change easily and those that cannot?
2. Identity salience. How are individual or group identities expected to change in importance based on the situation?

We will also discuss, in the context of prior examples, three related issues:

3. Identity and influence. How do individuals in a social network affect each others’ identity affiliations?
4. Ingroups and outgroups. How do identity groups define themselves in comparison to each other, and what are the resulting dynamics?
5. Relationships between identity groups. How do identity groups show affinity or rivalry with each other, and how does this affect alignments in conflict situations?

### 1.1 Identity permanence

There is an important distinction between descent identities such as ethnicity, which are relatively fixed, and flexible identities such as political party affiliation. Across cultures, an identity may vary in being assessed as descent or flexible.

**Descent identities** are identities that individuals are born with and that are difficult to impossible to change, especially in the short-term (cf. ‘stickiness’ in the political science literature, e.g. Chandra, 2006). Obvious examples of descent identities are ethnicity and race. Individuals who identify as African-American are going to have some connection to the African-American identity group their entire lives. While this identity may be nuanced or augmented, it cannot be changed to a completely different group

(e.g. Asian). Religion can also be treated as a descent identity. Although, technically individuals can convert from one religion to another, this is very difficult if not impossible in many parts of the world and usually carries a high cost, such that most models should regard this variable as permanent. Descent identities should not necessarily be considered exclusive, however. Conversion or intermarriage may tie a person to more than one identity group. A Caucasian woman with an African-American husband and children may adopt a strong affiliation to that identity group, even though it was not hers by birth. Descent identities are augmented, but not replaced. Even in the case of conversions or intermarriage, an individual’s original religious or ethnic identity still affects behavior. People carry multiple descent identities, although they often differ in salience, as will be discussed.

**Flexible identities** are those identities which individuals can change fairly easily with relatively low cost. The most commonly modeled flexible identities are political-party affiliations and occupation. It is usually possible to switch political parties or occupations, and usually the barriers are much lower than those related to changing religions. Ideologies that blend the social and political are a third common example of flexible identities: ‘environmental activist’; ‘evangelical conservative’; and ‘moderate Islamist’ might be examples. Not every belief constitutes an ‘identity’, (e.g. ‘Ford truck advocate’ probably does not need to be modeled as an identity group in most sociocultural models); but beliefs that connect people to larger groups with established norms and that affect a variety of behaviors may need to be modeled as such.

Some identities, such as social class, may need to be treated as descent identities in some settings and flexible in others. In regions of the world known to have strong class distinctions and low economic mobility, social class and even occupation may be a descent identity; but these should be treated as flexible in most parts of the developed world.

### 1.2 Determining salience: Accessibility x Fit

While every individual can hold multiple identities of multiple types, the importance of these identities can change radically from one circumstance to another.

Understanding when particular identities are salient is a critical capability. We will use the concepts of Accessibility and Fit, which are aspects of Turner's Social Categorization Theory (SCT; Turner et al., 1987, Bruner, 1957, Blanz, 1999) as a way of thinking about differences in salience. Salience is the product of a relatively permanent 'accessibility' parameter and a contextual 'fit' for a particular identity (Salience = Accessibility \* Fit).

Individuals have self-identities that are more or less salient. For one individual, their religion may be the most important component of their identity, while for another, an economic identity (e.g. 'successful businessman') may be most salient. In our research group's work modeling Nigeria, ethnic loyalties were thought to be particularly important. For this research, we benefitted from a data source which asked questions directly about salience. The data source is Afrobarometer ([www.afrobarometer.org](http://www.afrobarometer.org)), which is a repeated survey of a number of West and South African countries. In addition to collecting demographic information for each respondent, Afrobarometer asked each respondent: "Besides being (your country's nationality), which specific group do you feel you belong to first and foremost?" The answer to this question would be the non-national identity most salient at that moment. This data showed how salience varied across individuals, and also how it varied systematically across different segments of the Nigerian population. For example, religious identity was most salient for Muslim Hausas in Nigeria, while ethnic identity was most salient for Christian Igbos. There was also considerable individual variation-- each group included some individuals with strongly salient ethnic, religious, political, and economic identities.

**Accessibility** is the ease of retrieving a given identity to mind, similar to the 'availability heuristic' from cognitive psychology (Tversky & Kahneman, 1973). Identities that are more familiar or carry more emotional valence are more accessible. For example, it is relatively easy for Americans to retrieve the identities 'Christian' or 'Muslim,' and generally harder to retrieve some other religious identities (e.g. 'Rastafarian', 'Sunni', 'Shintoist'). The harder it is to retrieve a particular identity, the less likely a

person is to categorize either themselves or another into that category.

Some identity categories are also more accessible than others. Most individuals have ethnic, religious, and occupational identities, but they are not equally accessible. Research has shown that ethnic, religious, and political identities tend to be more accessible than occupational or relational identities (such as 'husband' or 'son', Deaux, Reid, Mizrahi, & Ethier, 1995).

One example where accessibility affects perception is in the American perception of 9-11 hijackers. Although 15 of the 19 were from Saudi Arabia, 'Saudi Arabian' has low accessibility for most Americans, so very few Americans noticed or remembered that the hijackers were Saudi Arabian. However, both 'Muslim' and 'Iraqi' had much higher accessibility, thus the identity of the hijackers was more easily perceived to be Muslim (which was accurate) or Iraqi (which was not). (This may help explain why over 40% of Americans felt that Iraq played a direct role in the 9-11 attacks, Wolf, 2007).

**Fit** is the degree to which a particular context activates particular identities. While accessibility is considered to be a relatively fixed feature of an identity for any given individual, contextual fit can vary widely. Current events can strongly interact with particular identities. We saw that in America after 9/11, people's American identity was more salient than their political identity, because of the 'fit' between the events and national identities. Lewis (2007) showed that identity affiliation in Nigeria changed markedly between 2001, 2003, and 2005, with ethnic identities significantly higher in the first and last. His explanation: elections were being held near the 2001 and 2005 data collection events, and Nigerian elections have often been seen as contests between ethnic groups. Using SCT terminology, the context of Nigerian national elections had a high degree of 'fit' with ethnic identities.

Fit can also be affected by a particular social context. Nigerian expatriates living in the U.S. may become particularly conscious of their Nigerian identity, especially in the company of other Nigerians. When we review Salzarulo's model, the use of metacontrast

ratios to quantify comparisons between identity groups will be relevant to this kind of fit.

Framing or re-framing of an event can attempt to change the ‘fit’ of events, and thus change which identities becomes salient. This is one of the techniques used by Al Qaeda to try to elicit sympathy for themselves, by portraying Al Qaeda actions done against specific American or European targets as part of a conflict between Muslims in general and Western powers in general. In the language of SCT, Al Qaeda tries to create a fit between specific events and identities that are highly accessible to their audience: Islam and the West.

## 2. Four models of social identity

We will review four modeling efforts where social identity plays a large role. We will describe each model’s unique strengths, and compare how they handle identity permanence (descent versus flexible identities) and identity salience (with components of accessibility and fit). Each model also brings in additional theoretical issues, which will be described in the context of each model.

Model	Types of identities	Data sources	Key features
SCIPR	Flexible: Political opinions	Grounded and validated with IRA attack data and voting results from Northern Ireland	Models influence using a bounded confidence model. Includes multiple overlapping identities and uses a simple social network for influence.
Salzarulo’s MetaContrast model	Flexible: Belief-based social categories	Synthetic	Illustrates how polarization and extremism can occur due to combination of attraction to ingroups and repulsion from outgroups
PS-I	Flexible and Descent: Political/cultural identity groups	Author’s regional expertise	Models geographic clusters, or ‘polities’, and spread of identities through a population
SILAS	Flexible and Descent: Ethnic, Religious, and political identity affiliations	2001 Afrobarometer survey of Nigeria used for grounding and validation	Models how internal conflicts between identities may be resolved; models ‘common enemy’ dynamic

Table 1.1 Overview of social identity models

**Aptima’s SCIPR** (Simulate Cultural Identities for Predicting Reactions to Events) is an agent-based model of opinion dynamics (Grier, Skarin, Lubyansky, & Wolpert, 2008). A collection of agents maintains a set of possible identities, where each identity is defined by a set of beliefs. Each agent also has a synthetic social network of associates, largely determined by geographic proximity. As the model runs forward in time, agents influence each other to try to draw others closer to their beliefs, and thus influence political party affiliation.

Central to the SCIPR model is a model of ‘bounded confidence’. Agents hold beliefs and also have a degree of confidence associated with those beliefs. This confidence strongly constrains how easily they can be influenced by other agents. When the model is running agents try to influence the other agents in

their social network, but can only be influenced by them if 1) the two agents are demographically similar, and 2) the influence message being sent is close enough to the receiving agent’s current beliefs to fall within that agent’s confidence parameters. Agents with less confidence are both more likely to listen to agents whose starting position is dissimilar to their own and more easily persuaded by new messages. Agents with very strong confidence are very resistant to changing opinions, although in the absence of reinforcing messages from similar agents, confidence does decay over time.

For the 2006 paper cited here, the SCIPR model was used to try to reproduce broad changes in opinion dynamics of Northern Ireland residents during ‘the troubles’ by comparing outputs with election results.

Aptima's model cites Salzarulo's work and its use of bounded confidence is similar.

**Salzarulo's Metacontrast model** also focuses on opinion dynamics (Salzarulo, 2006). Agents have positions on a single issue, with a continuous number representing their opinion. Similar to the SCIPR model, each agent has a bounded confidence which affects who the agent will listen to and how much they may be swayed by an alternate position. Persuasion in this model is equivalent to an agent moves along the continuum of opinions toward a different position held by another agent. A unique feature of Salzuro is that the model includes both attraction and repulsion forces; agents move toward 'identities', or opinion positions that they want to join, and also try to move away from opinion positions that they define themselves against.

Salzarulo uses the principle of meta-contrast from social categorization theory (Haslam & Turner, 1992; Turner et al., 1987) to judge similarity and cohesion of identity groups. Groups (Salzarulo calls them categories) form when a cluster of agents perceive that the differences between them are small, and the distance between them as a group and other individuals in a group is large. More precisely, agents calculate the mean pairwise difference between all individuals in the model and compare it to the mean different pairwise differences to a subset of agents that form a candidate group. Groups form from clusters with a low ratio of group differences to context differences.

Once these groups form, agents act to reinforce group membership. Groups observe which individuals are most central, or prototypical of the group, and move to reduce differences between themselves and their group prototype. At the same time agents seek to maximize the difference between themselves and agents outside of the group. This is consistent with prior psychological studies of identity dynamics (Tajfel & Turner, 1986); ingroups often consolidate their identity by trying to clearly differentiate themselves from other groups, referred to as 'outgroups'.

The Salzuro model produces three interesting effects that may be particularly useful for modelers. First, it produces polarization of opinions between groups.

Because Salzarulo's agents actively change opinions to move away from outgroups and toward the center of ingroups, they can result in groups clustered at the extreme ends of an opinion continuum, although this does not always happen. Polarization clearly happens in the real world, but often fails to happen in other influence models where over time agents become homogenized; Salzarulo provides a plausible mechanism for polarization to occur.

Second, Salzuro's model produces an effect where agents whose opinions are prototypical of their identity group have very high confidence in their opinions. Because other agents in the group are moving toward them as central figures, and no force is pulling them away from their own center, the confidence of prototypical agents increases. This again corresponds to the real-world observation: group leaders tend to be very certain of their opinions. Salzarulo does not use the term 'leader'; his model speaks only of more- or less-prototypical members; but it would be a natural extension to use his mechanisms to name these prototypical members as group leaders, and use these mechanisms to explain (at least partially) observed high levels of leader confidence.

Third, Salzarulo introduces a mechanism for context to affect identity. Salzuro's explorations show that the formation and differentiation of groups in the metacontrast model are strongly influenced by the profile of agents in the initial model, i.e. the social context. Salzarulo's explorations do not take the next step of varying the context within model runs, but one can easily imagine changing context within a larger model and observing the resulting effects on identity. This could model the strengthening of identity in a context where that identity is the minority; e.g. the previous example of Nigerian expatriates in the US context feeling a strengthening national identity.

Salzarulo's work is a pure modeling effort, so has not (to our knowledge) been grounded or validated against real-world datasets.

**Lustick's PS-I model**, is also focused on political opinions and persuasion, and particularly focused on regionally coherent 'polities', or identity groups. PS-I is intended as an open, general framework and has

been applied by the author (a Middle Eastern expert) in several settings. The example used in Lustick (2002) is a fictional country called ‘Middle Eastern Polity’ (MEP). MEP is represented by a rectangular grid populated with 2260 agents. Each agent represents a population aggregate, but behaves similarly to individuals in other models. There are 19 ‘identities’ present in the model, that vie for influence within and between agents. These are mostly political identities, but also include elements of religious, ethnic, and economic identities. Three examples are ‘Fundamentalist Islam’ (religious/political), ethnic Kurd (ethnic), and modernized Islam (religious/political). Each agent has a repertoire of 2-6 ‘identities’ that they hold. Only one identity is ‘active’ at a time, but the others may maintain lower levels of activation that are important to the model. A geographic cluster of agents with the same activated identity is referred to as a polity.

The pattern and initial activation levels of identities are how PS-I handles accessibility. The model also includes fit of contextual events. Model runs include disruptive, short-term events originating outside the model; e.g. a terrorist incident in a nearby country. The effects of these external events are determined by the existing pattern of activations moderated by tables of ‘bias’ specific to event types. These bias tables are what implement fit in PS-I. When the model runs, agents influence their neighbors and politics spread, shrink, or disappear across the landscape of the country. As in the other political influence models, similarity between agents determines influence. Lustick’s model also includes varied agent ‘personalities’ which are important to the influence dynamics but will not be described here.

PS-I has been used to study the volatility and common patterns of identities through simulated countries such as ‘Middle Eastern Polity’, and has also achieved some success validating against historical data. A focal point of study has been predicting regime instability. Other noteworthy strengths of this model are its ability to combine across identity types; and the ability to model of larger-scale geographic trends.

**SILAS (Social Identity Look-Ahead Simulation)**, is in development by the authors at Johns Hopkins University Applied Physics Laboratory. SILAS focuses on identity-based conflicts. It attempts to predict how individuals with multiple identity affiliations will align in a conflict that may activate more than one of their identity groups.

The model includes two layers: individual agents (people), and abstracted identity groups. Each individual agent is modeled on single respondent to the Afrobarometer 2001 survey. Each agent was also given a synthetic social network of other agents based on assumptions about levels of cross-ethnic and cross-religious affiliations in Nigerian society (no data was available for this).

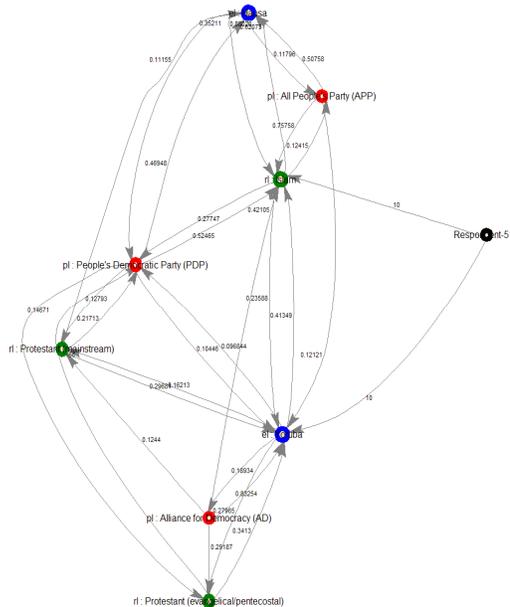


Figure 1.1. One individual (far right) and the network of identities joined by affinity links that the individual is affiliated with

Identities are modeled as objects that are separate from, but connected to individual agents by ‘affiliation’ links. Identities are arranged in separate hierarchies for three types: ethnic, religious, and political identities. Groups have affinity relationships with each other, both within and between hierarchies which are set by comembership data derived from Afrobarometer 2001 data. So, for example, the ‘affinity’ between the Hausa ethnic group identity

and the Muslim religion was set to correspond to the percent of Hausa Afrobarometer respondents were Muslim. (This was an asymmetric network; the affinity from Muslim to Hausa corresponds to the percent of Muslims who are Hausa.)

Each individual in this model was affiliated with multiple identity groups, usually there was one ethnic, one religious, and one political affiliation. As a default, the weight (accessibility) of each affiliation link was set to 1 to indicate membership with a group. We used the Afrobarometer data on most-favored identities to increase this weight to 2 when such a response was given (recall that each individual in the model is based on an actual Afrobarometer respondent). These permanent affiliation weights are SILAS's representation of identity accessibility.

Running the SILAS model begins with a conflict event between any two identity groups. (e.g. Muslim versus Christian; Igbo versus Ijaw; or Igbo versus Muslim). The groups do not have to be of the same type. The two groups in the conflict spread positive sentiment about themselves and negative sentiment about their opponent in the conflict. These sentiments spread through the abstracted identity model along affinity links. The strength of the affinity links was used as a multiplier of the strength of the sentiment. Sentiment, both positive and negative, spreads between identities and down to individuals. Spreading activation is limited to minimize feedback loops among identities. When the model is finished running, many individuals will have received positive and negative sentiment about the identities involved in the conflict. Some will have received both through separate channels, and will weigh the level of each to determine where they stand on the conflict. Some individuals will have received no sentiment messages, or equal positive and negative sentiment, and so will remain neutral. The temporary sentiment messages with varying levels of activation are how SILAS represents situational fit.

There are two notable features of SILAS. First, SILAS can predict the opinion of conflicted individuals; i.e. those that have identity links (direct or indirect) to multiple parties in a conflict, as described. A formative evaluation study used SILAS to predict political party affiliation based on an

individual's identity links. The model was constructed using known co-memberships, and then run on the same dataset with political affiliation links removed. (We chose to train on the entire set rather than reserve part for validation, because of the small cell size of some affinities). The conflict event was a simulated election between the three major political parties in Nigeria at the time of the 2001 Afrobarometer survey. The SILAS model correctly predicted 72% of known party affiliations. We compared this with a more conventional regression analysis, which predicted 76% correct. We were disappointed that SILAS did not outperform conventional regression, but pleased to be close. We hope to be able to improve the model with more highly localized data.

SILAS's second notable feature is reproducing the 'uniting against a common enemy' effect. The hierarchical arrangement of identities allowed inference beyond stated groups, e.g. the model knew that an individual who self-identified as a Baptist was a Christian. In a conflict between a Catholic group and a Muslim group, the Baptist will receive stronger sentiment messages from the Catholic than the Muslim identity groups because of the shared Christian identity. The dynamic of uniting against a common enemy is well documented in the real world, but previous models did not necessarily reproduce it, or produce it only as a byproduct of other kinds of similarity.

We are seeking, but have not yet found a dataset that could be used to test the validity of the 'common enemy' effects. We are also seeking to extend the SILAS model to reproduce the corollary 'sibling rivalry' effect. Sibling rivalry is an effect where, in the absence of a common enemy, peer groups in a hierarchy may be particularly prone to conflict. This may be useful in modeling 'horizontal inequality' conflicts between peer groups, which Stewart (2000) argues are one of the most common types of conflicts in third-world countries. We have experimented with adding negative affinity, or 'rivalry' links between peers in the identity hierarchy, but initial runs with these in the model yielded unsatisfying results.

Model	Accessibility?	Fit?
-------	----------------	------

SCIPR	✓	
MetaContrast model	✓	
PS-I	✓	✓
SILAS	✓	✓

Table 2. Model coverage of accessibility and fit

Modeling social identity is an important capability for valid socio-cultural models. The need to

#### References

Abdelal, R., Herrera, Y. M., Johnston, A. I., & McDermott, R. (December 2006). Identity as a variable. *Perspectives on Politics*, 4(4), 695-711.

Blanz, M. (1999). Accessibility and fit as determinants of the salience of social categorizations. *European Journal of Social Psychology*, 29(1), 43-74.

Bruner, J. S. (1957). On perceptual readiness. *Psychological Review*, 64(2), 123-152.

Chandra, K. (2006). What is ethnic identity and does it matter? *Annual Review of Political Science*, 9, 397-424.

Deaux, K., Reid, A., Mizrahi, K., & Ethier, K. A. (1995). Parameters of social identity. *Journal of Personality & Social Psychology*, 68(2), 280-291.

Grier, R. A., Skarin, B., Lubyansky, A., & Wolpert, L. (2008). SCIPR: A computational model to simulate cultural identities for predicting reactions to events. *Proceedings of the Second International Conference on Computational Cultural Dynamics*,

Haslam, S. A., & Turner, J. C. (1992). Context-dependent variation in social stereotyping 2: The relationship between frame of reference, self-categorization and accentuation. *European Journal of Social Psychology*, 22(3), 251-277.

Haslam, S. A., & Turner, J. C. (1995). Context-dependent variation in social stereotyping 3: Extremism as a self-categorical basis for polarized judgement. *European Journal of Social Psychology*, 25(3), 341-371.

Kuhn, M. H., & McPartland, T. S. (1954). An empirical investigation on self-attitudes. *American Sociological Review*, 19, 68-77.

synthesize a broad and diverse literature and the need for new modeling techniques make this a difficult but also very interesting problem.

The models reviewed in this paper were all narrowly focused on a few identity-related issues. Future models will also need to integrate with a broader set of behavioral, economic, and political dynamics, which should be a focus of current research.

Lewis, P. M. (2007). Identity, institutions and democracy in nigeria. *Afrobarometer Working Paper #68*. Retrieved from: <http://afrobarometer.org/papers/AfropaperNo68.pdf>

Lustick, I. S. (2002). PS-I: A user-friendly agent-based modeling platform for testing theories of political identity and political stability. *Journal of Artificial Societies and Social Simulations*, 5(3) Retrieved from: <http://jasss.soc.surrey.ac.uk/5/3/7.html>

Salzarulo, L. (2006). A continuous opinion dynamics model based on the principle of meta-contrast. *Jasss-the Journal of Artificial Societies and Social Simulation*, 9(1). Retrieved from: <http://jasss.soc.surrey.ac.uk/9/1/13.html>

Stewart, F. (2000). Crisis prevention: Tackling horizontal inequalities. *Oxford Development Studies*, 28(3), 245-262.

Tajfel, H. and Turner, J. C. (1986). The social identity theory of inter-group behavior. In S. Worchel and L. W. Austin (eds.), *Psychology of Intergroup Relations*. Chicago: Nelson-Hall

Turner, J. C., Hogg, M. A., Oakes, P. J., Reicher, S. D., & Wetherell, M. S. (1987). *Rediscovering the social group: A self-categorization theory* Oxford: Basil Blackwell.

Tversky, A., & Kahneman, D. (1973). Availability - heuristic for judging frequency and probability. *Cognitive Psychology*, 5(2), 207-232.

Wolf, Joerg. More Americans believe that Saddam was directly involved in 9/11. Retrieved from the web December 15<sup>th</sup>, 2009. <http://atlanticreview.org/archives/726-More-Americans-Believe-that-Saddam-Was-Directly-Involved-in-911.html>

# A Hybrid Model of Ethnic Conflict, Repression, Insurgency and Social Strife

Stacy Lovell Pfautz

Michael Salwen

NSI, Inc.

3811 N. Fairfax Drive, Suite 850-B

Arlington, VA 22203

spfautz@nsiteam.com, msalwen@nsiteam.com

Keywords: Human, social, culture, behavior modeling (HSCB), hybrid model, agent based model, system dynamics model, ethnic conflict

**ABSTRACT:** *Ethnic conflict, Repression, Insurgency, and Social strife (ERIS) is a multi-paradigm model of ethnic conflict at varying levels of analysis and implementation. ERIS attempts to address the complexity of micro and macro-level social interactions among a population and can be used to assess the effects and implications of social unrest and conflict.*

## 1. Introduction

Ethnic Conflict, Repression, Insurgency and Social Strife (ERIS) is a comprehensive, multi-level model of ethnic conflict that simulates how population dynamics impact state decision making and, in turn, respond to state actions and policies. Population pressures (e.g., relocation, civil unrest) affect and are affected by state actions. The long term goal of ERIS is to support operations development and analyses, enabling military planners to evaluate evolving situations, anticipate the emergence of ethnic conflict and its negative consequences, develop courses of action to defuse ethnic conflict, and mitigate the second and third order effects of U.S. actions on ethnic conflict.

## 2. Background

The current ERIS system is based on a macro-level model specified by Urdal (2008) and a micro-level model specified by Lim, Metzler and Bar-Yam (2007). Each model addressed a particular aspect of ethnic conflict, repression, insurgency or social strife, and could potentially be suitable for multi-level integration.

The Urdal model predicts conflict within a state based upon demographic inputs. The model by Lim et. al. simulates the movement of individuals desiring to cluster with those in their own ethnic group. Conflict is predicted in this model where islands or peninsulas of one ethnicity are surrounded by a sea of

another (Figure 2.1).

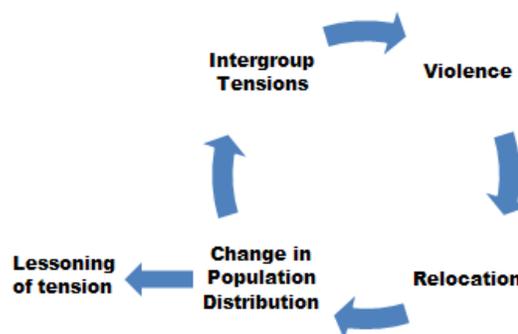


Figure 2.1. The geospatial distribution of the population both affects and is affected by the occurrence of conflict

## 3. The ERIS Model

The ERIS system integrates Urdal's state-level model as a systems dynamics (SD) model with a micro-level agent-based model (ABM) inspired by Lim et. al. Agents respond to conflict by relocating, which in turn causes the demographic composition of locations to change and alter the inputs to the macro-level model.

SD is an approach to understanding the behavior of complex systems that uses feedback loops, stock & flow diagrams, and delays that affect the entire system over time. SD models provide a high level of abstraction, have less detail than ABM, and are well suited to framing and understanding macro level issues and problems. ABM is a computational

approach for simulating dynamic interactions of autonomous agents (or individuals). Agent-based models provide a lower level of abstraction and are well suited for modeling micro level phenomena.

### 3.1 System Dynamics Model

The initial ERIS design and development focuses on four states in northern India: Jammu & Kashmir, Himachal Pradesh, Punjab, and Haryana, which together comprise 62 districts and 306 sub-districts.

The macro-level, system dynamics model (Figure 3.1) determines whether conflict occurs within a state based on demographic information. There is a SD model for each of the four Indian states, initialized with variables and parameters derived from the Urdal model.

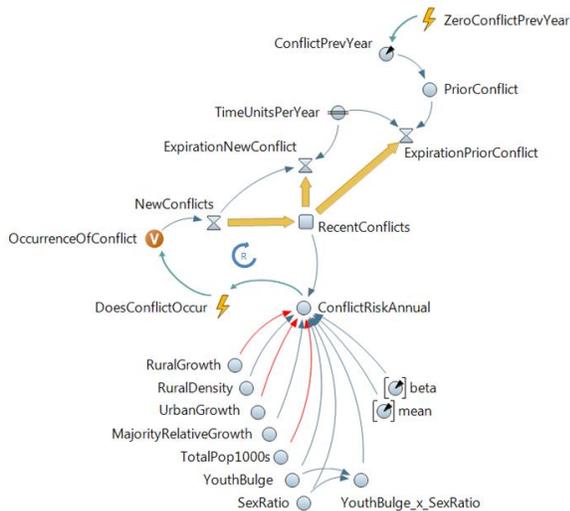


Figure 3.1. System Dynamics model in AnyLogic

The SD model outputs whether conflict occurs within the state as an input to the agent based model. At each time step, the probability of conflict based upon demographic measures derived from the micro-level model (ABM) is computed by the macro-level model (SD). A random draw weighted by this probability is then used to determine whether conflict occurs at the time step. If it does, the conflict stock variable is set to “true”; one year later this state decays and is reset to “false.” The system’s one-year memory for conflict aligns with a macro-level model input of an indicator of conflict within the previous year.

### 3.2 Agent Based Model

Agents move over a GIS Map—a shape file of India that includes polygonal representations of the state, district, and sub-district boundaries elected for use in the ERIS system. Agents use true latitude and longitude coordinates to move within the simulation space. Agents move between locations, currently defined as sub-districts. A location matrix determines the “cost” of moving between locations, and agents are allocated a budget that effectively determines their permitted extent of motion.

Agents represent 1000 individuals and are uniform with respect to religious affiliation. Agents are sampled with respect to age and sex ratio; however, skew sampling is used to create agents with different demographic profiles with respect to these attributes. Agents also have attributes to capture propensities to conflict and tolerance, which affect agent behavior and interact in the aggregate with the macro-level model to localize reports of conflict.

A homophily matrix measures tensions between ethnoreligious groups. This matrix is a property of location, and varies from place to place based upon local inter-group conditions and will, in subsequent implementations of the system, dynamically alter as the simulation unfolds. Homophily is used in concert with individual agent propensities to conflict or tolerance in localizing occurrences of conflict and by the logic governing agent movement.

Communication is enabled between agents in direct proximity of one another in anticipation of more complex information transmission and diffusion contemplated for future model development.

### 3.3 Hybrid Model

The SD model aggregates attributes from the ABM to calculate rural growth, rural density, urban growth, majority relative Hindu growth, total population, youth budge, and sex ratio as additional input variables that affect the probability of conflict occurring within the state. This drives agent movement behavior as intergroup homophily adapts to the presence or absence of conflict. During each time step (currently set to one week), agent tolerance, pressure to move and propensity for violence produce a subset of agents who may chose to change location. The choice of locations is constrained by the location

cost matrix and the maximum cost an agent can support. Agent movement logic is comprised of a probability measure that initially determines whether the agent is under sufficient pressure to shift location coupled to a location utility calculation. The utility calculation combines in-group/out-group considerations (the homophily matrix) with transit cost (from the location matrix) and time since instances of conflict at candidate locations. Figure 3.3 shows a snapshot of the hybrid model—the purple links indicate the macro-to-micro and micro-to-macro links.

District level input includes a unique id (Num), the state name (State), the district name (District), total population (TotalPopulation), urban population (UrbanPopulation), rural population (RuralPopulation), the number of males (Males), age ranges (Age0-14, Age15-24, Age25Up, AgeNotStated), and religion (Hindus, Muslims, Christians, Sikhs, Buddhists, Jains, Others, NotStated).

The shape file includes geometry for all the states and districts in India used in this version of the ERIS

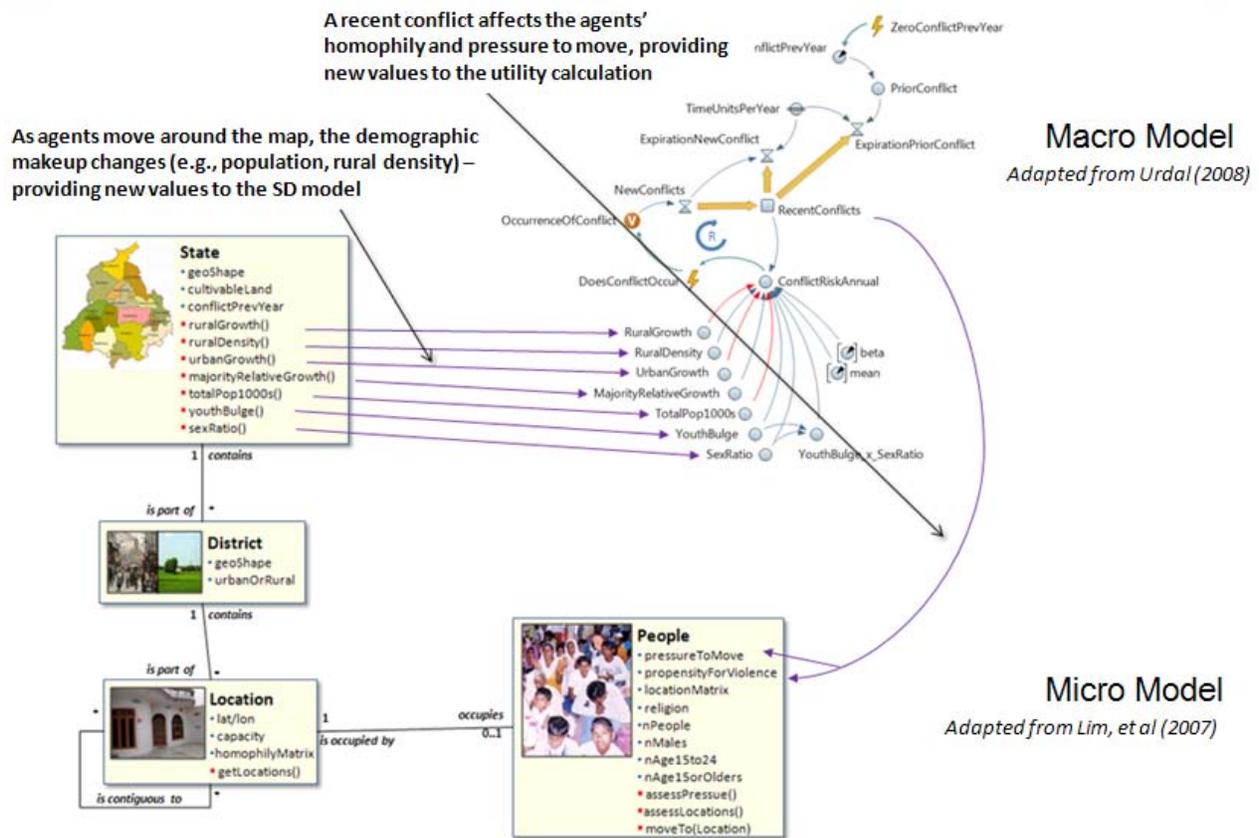


Figure 3.3 Hybrid ERIS model

## 4. Data, Interface, and Configuration

### 4.1 Data Design

State level input data includes a unique id (Num), the state name (State), whether or not there was conflict the previous year in that state (ConflictPreviousYear = 1 indicates the presence of conflict during the previous year) and the land available for cultivation (CultivableArea), in hectares.

system (Figure 4.1.1).

Data on Indian states and districts across sources is not consistent. This is particularly challenging for our model due to discrepancies between the census data and the shape file (e.g., shapes without corresponding data, census data for districts not included in the shape file), which forced decisions about those districts to include and those to exclude. The census map showed areas in India covered by the census,

with large portions of many districts left uncovered. We assume any districts omitted from the census data were ones where data collection was not physically possible. Population distribution by religion is known at the district level, but not the sub-district level. Much of the available geographic data is in non-geospatial file formats (e.g., tables or other media within PDFs, jpeg maps within documents, HTML tables). This type of data requires significant manual labor to extract into structured format and link to geospatial objects in shape files.

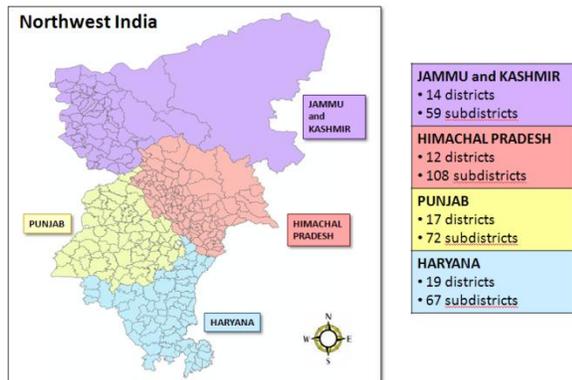


Figure 4.1.1. Geographic scope

## 4.2 Interface Design

The main interface (Figure 4.2.1) includes the GIS map (shape file of India) that shows agents moving from location to location. Sliders bars can be used to pan the map and there are buttons to zoom in and out. The buttons are used to navigate between the map

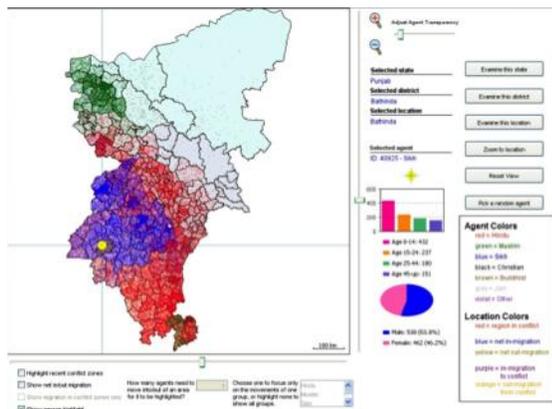


Figure 4.2.1. ERIS – main map interface

view, state view (SD model), district view, sub-district view, and person (agent) view.

## 4.3 Configuration Design

ERIS currently resides entirely on the analyst's laptop or desktop computer. The AnyLogic Model Development Environment serves as the execution environment for ERIS, providing a platform for model execution, data integration, and visualization and analysis. The ERIS Model, which captures the model's execution logic as well as the graphical analytic interface, is stored as an AnyLogic project file. The datasets for states and districts are stored as Microsoft Excel files, while the map data is stored in an ESRI shape file.

## 5. Conclusion

ERIS is an evolving project, now in its earliest stages. The development to date has served the dual purpose of advancing the cause of integrating highly nonlinear models of social behavior at multiple levels while unearthing many of the fundamental obstacles to creating such systems, in particular with respect to obtaining and incorporating empirical data suitable to hybrid combinations. This paper presented the design and execution of the current ERIS system and described some of the hurdles confronting this type of endeavor.

## 6. References

- Lim, K., Metzler, R.: Global Pattern Formation and Ethnic/Cultural Violence. *Science*. 317, 1540{1544 (2007)
- Urdal, H.: Population, Resources, and Political Violence: A Subnational Study of India 1956-2002. *Journal of Conflict Resolution*. 52(4), 590{617 (2008)

## Acknowledgements

We'd like to thank the ERIS team, specifically Pamela Toman, Kari Kelton, Bruce Bullock, and Elisa Jayne Bienenstock. This work was sponsored by the Combating Terrorism Technical Support Office (CTTSO).

## Author Biographies

**STACY LOVELL PFAUTZ** is a senior scientist at NSI, Inc. In this capacity, she provides innovative, research-driven solutions to the challenges in human, social, and cultural behavioral modeling. Ms. Pfautz has led research and development projects focused on the application of systems engineering, artificial intelligence, and cognitive/social/information science to complex sociotechnical systems. Her interdisciplinary experience includes developing decision support tools, techniques for modeling and predicting complex organizations, human-computer interaction, applied artificial intelligence, and cognitive support to intelligence analysis. Ms. Pfautz received a M.S. in Computer Systems Engineering from Northeastern University and a B.S. in Computer Science from the State University of New York. She is a member of the Institute of Electrical and Electronics Engineers (IEEE), the Human Factors and Ergonomics Society (HFES), and a Director on the Executive Board of the HFES New England Chapter.

**DR. MICHAEL SALWEN** is a senior research scientist at NSI, Inc. Prior to working at NSI, Dr. Salwen's professional academic experience centered on mathematics education, with specific focus on the mathematical training of teachers. He served as a Senior Research Associate at MetroMath@CUNY, a Center for Learning and Teaching funded by the National Science Foundation. Outside of academia, Dr. Salwen worked in actuarial research at the Insurance Services Office in New York, where among other projects he was the chief architect of a genetic algorithm-based search of high-dimensional parameter space for insurance premium pricing models. He has also consulted independently on statistical data analysis and online analytic processing and other business intelligence applications. Dr. Salwen holds a Ph.D. in Mathematics from The City University of New York Graduate Center, a B.A. in Mathematics from Hunter College and a B.A. in Politics from New York University.

# Modeling Situation Awareness for Army Infantry Platoon Leaders Using Fuzzy Cognitive Mapping Techniques

Rashaad E. T. Jones

Erik S. Connors

Mary E. Mossey

John R. Hyatt

Neil J. Hansen

Mica R. Endsley

SA Technologies, Inc.

3750 Palladian Village Dr., Building 200

Marietta, GA 30066

770-565-9859

{rashaad.jones, erik.connors, marymossey, mica}@satechnologies.com

## Keywords:

Situation Awareness, Fuzzy Cognitive Maps, Army Infantry Platoon, METT-TC, Modeling and Simulation

**ABSTRACT:** *This paper describes work on the development of an actionable model of situation awareness for Army infantry platoon leaders using fuzzy cognitive mapping techniques. Developing this model based on the formal representation of the platoon leader provided by the Goal-Directed Task Analysis (GDTA) methodology advances current cognitive models because it provides valuable insight on how to effectively support human cognition within the decision-making process. We describe the modeling design approach and discuss validating the model using the VBS2 simulation environment.*

## 1. Introduction

This paper describes our novel approach to providing an actionable model of SA using fuzzy cognitive maps (FCM) that encompasses all three levels of situation awareness (SA) (i.e., perception, comprehension, and projection). Our cognitive model, the SA-FCM model, is built directly from the goals, decisions, and essential information requirements associated with effective decision-making in a domain. As such, the SA-FCM represents a computational naturalistic decision-making model.

Traditional approaches in cognitive modeling relied upon presumptive and assumptive principles derived from basic rational behavior. For example, cognitive architectures, such as ACT-R (Anderson and Lebiere, 1998), SOAR (Newell, 1990), COGNET (Zachary & Le Mentic, 1999), and CoJACK (Evertsz, Pedrotti, Busetta, Acar, & Ritter 2009) provide structural properties of a modeled system that instantiates cognitive models developed from rule-based logic, decision trees, or production and planning rules.

Alternatively, Task Network modeling tools, such as Micro Saint and C3TRACE provide a framework for representing human behavior as a decomposition of operator tasks (Warwick, Archer, Hamilton, Matessa, Santamaria, Chong, Allender, & Kelley, 2008). Finally,

intelligent agent-based systems, such as the Beliefs, Desires, and Intentions (BDI; Bratman, 1987) framework and R-CAST (Fan, Sun, & Yen, 2005) require a priori knowledge and prior experience.

While these cognitive models have advanced the artificial intelligence community, a notable shortcoming of these approaches is that the decisions represented by these models are primarily driven from inferences, behaviors, and rules that do not generally include situation awareness as a cognitive factor. Extensive research has identified SA as a major factor behind the quality of the decision process (see Endsley & Jones, 1997; Klein, 1989; Kaempf, Wolf, & Miller, 1993; Cohen, 1993).

Accordingly, recent prior approaches to computationally modeling SA have been examined, such as dTank (Ritter, Kase, Bhandarkar, Lewis, & Cohen, 2007) and CoJACK (Evertsz, et. al, 2009). However, we have found that these efforts only model the perception construct of SA (i.e., Level 1 SA), and generally do not include the comprehension (Level 2 SA) and projection (Level 3 SA) levels of situation awareness. In order to effectively model decision-making that reflects real world conditions, these higher-level SA constructs should be considered.

Thus, our SA-FCM model is an advancement to cognitive modeling because it incorporates not only Level 1 SA, but higher-levels of SA that is required to make decisions in a

complex world. This is critical in domains such as military command and control, where sufficient data is not always available for developing a cognitive model that provides a realistic representation of the behaviors of the people involved (e.g., friendly forces, insurgents, and civilians).

The next sections describe the design of the FCM model. The following section discusses using VBS2 to validate the model. The paper concludes with preliminary results and highlights the strengths and weaknesses of modeling SA using a FCM. The significance of this effort is that it provides a modeling approach that utilizes SA as the primary driving force for effective decision-making and overcomes some of the limitations of rules, learning algorithms, and behavior moderators that are essential for other cognitive modeling systems.

## 2. Designing the SA-FCM Model

Our current work focuses on improving the representation of situation awareness through the use of Fuzzy Cognitive Mapping techniques. Our objective is to develop a model that replicates human cognition as it relates to SA. The SA-FCM model is designed from the relationship between goals, decisions, and SA requirements as represented by a Goal-Directed Task Analysis (GDTA) hierarchy (see Endsley, Bolté, & Jones, 2003).

Based on the theoretical model of SA provided by Endsley (1995), the GDTA process has been used in many domains to detail SA requirements. As such, it forms an exemplary template for incorporating human cognition into an actionable model by describing in detail not only a user's information data needs (Level 1 SA), but also how that information needs to be combined to form the needed comprehension (Level 2 SA), and projection of future events (Level 3 SA) that are critical to situation awareness thus providing a critical link between data input and decision outputs.

### 2.1 Fuzzy Cognitive Mapping

Conceptually, a FCM can be thought of as a combination of fuzzy logic and concept mapping. Fuzzy logic is derived from fuzzy set theory dealing with reasoning that is approximate rather than precisely deduced from classical predicate logic. It provides the application side of fuzzy set theory dealing with well-conceived real world expert values for a complex problem (Klir, St. Clair, & Yuan, 1997). FCMs use predefined knowledge, or constructs of the causality of concepts (represented as nodes), to define a system. FCMs are especially applicable in soft knowledge domains through their use of (symbolic) knowledge processing.

In a sense, the FCM provides an adaptive structure that affords qualitative reasoning as assessed from the current levels or states of a complex system along with quantitative elements (i.e., causal algebra). At the heart of a FCM is a graphical structure with variable concepts connected via cause/effect relationships. The strength of the causal connection is represented by a numerical quantity defined on the interval  $[-1, +1]$ , with  $-1$  representing an inverse causality and  $+1$  meaning direct causality (Kosko, 1987). Additionally, fractional values are used for the causal connection when combinations of multiple nodes lead to an effect (e.g., a many-to-one relationship).

FCMs provide an efficient soft computing tool that supports adaptive behavior in complex and dynamic worlds (Siraj, Bridges, & Vaughn, 2001; Stylios & Groumos, 2000) as well as reasoning characteristics that make it a significant support aid for analysts and decision-makers. A main advantage of FCMs is their flexibility in system design, modeling, and control (Papageorgiou & Groumos, 2004). Their benefit lies in their capability to represent dynamic systems that can evolve over time, supporting dynamic timeline structures. Unique to FCMs is their ability to incorporate attributes as qualitative states, rather than hard numerical characteristics. FCMs are thus useful for constructing models of dynamic feedback systems, reducing the semantic gap between a system and the model of the system, and predicting the future state (i.e., Level 3 SA, projection) of a system, based on knowing the present state (Level 1 SA, perception).

### 2.2 The SA-FCM Model

The diagram below (see Figure 2.1) illustrates a high-level overview of the SA-FCM model. The model utilizes both top-down (i.e., goal driven) and bottom-up (i.e., data-driven) approaches.

Specifically, the top-down approach begins at the *Goal node*, which influences what the operator perceives from the available data in the world (i.e., the *Level 1 SA node*). Similarly, the operator's goal also influences the *Level 2 SA node* through (1) how much is comprehended (quantity) and (2) which data items are comprehended (quality), thereby effecting the nature of the comprehensions (i.e., the "so what" of the data). Furthermore, the operator's goal also has the same influence on projections (i.e., the *Level 3 SA node*). Collectively, these three nodes represent the SA Requirements submap of the overall SA-FCM model (see Figure 2.1), the content of which is derived directly from the GDTA hierarchy.

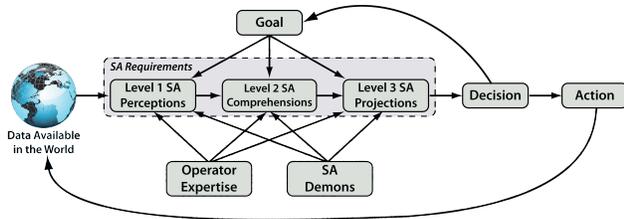


Figure 2.1: High-Level SA-FCM Model

The aggregate SA from these nodes affects the decision of the operator, which then influences the actions of the operator, and may influence the selection of the current goal of the operator. The *Operator's Expertise* and the presence of factors we have dubbed the *SA Demons* are nodes that can degrade or enhance the operator's SA in this process. For example, a novice operator may have trouble achieving the same level of high SA as an experienced operator given the same conditions (as they likely will not have the same models or schema for processing information). Additionally, the presence of certain SA Demons (such as data overload, requisite memory traps, misplaced salience, attentional narrowing, workload, fatigue and other stressors, complexity creep, errant mental models, or the out-of-the-loop syndrome) will limit the SA of the operator, (see Endsley, Bolté, & Jones, 2003 for more information on SA Demons).

Processing in this model can be either bottom-up or top-down, often in an alternating fashion. The bottom-up approach begins at the data node (i.e., *Data available in the world*). The available data determines the goal, which then influences each level of SA. Similar to the top-down approach, the operator's SA is affected by the *Operator's Expertise* and *SA Demons* nodes. The resulting decision is directed by the operator's SA, which then influences the current goal as well as actions taken. Moreover, each top-level node represents a submap that contains concepts and relationships that determine the output of its map. For brevity, only a brief description of the Goals submap, and the SA Requirements submap are provided.

### 2.3 FCM Algorithm

A fuzzy cognitive map is comprised of concepts and weights that can be categorized into three types of layers. First, the input layer contains the concepts that are directly connected to the external world. The middle layer of the FCM serves as a processing layer that integrates concepts from the input layers and directs them to the output layer. Complex FCMs (e.g., those with sub-FCM structures) can have multiple middle layers. The final layer is the output layer whose values are directed back into the external world, or back into the input layer if the FCM incorporates feedback explicitly. The FCM for this project is considered a complex FCM; the concepts on the

middle layer are formed from multiple sub-map structures that contain additional middle layers that are directed to an output layer. Concepts that reside on the middle and output layers have activation functions that determine the output value of the concept. The activation function of a concept node (e.g., *Concept A*) is determined by (1) the value of each input concept that is directed into *Concept A*, and (2) the *influence* that each input concept has on *Concept A*. The activation function can be a global function (i.e., all concepts use the same function) or each concept can have a unique activation function. For example, a binary-state FCM will have a concept value of 1 if activated and a 0 if deactivated. Formally, the activation function is the summation of each input concept multiplied by its weight value minus a threshold value (see equation below). For a complete description of the mathematical process, see Kosko (1987).

$$A_x = (SA_{in} * w_{in}) - t_x$$

### 2.3 Goals Submap

The Goals submap defines the relationships of the main goal, its subgoals, and how each goal influences the other goals (i.e., the activation of one goal can cause the activation of other related goals). For example, the platoon leader GDTA hierarchy (see Figure 2.2) features seven goals under the main goal *attack, secure and hold terrain*. The overall FCM (Figure 2.3) details the causal relationships between these main goals, with each goal representing a node in the map. A total of 15 causal relationships (represented as arcs) with preliminary weight placeholders (e.g.,  $w_{16}$ ) were mapped between the nodes. For each of the seven goals, we created additional "sub-FCMs" using the subgoals as nodes and defined the causal relationships between sub-goal nodes.

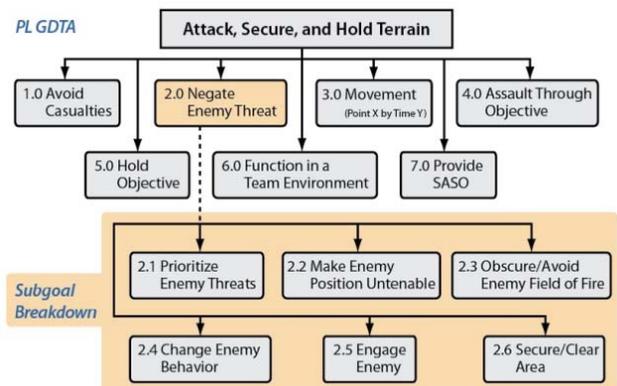
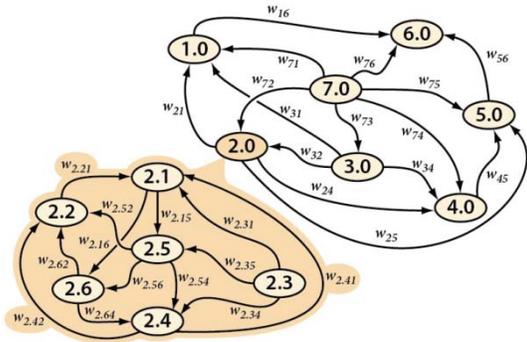


Figure 2.2: Platoon Leader GDTA, showing the main goal and subgoals

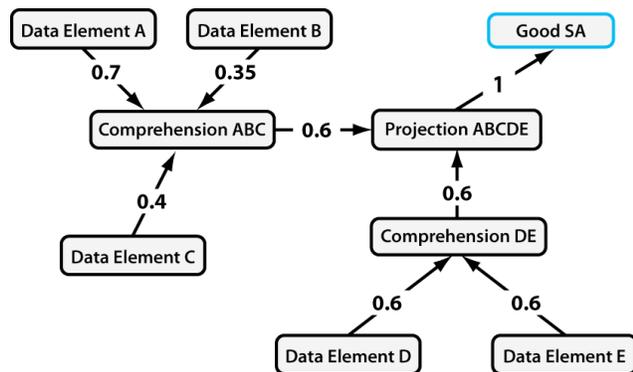


**Figure 2.3: Overall FCM developed for platoon leader with sub-FCM representing sub-goal 2.0**

**2.4 SA Requirements Submap**

The SA Requirements submap can be used to compute the amount of SA the operator has at each level for each SA requirement. The model accomplishes this by maintaining the hierarchical relationship of each SA requirement identified in the GDTA hierarchy and providing a SA score at each level. Consider the simple example submap shown in Figure 2.4. The nodes for this FCM would be obtained directly from the GDTA hierarchy. For example, the GDTA hierarchy identifies *Data Element A, B, and C* as Level 1 SA requirements tied to the Level 2 SA element *Comprehension ABC*.

The specific weights for this map are obtained from discussions with subject matter experts (SMEs). The SMEs are not asked to assign weight values, but rank the importance of each concept, from which the researcher develops the weighting scheme. For example, *Comprehension ABC* can occur if *Data Element A* is available and either *Data Element B* or *Data Element C* is available.

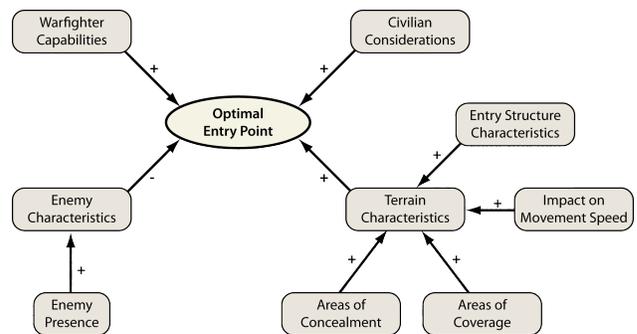


**Figure 2.4: Conceptual SA Requirements submap of FCM translated from GDTA hierarchy (with sample weights)**

From Figure 2.4, in order to have good SA, *Projection ABCDE* must be active. *Projection ABCDE* is only active if *Comprehension ABC* and *Comprehension DE* are both active. Since this is a simple sample case, it is easy to see that from the sample weight values, *Data Element A, D, and E* are the most the significant concepts. Thus, in this particular instance, it is impossible to have good SA without those data elements being presented to the user in a meaningful way.

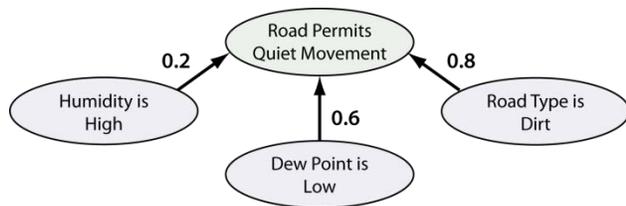
**3. SA-FCM Model in Practice**

The SA requirements outlined in the GDTA encompass the militarily relevant aspects of the environment or background against which a military operation occurs known as Mission, Environment, Terrain and Weather, Troops, Time Available and Civil Considerations (METT-TC factors), the accurate depiction of which is necessary for good decision-making. The SA-FCM model incorporates those METT-TC factors and establishes relationships linking specific considerations to a decision as defined in the GDTA (see Figure 3.1).



**Figure 3.1: Abstracted version of the FCM of METT-TC factors influencing *Optimal Entry Point***

We provide an example to demonstrate how the weights were determined using the methodology defined by Kosko (1987). Our procedure parallels the methodology employed in the development of a FCM that modeled the behaviors of dolphins, fish, and sharks in an undersea virtual world (Dickerson & Kosko, 1994). For terrain considerations, specifically understanding areas of concealment, an Army Infantry Platoon Leader may want to know the following factors: humidity, type of road, and dew point. The infantry platoon leader interprets this information to understand if the road is traversable for covert and stealth operations. A lower dew point combined with a high humidity generally means that a dirt road would more than likely be wet, and therefore quieter, which is preferable for stealth operations. An example of how the SA-FCM represents this relationship is presented in Figure 3.2.



**Figure 3.2: Example FCM detail showing METT-TC (terrain) factors**

The weights are relative values, which are determined in conjunction with our SME, who prioritized the terrain-related factors. For this particular example, the critical factor to stealth movement is identifying the type of the road. Once it is established that a road is a dirt road, the platoon leader can then consider the dew point and humidity as factors, and the impact of those on stealth movement. As explained by the SME, even though the dew point and humidity are related, the platoon leader is more interested in the dew point, and only cares about the humidity in extreme situations. Thus, the condition for conducting stealth movements is primarily dependent upon the road type being dirt and the dew point being low. Consequently, the weight values for those factors are set such that if the nodes for *road type is dirt* and *dew point is low* are true, the *road permits quiet movement* node will be activated.

It is important to note that this process of prioritizing factors parallels the cognitive processes that humans naturally employ. It is easier to characterize an event by prioritizing the conditions that must be present for an event to occur. Conversely, the use of traditional modeling approaches, such as Bayesian Nets, requires quantifying events in terms of probabilities by associating an event to a set of conditions. For example, using a Bayesian approach, the SME would be required to provide the likelihood that the road permits quiet movement given the conditions that the humidity is high, the dew point is low, and the road type is dirt.

#### 4. Validation

The SA-FCM model represents an actionable model of SA that is designed to mimic effective decision-making. The model is derived from a specific GDTA that establishes the goals, decisions, and SA requirements for a given role, in this case infantry platoon leaders. As such, the model considers the following information derived from the METT-TC factors: location(s) to position warfighters for engagement, area(s) for stealth movements, warfighter (i.e., Blue Forces), capabilities enemy capabilities, and Rules of Engagement (ROE) considerations (e.g., places to avoid due to civilian

presence). The current output of the SA-FCM model will be a plan based upon those considerations. Thus, the SA-FCM model represents the SA for an infantry platoon leader whose plan is based upon information that has been gathered in the field. The effectiveness (i.e., success or failure) of the infantry platoon leader’s plan will be primarily predicated on their SA level as represented in the SA-FCM model.

A VBS2 simulation was utilized to validate the SA-FCM model. Working with the SME, we narrowed the platoon leader GDTA down to one subgoal, *Determine Entry Point*, for the purpose of validation. Our Army SME identified this subgoal as one of the more critical for missions that are important to the Army. Additionally, this goal allowed us to quickly develop and implement the SA-FCM model for the validation exercise. The decisions and information requirements associated with this subgoal can be best represented by an infiltration operation that requires an understanding of the terrain and enemy locations and their capabilities in order to choose the correct entry location.

The simulation features a scenario where the warfighters’ goal is to successfully enter a building. Depending upon troop size and capabilities, enemy size and capabilities, and the presence of civilians in public places, the model will need to determine where to strategically position Blue Force assets and avoid major civilian injuries. The scenario development was guided by a SME and is regarded as a representative of common modern Army operations involving clearing a building. The scenario is played in a default town that is available with the VBS2 simulation and it is populated with building architectures and non-playable characters (NPCs) that are common to a Middle Eastern setting.

Two SMEs with different areas of expertise were chosen to assist in the creation and validation of the SA-FCM. One SME, whose area of expertise is intelligence, focused on the information-gathering phase of the mission. Specifically, we discussed the intelligence that would be provided to infantry platoon leaders. The second SME has a background in maneuver and combat, and described how the intelligence would be used to devise a plan in accordance with the Army Combat Manual. Additionally, the second SME explained how specific METT-TC factors, such as areas of concealment and coverage, needed to be established prior to executing the mission. Each was interviewed at length with respect to their area of expertise. The resulting weights for the SA-FCM model and components for the VBS2 scenarios were developed independently of the SMEs.

A Turing test was completed to validate the model. The validation plan involved a SME serving as a confederate

(SME-A). SME-A was given information about a scenario outlined in the METT-TC factors. The same information was provided to the SA-FCM model. Both SME-A and the model produced a plan, which was translated into VBS2. The other SME (SME-B) reviewed the execution of each plan using the After Action Review (AAR) feature of VBS2. The AAR also provided performance measures that were collected for each run. Trial runs were conducted that varied the number of insurgents guarding the building. SME-B evaluated each plan by reviewing avenues of approach and avenues of departure, entry location to building, and how the Blue Forces were deployed. SME-B was unable to distinguish the plans devised by the SA-FCM from the plans devised from SME-A. These preliminary results suggest that the SA-FCM model was successful in developing plans that are consistent with Army procedure.

## 5. Discussion

The significance of the SA-FCM model is twofold. First, the model directly represents the SA requirements for army operations in terms of their relationship as METT-TC factors. Thus, the model is based upon the same information that a warfighter would need to make a decision. Secondly, the SA-FCM model represents decisions in real-time (or near real-time) by effectively comprehending and projecting a scenario based upon the METT-TC factors that is used by a human decision maker.

The following scenario provides an example of how the SA-FCM model can be used to support the warfighter. A platoon of Blue Force warfighters is traveling in a helicopter to a location close to an insurgent hotspot. The warfighters are commanded to clear a building occupied by the insurgents. The platoon leader is provided with a map and intelligence gathered about the area that includes information about the insurgents, terrain, and civilians (i.e., METT-TC factors). Ideally, an infantry platoon leader would prefer sufficient time to devise a plan that may include a detailed process of examining multiple courses of action (COAs). However, in this case, the platoon leader has to develop a plan before the helicopter lands. Thus, the platoon leader attempts to comprehend and make projections from data obtained from various sources, which can be a daunting challenge given the severe time constraints. The SA-FCM would be used to support the decision-making of the infantry platoon leader by mapping the relationship of the METT-TC factors, displaying the relevant considerations appropriately and recommending a plan. Consequently, an immediate area in which the SA-FCM model would prove beneficial is the planning phase of missions; the model could quickly develop and display a recommended plan that effectively

supports the SA requirements for the infantry platoon leader.

### 5.1 Benefits of FCM Approach

An advantage to modeling SA with a FCM from the GDTA is that it allows for higher-level SA to be expressed explicitly. Neural networks, ACT-R, and intelligent agents generally can only model the relationship between input (i.e., perceived elements in the world) and output (i.e., decisions, behaviors, or actions). In these models, how Level 1 SA leads to a decision is unknown to the user as the computational processes are hidden in a “black box.” FCMs built on GDTA hierarchies, on the other hand, include Level 2 and Level 3 SA and are capable of modeling the relationship of how perceived elements (Level 1 SA) lead to comprehension (Level 2 SA), and how that leads to projection of future events (Level 3 SA) which are understandable to the user.

Thus, the SA-FCM will be tailored to fit and encompass the cognitive elements of the decision-making process. The SA-FCM model will incorporate warfighters’ decisions that are made when incomplete information is present (i.e., the platoon leader does not have enough information to make a decision) or when warfighters have information of questionable quality. In both cases, the model identifies the SA requirements that are essential to making the correct decision. Thus, we believe that this model provides a direct way of representing the user because it defines the user’s cognition using subjective terms rather than mathematical expressions. Consequently, the SA-FCM is a valuable approach for modeling goals, decisions, and SA requirements across the three SA levels and then translating that information into a complete actionable model.

### 5.2 Limitations of FCM Approach

A drawback with this methodology is that it solely relies upon the expert’s understanding of the work domain. This understanding can include not only the expert’s knowledge, but their ignorance, prejudice, or biases. Fortunately, FCMs can contain multiple experts’ perspectives by merging each expert’s FCM to create a new FCM that can represent the views of a number of experts in a unified manner.

Translating the GDTA to a FCM is also a challenge. It requires an elicitor that can form a very developed GDTA that contains unique goals and decisions. Since the translation is purely qualitative, the translation process also requires consistency amongst terms. For example, interchanging terms such as speed and velocity can become problematic because it may result in 2 separate

FCM nodes (i.e., one for each term), where they are the same concept.

### 5.3 Future Work

Future work for this effort will include the development and validation of a FCM for all of the remaining subgoals and goals described in the platoon leader GDTA hierarchy. The presence of multiple goals poses additional challenges because the model must also correctly represent the relationships between goals.

A related research direction we wish to pursue is how to represent and incorporate uncertainty within the SA-FCM model. An important feature of FCMs is their capability for addressing uncertainty. Thus, identifying and understanding the sources of uncertainty as it relates to SA is critical to resolving data with different degrees of uncertainty.

Additional future work also includes integrating the SA-FCM in an adaptive environment, so that the model can perform real-time decisions based upon real-time information. For example, the model will produce a plan and can modify it based upon real-time information that is gathered throughout the simulation. Currently, this type of real-time adaptable environment is not supported within VBS2.

## 6. Acknowledgments

Work on this paper was partially supported by funding through participation in the Advanced Decision Architectures Collaborative Technology Alliance sponsored by the U.S. Army Research Laboratory (ARL) under Cooperative Agreement DAAD19-01-2-0009. The views and conclusions contained herein, however, are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the ARL, U. S. Army, U. S. Department of Defense, U. S. Government, or the organizations with which the authors are affiliated.

## 7. References

Anderson, J. R. & Lebiere, C. (1998). *The atomic components of thought*. Mahwah, NJ: Lawrence Erlbaum Associates.

Bratman, M. (1987). *Intentions, Plans, and Practical Reason*. Cambridge, MA: Harvard University Press.

Cohen, M. S. (1993). Metacognitive strategies in support of recognition. In *Proceedings of the Human Factors and Ergonomics Society 37th Annual Meeting*. Santa Monica, CA: Human Factors and Ergonomics Society.

Dickerson, J. A., & Kosko, B. (1994). Virtual Worlds as Fuzzy Cognitive Maps. *Presence*, 3 (2), 173-189.

Endsley, M. R. (1995). Toward a theory of situation awareness in dynamic systems. *Human Factors*, 37(1), 32-64.

Endsley, M. R., & Jones, W. M. (1997). *Situation awareness, information dominance, and information warfare* (Tech Report 97-01). Belmont, MA: Endsley Consulting.

Endsley, M. R., Bolté, B. & Jones, D. G. (2003). *Designing for situation awareness: An approach to user-centered design*. New York: Taylor & Francis

Evertsz, R., Pedrotti, M., Busetta, P., & Acar H. (2009). Populating VSBS2 with Realistic Virtual Actors. *Proceedings of the 18<sup>th</sup> conference on Behavior Representation in Modeling and Simulation*. Orlando, FL: U. of Central Florida.

Evertsz, R., Ritter, F. E., Busetta, P., Pedrotti, M., Bittner, J. L. (2008). CoJACK- Achieving Principled Behavior Variation in a Moderated cognitive Architecture. *Proceedings of the 17<sup>th</sup> conference on Behavior Representation in Modeling and Simulation*. Orlando, FL: U. of Central Florida.

Fan, X., Sun, S., and Yen, J. (2005). On Shared Situation Awareness for Supporting Human Decision-making Teams. *Proceedings of 2005 AAAI Spring Symposium on AI Technologies for Homeland Security*.

Kaempff, G. L., Wolf, S., & Miller, T. E. (1993). Decision making in the AEGIS combat information center. In *Proceedings of the Human Factors and Ergonomics Society 37th Annual Meeting*. Santa Monica, CA: Human Factors and Ergonomics Society.

Klein, G. A. (1989). *Recognition-primed decisions*. In W. B. Rouse (Eds.), *Advances in man-machine systems research*. Greenwich, Conn: JAI Press, Inc.

Klir, G., St. Clair, U. H., and Yuan, B. (1997). *Fuzzy Set Theory Foundations and Applications*.

Kosko, B. (1987). Adaptive inference in fuzzy knowledge networks. *Conference on Neural Networks*, 2, 261-286.

Newell, A. (1990). *Unified Theories of Cognition*. Cambridge, MA: Harvard University Press.

Papageorgiou, E. I. and Groumpos, P. P. (2004). Two-state learning algorithm for fuzzy cognitive maps. *Proceedings of the IEEE International Conference on Intelligent Systems*.

Ritter, F. E., Kase, S. E., Bhandarkar, D., Lewis, B. & Cohen, M. A. (2007). dTankUpdated: Exploring Moderated Behavior in a Light-weight synthetic Environment. *Proceedings of the 16th Conference on Behavior Representation in Modeling and Simulation (BRIMS)*. Orlando, FL: U. of Central Florida.

Siraj, A., Bridges, S. M., and Vaughn, R. B. (2001). Fuzzy cognitive maps for decision support in an intelligent intrusion detection system. *North American Fuzzy Information Processing Society*.

Stylios, C. D. and Groumpos, P. P. (2000). Fuzzy cognitive maps: A soft computing technique for intelligent control. *Proceedings of the 2000 IEEE International Symposium on Intelligent Control*, 97-102.

Warwick, W., Archer, R., Hamilton, A., Matessa, M., Santamaria, A., Chong, R.S., Allender, L. & Kelley, T. (2008). Integrating Architectures: Dovetailing Task Network and Cognitive Models. *Proceedings of the 17th Conference on Behavior Representation in Modeling and Simulation (BRIMS)*. Orlando, FL: U. of Central Florida.

Zachary, W., & Le Mentec, J.-C. (1999). A framework for developing intelligent agents based on human information processing architecture. *Proceedings from IASTED International Conference ASC*.

battalion staffs and is a resident graduate of the U.S. Army Command and General Staff College.

**NEIL J. HANSEN** is a military subject matter analyst for SA Technologies, Inc. in Marietta, GA. He has over 15 years of experience as an Army Intelligence analyst.

**MICA R. ENDSLEY** is the President of SA Technologies, Inc. in Marietta, GA. Her career has focused extensively on cognitive engineering, situation awareness, workload, automation, design, and training issues in multiple domains. She is the editor-in-chief of the *Journal of Cognitive Engineering and Decision Making*.

## Author Biographies

**RASHAAD E. T. JONES** is a Research Associate II at SA Technologies, Inc., in Marietta, GA. His current work focuses on developing cognitive models of situation awareness that can be employed in various applications, including modeling SA for urban warfare operations and modeling SA for the data fusion process. His professional experience also includes user-centric research and technology development for teams operating in command and control, urban warfare, and other militaristic work domains, as well as humanitarian efforts such as the Red Cross.

**ERIK S. CONNORS** is a Research Associate III at SA Technologies, Inc., in Marietta, GA, where he leads research projects on situation awareness for domains such as military command and control, electric power transmission and distribution, and cyber defense. His research interests include user interface design, technology integration, collaborative tools for complex teams, and cognitive modeling.

**MARY E. MOSSEY** is a Research Assistant at SA Technologies, Inc. in Marietta, GA. Her recent work has focused on developing an interface for Red Cross volunteers and research on wind/solar power visualization technology. Her research interests also include SA and human factors issues related to driving.

**JOHN R. HYATT** is a military subject matter analyst for SA Technologies, Inc. in Marietta, GA and an infantry Lieutenant Colonel in the U.S. Army Reserves. With 15 years of active service as a Regular Army Infantry Officer, he has experience working in the small unit infantry environment as well as on battalion and brigade headquarters staffs supporting infantry and armor forces. He has served in positions on infantry brigade and

# Improving Usability and Integration of Human Behavior Representation Engineering across Cognitive Modeling, Human Factors, and Modeling and Simulation Best Practices

*Sylvain Pronovost*

Human Factors Dept., CAE Professional Services  
Institute of Cognitive Science, Carleton University  
Ottawa, ON, Canada  
spronovost@gmail.com

*Dr. Robert West*

Institute of Cognitive Science, Carleton University  
Ottawa, ON, Canada  
robert\_west@ccs.carleton.ca

## Keywords:

ACT-R; cognitive architecture; cognitive modeling; CDB; GOMS; human behavior representation; human factors; IPME; knowledge representation; macrocognition; microcognition; sociotechnical systems; SoHBeR; task analysis; XML.

**ABSTRACT:** *Models of human behavior and cognition differ greatly in breadth, level of detail, and ultimately on the features and criteria of interest relative to the intents and goals of the modelers and their field of expertise. On the one hand, cognitive modeling in general, and cognitive architectures specifically, are interested in microcognitive models of mental processes and fine-grained behavioral outcomes, pitched at a fundamental level of theoretical interest, whereas human factors and cognitive ergonomics modelers focus on performance and workload measures at a coarse macrocognitive level of interaction between multiple agents and their sociotechnical environment. There has traditionally been a gap between micro- and macrocognitive modeling endeavors, reinforced by skepticism on the possibility of reconciling what is seen as fundamental differences between their respective levels of description. The purpose of this paper is to present the progresses of the authors' research project aiming to bridge microcognitive and macrocognitive models of cognition, from cognitive architectures to task analysis. Herein are presented a methodology and a conceptual framework aimed at streamlining the process of cognitive and behavior modeling, focusing on the issues of usability and integration in the development and use of models.*

## 1. Introduction

The research presented here endeavors to integrate human factors models and other cognitive/behavioral modeling efforts, focusing on knowledge representation (KR hereafter), as well as on linking theoretical and applied research issues. On the issue of *knowledge representation*, the aim is to establish necessary and sufficient conditions for (i) satisfying the constraints of known design and processes concerning brain, cognition, and behavior on the one hand, and (ii) for satisfying the integration of such KR with other types of representations used in modeling and simulation (M&S) practices. The second focus is on *linking theoretical issues with applied issues*, with an emphasis on what features of models of individual agents are necessary to model their interactions with technologies, environments, and other agents, and what additional requirements are needed to make them scalable to such larger complexities.

Two interrelated solutions that are currently in development to address the aforementioned objectives are presented in this paper: the first is the development of a concept for the integration of scalable cognitive models (where scalability is meant as an architecture design bridging micro- and macro-level cognition and behavior) with human behavior

representation (HBR) models, which are engineering models designed for M&S products and services. There have been numerous attempts to link low-level cognitive architectures to human-technology interaction (HTI) and multi-agent interaction models – all such models now generally fall under the label of sociotechnical systems (STS) modeling. We propose SoHBeR (Sociotechnical Human Behavior Representation), a tripartite model combining the ACT-R cognitive architecture, a sociotechnical systems model bridging ACT-R with a macro-cognitive framework, and task network models obtained from human factors best practices used in discrete-event simulations of performance and workload.

The second solution is the automated re-use of modeling data in HBR via the standardization of HBR taxonomy and structure. This research interest stems from the idea of reusing human factors models generated via all sorts of task analyses, to be translated as direct extensions of HBR models of synthetic agents. This amounts to transferring the knowledge gathered from human factors analyses into working models of intelligent agents. Some compromises have to be made by the concerned subject matter experts, such as in the way human factors analyses are conducted and data is compiled, as well as how HBR-specific programming is conducted. On the human factors side,

knowledge representations of goals, tasks, functions, etc. will have to follow a strict language to satisfy formalism constraints such as explicitness, completeness, and decidability, while on the HBR programming side, extensions will have to be created to accommodate higher-level constructs such as goals, operators to reach such goals, selection rules, planning schemas for networks of subgoals and subtasks, etc. The end product would be an automated human factors model-to-HBR script to generate on-the-fly intelligent agents in synthetic environments, fulfilling roles, functions, and goals gathered from human factors analyses. The extensions for the HBR modeling specification would be a candidate choice for inclusion in the Common Database (CDB) standard in the M&S community, such as XML metadata files to be seamlessly accessed via CDB development and use.

### 1.1 From Micro to Macro Cognition

There are multiple approaches to modeling human behavior and cognition, from artificial intelligence (AI) to cognitive modeling, to engineering models. While such approaches exhibit considerable variability in the features and techniques they select to further their ends, it is mostly through such ends that they can be established as distinct research endeavors. The widespread use of production rules (“if-then” or “condition-action” clauses) and artificial neural networks, for example, may obfuscate what roles and functions such specific algorithms are meant to implement.

*Artificial intelligence’s* stakes in cognitive modeling have been the most diverse, considering its pragmatically-driven nature. Simulation of cognition and behavior have been accomplished in “game AI”, via anything from physics engine algorithms (such as line of sight and collision detection algorithms) to scripting and heuristics, and are nowadays reaching sophisticated levels akin to the implementation of techniques borrowed from theoretical and applied AI research as found in Russell and Norvig (2009). Orkin’s (2006) review of the state of the art AI algorithm in the F.E.A.R game engine exemplifies this transition, from traditional finite state machines scripts to the more elegant STRIPS framework, the Stanford Research Institute Problem Solver for intelligent planning.

*Cognitive modeling*, in its purest academic and theoretical endeavors, uses biologically- and psychologically-inspired algorithms to simulate neural and mental processes in order to test theories of cognition. Production systems, neural networks, and hybrid cognitive architectures represent decades of research in an open community where a crosspollination of ideas helps fine-tune simulations in order to achieve more descriptive and predictive matches between experimental data and model outputs. The most successful and popular cognitive architectures are Anderson, Matessa and Lebiere’s ACT-R (1997), Kieras and Meyer’s EPIC (1997), and Laird, Newell and Rosenbloom’s SOAR (1987).

*Engineering Models* of “human behavior representations” (Pew & Mavor, 1998; Zacharias, MacMillan & Van Hemel, 2008) are pitched at task-level, human-environment interactions, by approximating through mathematical parameters and variables the impact of cognition and perception on agent performance and behaviors. By using discrete-event simulations, i.e. process simulations of state changes in a complex system, coupled with such mathematical constructs, commonly referred to as performance-shaping factors (Blackman, Gertman & Boring, 2008), task flows are simulated with degrees of input variability, and a range of process and output data are generated in order to assess human and technology interactions with regards to performance, effectiveness, workload, etc.

Some attempts at hybridization of various cognitive and behavioral modeling approaches have yielded a certain degree of success, promising more constraints and credibility in their claims by bridging gaps between agent-level model, component models such as neural networks for visual perception, and synthetic environment models. One such remarkable success story is SAL (figure 1), the Synthesis of ACT-R and LEABRA, a cognitive architecture and an artificial neural network programming architecture (Jilk, Lebiere, O’Reilly et al, 2008). The SAL model was successful in modeling multi-agent tactical activities in the UNREAL Tournament™ video game environment, by combining high-level planning and low-level perceptual elements of cognitive and neural architectures.

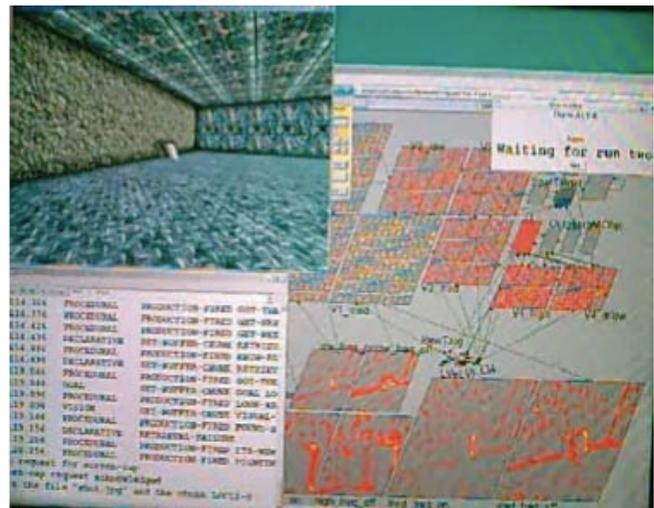


Figure 1: SAL (ACT-R architecture with a LEABRA visual perception module) in Unreal Tournament

Various attempts at integration between cognitive architectures and engineering models have also been made, from ACT-R and IPME – the Integrated Performance Modeling Environment, a discrete-event simulator modeling operator performance via task network models (Archer,

Lebiere, Warwick, et al, 2002), to Kieras' combination of EPIC and the GOMS approach (the HCI methodology of Card, Moran, and Newell, 1983, explained in section 2) into GLEAN, a tool to evaluate user interface design usability (Kieras, Wood, Abotel et al, 1995).

## 1.2 Limitations

Crystal and Ellington (2004) reviewed task analysis models and techniques in the area of human-computer interaction and observed two major issues shared by modeling approaches when it comes to human activity: they require increased *usability* and a higher degree of *integration*. The former is necessary because traditional task analyses are too long and/or complex to learn, difficult to perform, and once data is generated, it is hard to analyze and interpret. The latter issue concerns the tradeoff between efficiency (factoring usability, among other criteria) and effectiveness (factoring breadth and depth) of modeling techniques, with the assumption that specialized models could be combined to yield richer data than in isolation, yet having to remain tractable and usable. Those two sources of criticism of models of human activity can be leveled at the present topic of micro- and macro-cognitive modeling endeavors. We propose four problem areas for current practices in computational modeling of human behavior and cognition:

**Scope** Traditional modeling approaches are pitched at a specific level, whether neural, cognitive, behavioral, physical interactions with environment, swarm behavior, sociotechnical systems, or even models involving economics and politics. Trespassing on some of those boundaries would allow richer representations and more heuristic models to produce more realistic individual and multi-agent performances and predictive data.

**Interoperability** The isolated development of oftentimes proprietary algorithms aiming to model a subset of phenomena related to HBR hinders not only the transfer of knowledge from one modeling paradigm to another, but also that possibility of sharing data and bridging systems to be syntactically and semantically interoperable. A unified modeling approach, coupled with data format, validation, and interchange standards, specifically aimed at HBR interoperability, is needed to overcome the isolation of current and future HBR modeling practices.

**Reusability** HBR modeling paradigms are pursued in a fashion whereby models and data are tightly coupled together, thereby lacking "plug-and-play" capabilities: the overall architectures and algorithms, as well as the more specific models engineered through them, and data structures used to specify inputs are amalgamated or fused together, lacking modularity. In the words of Jones, Crossman, Lebiere, et al (2006), this could be done by "creating a clean distinction between the parts of a model

*that depends on the unique aspects of the architecture and those that do not*", among other strategies.

**Ergonomics** The learning curve to develop sufficient skills to understand, analyze, and tweak cognitive models is steep, let alone to develop one's own model. One needs to learn the capabilities and limitations of all aspects of the modeling architecture, the subtle differences between modeling paradigms, and comparing how a model fares with regards to other architectures requires the researcher to rewrite models from one modeling language to another.

## 1.3 Solutions Under Development

Our research proposes two solutions to overcome the limitations of current modeling approaches: (i) a unified modeling taxonomy and modeling framework, and (ii) the technological means to standardize such endeavors. The SoHBeR framework, Sociotechnical Human Behavior Representation, is aimed at multi-agent, flexible, and scalable HBR modeling, and is presented in section 2. A standardized, computational knowledge representation approach is presented in section 3, detailing SoHBeR XML data representation, validation and tools for interoperability. The modeling framework and standardization techniques rely on existing technologies and concepts from the literature in cognitive science, human-computer interaction, and human factors and ergonomics. Of interest to us are the ACT-R cognitive architecture, the GOMS modeling approach, the IPME software, the extensible markup language (XML), and the common database standards (CDB), some of which are also detailed below.

## 2. SoHBeR Modeling

The SoHBeR modeling framework is a conservative extension of the original GOMS technique to model operator tasks and behaviors from Card, Moran, and Newell's seminal work in the study of HCI, as presented in *The Psychology of Human-Computer Interaction* (1983). The scientists had developed a framework to analyze routine, expert-level use of a technology for a human operator by breaking down the task flow in goals, operators, methods, and selection rules (figure 2). Note that in GOMS, "operators" were merely a label to refer to a task or activity, while methods referred to compound tasks.

While HCI benefited greatly from GOMS models and analyzes for user interfaces and other workstation studies, with an emphasis on human error, performance, etc., the modeling framework has significant limitations: it does not address unpredictability in less straightforward and non-routine tasks, it is very much oriented towards the study of usability, not focused on functionality, and it requires extensive training to learn GOMS analysis. GOMS is thus geared towards routine, sequential tasks modeling, with a single operator, and does not fare well in the pursuit of HBR

involving dynamic, uncertain, and cooperative/competitive human activity, which involve decision-making, learning, task scheduling and prioritizing, and coordination between agents.

```

GOAL: DELETE-FILE
.
.
.
GOAL: SELECT-FILE
.
.
.
[select: GOAL: KEYBOARD-TAB-METHOD
.
.
.
GOAL: MOUSE-METHOD]
.
.
.
VERIFY-SELECTION
.
.
.
GOAL: ISSUE-DELETE-COMMAND
.
.
.
[select*: GOAL: KEYBOARD-DELETE-METHOD
.
.
.
PRESS-DELETE
.
.
.
GOAL: CONFIRM-DELETE
.
.
.
GOAL: DROP-DOWN-MENU-METHOD
.
.
.
MOVE-MOUSE-OVER-FILE-ICON
.
.
.
CLICK-RIGHT-MOUSE-BUTTON
.
.
.
LOCATE-DELETE-COMMAND
.
.
.
MOVE-MOUSE-TO-DELETE-COMMAND
.
.
.
CLICK-LEFT-MOUSE-BUTTON
.
.
.
GOAL: CONFIRM-DELETE
.
.
.
GOAL: DRAG-AND-DROP-METHOD
.
.
.
MOVE-MOUSE-OVER-FILE-ICON
.
.
.
PRESS-LEFT-MOUSE-BUTTON
.
.
.
LOCATE-RECYCLING-BIN
.
.
.
MOVE-MOUSE-TO-RECYCLING-BIN
.
.
.
RELEASE-LEFT-MOUSE-BUTTON]
.
.
.
*Selection rule for GOAL: ISSUE-DELETE-COMMAND
If hands are on keyboard, use KEYBOARD-DELETE-METHOD,
else if Recycle bin is visible, use DRAG-AND-DROP-METHOD,
else use DROP-DOWN-MENU-METHOD
    
```

Figure 2: original GOMS modeling example

### 2.1 SGOMS, and S<sub>2</sub>GOMS

The study of sociotechnical systems, i.e. the complex interactions between humans and technological environments, is a natural extension of HBR research endeavors, albeit a far more complex one. Macrocognitive models have barely been explored outside of the kingdom of artificial intelligence (see Sun, 2005, for a recent account of the state of the art in macrocognitive modeling research). West and Nagy (2007) set out to explore the possibility of reconciling micro- and macrocognitive modeling approaches by laying out a framework extending GOMS into the world of macrocognition, an endeavor which would combine the analytic power of GOMS concepts, methods, and results on the microcognitive level, with the potential of sociotechnical systems-level analysis of complex, multi-agent interactions.

Their SGOMS model (Sociotechnical systems GOMS, see figure 3) resulted in the realization that additional concepts and an extended theoretical framework were needed to bridge micro- and macrocognitive levels of analysis. Most significant of these concepts were that SGOMS requires the analysis of complex human activity in terms of planning units and unit tasks (where a planning unit is a super-ordinal construct via which unit tasks are organized and sequenced), with theoretical extensions for scheduling and coordination. Also worth noting is that the SGOMS model only makes accurate predictions when planning units may be interrupted, shed, and resumed, for coordinated activities.

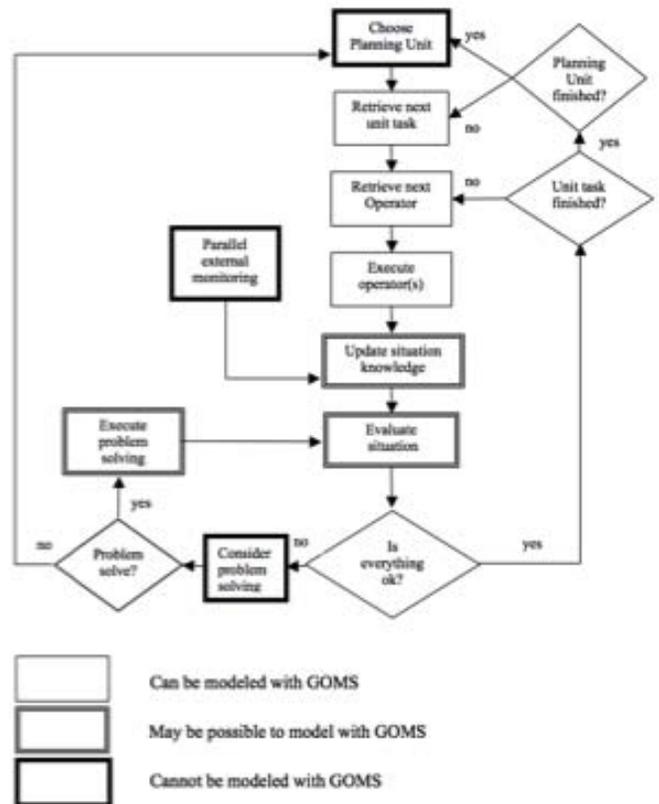


Figure 3: the SGOMS framework

Pronovost and West (2008ab) extended the SGOMS framework to account for strategic activities. The S<sub>2</sub>GOMS model (Strategic Sociotechnical systems GOMS) is applicable to strategic multi-agent interactions modeling, including cooperative and competitive interactions modeling, decision-making under uncertainty, and was tested in a low-fidelity synthetic environment in the form of a World of Warcraft™ video game scenario (with the additional goal of validating and promoting low-fidelity synthetic environments as computationally viable testbeds for academic research in HBR).

S<sub>2</sub>GOMS not only confirmed the theoretical claims and conceptual extensions of SGOMS by predicting the performance of unit tasks within planning units (figure 4), it also deliberately reduced the complexity of modeling decision-making processes by including decisions as planning units, following the rationale of Schultz (1997) in mapping the “estimate process” as specified in the military decision-making process (MDMP) of the US Armed Forces Joint Doctrine for Joint Operations, with the theoretical constructs of prospect theory in the cognitive psychology of decision-making (Kahneman & Tversky, 1977). West and Pronovost (2009) further demonstrated that it was theoretically possible for SGOMS models to be translated into ACT-R models, thereby allowing a microcognitive

theory in the form of a cognitive architecture to model macrocognitive, sociotechnical systems-level phenomena.

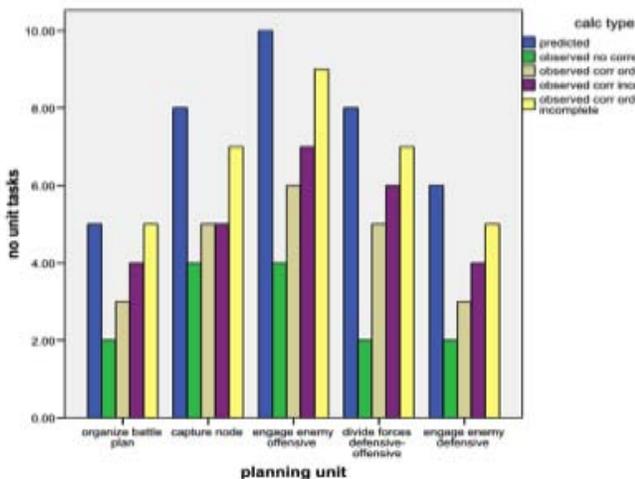


Figure 4: an example of S<sub>2</sub>GOMS’ predictive power

## 2.2 SoHBeR

SoHBeR, the Sociotechnical Human Behavior Representation modeling framework under development, is an attempt to unify traditional cognitive modeling with a sociotechnical systems (STS) theory and human behavior representation (HBR) engineering approaches. By bridging and combining the ACT-R cognitive architecture and the IPME task network modeling suite, guided by the S<sub>2</sub>GOMS framework presented above, it is hoped that HBR best practices would satisfy the requirements laid out in section 1.2, namely *scope*, *interoperability*, *reusability*, and *ergonomics*. The following section details how SoHBeR may provide the conceptual and technological means to implement this HBR modeling framework.

## 3. SoHBeR Standardization

While HBR models from all approaches achieve ever-increasing levels of complexity, augmenting in breadth and depth, we argue, along with other scientists (Crystal & Ellington, 2004, Jones et al, 2006) that they still don’t play well together because of *taxonomical* issues. All three approaches (AI, cognitive modeling, and engineering models) do not possess the necessary and sufficient theoretical framework and taxonomy to produce coordinated, multi-agent behavior in total interoperability, or even allow the transfer of a specific model and its data (inputs and outputs) from one modeling approach to another. How do we get various models of routine-like, expert, individual agency to scale up to models of dynamic and strategic, multi-agent behaviors under uncertainty?

What we need is to streamline the efforts towards integration and interoperability by means of establishing a common, abstract taxonomy to account for complex behavior (Jones et al, 2006), and we argue that this should be done via standardization across modeling and simulation (M&S) communities (Pronovost, 2009). Let us address the first question of interest raised by this previous statement: what are those taxons, exactly, and where do we find them? In artificial intelligence, they are broad in scope, vague in conceptualization, and scattered heterogeneously – from the procedural finite state machines consisting of sets of conditions-actions, to the planning AI incorporating goals, hierarchical structures for complex actions, etc. (Orkin, 2006). Cognitive Modeling generally yields more principled taxonomies and sets of “primitives” by virtue of being dependent on cognitive theories that are the underlying assumptions of cognitive architectures like ACT-R, EPIC, and SOAR. They use a mechanistic model where production systems determine behavioral outcomes based on productions rules coupled with inputs and past experience (declarative and procedural memories) (see Polk & Seifert, 2002, for a comprehensive overview of cognitive modeling). And engineering models, as we have seen in section 1.1, possess abstractions dealing with performance, workload, operator resources, and performance-shaping factors to express behavioral variability (Zacharias et al, 2008).

How do we go from there to achieve SoHBeR standardization? The commonalities in abstract, conceptual primitives found in modeling paradigms can be reduced to a small set of universals spanning from latencies, workload metrics, conditions and actions, goal-oriented behavior, etc., all of which can be in turn subsumed via hierarchical structures as found in human factors best practices, e.g. HGA (hierarchical goal analyses), MFTA (mission-functions-tasks analyses), unsurprisingly similar to HCI techniques such as GOMS. Once we decide which primitives are necessary and sufficient for a common modeling framework, as well as on a common structure to organize them, we can then move on to a translation of this taxonomy and this framework into XML data structures.

### 3.1 XML Knowledge Representation

SoHBeR representations, i.e. the data about goals, tasks, performance metrics, operator allocation, latencies, etc., need to be standardized in one format or another, and multiple options are available to this end. XML, the eXtensible Markup Language, already has more than a decade of history as a standard used to structure, store, and transport information. XML doesn’t “do” anything, it merely specifies a set of guidelines to follow to encode documents in a structured, digital representation of data, where the structure of the knowledge domain itself is arbitrarily defined hierarchically, with properties and relations, but has to make use of XML constructs such as markup notation and operators. Its syntax is simple, and

XML happens to be a candidate format for many types of software architecture outputs used across a variety of scientific and engineering applications. For our intents and purposes, XML happens to be the format of IPME outputs, of metadata in Common Database (CDB, reviewed in section 3.4 below) compliant files, is compatible with various tools used in human factors modeling such as Microsoft Visio, mind-mapping software, and finally, can be accessed by existing programming language libraries for Python, Java, and LISP, for which three different implementations of the ACT-R cognitive architecture have been produced. Figure 5 is an example of three tasks framed in an XML-compliant format using an XML editor.

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <Task_Flow xmlns="SoHBeR"
3   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4   xsi:schemaLocation="SoHBeR
5 <Task xmlns=""
6   <ID>0001</ID>
7   <Label>Setup Operator Workstation</Label>
8   <Operator>Radar Operator</Operator>
9   <Goal>Picture Compilation</Goal>
10  <Latency>5.0</Latency>
11  <TaskType>Discrete</TaskType>
12  <Verbal>0</Verbal>
13  <Auditory>1</Auditory>
14  <Cognitive>6</Cognitive>
15  <Psychomotor>7</Psychomotor>
16 </Task>
17
18 <Task xmlns=""
19   <ID>0002</ID>
20   <Label>Communicate Contact to Officer of the Watch</Label>
21   <Operator>Radar Operator</Operator>
22   <Goal>Picture Compilation</Goal>
23   <Latency>0.5</Latency>
24   <TaskType>Discrete</TaskType>
25   <Verbal>7</Verbal>
26   <Auditory>5</Auditory>
27   <Cognitive>5</Cognitive>
28   <Psychomotor>1</Psychomotor>
29 </Task>
30
31 <Task xmlns=""
32   <ID>0003</ID>
33   <Label>Track Contact</Label>
34   <Operator>Radar Operator</Operator>
35   <Goal>Picture Compilation</Goal>
36   <Latency>0.5</Latency>
37   <TaskType>Discrete</TaskType>
38   <Verbal>0</Verbal>
39   <Auditory>1</Auditory>
40   <Cognitive>6</Cognitive>
41   <Psychomotor>6</Psychomotor>
42 </Task>
43 </Task_Flow>
44

```

Figure 5: a SoHBeR-compliant XML data file

### 3.2 XML Schema

A very dire consequence of creating knowledge representations for reusability, interoperability, ergonomics, and augmenting the scope of HBR models would be to have to manually validate the datasets to be input into another HBR model or architecture, or to have to manually verify the consistency and legitimacy of their outputs. This is where XML Schema comes into play. In order to validate not only the compliance of data to XML standards, but to further validate any HBR data in XML format, one needs only create a template XML Schema to automatically verify whether data is missing or is improperly formatted. This will be the very core of the SoHBeR standardization effort: compliance validation through an XML Schema, called the SoHBeR XML Schema, part of which can be seen in figure

6 below. An XML Schema specifies how an XML data file should be formatted with regards to a Document Type Definition (DTD), a set of markup declarations determining the syntax of a document. In the case of SoHBeR, the elements and attributes of various data types refer to the expected labels, types, and values of the taxonomy established through the SoHBeR modeling framework. For example, an element tagged as being a “Goal” in any HBR XML file that purports to be compliant to SoHBeR standards would have to be of the type “string”, and this would be automatically validated by the SoHBeR XML Schema, as seen by comparing figures 5 and 6.

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <!-- schema xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns="SoHBeR" targetNamespace="SoHBeR" -->
3
4 <!-- comments format -->
5
6 <xs:element name="Task_Flow">
7   <xs:complexType>
8     <xs:sequence>
9       <xs:element name="Task" minOccurs="unbounded">
10        <xs:complexType>
11          <xs:sequence>
12
13            <!-- BASIC DATA -->
14
15            <xs:element name="ID" type="xs:string" />
16            <xs:element name="Label" type="xs:string" />
17            <xs:element name="Operator" type="xs:string" />
18            <xs:element name="Goal" type="xs:string" />
19            <xs:element name="Latency" type="xs:decimal" />
20
21            <!-- TASK DESCRIPTION -->
22
23            <!-- Effects data -->
24            <xs:element name="IDInboundTask" type="xs:string" minOccurs="0" />
25            <xs:element name="IDOutboundTask" type="xs:string" minOccurs="0" />
26            <xs:element name="Queue" minOccurs="0">
27              <xs:simpleType>
28                <xs:restriction base="xs:string">
29                  <xs:enumeration value="FIFO" />
30                  <xs:enumeration value="LIFO" />
31                </xs:restriction>
32              </xs:simpleType>
33            </xs:element>
34            <xs:element name="DecisionNode" minOccurs="0">
35              <xs:simpleType>
36                <xs:restriction base="xs:string">
37                  <xs:enumeration value="Multiple" />
38                  <xs:enumeration value="Probability" />
39                </xs:restriction>
40              </xs:simpleType>
41            </xs:element>
42
43            <!-- Timing data -->
44            <xs:element name="Distribution" minOccurs="0">
45              <xs:simpleType>
46                <xs:restriction base="xs:string">
47                  <xs:enumeration value="Normal" />
48                  <xs:enumeration value="Gamma" />
49                  <xs:enumeration value="Exp" />
50                  <xs:enumeration value="Rectr" />
51                </xs:restriction>
52              </xs:simpleType>
53            </xs:element>
54            <xs:element name="TaskType" minOccurs="0">
55              <xs:simpleType>
56                <xs:restriction base="xs:string">
57                  <xs:enumeration value="Discrete" />
58                  <xs:enumeration value="Continuous" />
59                  <xs:enumeration value="Repetitive" />
60                </xs:restriction>
61              </xs:simpleType>
62            </xs:element>
63            <xs:element name="Workflow" type="xs:string" minOccurs="0" />

```

Figure 6: the SoHBeR XML Schema (fragment)

### 3.3 XML Data Binding, Queries, and Transformations

An even greater benefit of the XML format is the capabilities for integration with programming interfaces that have been created to take full advantage of the data structures represented. Such application programming interfaces (APIs) are worth noting here, with regards to the capabilities that we anticipate will be of great use for HBR modeling. The *Document Object Model* (DOM) API allows the navigation of an XML document as a radial structure (a tree-like outline), treating XML entities as objects and properties, which in turn allows the binding of XML

elements to object-oriented programming declarations for scripting. *XQuery* allows users to retrieve information from XML data in the form of collections, a useful tool for database creation and maintenance. Should there be a need to alter the very structure of any or all of the HBR XML-compliant datasets or even the SoHBeR XML Schema itself, *XSLT* allows alterations of XML structures into novel syntax and data.

Since SoHBeR-compliant XML data is accessible via scripting for many types of APIs, integration with software from all modeling paradigms would be greatly facilitated. Python and LISP have their own XML DOMs, which would be directly interoperable with ACT-R, while IPME can benefit from C++, JavaScript and Python XML DOMs in a similar fashion.

### 3.4 CDB XML Integration

One of the ideas under review for a full-blown capability for HBR modeling interoperability is the inclusion of the SoHBeR XML Schema specification into the Common Database (CDB) initiative, a standardization effort initiated by Presagis Canada/USA Inc., a business specialized in modeling and simulation software solutions. The CDB is “*an open synthetic environment database specification*”<sup>1</sup>, whose entities are represented via five data formats: TIFF, GEO-Tiff, OpenFlight, Shapefile, and XML. This last file format is the one of interest, where all the metadata associated with a CDB-compliant entity is stored. It is hoped that the extension of the CDB specification with the SoHBeR XML Schema as a standard for HBR modeling would allow greater interoperability with M&S technology and various defence-oriented assets such as SAFs and CGFs (Semi-Automated Forces and Computer-Generated Forces), within a common data repository.

## 4. Discussion

There are anticipated benefits and a few limitations to this research endeavor, some of which are readily assessable, while others are dependent on factors both theoretical and practical in nature. The benefits can be segregated in direct, anticipated, and collateral benefits. The *direct benefits* are the establishment of necessary and sufficient features for a framework bridging individual agency and sociotechnical systems modeling, thereby linking cognitive architectures, applied cognitive engineering, and even human factors best practices via a common modeling framework and common knowledge representations.

The *anticipated benefits* address the limitations and derived requirements established in the introduction: the *scope* of a common HBR modeling framework will increase, bearing scalability from simple to complex agent-environment and

agent-agent interactions. Greater *interoperability* will be achieved via common data structures, used as inputs and transfers between algorithms. Algorithm- and platform-independent, modular data will yield data and model *reusability*. Finally, greater *ergonomics* will be achieved via the standardization of data structures for HBR in that there will be less to learn about for each and every new architecture or synthetic environment.

A very interesting anticipated *collateral benefit*, besides a reduction in costs, time and resources, is the increased capacity to make a more rigorous science out of HBR modeling. Indeed, by using identical inputs as independent variables, common data structures shared by the algorithms involved, and testing via some constrained variability (such as through discrete-event simulations), we could then measure and benchmark different algorithms in a much simpler way, therefore achieving a level of commensurability as of yet much harder to obtain. See Gluck & Pew’s (2005) presentation of the AMBR project, the Agent-Based Modeling and Behavior Representation model comparison effort, for an in-depth account of the hardships of model comparison.

There are of course some anticipated difficulties in the pursuit of such far-reaching endeavors. One mostly controversial theoretical difficulty lies in the apparent absence of strong isomorphisms between cognitive architectures and HBR models when it comes to their taxons. Indeed, there is no easy way to decide which processes, elements, and relations at one level of description, say, the cognitive processes of interest in the ACT-R cognitive architecture, would match which other processes, elements, and relations at another level of description, such as the task-level of human factors models used in HBR engineering models. An isomorphism is a mapping representing a relationship between objects, properties or operations, and such isomorphisms must be either discovered or arbitrarily chosen in order to achieve a common modeling framework. This is precisely the aim of efforts into bridging micro- and macro-cognitive models and theories of cognition and behavior (West & Nagy, 2007, Pronovost & West, 2008ab, West & Pronovost, 2009).

The future of SoHBeR lies into the achievement of further validation in simulation models and synthetic environments, using various modeling frameworks and architectures of human behavior representation. Such validation efforts can be made using low-fidelity video game engines as experimental testbeds, as well as more sophisticated SAFs/CGFs, but they must also match the experimental data of research in cognitive psychology. Other areas of inquiry of possible interest are the development of an OWL- (Web Ontology Language) compliant specification, in order to make SoHBeR directly translatable into a markup language to share data using ontology engineering, which would be useful to manipulate knowledge representations in inference

<sup>1</sup> <http://www.presagis.com/products/standards/cdb/>

engines such as description logic-based systems, the semantic web, etc. Finally, it may turn out that XML is not the best candidate format for run-time environments, so the JavaScript Object Notation (JSON) is under consideration, a less verbose data interchange format compared to XML that reduces data entry and even data processing overhead significantly.

## 5. References

- Anderson, J. R., Matessa, M., Lebiere, C. (1997). ACT-R: A theory of higher level cognition and its relation to visual attention. *Human Computer Interaction*, 12(4), 439-462.
- Archer, R., Lebiere, C., Warwick, W., Schunk, D., and Biefeld, E. (2002). Integration of task network and cognitive models to evaluate system design. *1st Conference of the U.S. Army Research Laboratory Collaborative Technology Alliances*, 9 April, 2002.
- Blackman, H. S., Gertman, D. I., and Boring, R. L. (2008). Human Error Quantification Using Performance Shaping Factors in the SPAR-H Method. *Proceedings of the 52nd Annual Meeting of the Human Factors and Ergonomics Society*.
- Card, S.K., Moran, T.P. and Newell, A. (1983). *The psychology of human-computer interaction*. Hillsdale, NJ: Lawrence Erlbaum Associates.
- Crystal, A., and Ellington, B. (2004). Task analysis and human-computer interaction: approaches, techniques, and levels of analysis. *Proceedings of the Tenth Americas Conference on Information Systems, New York, New York*.
- Jilk, D., Lebiere, C., O'Reilly, R. and Anderson, J. (2008). SAL: An explicitly pluralistic cognitive architecture. *Journal of Experimental and Theoretical Artificial Intelligence*, 20(3), 197-218.
- Jones, R. M., Crossman, J. A., Lebiere, C., & Best, B. J. (2006). An abstract language for cognitive modeling. *In Proceedings of the Seventh International Conference on Cognitive Modeling* (pp. 160-165). Trieste, Italy.
- Gluck, K. A., and Pew, R. W. (2005). *Modeling human behavior with integrated cognitive architectures: Comparison, evaluation, and validation*. Mahwah, NJ: Lawrence Erlbaum Associates Inc.
- Kahneman, D., and Tversky, A. (1977). Prospect Theory: An Analysis of Decision under Risk. *Econometrica* 47: 263-91.
- Kieras, D. E. & Meyer, D. E. (1997). An overview of the EPIC architecture for cognition and performance with application to human-computer interaction. *Human-Computer Interaction*, 12, 391-438.
- Kieras, D.E., Wood, S.D., Abotel, K., & Hornof, A. (1995). GLEAN: A Computer-Based Tool for Rapid GOMS Model Usability Evaluation of User Interface Designs. *UIST'95 Proceedings of the ACM Symposium on User Interface Software and Technology*.
- Laird, J. E., , Newell, A., and Rosenbloom, P. S. (1987). SOAR: An Architecture for General Intelligence. *Artificial Intelligence*, 33, 1-64.
- Orkin, J. (2006). Three States and a Plan: The AI of F.E.A.R. *Proceedings of the Game Developer's Conference (GDC)*.
- Pew, R. W., and Mavor, A. S., (eds.) (1998). *Modeling Human and Organizational Behavior: Application to Military Simulations*. Washington, DC: National Academy Press.
- Polk, T. A., and Seifert, C. M. (2002). *Cognitive Modeling*. Cambridge: MIT Press.
- Pronovost, S. (2009). Bridging Cognitive Modeling and Human Behavior Representation Engineering. *Proceedings of the 9th Conference of the Australasian Society for Cognitive Science*.
- Pronovost, S., and West, R. L. (2008a). A GOMS Model of Virtual Sociotechnical Systems: Using Video Games to Build Cognitive Models. *Proceedings of the European Conference on Cognitive Ergonomics*.
- Pronovost, S., and West, R. L. (2008b). Bridging cognitive modeling and model-based evaluation: extending GOMS to model virtual sociotechnical systems and strategic activities. *Proceedings of the 52nd Annual Meeting of the Human Factors and Ergonomics Society*.
- Russell, S. J., and Norvig, P. (2009). *Artificial Intelligence, a Modern Approach, 3<sup>rd</sup> edition*. Englewood Cliffs, NJ: Prentice Hall.
- Schultz, Lt. Col. J. V. (1997) *A Framework for Military Decision Making under Risks*. Maxwell Air Force Base, Alabama: Air University Press.
- Sun, R. (ed.) (2005). *Cognition and Multi-Agent Interaction From Cognitive Modeling to Social Simulation*. New York: Cambridge University Press.
- West, R. L. & G. Nagy (2007) Using GOMS for Modeling Routine Tasks Within Complex Sociotechnical Systems: Connecting Macro Models to Microcognition. *JCEDM*, 1 (2): 186-211.
- West, R. L., and Pronovost, S. (2009). Modeling SGOMS in ACT-R: Linking Macro- and Microcognition. *JCEDM*, 3 (2), 194-207.
- Zacharias, G. L., MacMillan, J., and Van Hemel S. B. (eds.) (2008). *Behavioral Modeling and Simulation: From Individuals to Societies*. Washington, DC: National Academies Press.

## Author Biographies

**SYLVAIN PRONOVOST** is a PhD candidate in cognitive science at the Institute of Cognitive Science, Carleton University, as well as a human factors consultant in the defence and aerospace sector, working at CAE Professional Services.

**ROBERT WEST, PhD** is a professor of psychology associated with the Institute of Cognitive Science, Carleton University, as well as head of the Carleton Cognitive Modeling laboratory (CCM Lab), affiliated with the Centre for Applied Cognitive Research (CACR).

## Procedure Design and Validation by Cognitive Task Model Simulations

*Tina Mioch*  
TNO Human Factors  
Kampweg 5  
3796 DE Soesterberg  
The Netherlands  
[Tina.Mioch@tno.nl](mailto:Tina.Mioch@tno.nl)

*Tomasz Mistrzyk*  
OFFIS Institute for Information Technology  
Escherweg 2  
26121 Oldenburg, Germany  
+49 441 97 22 560  
[Tomasz.Mistrzyk@offis.de](mailto:Tomasz.Mistrzyk@offis.de)

*Frank Rister*  
Hapag Lloyd Airways  
Alphonsstrasse 10  
22043 Hamburg, Germany  
+49 40 760 80 797  
[Frank.Rister@hamburg.de](mailto:Frank.Rister@hamburg.de)

### Keywords:

Task Models, Human Behavior, Pilot Procedures Design, Communication

**ABSTRACT:** *Currently, the main means of communication between air traffic control and the cockpit is the voice. However, non-auditive datalink communication via the flight management system is increasingly applied for air-ground communication. In this paper, we show that the procedure to handle voice communication with air traffic control is not adequate for datalink communication, as it would lead to less feedback in the cockpit and less active monitoring. The procedure is analyzed by visualizing it through the semi-formal task model AMBOSS, which also makes it possible to simulate the procedure step by step to evaluate safety-critical tasks, e.g. tasks for which there does not exist a safety net within the procedure, such as active monitoring by the other pilot. We argue that the current procedure needs to be adjusted to the changed communication in the cockpit, and we suggest and evaluate a new procedure.*

### 1. Introduction

Human error plays an important role in aviation accidents. The Federal Aviation Administration (FAA) estimates that human error contributes to 60-80% of all airline incidents and accidents, with communication, the governing factor for multi-crew cooperation, being its foundation (Wiegmann & Shappell, 2003).

As research and practice reveal, auditory and visual perception in the cockpit is in imbalance (Gordon et al., 2004). The perception of an auditory channel in a working environment that greatly relies on visual cues, such as the flight deck, is of considerable saliency (Wickens, 2003), whilst the long term working memory cannot store this information (Bredenkamp, 1998). Apart from lacking saliency, visual communication bears the advantage to be longer retainable and that it can be stored by technical means which make this information readily recallable at any time (Lee et al., 1999). This is one reason why the

implementation of datalink air-ground communication, embedded into flight management systems is assessed since the Mid-Nineties (Parasuraman, 2001).

The translation into practice of the datalink air-ground communication in the flight management system is still at its beginning: modern aircraft enable controller-pilot-datalink communication (CPDLC), a derivative of the aircraft communication, addressing and reporting system. This technology is currently tested in a trial-phase in Eurocontrol - upper airspace and is already applied for the reception of ground clearances at larger airports as well as in the North Atlantic Track (NAT-track) scheme. (Eurocontrol, 2007).

Typically, the pilot flying (PF) has direct access to aircraft control, including the auto flight system and the flight management system (FMS). According to the standards for workload management, manifested in most procedural standards documentations of the airlines, the areas of responsibility of the pilot

monitoring (PM) include systems control, such as hydraulics, fuel and pneumatics; and he is the one to communicate with air traffic control (Rister, 2005). As a consequence of the datalink air-ground communication being embedded in the flight management system, the responsibility of the PF and the PM would change according to the above mentioned standards. The communication with air traffic control, before a task of the PM, is done via the flight management system, which is part of aircraft control and is thus the responsibility of the PF.

### 1.1. Problem description

Datalink communication is on its way of becoming the standard way of communicating with air traffic control in the cockpit. This has direct consequences on the execution of procedures, as we will show by means of an analysis of a particular air-ground communication in section 2. However, the procedure that was in place for auditory communication, when applied in this new situation without substantial modifications, leads to safety critical problems. Neither the CPDLC-operators, nor the aircraft manufacturers have developed flight deck procedures yet which could solve these problems.

In the following, we argue that not adapting the procedure to the changed circumstances in communication leads to less redundancy in the handling of the situation and thus is less probable to withstand errors. We suggest a modification of the procedure, which combines the advantages of both the auditory procedure and the communication via datalink to minimize (unrecognized) errors in the cockpit and to re-establish the monitoring function as an active involvement in the task with a higher potential for shared SA (Endsley et al., 2003, Sarter & Woods, 1995). This new procedure is then validated by simulation to show that the redundancy is back in place and errors are less easily possible.

## 2. Analyses of Procedures

In this section, the different procedures and communication types are analysed. First, the current procedure to handle auditory communication is described. Second, the current procedure as it would be used for datalink communication if applied without modification is depicted. In addition, it is shown that the different mode of communication leads to a less safe handling of the communication by the procedure. At the end, a modified procedure is described that combines the safety of the first handling of the communication with a datalink communication.

### 2.1. Auditive Communication

The main means of current communication between air traffic control and pilots is voice transmission (radio). In Figure 1, a schema that depicts the communication between the different communicational partners is given. An uplinked ATC voice message is received by both pilots via headphones. The message that is radioed to an airplane is controlled and read back by the PM. Only if the PF receives the same message and only if the PF agrees with its contents and the PM's readback, this message will lead to its execution. If the PF does not agree with the message or with the PM's readback (which would mean that the two pilots have different mental models that inhibit shared SA), the proceduralized task distribution acts as a safety net. The PF simply only executes any clearance if he receives an ATC voice message and a PM's readback he both agrees with.

In the following, we are looking into the procedure in more detail to evaluate for which reasons errors could occur and how these errors are foreseen and intercepted by the procedure. There are three communicational partners involved, and the procedure is described for each of the partners.

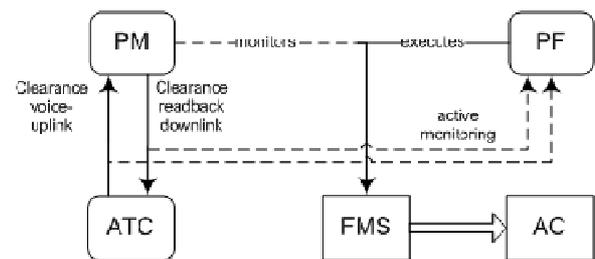


Figure 1: Two-Way Communication Rule with auditive communication for the task 'Handling an ATC Clearance'

**PM:** The PM receives the voice uplink. Voice has a high saliency (Wickens, 2003), so that an error that comes forth from not hearing the uplink is not very likely. In addition, as the PF also receives the uplink, he can counteract this unlikely error of the PM. The PM does a readback of the clearance. This means that the PM has to consciously process the input, as he has to reformulate and reproduce the heard information. This also includes a decision of whether this uplink makes sense and should be accepted. Only if acceptable, the readback is done by the PM. If the uplink is not acceptable, this is communicated to air traffic control, and the procedure starts again. It might be the case that the PM does not understand the air controller correctly. As he has to do a readback to PF, who also received the uplink, this possible error will be intercepted by the PF. The PM then monitors the execution of the clearance by the PF. As the PM has been actively involved in the task (i.e. through the readback and decision-making whether the uplink is

acceptable or not), the likelihood of consciously and actively monitoring the actions of the PF is high.

**PF:** The PF receives the voice uplink and hears the readback of the PM. The PF might have understood the uplink differently (either through interpreting it differently or through actually hearing something different). This error is intercepted in this step. If both pilots did understand the air controller wrongly, but both in the same way, this will not directly be caught by the PF, but by the air controller, who is also listening to the readback. The PF actively has to compare his own mental model with the readback of the PM, and makes the decision whether to execute the clearance. If the clearance is acceptable, he executes it.

**ATC:** The air traffic controller initiates the voice uplink. He hears the readback of the PM, and in the case of the readback being wrong, the controller can directly intervene and repeat the uplink.

The errors that can occur in the communication, monitoring or execution tasks of other steps in the procedure are all intercepted by a safety net that is implicit to the procedure. Every possible error is foreseen (or very unlikely) and is recognized either by the person making the error or by one of the other conversational partners.

This safety net also works when either the PF does not perceive or understand the message, or if the PF misses the PM's readback (absence of active monitoring, lower dotted arrow in Figure 1). Should the PM fail to perceive or understand the message (absence of the active, solid arrow between the PM and ATC), the PF would also refrain from executing any FMS changes, as he would lack the readback for proper comparison with the message (absence of upper dotted active monitoring arrow).

## 2.2. Non-auditive Communication

If the voice-messages are replaced by CPDLC, the received message is stored in the FMS. Using datalink has several advantages compared to voice communication. First, the pilots do not need to memorize the information provided by air traffic control. The information is set in the system, and is available at all times during task execution. If there is uncertainty about the uplink information, the pilot can just check the message again. Second, as the pilots do not need to memorize the information (and recall it when executing the procedure), the pilots experience less workload. If there is less workload, there is less probability of errors in retrieving the information (Wickens, 2003).

The FMS, in which the datalink messages are stored, is the same system with which the PF typically flies the airplane. For that reason, it is the PF who processes and executes the incoming messages, which then would have a direct effect on the airplane's trajectory.

In the following, we are looking in more detail into the procedure to evaluate for which reasons errors could occur and whether the errors are foreseen and intercepted by the procedure.

**PM:** The PM monitors the FMS and receives the data uplink. No action is involved for the PM when receiving the uplink. He (passively) monitors the execution of the uplink by the PF. If an error occurs at this point of the procedure, e.g. omission of the monitoring task, there is no safety net for intercepting this omission.

**PF:** The PF monitors the FMS and when receiving the data uplink, he has to decide whether to execute the clearance. Execution of a clearance is done by pressing the WILCO button, which represents compliance to the ATC's request). There are several errors that might occur. First, it is possible that the PF does not see the uplink. However, the likelihood of this error is not higher than for the current procedure, as all datalinks are additionally accompanied by an aural signal. As the PM is also monitoring the FMS, the probability of none of them seeing the uplink is small. Also, it is possible that the PF has a wrong interpretation of the uplink or that he makes an error in the decision-making process. Here, we can differentiate between the following possible consequences:

1. The PF makes a wrong decision. This only will be recognized by the PM if he is actively monitoring the execution of the uplink. If the PM is not monitoring the execution of the task (either not at all or only superficially), there is no safety net in this procedure to intercept a wrong decision of the PF. The PF does not know whether the PM is actively and reliably monitoring the PF's task execution.
2. The PF's wrong interpretation or decision-making of the uplink leads to the right decision. The wrong mental model is not recognized by the PM. This does not directly lead to a problem, as the action is correctly implemented by the PF, but it also does not lead to the recognition of the wrong mental model, which might lead to errors later on.

Note that it is solely the PF who has to exercise active, cognitive processing of the uplink. He is the only one involved in the clearance execution process. The readback, which should be understood as the acknowledgement of the uplink whether silent or aloud

as in the first procedure, is a task that rests solely by the PF. The PM's role becomes passive. Even though he still has the monitoring function, his possibilities to e.g. deliver his mental model for shared SA-building to the PF is restricted. The safety net becomes leaky. Neither does an active communicative action link the PM with ATC anymore (for reception and readback), nor does the PF have an opportunity for synchronization. A modification of the procedure which could allow the PM to operate the FMS would not help, as feedback would still be missing. The situation would be mirrored and the PM would involuntarily take over duties of PF which contradicts task distribution principles as laid down in the Standard Operating Procedures (SOP).

That means that even though there are some advantages of using datalink communication (e.g. that the information is available during task execution without having to memorize it), the procedure such as it is less safe, as just one pilot needs to make an active decision. As decision-making is an error-prone activity (it costs a lot of effort and is susceptible for shortcuts), there should be a safety net in place that includes active involvement of both pilots.

### 3. Procedure Design

In this section, the existing procedure is modified to account for the new technological circumstances and to close safety gaps. The resulting modifications are validated by simulation, producing a new flight deck procedure. But first of all, the purpose of task modelling in this context is discussed.

Task models are an elementary part of human-machine interaction. Models show which logical steps are necessary in a task to achieve a defined goal. Existing modelling approaches (e.g. K-MADe – Cafiau et al., 2008, VTMB – Biere et al., 1999, CTTE – Mori et al., 2002, Task-Architect- Stuart & Penn, 2004) allow for task and subtask specifications as well as for their relative timeframes to be set. The task hierarchy displays a detailed description of task allocations by one or more users in a complex environment. Hierarchical task models relate formally defined structures, such as hierarchy and temporal relations, with informal elements, such as additional description of a task.

For our procedure, we decided to use the freeware modelling environment AMBOSS (AMBOSS, 2009). Due to its enhanced concepts and flexible vantage points, AMBOSS represents a useful tool for task modelling in socio-technical and safety-critical systems (Giese et al., 2008). The modelling environment has been specially expanded for the specification of tasks in safety-critical systems and now allows for inspection of relevant aspects, first of all communication (Mistryk & Szwillus, 2008). In AMBOSS, it is possible to model communication between non-neighbouring tasks and to implement message objects. Message objects reveal how, why, by whom and for whom an information is being generated. Similar to other modelling tools (e.g. Cafiau et al., 2008, Biere et al., 1999, Mori et al., 2002), it enables to specify the roles of actors within a hierarchy. This allows for more transparency of the task-role-communication relationship than with any other modelling approach.

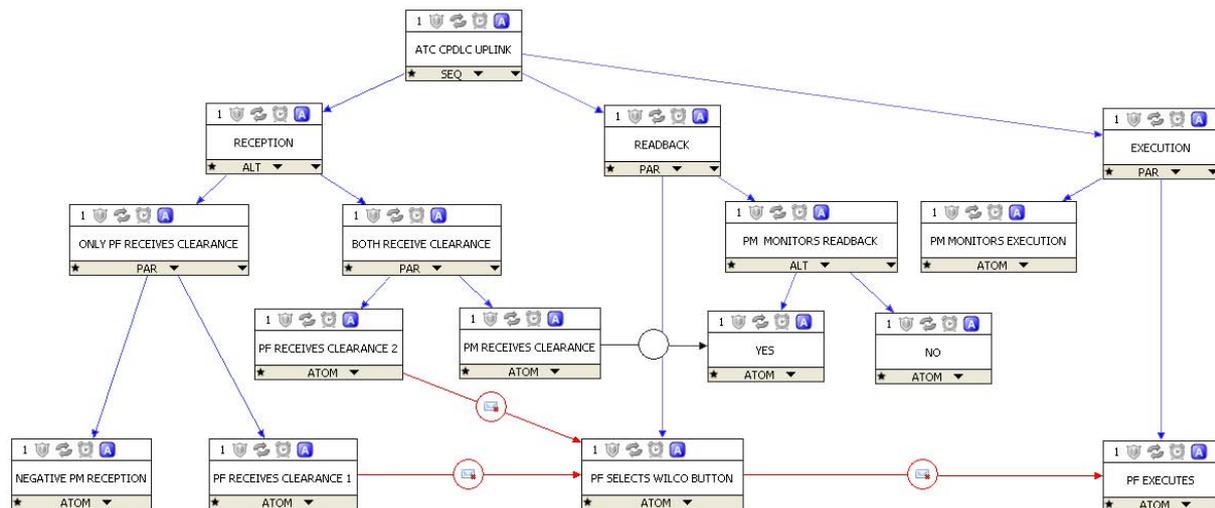


Figure 2: Non-auditive communication in a task model without active PM readback for the task 'Handling an ATC Clearance'

#### 3.1. Task modelling

AMBOSS allows to determine whether a communication event is classified as critical. Critical

communication events can be optically augmented. Furthermore, it can be determined whether a communication event serves as a trigger for a subtask. Additionally, it is possible to specify the necessity of feedback and to fill each event with detailed text.

Just as the approaches of K-MADe (Cafiau et al., 2008), CTTE (Mori et al., 2002) or VTMB (Biere et al., 1999), AMBOSS provides its own simulator which enables an interactive validation of contexts in a task model. Flow of information, triggers, as well as the task hierarchy and its temporal relations are considered by the simulation. The AMBOSS simulator is based on the concept of 'Enabled Task Sets' (Mori et al., 2002). This concept provides a presentation of executable tasks. The ability of AMBOSS to simulate task models enables the analysis of pilot interaction in a socio-technical safety-critical system step by step. Thereby, experts are able to simulate various scenarios of task models and to compare them. This kind of validation helps to check the correctness of a task model and to find weak points. In situations in which several tasks are ready to get activated, the user can determine the sequencing of tasks. This enables the modeller to thoroughly examine chosen sequences of the task model for potential problems. Such shortfalls occur, as model simulations reveal, due to incorrect task-sequencing, lack of information transfer, non-observability of problematic instances but also due to unreflected workload distribution amongst the actors as well as due to tense scheduling of the task processing.

### 3.2. Modelling of non-auditive communication

Figure 2 shows the graphical representation of a Task Model in a tree like format which depicts a procedure for non-auditive communication. One of the challenges related to modelling socio-technical systems is to introduce communication and its parameters in a model. In the model the communication is depicted as ovals. The red ovals symbolize critical communication, whereas white oval represent regular communication.

Transferring the communication models into task models, the non-auditive model's simulation results do not get influenced by the omission of redundant tasks and messages, such as the monitoring task of the PM (subtask: *PM RECEIVES CLEARANCE*). No matter which irregularities cause the disturbance of the PM's subtasks, the overall task (Handling an ATC clearance) will be executed anyway – the temporal relations as well as the trigger messages between the PM-subtasks do not necessarily guarantee the utmost necessity of the PM functions for this overall task (Figure 2). For example, if only the PF processes the uplink message, he is not be restricted by the PM at all, as there is no need to act for the PM. The reception has an alternative

temporal relation, allowing just one subtasks of several alternatives to be executed. The necessity of processing as well as the readback monitoring becomes obsolete. The stage is set for a PF solo. If both pilots perceive the received message, the PF processes the message in the FMS. The PM lacks the non-auditive means to monitor or intervene in the PF's performance. The task *PM MONITORS READBACK* comes with an alternative temporal relation, which is no prerequisite for completion of the entire task.

### 3.3. Overview of auditive communication

If one of the subtask branches of auditive communication is being destroyed, such as the reception of the uplink by the PM, the overall task, the handling of the uplink, remains incomplete. Both pilots, the PF and the PM, are dependent on reception before the PM is able to initiate a task-relevant readback. This requires that both subtasks, the reception of the uplink by both the PM and PF, have to be fulfilled before it can be proceeded; in an AMBOSS model, this would be reflected by a temporal parallel relation. Furthermore, trigger-messages that couple the subtasks of the reception of the uplink with the readback are necessary prior to initiation of the execution by the PF. Trigger messages represent the conscious processing of a received uplink. Without such cognitive processing, the subtask receiving the trigger message cannot be executed.

### 3.4. Description of the developed procedure

The simulation as well as the comparison of the previous two models leads to the conclusion that a new procedure shall actively re-insert the PM into the subtasks *RECEPTION* and *READBACK*. The new procedure developed by the authors focuses on dual access to the FMS by both pilots (Figure 3). We argue that this new procedure combines the advantages of both the other two procedures, and is thus safer than the datalink procedure that is currently implemented. The idea is to re-establish the monitoring function of the PM as an active involvement in the task.

In the following, we are looking in more detail into the new procedure, which is given in Figure 3, to evaluate for which reasons errors could occur and whether the errors are foreseen and intercepted by the procedure.

**PM:** The PM monitors the FMS. When an uplink is sent, The PM needs to act on this uplink. He needs to make a decision whether to accept the uplink, and consequently accept it. An error might occur because the PM does not see the uplink, e.g. because of focusing his attention elsewhere. This error-probability is minimized through introducing an aural signal when receiving an uplink, so that the saliency does not differ

from the other two procedures. Additionally, because the PF also receives the uplink and has to act on it, he will, after some time, point out to the PM that there is an uplink waiting for evaluation. Another error that might occur is that the PM interprets the uplink incorrectly or makes a wrong decision. In this case, again two different consequences can be identified:

this is not caught with this cross check. However, the probability of both pilots making an error in the same step is small, as both pilots are actively and likely cognitively, involved in executing the uplink.

By executing an uplink in the FMS, the PF automatically delivers the task-relevant area of his

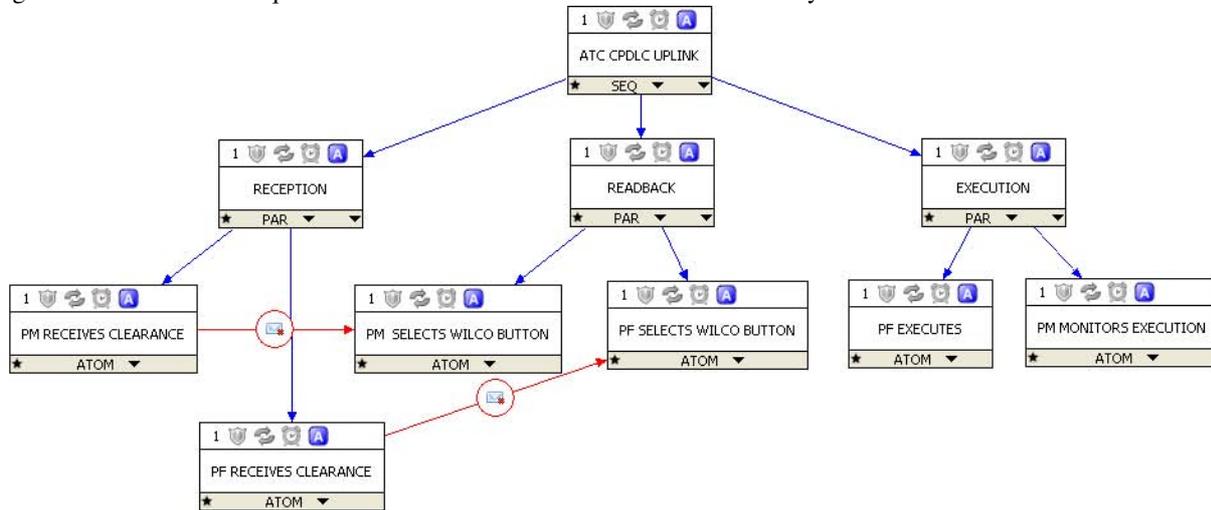


Figure 3: Non-auditive communication in a task model with active PM Feedback

1. The incorrect interpretation or decision leads to an error (either because the uplink is erroneously accepted or rejected). For this error, the PF is the safety net, as he executes the same task, and if he makes the correct decision, the difference will be found by the cross check of the system. This will lead to additional communication between the pilots.
2. The incorrect interpretation or decision does not lead to an error. The PM has a wrong mental model or makes the decision for the wrong reasons. As this does not lead to an error, it cannot be intercepted by the PM.

The PM has to actively decide whether the clearance should be executed. Here, the PM might make the wrong decision because of a wrong mental model or a bias in his decision-making process.

**PF:** The PF monitors the FMS. The procedure for the PF is the same as for the PM, and might lead to the same errors and has the same safety net. The actions are mirrored.

**System:** The task of the system is to cross check whether the two pilots have accepted (or not accepted) the uplink. This cross check intercepts possible errors that might occur in the actions of (one of) the pilots before. If both pilots make an error in the decision-making of whether to accept the uplink, and the uplink is accepted even though it should not be accepted,

mental model to the PM. As both pilots need to check, acknowledge and execute the uplink, it is assured that their mental models about this uplink do not contradict each other.

This procedure has the advantages of datalink communication and that both pilots are actively involved in the decision-making of accepting the uplink. The probability of errors decreases, as both need to come independently to a conclusion.

The new task model is safeguarded against inadvertent solos of the PF as the parallel relation of the two *RECEPTION* subtasks requires both pilots to receive the clearance in order to release trigger messages which are necessary for a successful completion of the sequence's subtasks, here *READBACK*. Without such, the last task, *EXECUTION* will miss in the overall sequence. The received message will not gain access to aircraft control.

The new procedure does not impair the PF's controllability of the airplane: the acknowledgement by the PM to execute a certain action, normally received verbally by the PF, remains silent; but as the PM needs to also press the WILCO-BUTTON and with it acknowledge and accept the uplink, the PF knows that the acknowledgement has been given.

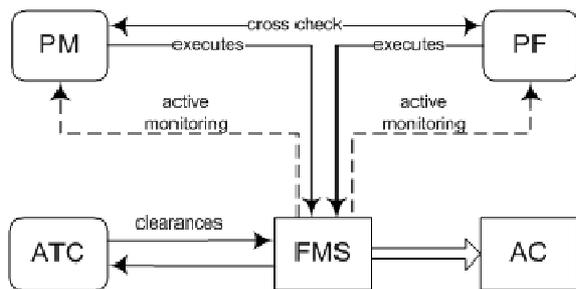


Figure 4: The new non-auditive communication procedure

Figure 4 provides a procedural visualization of the developed model. Here it becomes obvious that both pilots are required to show active monitoring as both need to accomplish an acknowledgement task.

Non-normal cases such as one pilot being either incapacitated, or simply not present on the flight deck, are covered by this procedure. For such a situation, the FMS has to be programmed to allow for dual execution out of the same seat (with a special reconfirmation bug to be programmed). This enables the PF and, regardless of his role, finally the commander to gain full and if needed sole authority over the aircraft whenever deemed necessary. The models in Figure 3 and Figure 4 remain unchanged as the PF in this special situation would simply take action in lieu of the PM which will complete the entire sequence of subtasks and finally the overall task.

#### 4. Discussion

We have shown that the current procedure for handling datalink communication is not sufficient to guarantee safety. We suggested modifications to the procedure, and showed that these suggestions lead to a safer procedure.

Our developed procedure can be operated independently of the accessible hardware and independently of the FMS's embedding grade. It requires no structural work, just software adjustments will become necessary and it complies with the Rules of Good Airmanship.

As described above, for several safety reasons, a dual access to trigger the WILCO BUTTON from either seat needs to be possible. This can be regarded as a shortfall, as only daily operation can reveal whether this feature will exclusively restrict to single pilot operations.

#### 5. Acknowledgement

The work described in this paper is funded by the European Commission in the 7th Framework

Programme, Transportation under the number FP7 – 211988.

#### 6. References

AMBOSS, (2009). Homepage of AMBOSS Project, University of Paderborn, Faculty of Engineering, Computer Science and Mathematics. (viewed December 1, 2009) <http://mci.cs.uni-paderborn.de/pg/amboss/>

Biere, M., Bomsdorf, B., Szwillus, G. (1999). Specification and Simulation of task models with VTMB. *In CHI 1999 Extended Abstract on Human Factors in Computing Systems*, New York CM Press, pp. 1-2.

Bredenkamp, J. (1998). *Lernen, Erinnern, Vergessen*. München: C.H. Beck

Cafiau, S., Girard, P., Scapin, L.,D., Guittet Laurent, S Sanou L., (2008). Assessment of Object Use for Task Modeling *In Proceedings of the International Workshop on Task Models and Diagrams*, TAMODIA, pp. 14-28

Endsley, M. R., Bolté, B., & Jones, D. G. (2003). Endsley, M. R., Bolté, B., & Jones, D. G. (2003). Designing for Situation Awareness: An approach to human-centered design, (pp. 193-261). London: Taylor & Francis.

Eurocontrol. Controller Pilot Datalink Communication (CPDLC) Implementation Plan. [URL:http://www.eurocontrol.int/link2000/gallery/content/public/files/documents/NPD\\_3.4](http://www.eurocontrol.int/link2000/gallery/content/public/files/documents/NPD_3.4). Pdf. Edition 3.4 (05/2007). Retained from Internet on August, 30<sup>th</sup>, 2009.

Giese, M., Mistrzyk, T., Pfau, A., Szwillus, G., von Detten, M. (2008) AMBOSS: A Task Modelling Approach for Safety-Critical Systems, *In Proceedings of the International Workshop on Task Models and Diagrams*, TAMODIA, 98-109

Gordon, S.E., Lee, J.D., Liu, Y., & Wickens, C.D. (2004). Cognition: Attention and Time Sharing. *In: An Introduction to Human Factors Engineering*. Upper Saddle River, NJ., Prentice Hall.

Lee, J.D., Gore, B.F., Campbell, J.L. (1999). Display Alternatives for In-Vehicle Warning and Sign Information: Message Style, Location and Modality. *In Transportation Human Factors Journal*, vol. 1 (4), 347-377

Mistrzyk, T., Szwillus, G. (2008). Modellierung sicherheitskritischer Kommunikation in Aufgabenmodellen, *In Zeitschrift für interaktive und kooperative Medien i-com*, Oldenburg Wissenschaftsverlag, 39-42

Mori, G., Paternò, F., Santoro, C. (2002). CTTE: support for developing and analyzing task models for interactive system design. *In IEEE*

- Transactions on Software Engineering* vol. 28 (8), 797-813
- Parasuraman, R., Hilburn, B., & Hoekstra, J. (2001). *Free Flight*. In W.Karwowski (Ed.), *International Encyclopedia of Ergonomics and Human Factors*. New York: Taylor & Francis.
- Rister, F.W. (2005). Procedural Standards - Workload Division. Operations Manual Part B, Volume 1.02.31.00 In *Flight Procedures Inflight*. 5-16
- Sanderson, P.M., Harwood, K. (1988). The Skills, Rules and Knowledge Classification: A Discussion of its Emergence and Nature. In Goodstein, L.P., Andersen, H.B., Olsen, S.E. (Ed.), *Tasks, Errors and Mental Models*. London: Taylor & Francis.
- Sarter, N.B. and Woods, D.D. (1995). *Strong, Silent, and Out-Of-The-Loop: Properties of Advanced (Cockpit) Automation and Their Impact on Human-Machine Coordination*. Cognitive Systems Engineering Laboratory (CSEL) Technical Report No. 95-TR-01.
- Stuart, J., Penn, R. (2004). Task-Architect: taking the work out of task analysis. In *Proceedings of the 3<sup>rd</sup> Annual Conference on Task Models and Diagrams, TAMODIA*, vol. 86, ACM Press, New York, 145-154.
- Swain, A.D., Weston, L.M. (1988). An Approach to the Diagnosis and Misdiagnosis of Abnormal Conditions in Post-Accident Sequences in Complex Man-Machine Systems. In Goodstein, L.P., Andersen, H.B., Olsen, S.E. (Ed.), *Tasks, Errors and Mental Models*. London: Taylor & Francis.

Wickens, C.D. (2003). Pilot Actions and Tasks: Selection, Execution and Control. In *Principles and Practice of Aviation Psychology*. Mahwah, NJ: Lawrence Erlbaum Associates.

Wiegmann, P.A., & Shappell, S.A. (2003). *A Human Error Approach to Aviation Accident Analysis: The Human Factors Analysis and Classification System*. Ashgate Publishing

## Author Biographies

**TINA MIOCH** is a researcher at TNO, department Human Factors. She received a M.Sc. in Agents and Computational Intelligence from Utrecht University. Her research is on cognitive modeling and artificial intelligence. More specifically, she is interested in human error modeling, cognitive architectures and agent modeling.

**TOMASZ MISTRZYK** is a Researcher at OFFIS, in the R & D division transportation. He received a Diploma in Business Information Systems from University of Paderborn. His research is on task analysis and communication in safety-critical computer systems. Particularly he focuses on task modelling and human error modelling.

**FRANK RISTER** is a training and acceptance test pilot at Hapag – Lloyd. He flies Boeing B737 and Airbus A320 aircraft and is involved in the development of training programmes, simulation scenarios and procedure design. He has an M.Sc. in Aeronautics and Human Factors.

## Process Modeling for the Study of Non-State Political Violence

*Olivier L. Georgeon*

*Jonathan H. Morgan*

The College of Information Sciences and Technology  
The Pennsylvania State University, University Park, PA 16802  
olg1@psu.edu, jhm5001@psu.edu

*John Horgan*

*Kurt Braddock*

The International Center for the Study of Terrorism  
The Pennsylvania State University, University Park, PA 16802  
jgh11@psu.edu, khb125@psu.edu

### Keywords:

Knowledge representation, Pattern recognition, Socio-cultural modeling, Terrorism

**ABSTRACT:** *Terrorism studies have and continue to face conceptual and analytic challenges that stem from the assumption that terrorism can be understood outside of its social and political context, as essentially a ‘state’ of being and/or set of personal qualities specific to the terrorist (Sageman, 2004; Taylor & Horgan, 2006). An under-explored alternative to this view is to see involvement in terrorism, at least in psychological terms, as a process rather than a state. One consequence of this is that we shift the focus away from individuals and their presumed psychological or moral qualities to an examination of process variables. These, by their nature, are more susceptible to change and thus form the basis of developing interventions. Interpreting these variables, such as changes in operational context or relationships between temporal events and individuals, requires tools capable of capturing time-sensitive semantic content. To date, there are few process-oriented tools and fewer analyses of terrorism data using these tools. In this paper, we present such a tool and offer an initial application for expanding and formalizing computationally our understanding of terrorism.*

### 1. Introduction

A major obstacle to greater conceptual development in the study of terrorism has been the assumption that we can understand terrorism outside of its social and political context. This has given rise to the view that terrorist acts essentially can be understood as stemming from an identifiable ‘state’ of being that can be analyzed to make predictions. Though popular, this assumption and the emphasis on static qualities that is implied by such an approach has proven ineffective, particularly in the development of meaningful counterterrorism initiatives (Horgan, 2009). Alternatively, it may be more valuable to consider involvement in terrorism (and political violence more broadly) as reflecting a complex process rather than a state.

Studying terrorism as a process makes us shift our focus from the individual and their presumed psychological or moral qualities to process variables. We can then begin to ask how changes in operational context, or how the relationships existing between events and the individual affects behavior (Taylor & Horgan, 2006). This is particularly important when considering how we might

formulate strategies for managing and controlling the extent of terrorist events (Horgan, 2009).

In addition, as Taylor and Horgan (2006) note, considering terrorism as a process would be consistent with the way we tend to study other forms of illegal behavior such as criminality. A further benefit that follows from this is that our attention transitions from addressing the qualities of individuals (i.e., personality or “evil traits”) that draw on intangible mentalistic concepts (that are, by definition, resistant to change and not visible) to identification of essentially tangible, practicable, and alterable matters. Moving our level of explanation away from properties to processes seems to offer tangible rewards beyond mere conceptual adequacy, and may offer a different approach, for example, to the development of more practical and efficient counterterrorism initiatives.

What then does assessing “terrorism as a process” imply? In this paper, we use the definition of process developed by Taylor and Horgan (2006) in that we are essentially describing a sequence of events, involving steps or operations that are usually ordered and/or interdependent. We therefore seek to understand terrorist activity as a set

of actions and reactions, often expressed in a reciprocal relationship in both an immediate and long-term sense between various actors. These actors can include but are not limited to: governments, terrorists, the media, the police and security services, politicians, and the civilians in general. As Taylor and Horgan explain, “the nature of that reciprocity may be expressed in a variety of ways, but it is important to note, however, that specifying or identifying the elements of the process does not necessarily imply a simple deterministic account, despite the ease with which such accounts may follow from post hoc analyses of events” (Taylor & Horgan, 2006, p 585).

In addition, describing activities as indicative of a process allows us to consider modeling events and their relationships. As Taylor and Horgan (2006) explain: “Modeling can take a variety of forms, and perhaps a continuum can be expressed between identifying and expressing mathematical or statistical probabilities about the relationships between events, and conceptual models of their relationships expressed as hypothetical constructs and intervening variables.”

In this paper, we introduce a tool and initial trace modeling approach for expanding and computationally formalizing our knowledge of terrorism processes. We first introduce trace-modeling approaches as means of addressing the growing data/knowledge gap found in the social sciences. We then discuss the limitations of classic activity analysis. We move on to discuss process modeling using trace-modeling methods, providing a brief specification and offering a process-oriented trace-modeling tool, ABSTRACT, to support the modeling of terrorist activities. We follow this discussion with a description and analysis of an example trace developed from the Global Terrorism Database<sup>1</sup> (GTD). We then conclude with a brief discussion and review, noting challenges and implications of this modeling approach.

## **2. Addressing the data/knowledge gap**

The data that may potentially inform us about terrorist processes is diverse. It can range from established sources such as intelligence reports and field work, case studies, and centralized logs of terrorism activity like the GTD to emerging media types such as chatroom logs, tweets, and other life streaming sources. For data, however, to inform us about a process, it must entail chronological information. Such data constitutes what we call a chronological activity trace. A chronological activity trace can be seen as a timeline of concrete or abstract events in which the analyst can find relations of causality between events, by referring to possible explanative theories.

Finding this network of abstract events and causal relations is challenging. This challenge raises a problem that we refer to as the data/knowledge gap. In essence, this challenge arises from an epistemological issue—the fact that to understand data we need previous knowledge, but to have previous knowledge we need to understand data. This is a general problem that is often related to Popper’s (1972) evolutionist theory of knowledge. In this article, we limit our focus to addressing two dimensions of this issue: a) the gulf between disciplines (primarily between toolmakers and tool-users), and b) the conceptual gap in our understanding of terrorism.

On one hand, we have high-level descriptions of terrorist activity formulated over multiple decades and drawing primarily upon interviews, court transcripts, and case studies coming from the direct experiences of researchers. These theories continue to offer insights, but their dependence upon a relatively small set of retrospective accounts limits their predictive power. From these sometimes inscrutable and always evolving accounts (a snapshot view), researchers attempt to identify the dynamics of a fluid, time-sensitive, and frequently reflexive set of processes.

On the other hand, there is a growing store of low-level granular data of multiple types. Finding patterns or processes in this kind of low-level data continues to be a challenging research area, as examples in other domains of human activity show, e.g, car driving activity (Georgeon, 2008). Though this data potentially offers a means of evaluating and refining our theories, constructing a useful interpretation of this data is not only a difficult challenge for the social sciences but also for the information sciences—a challenge neither community can surmount in isolation. Social scientists will require tools to interpret data; information scientists require the expertise of social scientist to ensure both the relevance and applicability of those tools and data.

Furthermore, the success of such tools is likely to vary in relation to the tractability of the process or sub-process we are studying. While online recruitment by terrorists generates large volumes of data, we are much less likely to fully capture the influence of idiosyncratic or contingent factors, or formulate a complete picture of processes whose participants systematically destroy or distort the data necessary to understand that process. For example, collected data seldom entails information about underlying social mechanisms. Consequently, social scientists must hypothesize, based upon incomplete information, the existence, relative significance, and operation of these processes (Hedström, 2005). We, therefore, must be realistic about our ability to predict terrorism, and rather confine ourselves to attempting understand and potentially predict certain terrorist

activities and processes.

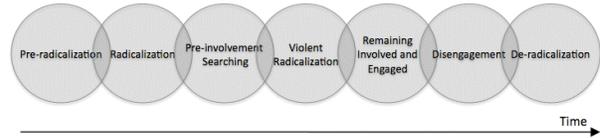
We address the data/knowledge gap by using an iterative and reciprocal top-down/bottom-up approach, drawing downwards from models proposed by experts and upwards from granular data. This approach can also be seen as a process of modeling activity traces by applying *abductive reasoning*, i.e. searching for hypothetical causes to explain observed consequences. In our case, the observed consequences are the events recorded in the data. The hypothetical causes can be either events already recorded in the data or abstract events that the expert adds to the trace. In both cases, the expert asserts the causal link based on his models or expertise. Notably, logicians consider abductive reasoning both as a non-logically-valid method, and as the only method of logical inference that can yield new knowledge. Once formed, the hypothetical causes and explanations need to be recorded in the trace. Then, the system should help the analyst ensure formal consistency and evaluate these hypotheses in terms of usefulness for making predictions. We call this process *expert-driven trace modeling*.

We will discuss one approach for conducting this trace modeling process throughout this paper. We start with a presentation of a top-down analysis in section 3. This presentation leads us to specify the requirements for an activity-trace modeling tool in section 4. We then present our prototype implementation of such a tool in section 5. We present our usage of this tool for expert-driven bottom-up modeling of field data in section 6. We then discuss how we imagine the two processes (top-down and bottom-up) could meet in the middle.

### 3. Top-down analysis

The literature provides us with diverse examples of top-level models of processes that lead to non-state political violence. Figure 1 depicts Horgan’s (2009) description of the phases of involvement and engagement in terrorism. Critically, Horgan, as do other authors (e.g., Sageman, 2004), makes a distinction between radicalization and engagement in actual terrorist activity. In Figure 1, the circles represent conceptually discrete but often overlapping phases of activity. We can break these phases down into organizational sub-processes, as we do in Table 1 with the violent radicalization phase. Such break downs

show the initial pathway to symbolic sequential modeling.



**Figure 1:** Pathway into, through, and out of terrorism (Horgan, 2009, p. 151).

From this break down, we have constructed a timeline representation of these different phases as shown in Figure 2. We have done so with an existing open-source visualization tool called Simile Timeline<sup>ii</sup>.

**Table 1:** Hierarchy of sub-processes of violent engagement drawn from Horgan (2009).

- (A) Decision and search activity - targeting and "pre-terrorism"
  - Plan
  - Have a leader
  - Connect to an organization
  - Search for suitable situations
- (B) Preparation and "pre-terrorist" activity
  - Target identification
  - Identification and selection of appropriate personnel
  - Training, general and specific to target
  - Design and manufacturing related to device construction
  - Device testing and preparation
- (C) Event execution
  - Bring device and manpower to the scene of the attack
  - Maintenance, surveillance, security of the operation
  - Dynamics of the event
  - Securing of weapons after attack
- (D) Post-event activity and strategic analysis
  - Destruction of evidence
  - Post-event evaluation

This modeling illustrates some of the limitations of available timeline visualization tools. Such tools require a precise timeline of events to represent events numerically—this proves unwieldy when modeling high level terrorist processes. As long as we do not know precisely at what timescales terrorist activities are operating (hours, days, weeks, months, years, or decades), we need to formalize the succession and relations between events as opposed to their real duration. Consequently, such a process model should be invariant through scale but should rather allow the analyst to express temporal relations such as sequentiality, concurrence, or overlap. In other words, we need a tool capable of supporting



**Figure 2:** High-level timeline drawn from Table 1.

pattern analysis on a more abstract level for datasets where few or no dates are available.

More broadly, we need tools that allow the analyst to represent events symbolically, identify symbolic patterns, and from those patterns develop new symbols that represent meaningful sequences of activity. These sequences, in turn, can indicate the emergence of processes over larger timescales. This approach addresses both dimensions of data/knowledge gap described above by (1) supporting the intelligible analysis of granular data that in turn can inform theory, and (2) by facilitating cooperation between knowledge-engineers and domain experts as they attempt to develop a meaningful trace.

#### 4. Process modeling tools

Current software tools for activity analysis (such as NOLDUS<sup>1</sup>, INTERACT<sup>2</sup> and MORAE<sup>3</sup>) do not meet our requirements in at least two ways (a review of such tools can be found in Hilbert and Redmiles (2000)). (a) Developed to analyze very detailed behavior data, such as a user interacting with a device, these tools typically only support sequential analyses spanning hours or days, as opposed to weeks, months, or years. (b) These tools also generally support data composed of low-level relatively simple events. They do not help the analyst manage the possibly evolving interpretation that he or she attributes to the events. Tools such as InfoScope<sup>4</sup>, on the other hand, do provide high-level data visualization, but do not offer symbolic timeline analysis.

Concerning tools specifically developed to model trends in terrorist activity, we must cite the GTD Data Rivers tool developed by Lee (2008). The GTD Data Rivers is an interactive visual exploratory tool that allows analysts to investigate temporal trends in terrorism found in the GTD. The GTD Data Rivers aggregates important variables from the database and visualizes them as a comprehensible stack chart as shown in Figure 3.

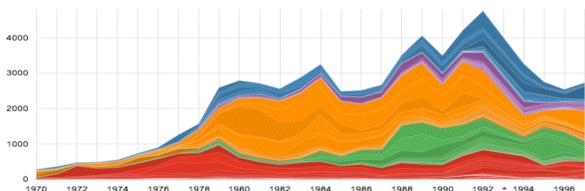


Figure 3: Number of events in the database differentiated by country (Lee, 2008).

<sup>1</sup> <http://www.noldus.com/>

<sup>2</sup> <http://www.mangold-international.com/en/products/interact.html>

<sup>3</sup> <http://www.techsmith.com/morae.asp>

<sup>4</sup> <http://www.macrofocus.com/public/products/infoscope/>

Figure 3 illustrates the rise and fall in the frequency of terrorist attacks for the years 1970 to 1996; the bands in this case represent targeted countries within six regions: Europe, Asia, South America, North America, Africa, and the Middle East. This tool enables us to analyze large chronological trends but it only supports numerical value visualizations, and does not support symbolic process modeling.

This review of tools helped us identify the need for a trace-modeling tool. These are summarized in Table 2.

Table 2: A specification for process-oriented trace-modeling tools.

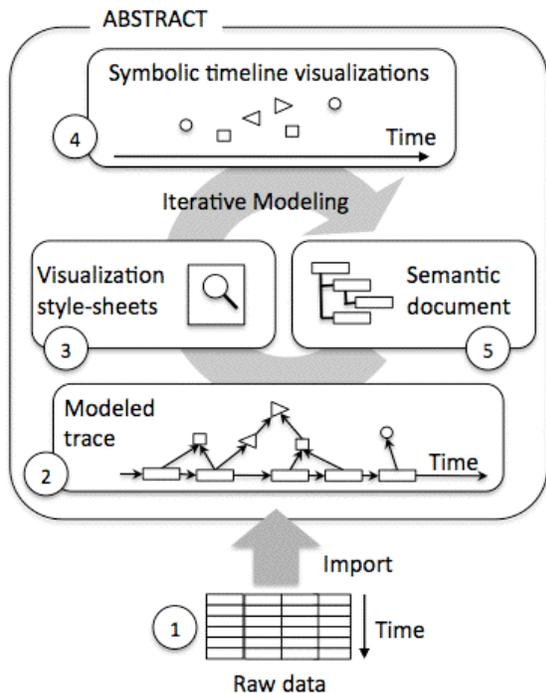
Modeling specifications	Sub-requirements
Model past activities (produce a representation of an activity that has occurred about which we have information)	Display symbolically what we know about particular events across multiple levels of abstraction including: location, time, actors involved, unique characteristics, etc.
Modeling current ongoing activities (produce a representation of an ongoing activity that we hope to control and/or predict)	Enable analysts to dynamically identify new events, meaningful sequences of events, and relations between events in order to find signatures of sequences that may lead to predictions.
Support the development of counter-factual scenarios from “abstractions” of real events	From these scenarios, develop inferences that inform the prediction of future events and suggest preventative courses of action.

#### 5. A tool for terrorism process modeling: ABSTRACT

To fulfill the requirements expressed in sections 3 and 4, we modified ABSTRACT<sup>5</sup>, a trace-modeling tool that we have designed in previous work (Georgeon, Henning, Bellet, & Mille, 2007). ABSTRACT enables the analyst to define transformation rules to process raw qualitative or quantitative data streams into abstract activity traces. These abstract activity traces are based upon symbols that the analyst can define and organize in an ontology. Analysts can then visualize these traces and iteratively refine the ontology, the transformation rules, and the visualization format. This iterative process helps the analyst make sense of the initially overwhelming behavioral data. This process and tool have been used in a road safety study to find patterns of interest in data collected with an instrumented vehicle (Henning, Georgeon, & Krems, 2007). Figure 4 illustrates the aspects of this modeling process as they apply to the

<sup>5</sup> <http://liris.cnrs.fr/abstract/>

present study. This process involves 5 steps represented in blocks (1) through (5).



**Figure 4:** Process modeling with ABSTRACT.

(1): The raw data is usually stored in a spreadsheet where each line represents an event, and where the different properties of these events are recorded in columns.

(2): This data is imported into ABSTRACT under the form of a graph structure (RDF graph). In this graph, each event is a node. The analyst can add new events as new nodes during the modeling process. He or she can also add relations between nodes, including hypothetical causal relations that he or she asserts. In the figure, the geometrical shapes symbolize the events: rectangles, squares, circles, and triangles. The arrows represent the relations between events. Events also have properties attached to them as elements of the graph.

(3): The analyst defines style-sheets to render the modeled trace as *symbolic timeline visualizations*. These style-sheets are XSLT (eXtensible Stylesheet Language Transformation), a language for transforming XML documents into other XML documents.

(4): The timeline visualizations are SVG (Scalable Vector Graphics) documents that are displayed by any SVG compatible browser such as Firefox. We present an example of this visualization in Figure 5. ABSTRACT makes this visualization interactive—the user can both scroll the timeline, as well as click on events to show their

properties and follow hypertext links to further documentation in a supporting wiki page.

(5): The analyst defines the types of events in the semantic documentation system. Within the system, he or she provides, on one hand, the textual documentation that explains each event category while on the other specifying the events' visualization properties, namely the geometrical shape, color, icon, and y position. Collectively, these event types form an event ontology that can appear in the traces. This ontology is exported as a RDFS graph (Resource Description Framework Schema). These graphs are then exploited by the style-sheets to render the visualization timeline.

To support the computational process modeling of terrorist activity, we modified ABSTRACT in two ways:

a): We implemented a server version that allows for concurrent modeling by multiple team members—typically a researcher in information sciences who focuses on tool and style-sheet development, and investigators in the domain of interest, in this case specialists in terrorism studies.

b): We have used a semantic wiki<sup>iii</sup> to implement ABSTRACT's ontologies and documentation system. Previous versions of ABSTRACT used Protégé<sup>iv</sup> as an ontology editor. Using semantic-media-wiki has several advantages. For one, the wiki principle offers a manageable and easy way for analysts to attach descriptions to event types. For another, wikis are sharable across the web and allow the construction of shared representations between different users. Finally, a semantic wiki supports the association of semantic properties to pages, in our case: a type/sub-type hierarchy and visualization properties.

## 6. Symbolic timeline representation of events collected from the field

Using ABSTRACT, we have obtained representations of terrorist activity like that shown in Figure 5. Figure 5 displays terrorist activity in the Republic of Ireland between 1970 and 2007 taken from 143 events. The upper half of this visualization represents a zoom consisting of a one hundred day interval, centered upon January 10, 1973. The lower half represents the entire (37 year) time-course. The interactive features of this representation are available online<sup>v</sup>. This visualization illustrates what we mean by symbolic timeline visualization and modeling. Unfortunately, this data does not include behind-the-scene information and does not inform us about the underlying processes that are happening. It is intended here as a

demonstration of a method equally applicable to more detailed, and thus more illuminating, data.

In Figure 5, each event is represented by an icon and possibly a second icon appended to it. The first icon is associated with the field "WEAPON\_TYPE". The three main weapon types are represented: "Firearms" (gun), "Explosive" (star) and "Incendiary" (flame). When the weapon type is unspecified, the event is represented as a gray circle. The second icon, representing a body outline, is appended when the "ATTACK" field is equal to "assassination".

The "y" position is associated with the field "PERPETRATOR". Meaning, the principal terrorist groups are each represented on a distinct line. Loyalist groups are represented above the central axis. Republican groups are represented below the central axis. Events whose affiliation is unknown are represented on the center axis.

The user can click on the event to show a tip window associated with it. The tip window displays the properties of the event. This tip window provides hypertext links to the definition of the different types in the semantic wiki.

By following these links, the analyst can change the visualization properties as well as the textual explanations, before generate new timeline visualizations. The "GTD\_ID" field gives a link to the GTD page that provides a comprehensive description of the event.

To illustrate the descriptive utility of this layout, let us consider the historical events associated with the Irish Troubles and how they are illustrated in Figure 5. For the group represented by the lower-most row on the y-axis (Group 11- the Irish republican Army), you'll notice that there are three sizeable lulls in activity toward the end of their campaign. After the second lull, there were two attacks that occurred in the first half of 1998. In April of 1998, several political parties (including Sinn Fein and its associated military force, the Provisional Irish Republican Army) came together to sign the Good Friday Agreement in an attempt to bring peace to the Ireland, Northern Ireland, and the United Kingdom. Although Sinn Fein was a signatory to the Good Friday Agreement, it is possible that some individuals within the IRA were opposed to the peace process and engaged in activity contrary to its stipulations.

One limitation of the dataset employed here is the lack of



Figure 5: Terrorist activity in the Republic of Ireland (1970-2007) represented with ABSTRACT.

representation for other notable dissident groups. For example, one group that is vehemently opposed to the peace process is the Real Irish Republican Army (RIRA). In response to what they deemed to be Irish submission in the form of a peace deal, some members of the Provisional IRA broke off to form a more violent faction. This faction became known as the Real IRA. Had they been represented more comprehensively in the GTD, Figure 5 would illustrate the extent to which violence struck Ireland, Northern Ireland, and Britain in the wake of the GFA (post April, 1998). In the weeks and months following the signing of the GFA, the Real IRA conducted several operations, including bombings and mortar attacks. Despite its lack of representation in the GTD, data concerned with the activities of the Real IRA could be effectively illustrated with ABSTRACT. Doing so would (a) further illuminate the extent to which dissident and paramilitary activity has pervaded Ireland, Northern Ireland, and the rest of the UK in past decades, and (b) show the relationships between contextual events (e.g. signing of the GFA) and attacks by dissident groups or paramilitaries.

## 7. Discussion and Conclusion

We have yet to explore the full potential of this approach with data that would contain more information about the full process of terrorism activity. We may consider extensive detainee history such as published by Bruning and Alexander (2008) or terrorist narratives like those assembled by Sageman (2004). Our work on the GTD data provides a high-level, relatively abstract, description of the events contained within the database. As we obtain more data, we expect we will be able to more readily identify persistent signature patterns of activity, and connect the bottom-up modeling and the top-down modeling together. Using GTD data has allowed us to make a start in that direction and to identify important features for future process-oriented trace-based approaches. We have found having an online tool invaluable for not only capturing semantic content but also facilitating cooperation between team members from different origins, namely terrorism study and information sciences. In addition, our experiences modeling GTD events underscore the importance of analyst-driven tools that readily support the creation and placement of new symbolic representations that in turn support the visualization of salient differences. Finally, this approach allows the data to speak for itself by enabling the user to visualize timeline of events represented by symbols and providing links to complementary information.

We have examined an approach for modeling process, an approach that acknowledges and attempts to address the data/knowledge gap emerging across the social sciences. We specifically address the modeling of terrorist activity,

however, we believe trace-based methods may be applicable to other domain areas where modeling emergence and reflexivity are important. For specialists in terrorism studies, we believe these methods will contribute to our understanding of data-rich processes and sub-process such as Improvised Explosive Devices (IED) development, online recruitment, and the movement of money and resources. We also believe that the insights we obtain from formalizing our understanding of the influence of low-level psychological and social factors may have implications for less tractable terrorist processes.

As we strive to deepen our understanding and formalize our knowledge, some analyses of processes describing events may integrate perspectives from a variety of contexts, others may focus on particular discipline or problem perspectives. It is possible that understanding some processes will necessarily draw on perspectives from particular disciplines or professions. The nature of the activity, the perspective taken, and the degree of conceptual complexity and understanding are all presumably variables that will affect the overall understanding of the phenomenon and its relationship to its environment and context.

The modeled traces that we obtain are sets of symbols and relations assembled as chronological representations. We must take these representations pragmatically (Wittgenstein, 1953), and assume that they are neither right nor wrong, neither true nor false: they are merely useful for the particular applications in which we apply them. These representations are also intended to evolve with our knowledge and with the data available. Our current level of analysis and the inherent assumptions we make about starting points for analysis and end products will influence further analysis.

We recognize the evolutionist and pragmatic aspect of this analysis, and attempt to support analysts operating in a variety of contexts and levels of analysis by synthesizing bottom-up and top-down approaches into a common framework. We, in fact, believe that a commitment to a pragmatic approach requires this from us while simultaneously obligating us to try to evaluate theory through the modeling of actual events. We believe this is not only possible but increasingly feasible as interdisciplinary communities cognoscente of data-mining and data-sharing tools emerge.

## 8. Acknowledgment

Support for this study was provided by ONR (contract N00014-06-1-0164 and N00014-08-1-0481) and DTRA (contract HDTRA 1-09-1-0054). We gratefully thank Dr

Frank E. Ritter for his useful comments on this study and report.

## 9. References

- Bruning, J., & Alexander, M. (2008). *How to Break a Terrorist: The U.S. Interrogators Who Used Brains, Not Brutality, to Take Down the Deadliest Man in Iraq*: Free Press.
- Georgeon, O. (2008). Analyzing traces of activity for modeling cognitive schemes of operators. *AISB Quarterly*, 127, 1-3.
- Georgeon, O., Henning, M. J., Bellet, T., & Mille, A. (2007). *Creating Cognitive Models from Activity Analysis: A Knowledge Engineering Approach to Car Driver Modeling*. Paper presented at the International Conference on Cognitive Modeling, Ann Arbor, MI, 43-48.
- Hedström, P. (2005). *Dissecting the Social: On the Principles of Analytical Sociology*. Cambridge, MA: Cambridge University Press.
- Henning, M. J., Georgeon, O., & Krems, J. (2007). *The quality of behavioral and environmental indicators used to infer the intention to change lanes*. Paper presented at the 4th International Driving Symposium on Human Factors in Driver Assessment, Stevenson, Washington USA.
- Hilbert, D. M., & Redmiles, D. F. (2000). Extracting Usability Information from User Interface Events. *ACM Computing Surveys (CSUR)*, 32(4), 384-421.
- Horgan, J. (2009). *Walking Away from Terrorism, Accounts of Disengagement from Radical and Extremist Movements*: Routledge.
- Lee, J. (2008). Exploring Global Terrorism Data: A Web-based Visualization of Temporal Data. *ACM Crossroads*, 15(2), 7-16.
- Popper, K. (1972). *Objective knowledge*. Oxford: Oxford University Press.
- Sageman, M. (2004). *Understanding terror networks*. Philadelphia, PA: University of Pennsylvania Press.
- Taylor, M., & Horgan, J. (2006). A Conceptual Framework for Addressing Psychological Process In the Development of the Terrorist. *Terrorism and Political Violence*, 18, 1-17.
- Wittgenstein, L. (1953). *Philosophical Investigations*.

---

iv <http://protege.stanford.edu/>

v [http://acs.ist.psu.edu/~georgeon/abstract/View\\_v10\\_v11/tI01/p01/3447](http://acs.ist.psu.edu/~georgeon/abstract/View_v10_v11/tI01/p01/3447)

---

i <http://www.start.umd.edu/gtd/>

ii <http://www.simile-widgets.org/timeline/>

iii <http://semantic-mediawiki.org/>

## Resistance is Futile: Winning Lemonade Market Share through Metacognitive Reasoning in a Three-Agent Cooperative Game

David Reitter, Ion Juvina, Andrea Stocco and Christian Lebiere

Department of Psychology  
Carnegie Mellon University  
Pittsburgh, PA

reitter@cmu.edu, ijuvina@cmu.edu, stocco@cmu.edu, cl@cmu.edu

Keywords:

Metacognition, Cognitive Modeling, Games, Cooperation

**ABSTRACT:** *The Lemonade Game is a three-player game in which players have to pick locations on a circular board, which are as far away as possible from those chosen independently by other players. Players may observe other player's moves and infer their strategies. The game was examined using a competition of cognitively motivated agents, which inherit properties of human memory and decision-making, and simplistic, yet effective agents. We argue that metacognition constitutes the unique attribute that allows sophisticated agents to adapt to unforeseen conditions, cooperators and competitors.*

### 1. Introduction

Unlike other species, humans are not optimized to any specific natural environment or task, but they are very good at many things. At least in the long run, generalists agents like humans seem to be superior to specialist ones. Agents that are optimized to a particular ecological niche might succeed in current conditions, but once their environment changes they are likely to be suboptimal and soon extinct. While there is no doubt that we owe our superior adaptability to cognitive rather than physical attributes, the precise source of that superiority has been the subject of some debate, and proposals have been made to precisely formulate and measure that capability (e.g., Anderson & Lebiere, 2003). Here we provide support for the notion that the flexibility and adaptivity that metacognition affords us is our main evolutionary advantage.

The same arguments can be applied to artificial as well as biological agents. In particular, the focus on optimality that dominates many fields of the cognitive

sciences can be seen as counterproductive, and indeed as the very source of their controversial pattern of reaching short-term objectives while making little or no progress toward their overall goal. Artificial Intelligence has met a number of high-profile challenges (a world champion chess player, or a vehicle that can drive itself semi-autonomously) but it seems no closer to the original dream of a generally intelligent artifact. Cognitive Psychology has seen the development of high-fidelity models that reproduce human behavior in highly controlled tasks, but none of these models can exhibit robust behavior in unforeseen situations. Finally, Machine Learning has produced algorithms that can use large amounts of data to adapt their performance, but only within the boundaries of their specific representations. The common thread of these approaches is narrow optimality within limited circumstances, and often disastrous behavior outside these confines.

## 1.1 The Lemonade Game

The question that arises is how to study the flexibility and adaptivity that might be the true magic of human cognition. One possibility is to adopt open-ended challenge tasks where agents are exposed to unforeseen situations. That was the approach chosen for the Dynamic Stocks and Flow Model Comparison Challenge (Lebiere, Gonzalez, & Warwick, 2009). Another possibility is to select an environment that highlights the complexity of the interactions of the agents that inhabit it. One such deceptively simple but subtly complex task is the Lemonade Game used in a recent challenge by Martin Zinkevich of Yahoo Research. In this game, three agents try to locate a fictional lemonade stand one of 12 possible locations (arranged in a circle and referred to as 0 through 11). The reward for each agent is the sum of the distances from the other two. A complete game consists of 100 consecutive trials. At the beginning of each trial, the three agents independently and synchronously decide the locations of their respective stands. The positions and rewards of all the agents are then calculated and revealed.

Many similar simple games feature either zero-sum competition (e.g., paper rock scissors; Billings, 2000) or the possibility of choosing between either cooperation or and competition (e.g., the prisoner's dilemma; Rapoport, Guyer & Gordon, 1976). The A unique feature of interest of this game is that it features permits a simultaneous combination of both cooperation (between two agents) and competition (against the third). As we will see, the emerging dynamics are quite interesting and prevent any notion of optimality. In order to succeed, the agents must adapt to the others' strategies, communicate their intent to cooperate and detect a similar willingness in others, and more generally encounter and adapt to patterns of behavior that cannot be derived from the environment but instead arise from the agents themselves and their interaction. We will start by outlining simple agents to

play the game and their limitations. Then, we will describe a more complex approach that depends upon a combination of action strategies, sequence-detection abilities, and (most importantly) meta-cognitive supervision that continually oversees the behavior of the agent.

## 2. Basic Decision-making Agents

These agents are "self-centered," in the sense that they ignore the actions of the other players. They correspond to basic approaches to the problem that can be used in isolation.

The *Random* agent chooses a random location independent of previous situations. The random agent is maximally unpredictable. This strategy can be successful in many games (e.g. zero-sum games such as in paper-rock-scissors (West & Lebiere, 2001) or adversarial games such as in the Prisoner's Dilemma (Lebiere, Wallach, & West, 2000)). In the Lemonade Game, however, randomness precludes cooperation and effectively ensures poor results. Indeed, the random agent often received the poorest score in our tournaments.

The *Sticky* agent selects its initial position at random, and then maintains it throughout the game. This agent is designed to be maximally predictable. In the lemonade game, predictability is a powerful invitation to cooperation; as a result the sticky agent outperforms the others, even when its opponents are much more sophisticated agents. The *Roll* agent is also easily predictable. At each trial  $i$ , it chooses a position  $p_i = p_{i-1} + c \pmod{12}$ , with  $c$  being an arbitrary constant. Similarly, the *SquareRoot* agent chooses  $p_{i-1}^{0.5+c}$ .

### 2.1 Evaluation

When self-centered agents play against each other, they do comparably well. No self-centered agent is clearly superior to the others. In particular, neither being maximally

predictable (sticky) nor maximally unpredictable (random) is inherently advantageous when playing against similarly self-centered agents, as shown in Table 1.

Table 1: Simple Agent Tournament Results

RANDOM	8.002
STICKY	8.002
ROLL	7.996

### 3. Metacognitive approaches

The term *Metacognition* refers to benefiting from awareness of each players performance and limitations, including one’s own.

#### 3.1 Basic Metacognitive Agents

Extending the basic agents with rudimentary metacognitive abilities created an initial set of metacognitive agents. *StickySmart*, an extension of Sticky, assumes that its opponents try to either maximize or minimize the distance from itself. Under the maximization assumption, it pays off to maintain your current location: the further your opponents are from yourself the higher your score. Under the minimization assumption, maintaining one’s current location is catastrophic: the closer one’s opponents are to yourself the lower one’s score. In this case, *StickySmart* moves to the opposite location (over the diagonal), which restores the situation under the maximization assumption.

*CopyCat* assumes that at least one of its opponents has an effective strategy, and it tries to copy it. Thus, *CopyCat* picks an opponent and always chooses its previous choice plus an increment  $c$ . The increment is needed to avoid the special case the opponent plays sticky, and thus both agents end up in the same location. The best constant increment is  $c=6$ , which ensures that a loss is avoided in case the opponent plays sticky, and it is neutral in other cases. *CopyBest* is a variation that also monitors whether copying an opponent is working; when it is not, it switches to copying the other opponent.

*Cooperator* takes a more active and constructive approach, and assumes that cooperation is the key to success. In order to establish a cooperative relationship, *Cooperator* initially issues a request for cooperation by making itself maximally predictable (i.e., playing “sticky”) and waits for an opponent to pick up the offer and cooperate (thus, become a partner). Two partners are said to cooperate if they maximize the clock-distance between themselves, that is, they select locations that lay on the opposite sides of a diameter. Thus, *Cooperator* plays “sticky” as long as it does not repeatedly lose points. Otherwise, it switches partners.

*StickySharp* is an extension of *StickySmart*. When the two opponents of *StickySmart* cooperate, any sticky agent will lose. *StickySharp* tries to find a way out by issuing an alternative cooperation offer toward its opponents by playing Roll. *StickySharp* succeeds if one opponent “helps the poor”, that is, cooperates with the lower-scoring player.

*Statistician* maintains a record of its opponents’ moves uses it to predict their subsequent moves. It then selects a location that is maximally distant from its opponents’ predicted moves. Its predictions are based on a weighted average of each opponents’ previous locations, where most recent choices are weighted more then less recent ones. Because it maximizes only its own payoff, *Statistician* plays aggressively rather than cooperatively.

*Strategist* extends *Cooperator*: it preserves cooperation and adds altruism. First, *Strategist* assesses its opponents’ predictability. If none of the two opponents is predictable, *Strategist* plays “sticky”, assuming that at least one opponent will accept the offer to cooperate, which in turn makes the behavior of this opponent predictable. If only one opponent is predictable, *Strategist* cooperates with it,

while continuing to assess the predictability of the other opponent. If both opponents are predictable, Strategist cooperates with either the weaker or the stronger of its two opponents depending on its own performance. If Strategist's performance has been consistently good, the weaker opponent is chosen; otherwise, the stronger opponent is chosen to cooperate with. This discretionary selection ensures that both principles of cooperation and altruism are enforced. Note that Strategist cannot always be altruistic without affecting its commitment to cooperation. Due to the zero-sum nature of the game, helping the weaker opponent would weaken the stronger opponent, which would eventually force Strategist to switch partners. These repeated switches make Strategist's behavior look less predictable to its potential partners, thus making it less attractive as a partner, and therefore less capable of cooperating.

### 3.2 A General Model of Metacognition

Cognitive models usually implement strategies to solve specific problems. The term *metacognition* stems from the realization that human problem-solvers have multiple strategies at their disposal, choosing and adapting them while carrying out the task: they are aware of their limitations. In the context of the Lemonade game, metacognition is especially relevant as strategies depend on the constellation of the players in the game. Some opponents may be willing to cooperate, or (at minimum) they are predictable and exploitable. For example, Statistician reliably outperforms Random because it can predict and cooperate with the third player, but it is defeated in games where this player is Roll.

We decompose the actions of metacognitive agents in each Lemonade trial into two steps. In the first step, predictions are generated for the other players in the game. These predictions depend on previously observed behavior of those players within the same game. A prediction can be represented as a

probability distribution over locations, indicating the estimated probability of a given opponent placing their lemonade stand at the given location in the next trial. The second step consists of making a decision about where to place one's own lemonade stand in the next iteration, in light of the expected payoff at each location, which can be calculated given the locations of all three stands. This step may be as simple as maximizing utility (joint probability and payoffs), but it may also include a strategy to induce future cooperation with a player or to hurt a specific player that may be performing too well.

Metacognitive agents can compare different strategies for both prediction and action. Each strategy's evaluation is updated immediately after each trial. We distinguish two possible monitoring mechanisms. Prediction strategies can be evaluated in parallel: all strategies may be used to predict each opponent's move, and they can all be evaluated after each trial. Action strategies, however, can only be evaluated one at a time if their long-term effects are to be considered. As a consequence, it is easier to converge on prediction strategies than on action strategies.

#### *Prediction Strategies*

Prediction strategies produce a probability distribution  $P(a)$  over the 12 locations for a given opponent. They use the decision history of that agent within the current game.

The prediction strategies use  $n$ -gram representation, where the opponent's moves there are recorded as series of  $n$  consecutive locations. This representation has been successfully used in sequence learning models (e.g., Lebiere & West, 1999) We provided a range of different algorithms by encoding relative and absolute movements of the agents separately. The Meta model, included different strategies are obtained by encoding series of  $n = 1, 2,$  or  $3$  choices, and encoding locations in absolute terms as well as relative movements from the previous agent location.

### Action Strategies

An action strategy uses the predictions (a probability distribution for each opponent) in order to determine the agent's move. We considered the following *elementary action strategies*.

**Utility optimization:** This strategy chooses the location with the highest immediate expected payoff. Assuming the point of view of player  $a$ , and its opponents as  $b$  and  $c$ , then the utility of  $a$  being at location  $l_a$  would be

$$u(a, l_a) = \sum_{l_b=0}^{11} \sum_{l_c=0}^{11} p'(b, l_b) p'(c, l_c) \text{payoff}(a, b, c)$$

$\text{payoff}(l_a, l_b, l_c)$  is the reward that  $a$  receives if players  $a, b, c$  are in positions  $l_a, l_b, l_c$ , respectively.  $p'$  are the probability estimates for one agent choosing a specific location.

The Sequence Learning agent in the tournament uses utility optimization as its action strategy.

**Offer to cooperate:** This class of strategies is designed to be as predictable as possible. It includes two instances of the *Sticky* action strategy that choose different, but constant, locations. Note that these strategies offer to cooperate, but do not cooperate themselves; the action meta-layer will switch strategy if one of them proves unreliable.

**Cooperation:** This action strategy identifies the opponent that is best performing while being predictable. Predictability is measured as a single location being predicted with probability  $> 0.85$ . If the better-performing opponent is not predictable enough, the worse performing opponent is chosen if any prediction is available. The strategy then cooperates by choosing the location opposite the predicted of that opponent. If no reliable prediction can be made (during the initial steps), the cooperator plays consistently the same location in order to offer cooperation to

another agent. Cooperation is the most successful one of the action strategies.

**Imitation:** As a further action strategy, we included the *Copy Cat* as described above.

### The Metacognitive Agent

The *Meta* agent implements a hybrid combination of the elementary strategies. The metacognitive layer combines all predictions and chooses an action strategy. This agent has a principled approach to choosing strategies, it is cognitively motivated, and was not optimized by hand to succeed in the task.

The agent's metacognitive layer evaluates both types of strategies using immediate feedback; in the case of prediction strategies, we evaluate the reliability of the estimates for the chosen location. In the case of action strategies, we use their immediate reward to update their overall payoff. To make the agent adaptive to changes in a strategy's payoff over time, we adopted a cognitively motivated approach known as *instance-based learning* (IBL, Gonzalez & Lebiere, 2003). This approach balances frequency and recency of the observed strategy performance. This approach is derived from the learning mechanisms in the ACT-R cognitive architecture. It has been applied both to both sequence learning paradigms (Lebiere & Wallach, 2001) and games like paper rock scissors (Lebiere & West, 1999) and baseball (Lebiere, Gray, Salvucci & West, 2003). The key intuition behind this approach is that more frequent and more recent memories provide more reliable information, since the environment is less likely to have changed since the memory was formed. In the Lemonade Game, this means that opponents are more likely to follow the same strategies within short periods of time.

IBL involves memorizing an *episode* every time a strategy  $s$  is evaluated for a specific agent  $a$ . The episodes encode  $t$  (time step at which it occurred),  $l$  (actual location chosen

by  $a$ ),  $p_l$  (probability predicted by  $s$  that  $l$  would be chosen in the next step). We then calculate a blend of the episodes, in which episodes are weighed by their relevance (did the strategy yield a high probability of the actual location?), their recency (a temporal decay is applied) and frequency.

We calculate a base-level activation value (as in ACT-R) for each episode, taking temporal decay into account. The activation is applied to the predicted probability for the chosen location in that episode:

$$c(a, s) = \sum_{\langle t, l, p_l \rangle} p_l e^{\frac{b_c + \ln((t_0 - t)^{-d})}{T}} + \varepsilon$$

$b_c$  is an ACT-R base-level constant (held at 4.0),  $t_0$  is the current time,  $T$  the Boltzmann temperature.  $d$  is a decay coefficient (0.5 in ACT-R models).  $\varepsilon$  is a term for noise, sampled from a pareto distribution. We arrive at a confidence value  $c(a, s)$  for given strategy  $s$  and opponent agent  $a$ .

To create a final, blended probability distribution  $P'(a)$  for an opponent agent  $a$ , the distributions from each prediction strategy  $P(a, s)$  are weighted by their confidence.

$$P'(a) = \frac{\sum_s c(a, s) * P(a, s)}{\sum_s c(a, s)}$$

The same method was used to evaluate the action strategies, except that rather than  $p_l$  we use the payoff as quality criterion for the strategy that is stored in each episode.

Parameters ( $T$ ,  $d$ ,  $n$ ) as well as the subset of action strategies were fit to optimize the Meta agent's performance against the basic and advanced agents discussed above. The final

parameter values were  $T=0.2$ ,  $d=0.7$ ,  $n=0.004$ .

#### 4. Evaluation

We evaluated the strategies in a tournament that ran games with 100 rounds each, running every combination of three different agents. (We aggregated data from several repetitions of each combination.) The outcome of each game strongly depends on the configuration of players. For instance, a combination of two agents may or may not end up cooperating, winning over the third player. We analyze three outcomes of agent pairings: the relative strength of the agents, their absolute performance, and the reliability of their performance with respect to changing third players. Figure 1 visualizes these measures. A + sign indicates that the Scored Agent (x-axis), on average, reaches higher payoffs than the 1<sup>st</sup> opponent (y-axis). Circle size indicates the payoff that the Scored Agent achieves on average when the 1<sup>st</sup> opponent is present in a game (large circles indicate higher payoffs). The shade of the circle visualizes the reliability of the Scored Agent's performance: dark circles indicate low variance across the different third agents. A column of large dark circles marks a strong, reliable agent.

Consider *CopyCat* as our target (Scored) agent. It defeats both *Statistician* and *Random*. *CopyCat* also tends to reach high scores when *Sticky* is present, exploiting *Sticky*'s predictability. However, it is also very susceptible to intervention by the third agent: cooperating with *Sticky* makes *CopyCat* equally predictable. This may be exploited by a third agent, which may choose to destroy *CopyCat*'s ambitions. In a game against *Random*, the winnings are more reliable.

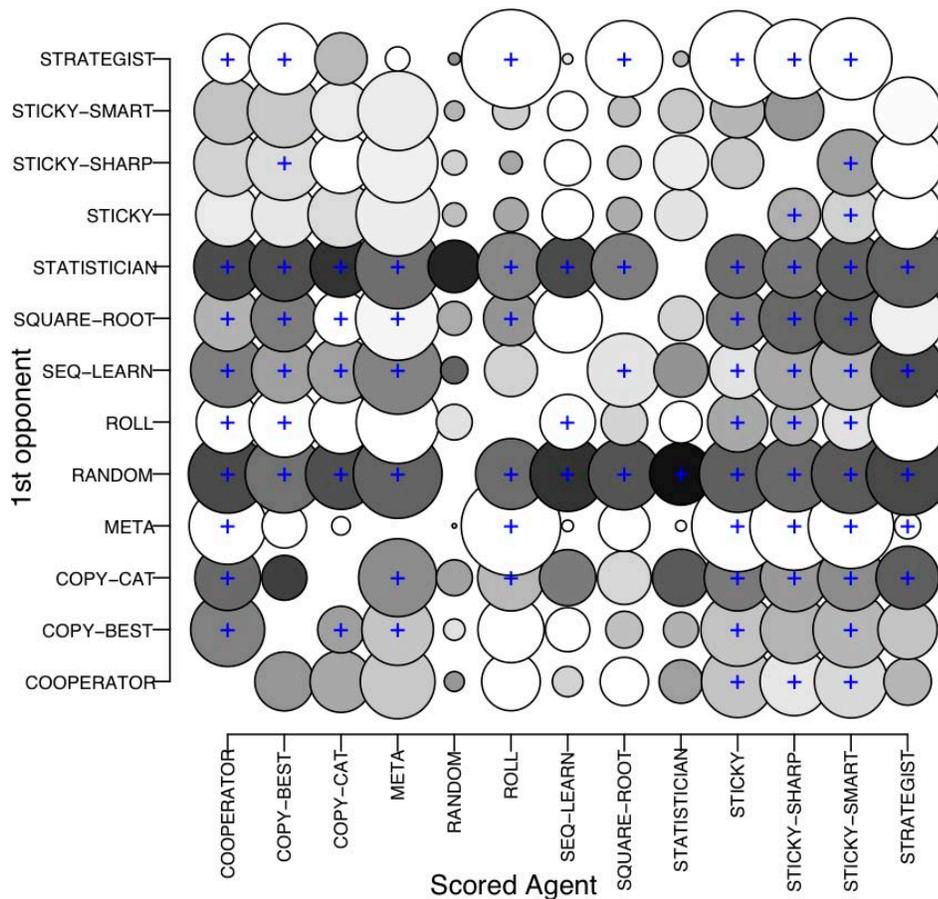


Figure 1: Performance of the strategies (x axis) when playing against other strategies (y axis). Sizes of circles indicate points achieved, while color of circles indicates variability of success depending on third player (dark: less variable). Plus signs indicate a numeric win of the scored agent over the 1<sup>st</sup> opponent.

Meta as well as some cooperating agents (Stick&friends, Cooperator) achieve high and reliable results. The development of *Meta* showed that its cooperative action strategy was crucial to its success; it differs from *Cooperator* only in its monitoring of the success of other players, cooperating with the more successful ones if predictable.

Meta as well as some cooperating agents (Stick&friends, Cooperator) achieve high and reliable results. The development of *Meta* showed that its cooperative action strategy was crucial to its success; that strategy differs from *Cooperator* only in its monitoring of the success of other players, cooperating with the more successful ones if predictable.

Monitoring also plays a role in several of the strategies, including *CopyCat* and *StickySmart*. *StickySmart* outperformed the non-metacognitive *Sticky*.

Table 2 gives the aggregated tournament results (250 rep.). *Meta* consistently outperforms all other agents. The *Meta* strategy was further evaluated by removing all but two basic prediction mechanisms (uni- and bigram models) and all action strategies except *Cooperation*. In a further tournament (200 rep.) did the resulting simplified agent perform worse than the full *Meta* strategy (8.205 vs. 8.432). This shows that the hybridization of strategies is beneficial.

### 5. Conclusion

From the viewpoint of cognitive modeling, this paper examined agent collaboration in a three-player game known as the Lemonade Game. The Lemonade Game differs from other paradigms (e.g., Paper, Rock, Scissors) in that both being predictable and collaborating with an opponent improves one agent's chances to succeed. A series of

<i>Meta</i>	8.432
<i>Sticky Smart</i>	8.311
<i>Sticky</i>	8.238
<i>Sticky Sharp</i>	8.222
<i>Cooperator</i>	8.214
<i>Strategist</i>	8.172
<i>CopyBest</i>	8.152
<i>Roll Clock</i>	8.039
<i>CopyCat</i>	7.948
<i>SquareRoot</i>	7.824
<i>Sequence Learning</i>	7.673
<i>Statistician</i>	7.602
<i>Random</i>	7.172

simulations has shown that most successful strategies include offers to collaborate by making oneself predictable (*Sticky*) or more direct forms of collaboration (*CopyBest*, *Cooperate*, *Collaborate*). We found that monitoring of one's own and the opponents; performance is crucial for making profitable choices. Yet, comparing the meta-cognitive *Meta* agent to some high-performing alternative agent, one would expect it to do slightly worse in some cases. Because of the inefficiency of its meta analysis, it will be worse than the fixed strategy in the cases when that one is appropriate (which could be many, if it is very good). Still, any fixed strategy is likely to be poor for at least some combinations of opponents, and that is where *Meta* profits. The overhead of *Meta* over the fixed strategy can be kept small, while the price of a fixed strategy in a poor match can be very high. That tends to favor *Meta* overall, even if those cases are few. This can be seen as a special case of a general argument against narrow optimization in the development of cognitive agents, since that optimization is only meaningful within limited circumstances and its cost in loss of robustness outside of those circumstances is often left unspecified.

The key to robustness in unforeseen situations, such as being matched with an agent that one has never encountered, is the ability for an agent to evaluate the effectiveness of *all* its strategies, modify them as needed and select them accordingly.

## References

- Anderson, J. R., D. Bothell, M. D. Byrne, S. Douglass, C. Lebiere, and Y. Quin. An integrated theory of mind. *Psychological Review*, 111:1036–1060, 2004.
- Anderson, J. R. & Lebiere, C. L. (2003). The Newell test for a theory of cognition. *Behavioral & Brain Sciences* 26, 587-637.
- Billings, D. (2000). The first international RoShamBo programming competition. *International Computer Games Association Journal* 23(1), 42-50.
- Gonzalez, C. and C. Lebiere. Instance-based cognitive models of decision making. In D. Zizzo and A. Courakis, editors, *Transfer of knowledge in economic decision making*. Palgrave MacMillan, New York, 2005.
- Lebiere, C., Gonzalez, C., & Warwick, W. (2009). A Comparative Approach to Understanding General Intelligence: Predicting Cognitive Performance in an Open-ended Dynamic Task. In *Proceedings of the Second Artificial General Intelligence Conference (AGI-09)*. Amsterdam-Paris: Atlantis Press.
- Lebiere, C., Gray, R., Salvucci, D. & West R. (2003). Choice and Learning under Uncertainty: A Case Study in Baseball Batting. In *Proceedings of the 25th Annual Meeting of the Cognitive Science Society*. pg 704-709.
- Lebiere, C., & Wallach, D. (2001). Sequence learning in the ACT-R cognitive architecture: Empirical analysis of a hybrid model. In Sun, R. & Giles, L. (Eds.) *Sequence Learning: Paradigms, Algorithms, and Applications*. Springer LNCS/LNAI, Germany.
- Lebiere, C., Wallach, D., & West, R. L. (2000). A memory-based account of the prisoner's dilemma and other 2x2 games. In *Proceedings of International Conference on Cognitive Modeling 2000*, pp. 185-193. NL: Universal Press.
- Lebiere, C., & West, R. L. (1999). A dynamic ACT-R model of simple games. In *Proceedings of the Twenty-first Conference of the Cognitive Science Society*, pp. 296-301. Mahwah, NJ: Erlbaum.
- Rapoport, A., Guyer, M. J., & Gordon, D. G. (1976). *The 2X2 game*. Ann Arbor, MI: The University of Michigan Press.
- Reitter, D. Metacognition and multiple strategies in a cognitive model of online control. *Journal of General Artificial Intelligence*, under review.
- West, R. L., & Lebiere, C. (2001). Simple games as dynamic, coupled systems: Randomness and other emergent properties. *Journal of Cognitive Systems Research*, 1(4), 221-239

# Taxonomy and Method for Handling Large and Diverse Sets of Interactive Objects in Immersive Environments

*David Pietrocola*

*Barry G. Silverman, PhD*

Ackoff Collaboratory for Advancement of the Systems Approach (ACASA)

Department of Electrical and Systems Engineering

University of Pennsylvania

Philadelphia, PA 19104

215-746-8314, 215-573-8368

dpiet@seas.upenn.edu, basil@seas.upenn.edu

## Keywords:

Knowledge representation, immersive environments, explainable agents

**ABSTRACT:** The growing interest in immersive 3D environments populated with intelligent agents has led to a flurry of approaches and products with particular focuses, including cultural awareness training, language training, and operations “what-if” scenarios. Human terrain data present challenges that require categorization efforts. We present a taxonomy and approach to handle physical structures realistic intelligent agents and players interact with, manipulate, and discuss. A short interaction with an explainable, socio-cognitive agent in a prototype cultural awareness training game called NonKin Village is reviewed and we outline next steps as well opportunities for research, collaboration, and standards development.

## 1. Introduction

Immersive virtual worlds are gaining traction as training tools for various applications and environments, most prolifically currently being the military. With an abundance of data on terrain, human and otherwise, it should seem plausible that detailed models of on-the-ground conditions can be constructed into a cohesive system that decision makers may inquire to visualize and assess potential courses of action along with their secondary and tertiary effects. Immersive environments that represent societies at the scale of villages and larger require not only believable intelligent agents and detailed behavior, but also realistic surroundings – most of which must be interactive with a player or small unit of players. This paper describes current efforts in developing a standardizing taxonomy as a means to effectively classify and categorize environment data suitable for reasoning by intelligent agents and players in an immersive environment. Using existing categorization schemas, we attempt to employ the taxonomy and demonstrate a prototype using NonKin Village, a training game framework built upon a socio-cognitive agent architecture.

### 1.1 From Data to Wisdom

Ackoff (1989) provides a framework that captures the heart of the taxonomy’s goal to facilitate a diffusion of environmental information among agents and players in an immersive world. Virtual worlds are stood up with

an abundance of detailed datasets to describe the terrain, populate the area, and inform agent models. The framework would categorize this as data, or observable facts. The next improvement, information, makes data useful and answers relational questions such as who, what, where, and when. Information can also bring about meaning and shed light on patterns or trends. At the very least, users of immersive environments and intelligent agents that exist in them should be able to obtain information. The goals of training, however, would reside in the attainment of knowledge and understanding. Knowledge can apply information through rules about what to do in situations revealed by information. At a more encompassing level, understanding is the assembly of the “big picture” situation one is in, and provides an appreciation of the “why” (Bellinger et al., 2004).

## 2. Related Work

While the field has largely focused on culturally relevant information for expressing behaviors and beliefs of cognitive agents, we are unaware of similar efforts in developing a rich and robust markup process for inanimate objects. Indeed, Barba et al. (2006), Hill et al. (2006), and Johnson et al. (2008) all describe various approaches to culturally-tuned and language-appropriate interactions between cognitive agents and players. The focus on rapport building at an individual level, though, is able to avoid developing the larger social and physical systems that we attempt here.

However, the U.S. military has invested resources into this area with its importance to counterinsurgency doctrine and human terrain analysis. The Counterinsurgency Field Manual outlines a review of key structure groups and capabilities as two components in its ASCOPE (Areas, Structures, Capabilities, Organizations, People, Events) assessment (Anon., 2007). Table 1 provides an overview of structure categories used in this framework.

**Table 1: Structures in ASCOPE framework**

- |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"> <li>• Government centers</li> <li>• Headquarters and bases for security forces</li> <li>• Police stations, courthouses, and jails</li> <li>• Communications infrastructure</li> <li>• Roads and Bridges</li> <li>• Ports of entry</li> <li>• Dams</li> <li>• Power stations</li> <li>• Sources of potable water</li> <li>• Sewage systems</li> <li>• Clinics and hospitals</li> <li>• Schools and universities</li> <li>• Places of religious worship</li> </ul> |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

The video game industry has also provided some insight into the problem of handling hundreds of interactive inanimate objects in sandbox-type immersive games with emergent gameplay. Coming from the point of view of ingame experience and enjoyment, developers focus on mechanics that can lead to interesting and immersive gameplay dynamics, thus resulting in some emotional response (Hunicke et al., 2004). Here we equate mechanics with potential interactions a player or agent may have with objects. Adjusting mechanics and rules in a game environment lead to better game dynamics and more player enjoyment and, while a developer's focus may be keeping players interested, it is similar to our goal of establishing relevant interaction capabilities for training and gameworld exploration. Additionally, developers have shown that exposing mechanics through objects in a rich immersive environment can lead to realistic emergence in gameplay (Smith et al., 2004).

### 3. Agent Framework Overview

PMFserv is a human behavior emulator that drives agents in simulated gameworlds. This software was developed over the past 11 years at the University of Pennsylvania as a "model of models" architecture to synthesize many best available models and best practice theories of human behavior modeling (Silverman et al., 2006). PMFserv agents are unscripted, using their micro-decision making to react to actions as they unfold and to plan out responses. A

performance moderator function (PMF) is a micro-model covering how human performance (e.g. perception, memory, or decision-making) might vary as a function of a single factor (e.g. event stress, time pressure, grievance, and so on). PMFserv synthesizes dozens of best available PMFs within a unifying mind-body framework and thereby offers a family of models where micro-decisions lead to the emergence of macro-behaviors within an individual. For each agent, PMFserv operates its perception and runs its physiology and personality/value system to determine coping style, emotions and related stressors, grievances, tension buildup, impact of rumors and speech acts, and various mobilization and collective and individual action decisions to carry out the resulting and emergent behaviors. None of these PMFs are "home-grown"; instead they are culled from the literature of the behavioral sciences. Users can turn on or off different PMFs to focus on particular aspects of interest. When profiling an individual, various personality and cultural profiling instruments are utilized with visual software tools and web interviews to elicit the parameter estimates from a country, leader, or area expert.

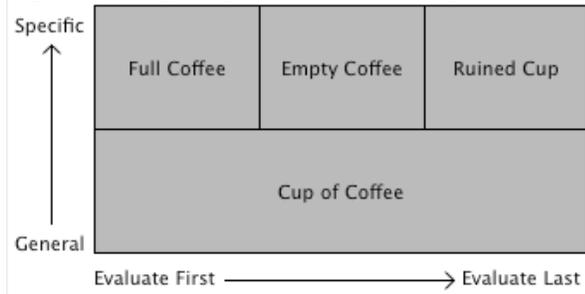
#### 3.1 Affordance Theory

A key concept in PMFserv that assists in modular modeling and object reuse is the implementation of affordance theory, introduced by psychologist James J. Gibson, to manage when and how agents and objects may be perceived and acted on (Cornwell et al., 2003). Each entity in the world – agents, inanimate objects, abstract objects, organizations -- applies perception rules to determine how it should be perceived by each perceiving agent. Entities then reveal the actions (and the potential results of performing those actions) afforded to the agent. For example, an object representing a car might afford a driving action that can result in moving from one location to another. A business might afford running it, working there, purchasing goods, and/or attacking and damaging it. These affordance markups permit PMFserv agents to perceive and reason about the world around them.

A simple example of a cup of coffee "marked up" for such perceptions is shown in Figure 1. Each gray box in the grid represents one way in which the object may be perceived. We call these perceptual types, or p-types. Rules on a p-type allow a modeler to establish appropriate contexts for the object to be viewed in that way. When active, p-types afford actions to the perceiving agent and the decision-making process can proceed. For example, an active "Full Coffee" p-type affords a "Drink" action with assured success, while an active "Empty Coffee" affords a "Drink" action with assured failure and a "Get Refill" action with arbitrarily

defined success and fail probabilities. It should be noted that the grid imposes an evaluation structure whereby more general p-types are evaluated at the bottom and the perception algorithm works its way up from left to right; p-types on the same row are mutually exclusive.

**Figure 1: Simple representation of a cup of coffee**



### 3.2 NonKin Village Overview

The use case for an object taxonomy is a training game, called NonKin Village, where the player(s) interacts with other virtual or real followers and leaders of contending factions at a local village level. These factions offer a corrupt sim-city type of world where one must convince various “crime” families to convert to legit operation. NonKin is also used to simulate insurgent operations in the village. The insurgent leader uses recruits to carry out missions. The player (s) has constrained resources, and must use them judiciously to try and influence the world via an array of Diplomatic, Intelligence, Military, and Economic (DIME) actions. The outcomes are presented as a set of intended and unintended Political, Military, Economic, Social, Informational, and Infrastructure (PMESII) effects.

The goal is to push the player through the three stages of counter-insurgency (COIN) theory: survey the social landscape, make friends/co-opt the agenda, and foster self-sustaining institutions so the player can safely depart (Anon., 2007). The player learns to use the given resources in a culturally sensitive way to achieve desired outcomes. All agents in the game are conversational and are able to explain their internal states, group grievances, relations/alignments, fears, and wants. The agents carry out daily life functions in the village in order to satisfy their internal needs for sleep, sustenance, company/belonging, maintaining relationships, prayer, etc. The village has places of employment, infrastructure, government and market institutions, and the leaders (agents) manage the economic and other institutional resources of their factions (Figure 2).

**Figure 2: Screenshot of NonKin Village in 2D**



### 4. Taxonomy Overview

Given the large set of possible objects needed in an immersive village gameworld, we first divide such objects into two categories, basic and functional. Basic objects are not essential to potential storylines or fundamental pieces of gameplay, but are important for immersion. Examples may include weapons or gas canisters. On the other hand, functional objects are important structures or items with major roles in gameplay and agent behavior. Hospitals, marketplaces, and homes are some examples of such objects. As functional objects are the most relevant to discussions of agent behavior and data categorization efforts, we will focus on them.

We divide functional objects into at least one of six distinct categories: military, religious, economy, government, media, and residential. This classification essentially permits implementation level details to be attached to objects when inserted into a simulation world. For example, in a certain context (e.g. cultural area), residential objects afford a set of actions and perceptions to agents. The goal is to separate input data from external, independent knowledge engineering efforts.

A major challenge of standing up a village-like environment are the countless structures in which people live and work, provide services, and produce or distribute resources. Such structures consist of two elements in this context, a physical layer and a services layer (Figure 3). As a physical entity in the world, a structure has an effect in terms of perception. It has dimensions in some configuration, is built with some material with inherent physical properties that lend to strength of the structure, and it may be owned by some individual, group, or government. Most structures,

however, are not empty; they have some function and satisfy a purpose. An optional services layer on top of a structure layer provides these affordances in a simulation. As Figure 3 highlights, our taxonomy categorizes service “packages” are broken into medical, transportation, communication, public/private works, emergency, and government. These groupings allow natural categorizations of terrain data from the modeler’s perspective and, like the structures layer, permit an independent augmentation of implementation specifics.

**Figure 3: Dual layer of services and structure**



**4.1 Affordances**

The markup of objects with detailed and extensive properties allows both agents and players to interact with them and through them. By “with them” we imply actions that are afforded by the objects and subsequently taken by an entity. Examples of this might include seeking shelter in a residence that is considered an entity’s home, or searching a building. All objects, basic or functional, are imbued with a starting set of actions (Table 2). However, the taxonomy remains independent of these action sets and the markup links to these action types may be exchanged depending on model demands and constraints.

**Table 2: Basic action types**

Action Type	Examples
Investigative	Search, raid, confiscate
Transactional	Buy, sell, give, exchange
Destructive	Destroy, detonate
Constructive	Build, repair, replace

Further action sets derive from object markups associated with property classes. Assigning a property

class immediately attaches relevant state properties and afforded actions to objects, available to entities in a simulation (Table 3). Functional objects, of course, will afford additional actions and capabilities associated with their service property classes.

**Table 3: Property classes**

Physical	Symbolism
<ul style="list-style-type: none"> <li>• Flammable</li> <li>• Can throw</li> <li>• Can shoot</li> <li>• Can enter/exit</li> <li>• Edible</li> <li>• Etc.</li> </ul>	<ul style="list-style-type: none"> <li>• Religious</li> <li>• Personal</li> <li>• Family</li> <li>• Tribe</li> <li>• Country</li> </ul>

While a rich set of actions help agents and players interact directly with the environment in a realistic manner, an equally important component in human terrain training settings is the non-kinetic, conversational aspects. Intelligence gathering efforts often highlight exposing relationships not only between individuals, but also between individuals and physical objects such as buildings, institutions, and offensive weapons. In other words, we seek a way to transform data – properties, numbers, symbols – into information, knowledge, and eventually an understanding of the entire area at all levels and perspectives (Ackoff, 1989).

The rich markups that are facilitated by the taxonomy permit the development of an utterance framework by which players may inquire agents about objects in the world. By way of simple, stored statement fragments with an ability to adapt to the subject of conversation, players acquire information, or relational connections between entities and facts (Silverman et al., 2010). This subsequently leads to acquired knowledge of entities (e.g. John Doe lives at address X and is the head of the household). Used in a training capacity, as this technology currently is, a player builds up knowledge of the area of operation and is encouraged to work toward an understanding of the environment, which would allow insight into behaviors and answer “why” questions.

**5. Implementation**

We assume that a large dataset of objects and structures has been tagged with the appropriate categories and classes to facilitate a semantic mapping to software instances in a simulation. Silverman (2009) outlines a method by which canonical templates of objects are created by a modeler and are subsequently combined with external metadata to automatically generate PMFserv-valid objects in the simulation world. Once

an object has been instantiated in the world, it is perceivable to agents and players inhabiting the world.

Transitioning from the conceptual level of the taxonomy and the dual-layer construction of functional structures, collections of p-types and their dependent state properties were created to associate categories with markups needed for perception in the simulation (Figure 4). Although categories in the taxonomy may consist of dozens of p-types at the implementation level, efforts are underway to create simple groupings to allow swapping out of category packages from objects. For example, consider an immersive game world in which insurgent agents have overtaken what was initially a school. Aside from dynamic changes to its structural properties (e.g. damage and fortification levels), it would be necessary to swap out the schooling services layer with a military layer. Such a change would have been traditionally accomplished by removing the object from the simulation entirely and reinstantiating the modified structure. However, this method is undesirable in an immersive environment; dynamic swapping of object components preserves information relationships with the larger terrain and social area.

When a player chooses to engage an agent in conversation, potential topics of discussion include objects in the world, including basic objects such as nearby weapons or functional structures such as a home

or local health clinic. By default all objects are available, but additional considerations (geometric, obstructions) can limit the scope in some cases. An agent may also not have much connection or awareness of a structure so transferrable information will vary.

A simple player-agent conversation example is illustrated in the sequence of screenshots in Figure 5. It should be noted that such interaction models are independent of visual platform and would proceed similarly in a 2D prototype platform (shown) or a 3D world such as VBS2 (shown in Figure 6). In this interaction, using a drop-down list for choosing available statements and questions, the player has approached an agent called Fakhir Badir-Aldin in the Heremat tribal area of the fictional village and, after learning his name, asks him about the tribe's area. Since buildings have been marked up universally according to the taxonomy, it is straightforward for the NonKin software to elicit information from these models and allow agents to reveal properties of known objects in a natural manner. In the first panel of Figure 5, the player may choose a building related to the area of interest. In the second panel we can see that the player has first inquired about a structure called ShameelHome but the agent has no connection to it. Once asked about HammoodHome, the agent responds by stating that he lives there along with his four family members. While this is a simple demonstration, efforts are underway to take advantage of a 3D visual

Figure 4: P-type representation of a media source building

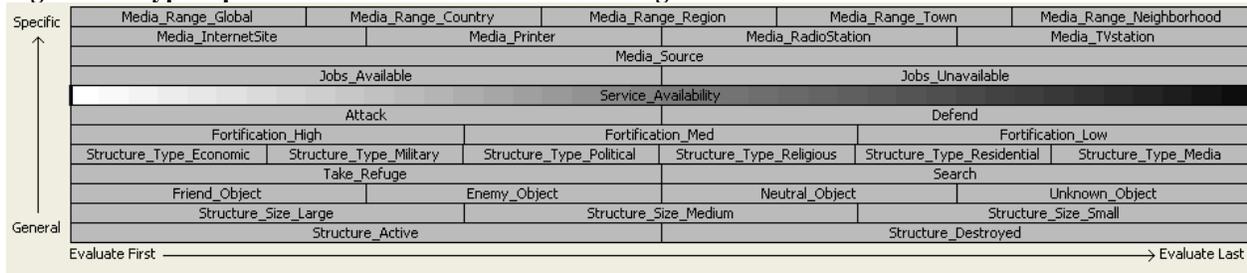
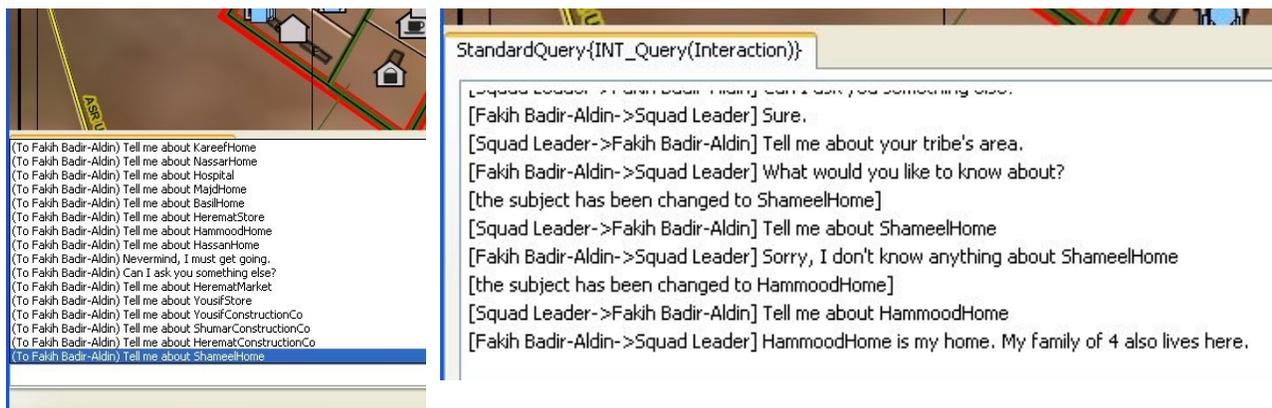


Figure 5: Screenshots of a player asking an agent about local buildings (left) and two responses (right)



environment for line-of-sight and spatially-related inquiries (e.g. “Do you anything about these new homes here?”).

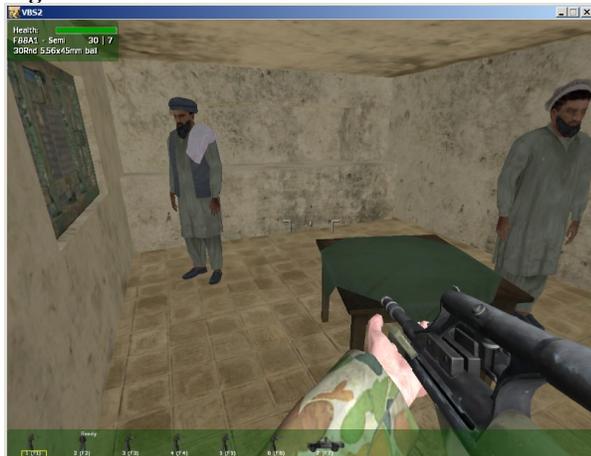
## 6. Discussion and Future Work

We have presented an approach for developing a rich collection of inanimate functional objects in an immersive environment populated with interactive and intelligent socio-cognitive agents. Having established a taxonomy by which arbitrary datasets may be tagged in a fruitful manner, we successfully brought to bear existing modeling techniques in the PMFserv framework to facilitate modular compositions of important objects in a simulation world.

It is a hope that this taxonomy may provide a common standard or lexicon for other modeling and simulation efforts in cultural awareness training in immersive 3D environments. With a foundation in place, extensions to the common language can assist in simulation interoperability and independence from virtual world representations.

As training via immersive environments continues to grow and mature, rapid scenario development will likely become critical in time-sensitive areas. The procedure from data gathering to scenario construction to in-game training may call for automatic generation of virtual objects. Consider a military application where a small unit has been given minimal notice on a mission in a particular area (e.g. an urban area or village). We foresee a capability in which prior information, intelligence, and terrain data contribute to automatic creation of the area along with a population of appropriately modeled socio-cognitive agents. Our current and future work takes steps toward this vision as we develop libraries and modular models that can tie into arbitrary virtual worlds (Figure 6).

**Figure 6: Screenshot of an encounter in VBS2**



## 7. References

- Ackoff, R.L. (1989). From Data to Wisdom. *Journal of Applied Systems Analysis*, 16, 3-9.
- Anon. (2007). *The U.S. Army/Marine Corps Counterinsurgency Field Manual, FM3-24*.
- Barba, C., Deaton, J.E., Santarelli, T., Knerr, B., Singer, M., Belanich, J. (2006). Virtual Environment Composable Training for Operational Readiness (VECTOR). Presented at 25<sup>th</sup> Army Science Conference, Orlando, FL.
- Bellinger, G., Castro, D., Mills, A. (2004). Data, Information, Knowledge, and Wisdom. Retrieved from the web January 2, 2010. <http://www.systems-thinking.org/dikw/dikw.htm>
- Cornwell, J.B., O'Brien, K., Silverman, B.G., Toth, J.A. (2003). Affordance Theory for Improving the Rapid Generation, Composability, and Reusability of Synthetic Agents and Objects. In *Proceedings of the 12<sup>th</sup> Conference on Behavior Representation in Modeling and Simulation*.
- Hill, R.W., Belanich, J., Lane, H.C., Core, M., Dixon, M., Forbell, E., Kim, J., Hart, J. (2006). Pedagogically Structured Game-Based Training: Development of the ELECT BiLAT Simulation. In *Proceedings of the 25<sup>th</sup> Army Science Conference*.
- Hunicke, R., LeBlanc, M., Zubek, R. (2004). MDA: A Formal Approach to Game Design and Game Research. Retrieved from the web October 27, 2008. <http://cs.northwestern.edu/~hunicke/MDA.pdf>
- Johnson, W.L. and Valente, A. (2008). Tactical Language and Culture Training Systems: Using Artificial Intelligence to Teach Foreign Languages and Cultures. In *Proceedings of the 20<sup>th</sup> Innovative Applications of Artificial Intelligence (IAAI) Conference*.
- Silverman, B.G., Bharathy, G., O'Brien, K. (2006). Human Behavior Models for Agents in Simulators and Games: Part II – Gamebot Engineering with PMFserv. *Presence*, 15, 2, 163-185.
- Silverman, B.G., Pietrocola, D., Weyer, N., Weaver, R., Esomar, N., Might, R., Chandrasekaran, D. (2009). NonKin Village: An Embeddable Training Game Generator for Learning Cultural Terrain and Sustainable Counter-Insurgent Operations. In F. Dignum, J. Bradshaw, B. Silverman & W. van Doesburg (Eds.), *Agents for Games and Simulations*, (pp. 135-153). New York: Springer.
- Silverman, B.G., Pietrocola, D., Weyer, N., Osin, O., Johnson, D. (2010). Toward a model-driven architecture for agents in immersive environments. Submitted for publication.

Smith, H. and Smith, R. (2004). Practical Techniques for Implementing Emergent Gameplay. Presented at the *Game Developers Conference*.

## Author Biographies

**DAVID PIETROCOLA** is a doctoral student in the Department of Electrical and Systems Engineering at the University of Pennsylvania. His current work explores the knowledge engineering and modeling challenges for the synthesis of socio-cognitive agents and realistic village behaviors in immersive virtual worlds. He graduated Phi Beta Kappa with a B.S. in electrical engineering from Trinity College in Hartford, Conn., is a member of IEEE and IEEE Computer Society, and became a volunteer for IEEE-USA after participating in the Washington Internship for Students of Engineering program.

**DR. BARRY G. SILVERMAN** (basil@seas.upenn.edu), is a Professor of Systems Engineering at the University of Pennsylvania. He is a Fellow of IEEE, AAAS, and the Washington Acad. of Science, and sits on the board of several organizations and journals in the intelligent systems fields. Barry's research over the past 34 years has largely been on socio-cognitive modeling of intelligent software agents able to interact as humans would do (ie, illustrating a descriptive, not normative, model of behavior). He has created models of human physiology, stress, emotion, personality, culture, factional, and relationship dynamics. These have been integrated to produce agent-based sims of ethno-political situations around the world; insurgency, crowd, and leader simulators; and several role playing games (RPGs). As a result of all this work, Barry is also the author of over 140 articles, 13 books/proceedings, over 100 technical reports, 7 copyrighted software systems, a boardgame, and several research and teaching excellence awards from ORSA, IEEE, AAAI, BRIMS.

## Sailing to the Model's Edge: Testing the Limits of Parameter Space and Scaling

*Amy Santamaria*

*Walter Warwick*

Alion Science and Technology

MA&D Operation

4949 Pearl East Circle, Suite 200

Boulder, CO 80301

303-442-6947

[asantamaria@alionscience.com](mailto:asantamaria@alionscience.com), [wwarwick@alionscience.com](mailto:wwarwick@alionscience.com)

Keywords:

Naturalistic Decision Making, Human Performance Modeling

**ABSTRACT:** *Using the MS/RPD integrated modeling approach, we have modeled a variety of tasks. We typically try to capture aspects of human performance and evaluate the qualitative and quantitative fit of model behavior to human data. A collection of individual models and demonstrations of fit to human data constitute an important validation of a modeling approach. However, there are problems with focusing solely on the “good fit” and “typical model” section of model complexity and parameter space. In this paper, we argue that as modelers, we need to examine our approaches in a broader context, going beyond the comfort zone of good fit and typical models. Using a very simple “generic” model, we examined a relatively small search space, with the goal of better covering and understanding a wider range of complexity and parameter values than our typical models utilize. We investigated scaling by systematically increasing the number of cues and COAs, and we investigated a range of values for three key model parameters. We learned something about limits of scaling. In our parameter exploration, the results underscored the importance of exploring the full range of possible values because parameter values did not always affect performance and learning in a monotonic way.*

### 1. Introduction

Over the past ten years, we have constructed and presented models of a variety of tasks using our MS/RPD approach (Warwick, McIlwaine, Hutton, & McDermott, 2001; Warwick & Hutchins, 2004; Warwick & Fleetwood, 2006; Warwick & Santamaria, 2006; Santamaria & Warwick, 2007; 2008). Our approach combines Micro Saint task network modeling (the MS component) with underlying learning and memory mechanisms that capture key aspects of recognition-primed decision making (the RPD component) in an integrated architecture. The MS component breaks down tasks into their constituent processes, creating a kind of “dynamic flowchart,” represented as a network of tasks. The RPD component uses a multiple-trace model of long-term memory, a similarity-based recall mechanism, and simple reinforcement-based learning to set values or determine the flow of control in the task network. Using this integrated modeling approach, we typically we focus on a single task, constructing a model, trying to capture aspects of human performance, and evaluating the qualitative or quantitative fit of model behavior to the human data.

A collection of individual models and demonstrations of fit to human data constitute an important validation of a modeling approach. However, there are bigger issues to take into consideration when developing, exploring, and evaluating a modeling framework. There are problems with goodness of fit as the sole criterion (see Roberts & Pashler, 2000, Collyer, 1985). But more critically, there are problems with focusing solely on the “good fit” and “typical model” section of model size and parameter space.

Several important points related to issues of scaling are brought out in Gluck et al. (2007). The authors describe three levels of theory that are implemented in models of cognition: architecture and control mechanisms (Type 1), internal component/module implementation (Type 2) and knowledge (Type 3). Gluck et al. point out that the parameter space for each of those levels is very large and that a typical modeling effort only selects a single point at the intersection of these spaces. From their paper:

A thorough search of even a modest portion of the total possible theoretical state space will require an unprecedented amount of computing

power because of the combinatorics associated with searching a multi-dimensional space...seemingly innocuous assumptions and implementation decisions can have dramatic consequences downstream in a complex system like a cognitive architecture that interacts with a simulation environment

The tendency in modeling is to focus on “pet problems” where the model succeeds. However, the potential parameter space for any given model is huge. We modelers need to examine our approaches in a broader context, not just the “good fit” space, or comfort zone. This problem is well laid out in Best et al. (2009):

The de-facto approach to cognitive modeling is more often a focus on maximizing fit to human data. This is done through either hand-tuning based on the intuition and experience of the modeler or automated optimizing of the fit...Any of these approaches can be sufficiently successful, but they provide little data about the performance of the model outside of the ultimate parameter values used in presenting the final fit.

Best et al. also point out the benefits of such exploration of parameter space:

Information about how a model performs outside the best-fitting parameter combination provides modelers with information about...the full range of behavior possible from the model and how different parameters interact to generate possibly complex behavioral dynamics.

Our modeling approach is simpler than the typical cognitive architecture of the type Gluck et al. and Best et al. describe (e.g., ACT-R or Soar), but issues of scaling still apply. For this paper, we examined a relatively small search space with a very simple model, but our goals were similar – to cover and better understand a wider space than our typical models explore.

In a recent paper (Santamaria & Warwick, 2009), we gave an overview of our MS/RPD modeling approach, the ground we have covered and tasks we have modeled, and our vision for the next steps to take. In our “next steps” section, we promised to “systematically investigate the computational limits of our algorithms, scaling up a simple model by adding cues and courses of actions.”

To follow through on this promise, we constructed a “generic” model without built-in assumptions about tasks or processes (and the expectations that come with them); the inputs to the model are cue 1 through cue n, and the values of these cues determine the selection of one of m

courses of action (COAs). We used this model to explore issues of scaling by systematically increasing the number of cues and COAs. We went beyond the typical size for MS/RPD models, on the order of 2 cues and 2 COAs, to explore up to 15 cues and 5 COAs. Using the same generic model, we also investigated a range of values for three key model parameters: the activation exponent, the COA selection mechanism, and confidence.

## 2. The Generic Model: A Testbed

The generic model was developed to explore scaling and parameter space issues. Why did we construct a generic model? In our models, closed form analytic solutions are not obvious or even tractable. Even the simplest cognitive models are fairly complicated pieces of software, and they need to be explored empirically. The generic model can be incrementally scaled up in the number of cues and the number of courses of action. In this section, we describe the underlying learning, memory, and recognition mechanisms and the construction and cue structure of the generic model.

### 2.1 Learning, Memory, and Recognition Mechanisms

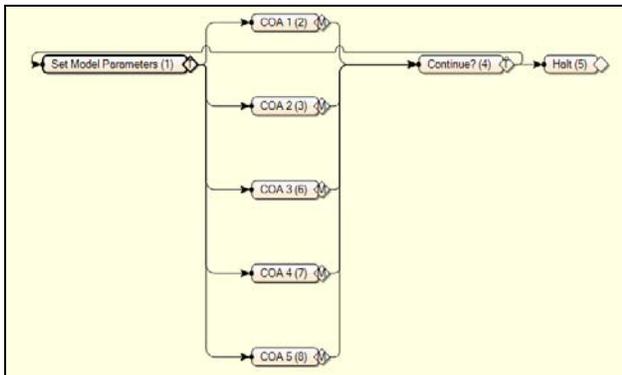
Our decision modeling mechanism was inspired by Klein’s theory of the recognition primed decision, or RPD (see Klein, 1998). It uses a multiple-trace mechanism based on the multiple-trace model of memory (see Hintzman, 1984; 1986a; 1986b). Following Klein, the major features of our modeling approach are cues and COAs, and the associations between them. Models learn the associations between cues and COAs through experience, and this accumulation of this experience can be modified by several recognition and learning parameters. These parameters include the activation exponent, the COA selection mechanism, and confidence, each of which is described in more detail below.

### 2.2 Construction and Cue Structure

The high-level task structure of the generic model is shown in Figure 1. The first task sets the model parameters, including number of cues, number of COAs, runtime, number of situations, and cue-to-COA mapping.

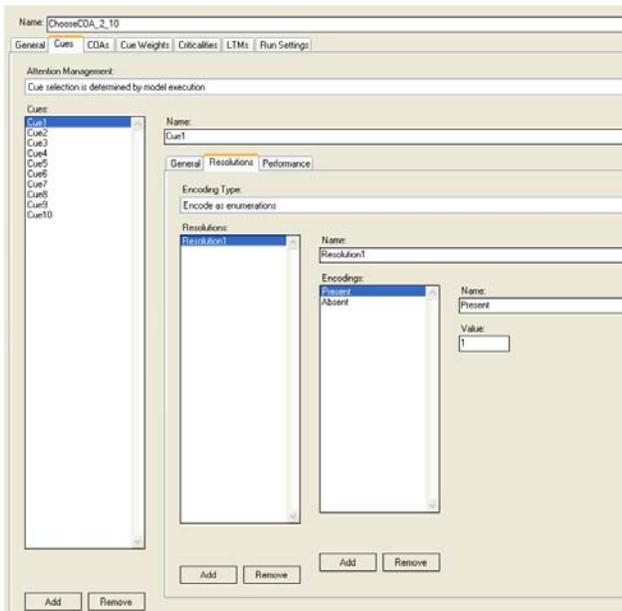
We explored several different cue-to-COA mappings in order to reduce the chance that we had hidden or “smuggled in” informative structure that essentially gave extra help to the model. Standard experimental paradigms are carefully crafted to have internal structure that is predictable and learnable. The model can latch on to certain kinds of structure, but what happens when the structure is completely arbitrary? We tested several mappings, including random assignment of cue combinations (situations) to COAs (“random”), a list-

based mapping covering all possible combinations (“alternating”), an offset list-based mapping (“offset”), and a mapping based on the location of cues in the situation vector (“left-right”). Results were similar for all mappings; the results reported in this paper used either the random or the alternating mapping.



**Figure 1. The task structure of the generic model.**

After setting model parameters, the task network model passes control to the RPD (decision) model, which selects a COA. Figure 2 shows the screen where cues are specified in the RPD model.



**Figure 2. Specifying cues in the decision (RPD) model.**

Next, the task network model resumes, goes to the “continue” task, and if runtime is not yet up, loops back to make another decision. There are no actual consequences in the task network model of choosing one COA over another other; that is why we call this a “generic” model that does not have built-in task assumptions.

### 3. Scaling Up Model Complexity

To test effects of scale and explore a wider range of model size than we typically investigate, we systematically changed the number of cues and the number of COAs in the model.

We tested all combinations of cues and COAs from one to five cues and from two to five COAs. To ensure that all cue situations deterministically predicted a COA, we omitted combinations with fewer cue situations than COAs. An example is the combination of three COAs and one cue (3-1); with one cue, there are two cue situations that cannot uniquely map to three different COAs. The combinations tested are listed in Table 1.

**Table 1. Combinations of cues and COAs tested.**

Cues	COAs			
	2	3	4	5
1	2-1	X	X	X
2	2-2	3-2	4-2	X
3	2-3	3-3	4-3	5-3
4	2-4	3-4	4-4	5-4
5	2-5	3-5	4-5	5-5

We tested each model holding confidence at medium and the activation exponent at 15. The cue-to-COA mapping was the “alternating” mapping and runtime was 500 trials. Figure 3 and Figure 4 show the results of these tests. They present the same data but group them differently, with Figure 3 showing the effect of number of COAs by grouping the models by number of cues, and Figure 4 showing the effect of number of cues by grouping the models by number of COAs.

Figure 3 shows the effect of number of COAs on learning for models with 2 cues (top left), 3 cues (top right), 4 cues (bottom left), and 5 cues (bottom right). Learning differences are very small for 2 or 3 cues. However, when the number of cues increases to 4 or 5, adding COAs slows learning. Tests with long runs showed that it takes much longer for model 5-5 to reach asymptote than for model 2-5 to reach asymptote.

Figure 4 shows the effect of number of cues on learning for models with 2 COAs (top left), 3 COAs (top right), 4 COAs (bottom left), and 5 COAs (bottom right). Again, learning differences are small for a small number of COAs but grow larger as the number of COAs increase.

### 4. Exploring Parameter Values

With our generic model, we explored three of the parameters that are available in the MS/RPD modeling

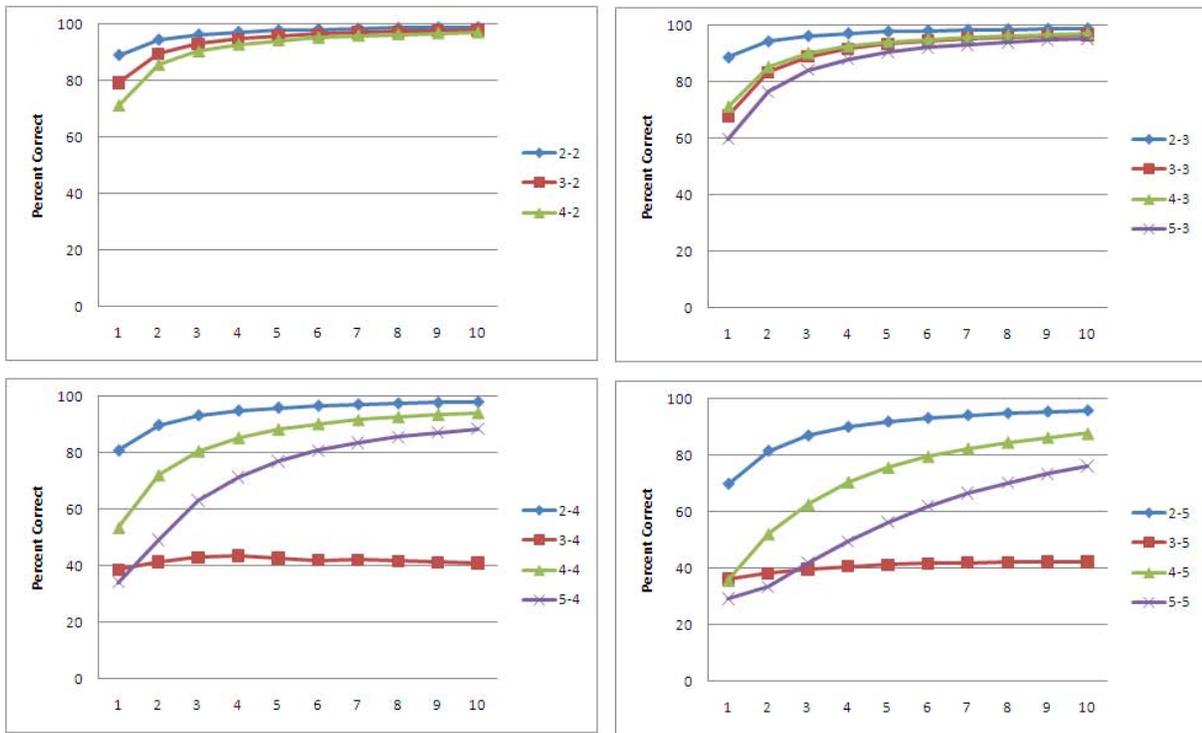


Figure 3. Effect of number of COAs on learning for 2, 3, 4, and 5 cues (left to right, top to bottom). Models are referred to as A-B, where A is the number of COAs and B is the number of cues. Time is on the x-axis (trial/50).

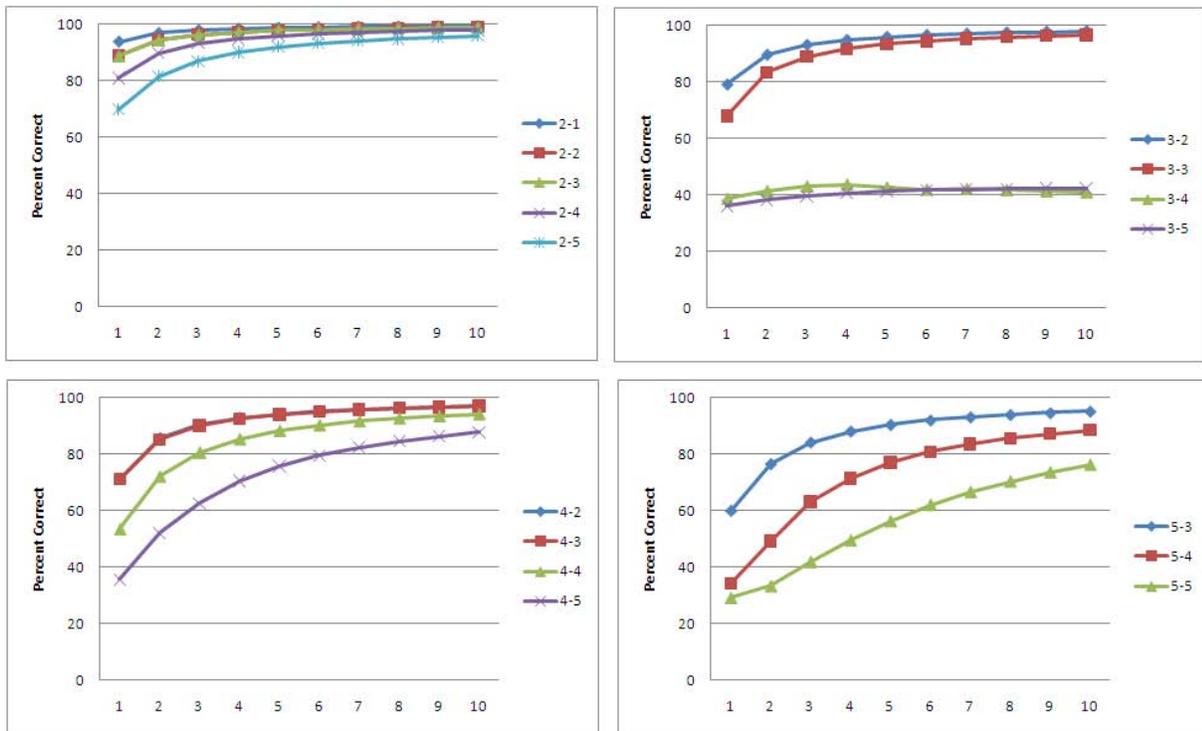


Figure 4. Effect of number of cues on learning for 2, 3, 4, and 5 COAs (left to right, top to bottom). Models are referred to as A-B, where A is the number of COAs and B is the number of cues. Time is on the x-axis (trial/50).

approach: activation exponent, COA selection mechanism, and confidence.

### 4.1 Activation Exponent

The first parameter we explored with the generic model was the activation exponent. Remember that the MS/RPD approach uses a similarity-based recall mechanism. The similarity value between the current episode and all the episodes in long-term memory is raised to a power, the activation exponent. The similarity value determines the proportion that each remembered episode contributes to the recognition process. A higher value for the activation exponent means that the match must be more exact for the remembered episode to contribute to the current decision.

We tested the 2-10 model (2 COAs and 10 cues), holding confidence at medium and COA selection at default. The cue-to-COA mapping was the “random” mapping, and runtime was 5000 trials. With 2 COAs, chance performance is 50 percent correct. As shown in Figure 5, all versions of the model performed above chance. A higher activation exponent yielded better performance and a faster learning curve.

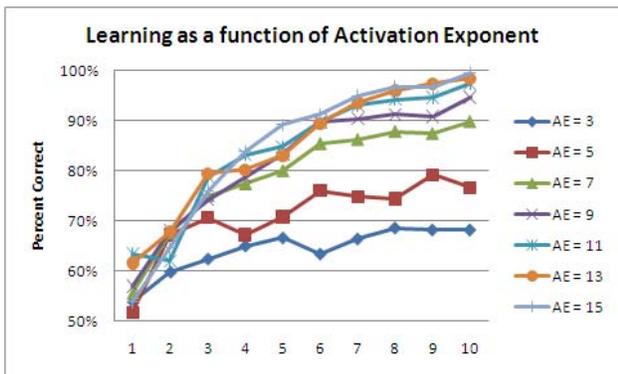


Figure 5. Learning (percent correct over time) as a function of activation exponent for the 2-10 model, for a run of 5000 trials. The x-axis is trial/500.

For this model, activation exponent is an important parameter. Holding everything else constant, it can improve overall performance from 64 percent correct to 85 percent correct. Figure 6 shows overall percent correct (across all trials) for the 2-10 model for activation exponent values of 3 to 15.

### 4.2 COA Selection Mechanism

The second parameter we explored with the generic model was the COA selection mechanism. The COA selection mechanism controls how the model will choose among recognized courses of action. By default, the model will always choose the COA most strongly

recognized as successful among those that exceed a recognition threshold; conversely, the model will not choose any COAs that have been recognized as unsuccessful. This selection strategy is referred to as “default”.

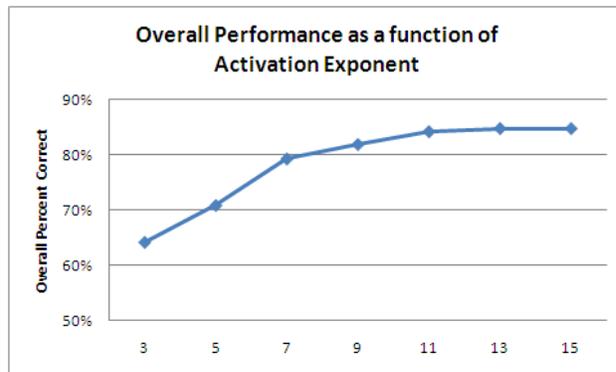


Figure 6. Overall percent correct as a function of activation exponent for the 2-10 model, for 5000 trials.

The default strategy is intended to steer the model toward the most successful COAs. The model can also employ a “fuzzy” selection strategy where it tends to choose the COA recognized as most successful, but not always. The fuzzy option uses a probabilistic draw weighted with respect to the normalized strength of recognition for each COA.

We tested the 2-10 model (2 COAs and 10 cues), holding confidence at none and COA selection at default. The cue-to-COA mapping was the “alternating” mapping. The effect of COA selection mechanism on learning for the first 200 trials is shown in Figure 7. Both default and fuzzy mechanisms result in similar performance, but they differ in the initial spin-up over the first 50 trials. On average, across a batch of ten runs, the model using the default mechanism spins up more quickly.

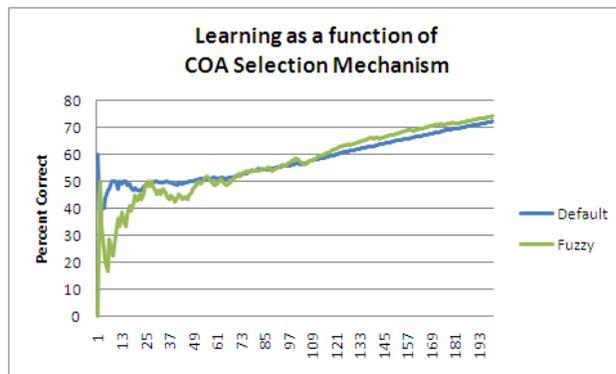


Figure 7. Effect of COA selection mechanism on learning for the first 200 trials. (Default and fuzzy are each averaged over 10 runs.)

### 4.3 Confidence

The third parameter we explored with the generic model was confidence. Confidence sets a threshold above which the model will recognize a COA. The lower the threshold, the less “confident” you can be that the recognition is due to systematic associations in long term memory between situations and COAs rather than the noise inherent in the similarity-based recognition process. Viewing long-term memory as a “population” of experience, the threshold corresponds to the number of standard deviations from the mean recognition value one would expect from a population of random experiences. Low confidence corresponds to one standard deviation, medium to two standard deviations, and high to three standard deviations.

The effects of confidence should show in early trials, as the model spins up. Early trials are especially important in models that are very sensitive to noise and initial effects. We have seen confidence affect early performance and spin-up in other models. However, our tests did not reveal differences in the generic model for different levels of confidence across a variety of conditions (specific results are not reported here).

## 5. Discussion

We used the generic model to investigate 1) scaling beyond our typical model size and 2) a range of values for several key model parameters. In the exploration of scaling, we found that we could increase either cues or COAs with only a very minor slowing of learning, but that increasing both beyond three led to a much larger slowdown in learning.

These results demonstrate the syntactic nature of the model. It is not learning anything about specific COAs or cues; it is learning about the combination of COAs and cues. This is evident in the symmetry of the effect of scaling up in number of cues and COAs on performance. It doesn't matter if the increase in decision space size is due to cues or COAs; the model is sensitive to the size of the decision space, not the source of the complexity.

In addition to the results presented here, we built models that scaled up even further: a 2 COA, 10 cue model (2-10), a 2 COA, 15 cue model (2-15), and a 5 COA, 10 cue model (5-10). The 2-10 model was able to learn to asymptote, although it took longer to reach asymptote than did models whose number of cues/number of COAs were capped at 5. The 2-15 and 5-10 models were not able to converge, even with runtimes of 25,000 trials. This was because of the very large space to learn (all combinations of cues were possible and had an assigned “correct answer”). For example, the 2-10 model had  $2^{10}$ ,

or 1024, possible cue combinations. The 2-15 model had  $2^{15}$ , or 32,768, and the 5-10 model had  $5^{10}$ , or 9,765,625! When we limited the number of possible cue combinations the model could face (to 50, 100, even 500), the 2-15 and 5-10 models were able to learn without a problem. So scaling up the cue and COA space and scaling up the situation space are actually separate issues.

Two of the parameters we examined provided interesting results: activation exponent and the COA selection mechanism. The value of the activation exponent made a substantial difference in the model's learning and performance. The higher the activation exponent, the faster the learning. Differences were largest among smaller activation exponents (3 to 7), and learning curves became more similar for higher values (9-15). Overall performance (percent correct) also improved as activation exponent increased, with the largest differences at the small end of the parameter scale.

It was important to explore the full range of possible activation exponent values because they did not uniformly affect performance and learning. The lesson from our exploration of this parameter is that you need to make sure the activation exponent is high enough (maybe 7 or higher), but beyond a certain point, it does not make much of a difference in the model's performance.

The COA selection mechanism showed a difference in learning but not performance. On average, the model reached similar levels of accuracy with default and fuzzy mechanisms, but it learned faster with default, showing better performance than fuzzy on the first 50 trials.

There were two puzzling results with the generic model that have not yet been explained. The first puzzling result was that model performance on the 3 COA, 4 cue (3-4) and 3 COA, 5 cue (3-5) models stagnated at chance performance. We suspect this is an anomaly resulting from the way cues were mapped to COAs (the “right answers” for which the model was reinforced).

The second puzzling result was the absence of a result for confidence. Earlier models have shown effects of confidence, particularly on early performance and spin-up. However, the generic model failed to show an effect of confidence under a variety of conditions. An effect of confidence should show up where there are systematic associations over and above the noise present. However, in the generic model, we deliberately built random cue-to-COA mappings - this is only noise! So there are no systematic associations inherent in cue structure. Finding no effect of confidence in this model is actually a validation that we haven't smuggled in any informative internal structure or biases, providing a purer test of the model's ability to learn essentially arbitrary relationships.

## 6. Conclusions

In this paper, we have described our integrated modeling approach and our attempts to push its boundaries a bit. While it is important for a modeling approach to build a repertoire of single-task models validated with human performance data, we have argued that it is also important to explore beyond the "good fit" areas of parameter space and the "typical model" areas of complexity space/scale.

Examining a relatively small search space with a very simple "generic" model, we attempted to gain a better understanding of a larger space than we typically explore with our models. We learned some interesting things as we tried to scale up the model and systematically move across parameter space.

This is just the beginning of this effort. It is critical to go beyond holding all parameters but one constant in order to explore the intersection of parameter space and to understand how model parameters interact. These efforts are a very small step in an enormous and intimidating effort that is emerging in the modeling community: putting our modeling endeavors in a broader context and moving outside our modeling comfort zones.

## 7. References

- Best, B. J., Furjanic, C., Gerhart, N., Fincham, J., Gluck, K. A., Gunzelmann, G., & Krusmark, M. A. (2009). Adaptive mesh refinement for efficient exploration of cognitive architectures and cognitive methods. *Proceedings of the ?? International Conference on Cognitive Modeling*.
- Collyer, C. E. (1985). Comparing strong and weak models by fitting them to computer-generated data. *Perception & Psychophysics*, 38, 476-481.
- Gluck, K., Scheutz, M., Gunzelmann, G., Harris, J., & Kershner, J. (2007). Combinatorics meets processing power: Large-scale computational resources for BRIMS. In *Proceedings of the Sixteenth Conference on Behavior Representation in Modeling and Simulation* (pp. 73-83). Orlando, FL: Simulation Interoperability Standards Organization.
- Hintzman, D. L. (1984). MINERVA 2: A simulation model of human memory. *Behavior Research Methods, Instruments & Computers*, 16, 96-101.
- Hintzman, D. L. (1986a). *Judgments of Frequency and Recognition Memory in a Multiple-Trace Memory Model*. Eugene, OR: Institute of Cognitive and Decision Sciences.
- Hintzman, D. L. (1986b). "Schema Abstraction" in a Multiple-Trace Memory Model. *Psychological Review*, 93, 411-428.
- Klein, G. (1998). *Sources of Power: How People Make Decisions*. Cambridge, MA: The MIT Press.
- Roberts, S. & Pashler, H. (2000). How persuasive is a good fit? A comment on theory testing. *Psychological Review*, 107, 358-367.
- Santamaria, A. & Warwick, W. (2007). A naturalistic approach to adversarial behavior: Modeling the prisoner's dilemma. In *Proceedings of the 16<sup>th</sup> Conference on Behavioral Representations in Modeling and Simulation*.
- Santamaria, A. & Warwick, W. (2008). Modeling probabilistic category learning in a task network model. In *Proceedings of the 17<sup>th</sup> Conference on Behavioral Representations in Modeling and Simulation*.
- Warwick, W. & Fleetwood, M. (2006). A bad Hempel day: The decoupling of explanation and prediction in computational cognitive modeling. In *Proceedings of the Fall 2006 Simulation Interoperability Workshop, Orlando, FL, SISO*.
- Warwick, W. & Hutchins, S. (2004). Initial comparisons between a "naturalistic" model of decision making and human performance data. In *Proceedings of the 13th Conference on Behavior Representation in Modeling and Simulation*.
- Warwick, W., McIlwaine, S., Hutton, R. J. B., & McDermott, P. (2001). Developing computational models of recognition-primed decision making. In *Proceedings of the 10th Conference on Computer Generated Forces*.
- Warwick, W. & Santamaria, A. (2006). Giving up vindication in favor of application: Developing cognitively-inspired widgets for human performance modeling tools. *Proceedings of the 7th International Conference on Cognitive Modeling*.

## Author Biographies

**AMY SANTAMARIA** is a Senior Cognitive Scientist at Alion Science and Technology. Her research focuses on modeling human behavior and cognition and experimentation for robotics interfaces. She received her Ph.D. in Cognitive Psychology and Neuroscience and an M.A. in Cognitive Psychology from the University of Colorado Boulder.

**WALTER WARWICK** is a Principal Systems Analyst at Alion Science and Technology. He is working on several projects having to do with the modeling and simulation of human behavior. He received his Ph.D. in History and Philosophy of Science, an Area Certificate in Pure and Applied Logic, and an M.S. in Computer Science from Indiana University.

## Results and Lessons Learned from the 2009 DSF Model Comparison Challenge

### Panel Chair

Walter Warwick

MA&D Operation, Alion Science and Technology

4949 Pearl East Circle, Suite #200

Boulder, CO 80301

303-442-6947

[wwarwick@alionscience.com](mailto:wwarwick@alionscience.com)

At the 2009 BRIMS conference, we announced a model comparison challenge (Warwick, 2009; Lebiere, Gonzalez & Warwick, 2009). The challenge was based on modeling human performance on the dynamic stocks and flows (DSF), a generic control task that captures many of the complexities of dynamic decision making (Dutt & Gonzalez, 2007; Gonzalez & Dutt, 2007). The DSF was designed to be as simple and accessible as possible to computational modelers while focusing on two key ubiquitous components of general intelligence: the control of dynamical systems and the prediction of future events. A general call for participation was submitted to invite independent developers, of distinct computational approaches, to simulate human performance on the DSF task.

Nine different individuals or teams chose to participate in the challenge by developing computational models to simulate human performance on the DSF task in a variety of conditions. All participants were provided a description of the DSF task and samples of detailed human data that had been collected and reported in previous studies (Dutt & Gonzalez, 2007). In addition, sample software was provided to facilitate a socket-based connection between the models and the DSF simulation environment. The stated goal of the comparison challenge was to reproduce human behavior, including learning, mistakes, and limitations in a way that their models would generalize to *new* conditions of the task undisclosed to the participants. Results from three of the models were selected for presentation at the 2009 International Conference on Cognitive Modeling (Lebiere, Gonzalez, Dutt & Warwick, 2009). In addition, after the challenge was complete, we issued a call for papers for a special issue of the Journal of Artificial General Intelligence devoted to the challenge and its implications for advancing cognitive science and Artificial General Intelligence. The human performance data and the output from each model under every condition are available on the challenge web site:

<<http://www.cmu.edu/ddmlab/modeldsf>>

The goal of this panel discussion is to present our experiences in conducting the DSF comparison challenge

and to reflect on the enterprises of model comparisons and modeling challenges in general. Walter Warwick will begin by discussing the motivation for this challenge and some of the issues faced in organizing it.

Next, we will turn to Varun Dutt of Carnegie Mellon University who will briefly review the DSF task itself, touching on both human performance in the laboratory and how he extended the experimental software to allow participants to link any model to the task environment supporting model comparison. He will also describe some of the challenges we faced, as organizers, in understanding the human performance and drawing meaningful comparisons among models. It became clear only after the fact that traditional measures of fit would not illuminate important performance differences among models on the DSF task.

The third panelist will be Kevin Gluck of the Air Force Research Laboratory. Gluck served in the role of Commentator in the previous panel on the DSF Comparison Challenge at BRIMS 2009. In that role, he recommended systematic exploration of the relative contributions of key mechanisms in all of the models that would be submitted, in order to establish the necessity of those mechanisms for predicting the transfer data. For any of several understandable reasons this did not happen as part of the standard process within the DSF Comparison Challenge. However, Gluck and colleagues at AFRL took on this and more as an independent set of supplementary analyses, exploring the complex interactions among architectural mechanisms, knowledge-level strategy variants, and task conditions. The general point motivating these efforts and to be summarized in Gluck's panel presentation is that the behavioral and cognitive modeling communities may reap greater scientific return on research investments – may achieve an improved understanding of architectures and models – if there is more emphasis on systematic sensitivity and necessity analyses during system development, evaluation, and comparison.

Finally, we will offer the first-hand experience of one of the participants. David Reitter, of Carnegie Mellon

University, submitted a cognitive model to the DSF challenge that generalized to yield the most accurate predictions of unseen data in novel conditions. He will report on the insights gained from his participation and from Gluck's subsequent parameter optimization and comparison with a competing model, pointing out three aspects of desirable progress in model evaluation: 1) generalization through prediction as opposed to post-hoc evaluation; 2) goodness-of-fit measures in numeric spaces other than the direct empirical measures obtained, yet; 3) the undesirable effect of "teaching the test" in competitions in other fields, such as Automatic Document Summarization and Machine Translation.

Although many of the issues we broach will be familiar to members of the BRIMS community, the challenges they present are no less urgent. In particular, this panel will provide a concrete, first-hand context for discussing questions about the representation of human variability in model performance, the need for task-specific quantitative measures of fit, the difficulty in expressing model content, the role of architectures in model development and the challenge in capturing the human cognitive ability to adapt to entirely new experiences. But more important than addressing those specific issues, we hope that the discussion will help us understand as a community what is needed to transition model comparison from an occasional and idiosyncratic exercise to a foundational research enterprise. Indeed, we see organized modeling comparisons and challenges as essential activities for advancing the science of human behavior representation but we cannot realize this vision without widespread community engagement.

## Panelists

**Walter Warwick** (Chair) – Alion Science

**Varun Dutt** – Carnegie Mellon University

**Kevin Gluck** – Air Force Research Laboratory

**David Reitter** – Carnegie Mellon University

## References

Dutt, V., & Gonzalez, C. (2007). Slope of inflow impacts dynamic decision making. In Proceedings of the 25th International Conference of the System Dynamics Society (pp. 79). Boston, MA: System Dynamics Society.

Gonzalez, C., & Dutt, V. (2007). Learning to control a dynamic task: A system dynamics cognitive model of the slope effect. In Lewis, Polk, & Laird (Eds.), 8th International Conference on Cognitive Modeling (pp. 61-66). Ann Arbor, MI.

Lebiere, C., Gonzalez, C., Dutt, V. & Warwick, W. (2009). Increasing Generalization Requirements for Cognitive Models: Comparing Models of Open-ended Behavior in Dynamic Decision-Making. In Proceedings of the 9th International Conference on Cognitive Modeling. Manchester, England.

Lebiere, C., Gonzalez, C., & Warwick, W. (2009). A Comparative Approach to Understanding General Intelligence: Predicting Cognitive Performance in an Open-ended Dynamic Task. In *Proceedings of the Second Artificial General Intelligence Conference (AGI-09)*. Amsterdam-Paris: Atlantis Press.

Warwick, W. (2009) Comparing the Comparisons. In Proceedings of the Behavior Representation in Modeling and Simulation (BRIMS-09). Sundance, Utah.

# Applying the Human Behavior Architecture in the Simulation of a Networked Command, Control and Communication Structure

*Walter Warwick*

MA&D Operation, Alion Science and Technology  
4949 Pearl East Circle, Suite #200  
Boulder, CO 80301  
303-442-6947  
[wwarwick@alionscience.com](mailto:wwarwick@alionscience.com)

Keywords:

Task Network Modeling, Cognitive Modeling, Model Integration

## 1. Background

At the 2008 BRIMS conference, we introduced the Human Behavior Architecture (Warwick et al., 2008). The HBA is the culmination of several efforts to integrate task network and cognitive modeling within a unified development and simulation environment (Lebiere et al, 2002; Lebiere, Archer, Warwick and Schunk, 2005; Lebiere, Best, Archer and Warwick 2005). As we described in 2008, the HBA has been developed to effect a deep integration between two modeling approaches that are often, and mistakenly, regarded as incompatible. In fact, both task network models and production-based cognitive architectures are, essentially, systems for representing transitions between discrete states. The HBA thus supports a unified approach to modeling by representing productions as nodes within a “cognitive sub-network” where the production cycle is driven by the same clock and event queue that controls behavior at the task network-level. In this way, cognitive processes, as represented by a reimplementations of the core functionality of the ACT-R cognitive architecture, can be developed directly within the C3TRACE task network modeling environment.

By 2008 we had verified the function of the ACT-R reimplementations against the tutorial models (see: <http://act-r.psy.cmu.edu/actr6/>) and developed demonstration models to show off the perspicuous relationship between the cognitive and task network components. In the time since, we have been verifying function in more complex models. In particular, we have taken a C3TRACE model that was developed by the Army Research Laboratory to study the flow of communication in a Future Combat System and attempted

a “cognitive retrofit.” This exercise had several goals. First, it provided a new opportunity to verify HBA function under the load of a very complicated, independently developed task network model. The complexity of the retrofitted model far outstripped any of the previous test models we had developed. Second, we wanted to demonstrate how additional cognitive fidelity could make a marked but plausible impact over the predictions made by the unmodified model. Third, we wanted to see for ourselves what it would be like to work within the unified development environment of the HBA. It is one thing to note that the perceived incompatibility of task network and cognitive modeling is an unfounded prejudice, it is quite another to simultaneously and successfully engage both approaches. Finally, we used this exercise to lay the ground work for further integration work we are currently performing under the Army’s Communications-Electronics Research, Development, and Engineering Center THINK Army Technology Objective. This effort will take the integration one level higher, where HBA itself serve as a component to be integrated with social network analysis tools and techniques for assessing team performance.

## 2. Progress to Date and Outstanding Issues

Though we have been nominally successful in meeting all of our goals, this retrofitting exercise has revealed some interesting modeling challenges and has prompted a few modifications to the HBA. First and foremost, the exercise has reminded us how important good debugging tools are. Task network models are, by their very nature, complex while cognitive models can give rise to some very subtle emergent effect. Verifying the behavior that results from potentially emergent effects in a complex

model is very difficult and it is nearly impossible once stochastic variability is added to a model. This has led us to modify the HBA to allow more selectable switching of stochastic effects at the task network level and to identify specific output reports that can be used to isolate the effects of the cognitive model within the HBA.

A second, less obvious modification followed from the realization that the inherent parallelism of a task network model leads to a more distributed representation of the modeled human. This makes it harder to specify a single “interface” between the cognitive model of the human and the tasks that the modeled human is performing. The challenge became clear as we tried to develop a cognitive model of message handling. Although the C3TRACE model explicitly represented the different tasks an FCS-enabled operator would perform upon receiving a message, there was no single point in the model where we could “sniff” all off the message traffic destined for that particular operator. This forced us to implement a fairly complicated queuing structure so that we could continually sample and buffer messages flowing in parallel so that they might be processed serially by the cognitive model. Although we have since modified C3TRACE to support an event-driven polling of messages, thereby eliminating the need for the message queuing, this modification does not reduce the inherent tension that exists when reconciling the parallel representation of task activity with the serial execution of a cognitive model.

Finally, as we look toward our ongoing work to integrate the HBA with social network analysis tools and techniques for assessing team performance we confront questions about the usual semantics within an HBA model. The original motivation of the HBA was to support a “cognitive level” of decomposition within a task network model so that we might make better predictions about task times and decision making. In the context social network analysis and team performance, however, the nodes of the network often represent individual actors, rather than the specific task an actor performs. Similarly, the edges in the graph of a social network can represent any number of relationships between nodes, rather than just specifying the flow of control among tasks. Although a task network might bear an obvious resemblance to the graph, serious ambiguities often lurk behind the familiar. As part of our THINK ATO work we have begun to identify specific points of contact between the analysis of a social network or team performance and the predictions that can be made using a task network model.

### 3. An Opportunity to Engage the BRIMS community

Rather than present results or specific recommendations by way of a formal paper presentation, our intent is to display the basic capabilities of the HBA and to discuss some of the foregoing issues and future work with BRIMS attendees. We hope that this dialogue will help us meet some of challenges while simultaneously making practitioners aware of the new possibilities that HBA affords.

### 4. References

Archer, R. D., Lebiere, C., Warwick, W. (2005). Design and evaluation of interfaces using the GRaph-Based Interface Language (GRBIL) tool. *Human Systems Integration Symposium*, June 20-22, 2005, Arlington VA

Lebiere, C., Archer, R., Warwick, W., & Schunk, D. (2005) Integrating modeling and simulation into a general-purpose tool. In *Proceedings of the 11<sup>th</sup> International Conference on Human-Computer Interaction*. July 22-27, 2005. Las Vegas, NV

Lebiere, C., Best, B. J., Archer, R., & Warwick, W. (2005). Integrating task network models and cognitive architectures in dynamic, information-rich environments. *Human Systems Integration Symposium*, June 20-22, 2005, Arlington VA

Lebiere, C.; Biefeld, E.; Archer, R.; Archer, S.; Allender, L.; Kelley, T.D. (2002). IMPRINT/ACT-R: Integration of a task network modeling architecture with a cognitive architecture and its application to human error modeling. In *Proceedings of the 2002 Advanced Simulation Technologies Conference*, San Diego, CA, Simulation Series 34 (3), 13-18.

Warwick, W., Archer, R., Hamilton, A. et al. (2008). *Integrating Architectures: Dovetailing Task Network and Cognitive Models*. Proceedings for the Seventeenth Conference on Behavior Representation and Simulation. Providence, RI SISO.

**WALTER WARWICK** is a Principal Systems Analyst at Alion Science and Technology. He has worked on several projects having to do with the modeling and simulation of human behavior. He received his Ph.D. in History and Philosophy of Science, an Area Certificate in Pure and Applied Logic, and an M.S. in Computer Science from Indiana University

# An Agent-Based Model of Conflict in East Africa And the Effect of Watering Holes

William G. Kennedy

Atesmachew B. Hailegiorgis

Mark Rouleau,

Jeffrey K. Bassett

Mark Coletti

Gabriel C. Balan

Tim Gulden

Krasnow Institute for Advanced Study

George Mason University

4440 University Avenue

Fairfax, VA 22030

703-993-9291

wkennedy@gmu.edu, ahailegi@gmu.edu, mdroulea@mtu.edu, jbassett@cs.gmu.edu, mcoletti@gmu.edu,  
gbalan@cs.gmu.edu, tgulden@gmu.edu

Keywords:

agent-based modeling, MASON, conflict modeling

**Abstract:** *An agent-based model conflict between herders in east Africa using the MASON agent-based simulation environment is presented. Herders struggle to keep their herds fed and watered in a GIS-based, spatially diverse environment with data-driven seasonal cycles. The model produces realistic carrying capacity dynamics and basically plausible conflict dynamics. With the rather basic set of behaviors, herders come into conflict over limited resources and one clan is eventually eliminated. We find that greater environmental scarcity leads to faster domination by a single group. At the same time, we note that there is tremendous variability from run to run in the rate and timing of the transition from a conflict-prone, multi-clan environment to hegemony of a single group.*

## 1. Introduction

The Mandera Triangle of East Africa is a complex environmental and human social area. Our research uses Agent-Based Modeling (ABM) to gain a better understanding of herder behavior in response to the environmental stresses and the introduction of new actors (i.e. farmers), the feedback from these actors through the natural environment (i.e., land-use practices), and the resulting sources of tension and conflict. Our multidisciplinary research team brings together knowledge from cognitive science, ethnography, political science, geography, and computer science to produce a model of conflict inspired by Mandera. The model's natural environment is constructed using data from Geographic Information Systems, including information on ground cover, resource variance, weather patterns, and hydrology (Keya 1998; Lenhart & Casimir 2001; Little, McPeak, Barrett, & Kristjanson 2008; MacOpiyo et al 2006; Parker 2001; Weinstein et al 1983). Agent decision-making within the model's social environment is supported by ethnographic research of social customs (Axtell et al 2002; Bah et al 2006; Johnson & Anderson 1988; Johnson 1983; Marshall 1990; Oba 2001), mechanisms for alliance formation and conflict resolution (Ellis & Swift 1988; Ensminger & Rutten

1991), and regional studies of conflict mediation conducted by both political scientists and policy makers (Bouh & Mammo 2008; Brockhaus 2003; Kuznar & Sedlmeyer 2005; Mace et al 1993; Mahmoud 2008; Saqalli 2008; Scoones & Graham 1994). The resulting model highlights the current socio-natural flashpoints in Mandera and provides the opportunity to experiment with future "what if" scenarios shaping the behavior of herders in response to land-use decisions.

This paper describes one of a series of experiments: the impact of changing one environmental variable, the number of watering holes. Water is a vital resource in the subject region and building wells may be one way to improve the areas carrying capacity and reduce conflict. The research question is whether adding wells improves conditions. For this work, we define improving conditions in terms of increased carrying capacity and reduced incidents of conflicts.

## 2. Background

The Mandera Triangle – an area of East Africa encompassing a roughly triangular area bordering Somalia, Kenya, and Ethiopia (see Figure 1)– has served as the traditional home for several well-established nomadic herding groups. This zone and its

populace were once coupled in a self-regulated socio-natural system developed over countless generations as a response to their sparse and seasonally changing environment. The herders of Mandera have constructed an elaborate social alliance structure to cope with various environmental shocks such as drought or flooding. Herders in today's Mandera face more socio-natural complexity in their lives due to the advancement of government supported private landowners (i.e. farmers). Without sufficient time or resources (i.e. the low carrying capacity of the land) to evolve, this new socio-natural system has become highly conflict ridden.



Figure 1. Area of East Africa Modeled

Pastoralism in Mandera was largely an adaptive response to both short- and long-term environmental cues. In the short-term, pastoralism offered the greatest return on effort in a semi-arid region that was not especially hospitable to agriculture. In the long-term, a mixture of agro-pastoralism, primarily dominated by herding, proved a flexible option for survival in a rather unpredictable and, at times, lean environment. Thus, societal evolution led the pastoralists of the Mandera Triangle to weave themselves into the fabric of the surrounding natural environment with its particular ebbs and flows (Smith 1984 and Smith 1992).

From this perspective it is possible to identify environmental constraints on survival, such as floods or droughts restricting access to grazing land, as potential triggers for conflict within these pastoralist groups. Consequently, institutional structures evolved to manage and accommodate these restrictions. One critical institutional development was the introduction of a customary system of shared resource access (Torry 1976 and Johnson 1988). This quasi-formal agreement among Mandera's pastoral groups permitted herders to mutually graze lands while traveling through one another's zone of influence or in times of desperation. Without this arrangement, pastoral life in Mandera would have been much more difficult if not impossible

to sustain for all but a handful of groups (Mace 1993).

The sparse and seasonally changing landscape of this region meant that intrusion onto another's land was likely to occur in transit but particularly when marginal land faced adversity. Thus, mutual access agreements were implemented under the condition that common customs were respected – such as the grazing of cattle in the highlands and camel in the lowlands – and such rights were not abused. Although these agreements did not eliminate conflict among pastoralists, they did provide an authoritative framework for conflict resolution that centered upon a common understanding of socio-natural interactions (Torry 1976 and Wario 2006). When inter-herder conflict did occur, it typically took the form of a symbolic gesture of economic redistribution rather than an attempt to annihilate the other party (Torry 1976). This is how Mandera came to cope with its complex socio-natural environment for hundreds, if not thousands, of years. However, in the past number of decades, this picture has begun to change and, with it, the nature of conflict, as those in Mandera have traditionally known it.

The situation in the Mandera Triangle provides a unique opportunity to examine the behavioral roots of conflict. Given that conflict was historically “well-regulated” prior to the introduction of states, it is reasonable to speculate that the entrance of new actors, in the form of landowning farmers, has had a significant impact on the nature of conflict. The case of Mandera is a good example of the impact of institutional collision leading to the upset of a longstanding symbiotic socio-natural relationship. Moreover, it is possible to sift out behavioral drivers from these changed circumstances by observing differences between the new herder-farmer interactions and the traditional behavior of pastoralists attempting to meet the age-old demands of the natural environment. Our study seeks a better understanding of this change, its influence on herder behavior, the impact on the socio-natural system, and the complex feedback driving a new form of conflict in Mandera.

## 2. Model Description

Our agent-based model (ABM) simulates interactions and conflict between herders with different ethnic identities and herders and farmers over the use of land resources. The model mainly focuses on the tension between different herder groups over the utilization of the common grazing land and water resources and the emergence of conflict related to their use.

The model is developed within the MASON simulation environment (Luke et al. 2005). MASON is a multi-purpose simulation library for the Java programming language. The system provides the necessary modeling

tools, such as agent scheduling and visualization, for the development of customized ABM simulations. As is typical for ABM simulations, MASON models are dependent upon the implementation of three critical components: agents, the environment, and the rules of interaction. We model the environment based on 1km by 1km land parcels, each time step represents one day, and each agent represents a family unit.

The model consists of two kinds of agents, herders and farmers (Figure 2). Because herders are in the focus of this model, their behavior is represented in significantly greater detail. Each herder is represented as a single agent with combined characteristics of the herder, herder's family, and the herd animals. Two groups of herder agents who are ethnically different are represented. Herders' relation with their ethnic group allows them to share scarce resources in time of need and to cooperate in time of conflict.

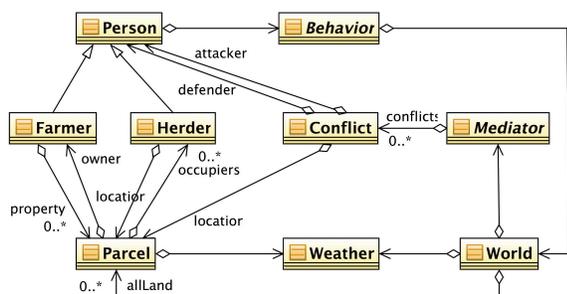


Figure 2: UML Diagram of Herders and Farmers

Herders are entirely dependent on their herds and manage their herds in each time step. They make decisions considering their movement depending on the herd's level of hunger, thirst, the distance to the current water source, and the quality of grazing nearby. Herders evaluate visible parcels' pasture and water ability to satisfy the needs of their herds. At any given time, each herder has a base camp near a water source. The herd must return to that water source to drink as its metabolism and the herd's movement priorities dictate. The herd continues to graze and water in the vicinity of this base until its needs for either food or water are no longer met. When the herd runs short of either food or water, the herder shifts the base camp to a nearby water source.

Herders share the common resource if they belong to the same ethnic group and compete with other herders or farmers if they are different. Herders minimize conflict by preferring to move to unoccupied parcels when they can. However, this is not always possible since the resource is limited. In such circumstance, they engage in conflict. The conflict can escalate by involving other herders within their ethnic group who share the burden through cooperation to increase their rate of survival.

The herders' knowledge to their environment depends on their vision, i.e., the range over which they can consider moving in a single day. Vision range, in km, can affect their success in surviving the environmental challenges. The availability of pasture and water determines the level of herd reproduction. If the environment is harsh, herds will be stressed by starvation or dehydration. Starving herds don't reproduce, nor do critically dehydrated ones. If they surpass the stress threshold, they will eventually die.

When a herder agent survives and grows and the herd reaches a specified size, the herder and herd split in to two and a new herder family is introduced. The movement decision characteristics of the newly formed herder agent depend on parameters values of its parent with some noise introduced.

To avoid overcomplicating our model from the outset, we have left the farmer agent as a simple, passive owner of territory. Farmer agents essentially occupy viable grazing land and increase the fertility of these parcels through their efforts. In this model, we assume that farmers are engaged in sedentary subsistence agricultural production and can produce enough food to meet the need of their family from their parcel on land. What is important to this behavior is that farmers occupy parcels with a high agricultural fertility and, once occupied, farmers have a stake in defending these high-demand parcels from herder intrusions and can cause damage to herders. However, in this model, farmers will stay unaffected by any incident or conflict and their property will be inherited to the next generation with out any transformation or damage.

The environment has a spatial extent of 150 km by 150 km, and is comprised of parcels, weather and water holes. The parcel is the central feature of the environment, serving to consolidate the interactions between agricultural fertility, vegetation production, waterhole location, population density, and ownership. We model the environment with three components: land, which is divided into a regular grid of 1 km by 1 km parcels, waterholes, and weather. Land parcels are of differing quality, which is represented by differing maximum amounts of vegetation they can support in the absence of grazing and under optimal weather conditions. We estimate this maximum vegetation level using GIS data on land use and slope. Parcels grow vegetation based on the parcel's maximum level of vegetation, its current level of vegetation, and the current rainfall. A minimum amount of rainfall is required to maintain the current level of vegetation – below which the growth rate is negative and the grass dies off even without grazing. Farmed parcels are capable of producing a maximum level of vegetation that is twice what it would be in the absence of a farmer.

We represent weather over the entire region as a single variable amount of daily rainfall in millimeters using monthly averages for the study area. This rainfall information drives vegetation growth and re-filling of watering holes. Model runs start in January and use the same rainfall values each year. In addition to data driven monthly rainfall, we can change rainfall to address droughts by using a drought parameter. Waterholes are located in randomly assigned parcels. A waterhole can be exhausted with high herd consumption and refilled again based on rainfall.

The main simulation loop consists of herder agents adapting to the seasonally driven changes in the grazing environment. Seasonal changes in weather, in the form of the amount of rainfall, determine the current state of any given parcel according to that parcel's maximum fertility. Each time step is equivalent to a day and the herder agent's utilized of its current parcel is pegged to this time increment. As the environment permits, herder agents avoid other herders and farmers to move from parcel to parcel to obtain vegetation and water to maintain their health. Parcel regrowth occurs but at a much slower rate than the herders' grazing reaps from them. This has the potential to drive herders onto farmer land during times of crisis. For example, if a herder agent's health reaches the desperate stage due to the lack of viable graze land or water, herder agents will then seek the nearest parcel with available resources regardless of the presence of another agent. It is these trespassing events that are considered conflict and the results of all the conflicts are determined at the end of each day.

At each time step (i.e. day), we update the vegetation on each parcel (vegetation regenerates as a function of current level of grazing and rainfall); we activate each herder (in random order); then finally, we resolve conflicts. As previously stated, we update the weather monthly, specifically every 30.4375 days. Droughts can be programmed to occur in any of the years with a fifteen-year cycle. This process is then repeated, resulting in herd movements, resulting in conflict dynamics. Other processes will be activated under certain circumstances. For instance, splitting of herds and formation of new herder family depend on the success of the herder to accumulate a specified herd size. Deaths of animals within herds results from thirst and hunger and when all the animals have died, the herder agent is removed.

Conflict is analyzed by checking herder movement and detecting of occurrence of trespassing incident. We consider an incident as a combat (or opportunity for combat) between a herder and either another herder or a farmer. Conflict is modeled as two agents in the same parcel at the end of the movement part of a time step. Incident(s) can grow over time and potentially involve

multiple herders and farmers. Consequences of an incident depend on participants. When it is between two herders of the same clan, the incident is resolved peacefully by averaging hunger and thirst values between both herders helping one and hurting other. When the conflict is between herders of different clans, the defender's herd size is reduced by damage ratio (a parameter) while the attacker's herd is increased by those animals. In the mean time, the attacker's hunger is also reduced based on the captured resources. However both the attacker and defender thirst is not changed. In farmer and herder situation, farmer is unaffected by conflict and only herder's herd size is reduced by a damage ratio percentage.

Escalation of conflict occurs only between herders and farmers when the incident persists over a specified number of steps. As all herders track their last combatant, and the duration (number of steps) that the most recent combat has persisted uninterrupted, which is when (if) the duration reaches a specified number of steps, escalation of conflict is initiated. Consequently all allied herders within a specified range are identified. The resources (hunger and thirst) of all allied herders are averaged.

In the current design of our model, only a single previous combat/combatant is remembered. This works well with herder-farmer conflicts since a herder can never fight more than one farmer at a time. If we model herder-herder escalation, we will need to consider that a herder can fight several other herders in a time step. Similarly, if we model farmer sharing of resources, we will need to consider that a farmer can fight several herders in a time step. However, at this stage of our model, we prefer to consider very simple behavior.

### 3. Experiment Description

Our model is runs and provides us the ability to experiment with different parameters to see the fidelity of the model in relation to real world phenomena. To start simple, we have limited our experiments to the relationship between the number of watering holes, the total population, and the level of dominance of one ethnic group. For this experiment, we omitted farmers. We did vary one parameter, namely the number of watering holes, in six steps between 50 and 300. For each number of watering holes, we conducted five 100 year-long runs.

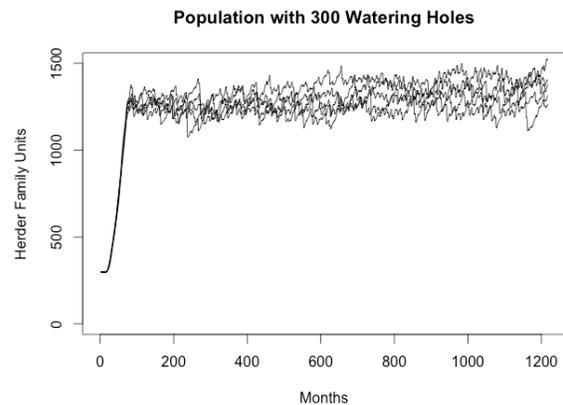
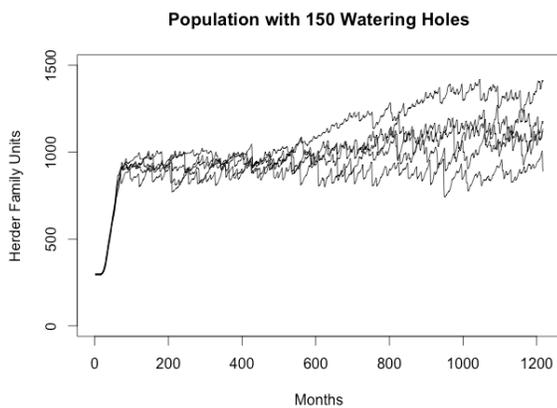
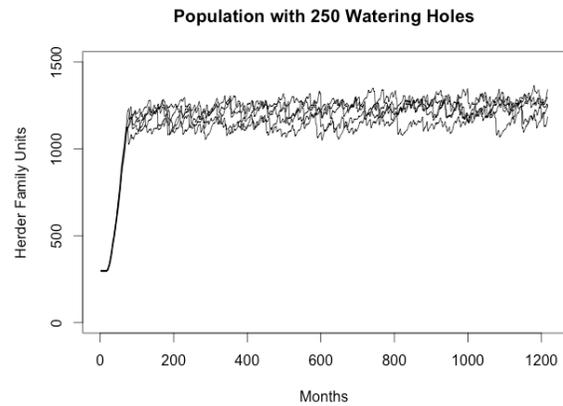
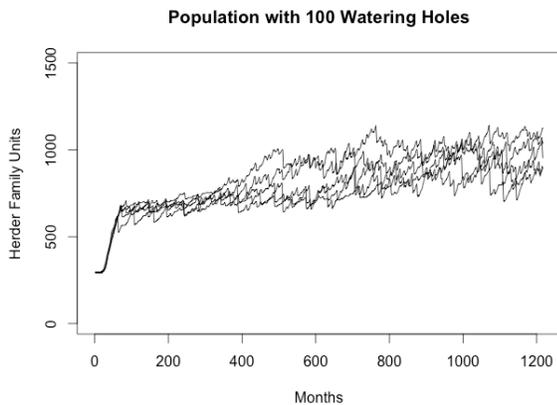
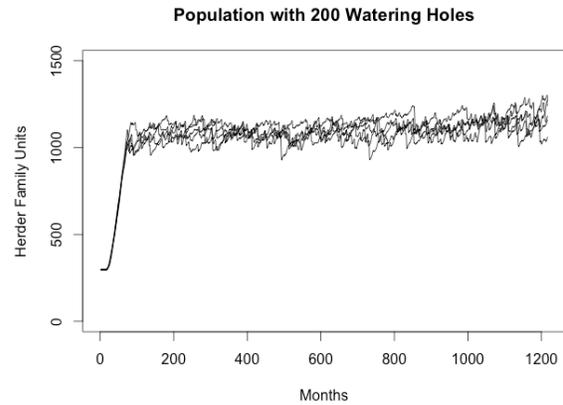
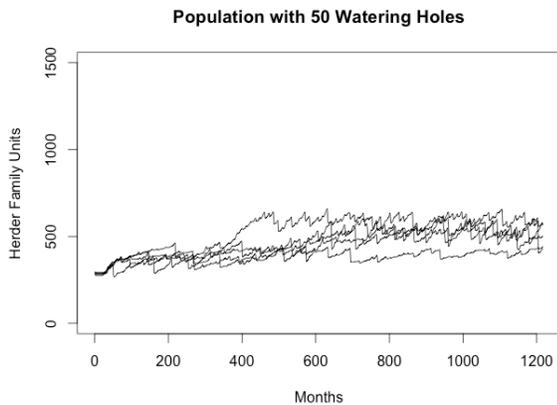
We started each run with 300 herders randomly assigned to one of the two tribes. Visibility was set at 10 km. This set the maximum distance from the current location that was considered at each step. Waterholes were placed randomly in each run, with the probability of their placement in a given parcel proportional to the fertility of that parcel

## 4. Experimental Results

### 4.1 Watering Holes and Carrying Capacity

Starting with 300 herder family units, the number of herders grows steadily for about the first 5 years (60 months) as the population reaches the environmental carrying capacity as seen in Figure 3a through e. Increasing the number of watering holes increases the carrying capacity, though not in a linear manner.

While in the lower ranges (between 50 and 100, for Population with Watering Holes example) the increase is nearly proportional (from around 400 to over 700), the proportionality breaks down with higher numbers of water holes. In going from 150 to 300 water holes, the initial carrying capacity increases from 900 to only around 1,300 – an increase of only about 50% as opposed to the 100% increase in watering holes. This fall off in the rate of increase in carrying capacity is because water is not the only limiting resource. When



Figures 3a, b, and c. Population with Watering Holes

Figures 3d, e, f. Population with Watering Holes

the number of watering holes is small, each new watering hole opens up grazing land that was previously too far from water to be useful. As more watering holes are added, however, their areas of influence begin to overlap and grazing land starts to become an additional limiting factor.

### 4.2 Ethnic Hegemony

Figure 4 and 5 compare two representative runs from the case with 100 watering holes (cf. Figure 3b). Both of these runs show four distinct phases: 1) a short period of initial growth toward carrying capacity characterized

by low levels of conflict and little change in ethnic composition, 2) a period of coexistence and competition for resources where the ethnic balance is relatively stable, 3) a period of relatively rapid and essentially monotonic increase in one clan at the expense of the other, and 4) a period of complete hegemony once the dominant clan has eliminated the competition.

The monotonic nature of the transition here is striking, as is the variability in its timing. Once one clan gains the upper hand, it almost always wins out. Though it may suffer setbacks lasting a few years, the progression

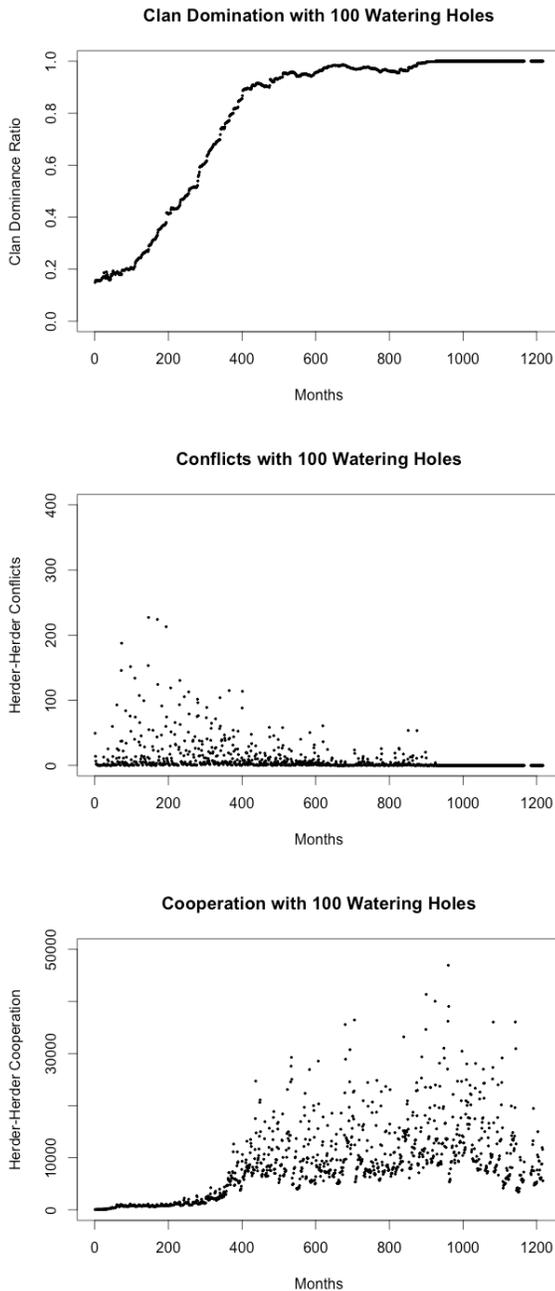


Figure 4. Run C with 100 Watering Holes

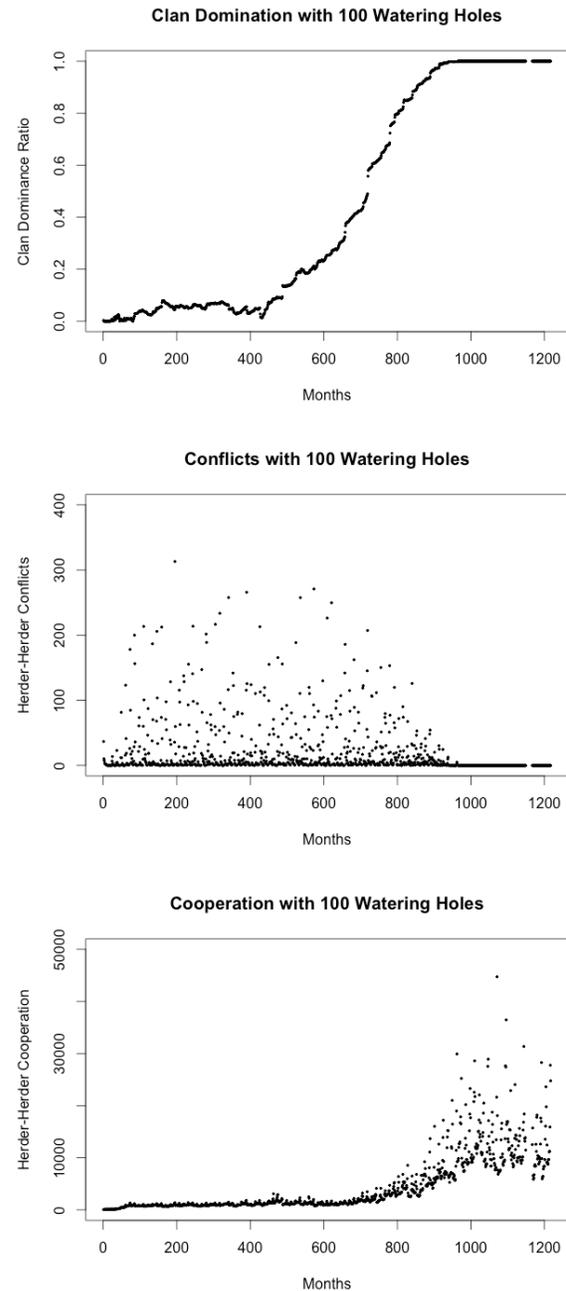


Figure 5. Run D with 100 Watering Holes

to dominance by the larger group is almost never reversed once the ratio goes beyond a tipping point. The timing of the transition is much less certain. In runs differing only in their random seed (resulting in slight differences in initial population ratio and major differences in the placement of watering holes), the transition may begin almost immediately and be essentially complete by month 400, or may not begin until approximately month 400 and not be complete until nearly the end of the 100 year simulation.

## 5. Conclusions

Although the large number and types of agents and phenomena included complicate our model, increasing the number of watering holes increases the population, as expected. However, considering only the total population plots misses the fact that there is competition between the two modeled clans. With the stress of fewer watering holes, one clan comes to dominate earlier than when there are more watering holes. Along the way to this hegemony, conflict between clans continues until one clan is eliminated. After total hegemony, inter-clan conflict ceases (by definition) but cooperation between members of the same clan increases dramatically

We can also draw conclusions concerning the behavior representation in modeling and simulation. In our work, the data-driven modeling of behavior has shown that environmental resources can result in disproportionately large variations in the frequency of conflict and cooperation.

Even the simple rules described here result in interesting macro-level behavior. We therefore find that this agent-based modeling framework is a rich approach for exploring the various complexities resulting from the interaction of purposive individuals in a spatially and temporally diverse natural environment. As a result, we believe agent-based modeling is the most effective modeling approach for the study of potentially chaotic systems.

### *Acknowledgments*

This work is supported by the Center for Social Complexity of George Mason University and by the Office of Naval Research (ONR) under a Multi-disciplinary University Research Initiative (MURI) grant no. N00014-08-1-0921. The authors would like to acknowledge input from Claudio Coffi-Revilla and the Mason-HRAF Joint Project on Eastern Africa (MURI Team). The opinions, findings, and conclusions or recommendations expressed in this work are those of the authors and do not necessarily reflect the views of the sponsors.

## References

- Axtell, Robert et al. (2002) "Population Growth and Collapse in a Multiagent Model of the Kayenta Anasazi in Long House Valley," *Proceedings of the National Academy of Sciences*, 99(3).
- Bah, A. et al. (2006) "An Agent-Based Model to Understand the Multiple Uses of Land and Resources around Drillings in Sahel," *Mathematical and Computer Modeling*, Vol. 44, pp. 513-534.
- Bouh, Ahmed M. and Mammo, Yared (2008) "Indigenous Conflict Management and Resolution Mechanisms on Rangelands in Somalia Regional State, Ethiopia," *Nomadic Peoples*, 12(1), pp. 109-121.
- Brockhaus, Maria et al. (2003) "The Role of the State in the Management of Farmer-Herder Conflicts in South West Burkina Faso," *Deutscher Tropentag*, University of Goettingen.
- Ellis, James E. and Swift, David M. (1988) "Stability of African Pastoralist Ecosystems: Alternate Paradigms and Implications for Development," *Journal of Range Management*, 41(6), pp. 450-459.
- Ensminger, Jean and Rutten, Andrew (1991) "The Political Economy of Changing Property Rights: Dismantling a Pastoral Commons," *American Ethnologist*, 18(4), pp. 683-699.
- Johnson, Douglas H. and Anderson, David M. (1988) *The Ecology of Survival: Case Studies from Northeast African History*, I.B. Tauris.
- Johnson, Gregory A. (1983) "Decision-Making Organization and Nomad Camp Size," *Human Ecology*, 11(2), pp. 175-199.
- Luke S, Cioffi-Revilla C, Sullivan K, Balan G .2005. MASON: A Multiagent Simulation Environment. *Simulation*, 81(7), pp. 517-527.
- Keya, George A., (1998) "Herbaceous layer production and utilization by herbivores under different ecological conditions in an arid savanna of Kenya". *Agriculture, Ecosystems and Environment* 69 pp. 55-67.
- Kuznar, Lawrence A. and Sedlmeyer, Robert (2005) "Collective Violence in Darfur: An Agent-Based Model of Pastoral Nomad/Sedentary Peasant Interaction," *Mathematical Anthropology and Cultural Theory*, 1(4).
- Lenhart, Lioba and Casimir, Michael J. (2001) "Environment, Property Resources and the State: An Introduction," *Nomadic Peoples*, 5(2), pp. 6-20.
- Little, Peter D., McPeak, John, Barrett, Christopher B., and Kristjanson, Patti, (2008) "Challenging Orthodoxies: Understanding Poverty in Pastoral Areas of East Africa". *Development and Change*, 39(4) pp. 587-611.

- Mace, Ruth (1993) "Nomadic Pastoralists Adopt Subsistence Strategies that Maximize Long-Term Household Survival," *Behavioral Ecology and Sociobiology*, 33(5), pp. 329-334.
- Mace, Ruth et al. (1993) "Transitions between Cultivation and Pastoralism in Sub-Saharan Africa," *Current Anthropology*, 34(4), pp. 363-382.
- MacOpiyo, Laban et al. (2006) "Use of GIS and Agent-Based Modeling to Simulate Pastoralist Mobility in the Rangelands of East Africa," *GSDI-9 Conference Proceedings*.
- Mahmoud, Hussein A, (2008) "Seeking Citizenship on the Border: Kenya Somalis, the Uncertainty of Belonging, and Public Sphere Interactions". Council for the Development of Social Science Research in Africa (CODESRIA), 12th General Assembly, Yaounde, Cameroun, December 7-11, 2008.
- Marshall, Fiona (1990) "Origins of Specialized Pastoralist Production in East Africa" *American Anthropologist*, 92(4), pp. 873-894.
- Oba, Gufu (2001) "The Importance of Pastoralists' Indigenous Coping Strategies for Planning Drought Management in the Arid Zone of Kenya," *Nomadic Peoples*, 5(1), pp. 89-119.
- Parker, Dawn C. (2001) "Agent-Based Models of Land-Use and Land-Cover Change," *Report and Review of an International Workshop*, LUCC Report Series No. 6.
- Saqalli, Mehdi (2008) "Populations, Farming Systems and Social Transitions in Sahelian Niger: An Agent-Based Modeling Approach," *PhD Thesis*, Université Catholique de Louvain.
- Scoones, Ian and Graham, Olivia (1994) "New Directions for Pastoral Development in Africa". *Development in Practice*, 4(3), pp. 188-198.
- Smith, Andrew B. (1984) "Environmental Limitations on Prehistoric Pastoralism in Africa" *The African Archaeological Review*, Vol. 2, pp. 99-111.
- Smith, Andrew B. (1992) "Origins and Spread of Pastoralism in Africa" *Annual Review of Anthropology*, Vol. 21, pp. 125-141.
- Torry, William I. (1976) "Residence Rules Among the Gabra Nomads: Some Ecological Considerations," *Ethnology*, 15(3), pp. 269-285.
- Wario, Hussein T. (2006) "Historical and Current Perspectives on Inter-Ethnic Conflicts in Northern Kenya" *Masters Thesis*, University of Life Sciences, Norway.
- Weinstein, D. A. et al. (1983) "Energy Flow and the Persistence of a Human Population: A Simulation Analysis," *Human Ecology*, 11(2), pp. 201-225.

## Author Biographies

**WILLIAM G. KENNEDY** is a Research Assistant Professor in the Department of Computational Social Science at George Mason University. His research interests are in cognitive modeling, "like-me" theory of mind, and integrating cognitive modeling into social simulations. Previously he was with the Naval Research Laboratory working in Cognitive Science and cognitive robotics.

**ATESMACHEW B. HAILEGIORGIS** is a PhD student in the Department of Computational Social Science at George Mason University studying complex social systems in east Africa.

**MARK ROULEAU** is a PhD candidate in the Department of Computational Social Science at George Mason University. Mark has been the lead programmer on numerous research projects at the University of Delaware, George Mason University, and the US Department of Agriculture. Projects include: modeling climate change negotiations, optimizing automated voter redistricting, assessing the feasibility of water quality-trading markets, and the above-mentioned RebeLand modeling, the development of civil unrest.

**JEFFREY K. BASSETT** is a PhD candidate in Computer Science, Volgenau School of Information Technology and Engineering, George Mason University. Most of his research to date has focused on using Evolutionary Computation as a learning technique, with particular emphasis on using rule-sets to represent behaviors in teams of robots.

**MARK COLETTI** is a PhD candidate in Computer Science, Volgenau School of Information Technology and Engineering, George Mason University. Most of his research to date has focused on graphical information systems (GIS).

**GABRIEL C. BALAN** is a new PhD (January 2010) in Computer Science, graduating from the Volgenau School of Information Technology and Engineering, George Mason University.

**TIM GULDEN** is a Research Assistant Professor with in the Department of Computational Social Science at George Mason University. His work applies complex systems theory and agent-based modeling to policy-relevant subjects such as counter-insurgency, international trade, and the role of urban systems in globalization

# Extracting the Ontological Structure of OpenCyc for Reuse and Portability of Cognitive Models

*Bradley J. Best*

*Nathan Gerhart*

Adaptive Cognitive Systems

1942 Broadway Street

Boulder, CO 80302

303-413-3472

{ bjb, ngerhart } @adcogsys.com

*Christian Lebiere*

Carnegie Mellon University

5000 Forbes Ave

Pittsburgh, PA 15213

[cl@cmu.edu](mailto:cl@cmu.edu)

Keywords:

OpenCyc, Ontology, Model Portability

**ABSTRACT:** *Large scale general-purpose knowledge ontologies, such as OpenCyc, have been suggested as a means of increasing the portability and reuse of cognitive models through a mapping onto domain-independent language. Previous efforts have revealed that this mapping process is difficult to perform due to several factors including the difficulty of understanding the underlying structure of the ontology and mismatches in representation between the target cognitive modeling architecture and the source ontology. We present a method of extracting, pruning, and visualizing the structure of OpenCyc localized around a given set of related terms and explore a set of examples targeted at the representational assumptions of the ACT-R cognitive architecture. Furthermore, we discuss the implications of both a quick-and-easy mapping method and a more robust methodology. The work described, though in its early stages, provides assistance in both rapid understanding of the OpenCyc structure and the process of mapping domain-dependent terms to a general ontology.*

## 1. Introduction

A central issue in developing a general-purpose layer between simulation environments and cognitive architectures is the representation to be used and its implications for further architectural processing. To attain generality with respect to the simulation environment, commitment to a common, general representation framework is necessary. An additional advantage of this approach is that it should foster on the cognitive architecture side much greater reuse of models than is currently the case. Even for closely related situations, models are usually not reused but instead re-engineered completely to accommodate a different environment. One potential source for such a representational commitment are general ontologies, such as Cyc, that have attracted much investment in recent decades. However, ontologies are fundamentally logic-based formalisms that might not be consistent with the representational, computational, architectural and behavioral commitments made by existing cognitive architectures.

To avoid having the ontological tail wag the architectural dog, it is essential to design a mapping from ontology to representation that is consistent with architectural practice and that leverages the key mechanisms of the target architectures. Ball, Rogers, and Gluck (2004) suggested that the creation of such a layer – the integration of cognitive architectures with general ontologies such as OpenCyc – might provide a remedy to some of the issues involved in cognitive modeling, but they did not go as far as actually implementing such a layer. Best and Lebiere (2009) described a series of issues in integrating intelligent agents into virtual environments and a corresponding set of solutions, some realized, some proposed, that related directly increasing the range and portability of cognitive models, and similarly proposed the integration of large-scale general knowledge ontologies, and OpenCyc in particular, as a means for addressing this issue. This paper describes the current state of our continuing research on this topic, including a functioning implementation of a mapping layer that will be explained in the context of multiple examples. For

specificity's sake, we will focus on the mapping to the ACT-R cognitive architecture (Anderson & Lebiere, 1998; Anderson et al. 2004), but our approach is general enough to apply to related architectures, especially production systems and other symbolic architectures featuring structured representations.

## 2. OpenCyc

OpenCyc is the open-source version of the Cyc general knowledge base, a large-scale ontology containing both broad general knowledge (e.g., facts relating objects like chairs to their purpose as seating furniture) and specific facts tied to domains (e.g., facts relating specific Army terrain mapping types to the cover and concealment they provide). OpenCyc, created by Cycorp, is written in a proprietary Lisp-like language and includes JAVA and ASCII APIs, as well as a command-line and web-based interface. For more details about Cyc/OpenCyc, see Matuszek et al (2006) and the OpenCyc homepage ([www.OpenCyc.org](http://www.OpenCyc.org)).

## 3. Extracting Information from OpenCyc

Ontologies are primarily constituted of three types of information: 1) basic terms and their types including hierarchical organization, 2) relations between these terms, and 3) inference rules applying to these terms and relations. Mapping terms and types into ACT-R chunks and their types is reasonably straightforward, but the issue of multiple inheritance across types is much more complex because cognitive architectures typically do not support this mechanism, often limiting themselves, as in the case of ACT-R, to the simpler single inheritance mechanism, for reasons both practical such as efficiency of implementation and theoretical such as cognitive plausibility (e.g., limits on the size of a unit of representation). Basic options to address this issue within the context of the ACT-R architecture include:

- Leveraging the simpler, single inheritance architectural mechanism and treating multiple inheritance in a separate way
- Leveraging other architectural mechanisms such as subsymbolic partial matching and activation spreading mechanisms
- Representing the terms and their types explicitly and requiring that the architecture perform type inferences in an interpretive rather than automatic way

These approaches are potentially complementary, but their implications for processing are fundamental. For instance, the simpler, more explicit and modular representation schemes also impose the most demanding processing requirements upon the architecture. Our research approach is to be strongly guided by behavioral and neural knowledge of representation to derive a robust and effective compromise between these options.

Relations between terms are potentially straightforward to represent but inferences are not. Like terms, there is a natural trade-off between the complexity of the representation and the efficiency of the architectural processes that can apply to it. One possibility is to focus on purely representational issues and consider knowledge-based inferences to be beyond the scope of an interface between environments and architecture. That is often the approach taken in modeling where knowledge and control are tightly intertwined and optimized to the task at hand, but the generality of the representation commitment in this case imposes additional constraints on the necessity to be able to reason upon the knowledge in order to compensate for the lack of hardwired control.

The approach we have taken has 3 main steps, 1) determining an appropriate mapping, 2) pruning an extracted hierarchy, and 3) visualizing the results, each of which are described in detail below. All examples use domain-specific terms from the dTank virtual environment (Morgan et al 2005).

### 3.1 Determining Appropriate Term Mapping

For any domain, the first (and potentially the most difficult) step is to determine an appropriate mapping from domain-specific terms to the general OpenCyc vocabulary. In section 4, we discuss the implications of two ends of the mapping spectrum: a simple lookup vs. an in-depth exploration of the OpenCyc structure and implied meaning.

Cycorp provides a web-based browser (the KB browser) for exploring and manipulating OpenCyc. Using the KB browser, one can find close matches based on English "pretty strings" (e.g., a search for "tank" returns links to the OpenCyc constants Tank-Vehicle and LiquidStorageTank). Stopping at this result is what we refer to as the simple lookup. Note that the simple lookup mapping procedure uses the domain-specific name as the most important (i.e., the only) criteria.

To perform a more accurate search, one would use the simple lookup as a starting point and dig for more specific constants. It is important to mention here that the full meaning of an OpenCyc term is best understood as a combination of 1) the name, 2) the related (more general/specific) terms, and 3) the "comment" tag associated with the term. For a walk-through of the general search procedure, see the tutorial (Cycorp 2002).

However, a question remains: what feature of the search term is most important? Is it the visual representation of the term in the environment? Is it the name of the term in the environment? Or is it the behavior of the term in the environment? The speed and accuracy of mapping terms onto OpenCyc are impacted by the choice of the most important feature. For example, consider the terrain feature "Woods" from the dTank environment. When interacting with dTank, there is a terrain object that

appears to be made of pine trees and is the same size as the tank. It is named "Woods" by the dTank authors. When an agent is touching the "Woods" object, several things happen: 1) the agent can only travel at a fraction of their maximum speed, 2) projectiles are less likely to hit and damage the agent, 3) the amount of the map that the agent can see is restricted, and 4) the agent is less likely to be visible to other agents (the amount depends on the terrain the other agents occupy).

All four of these features define the "Woods" object in dTank, but it is highly unlikely that we can find a term in OpenCyc that matches all of these features exactly. Therefore, we have to choose the level of accuracy that is sufficient for our purposes. As an illustration, however, we present the process of determining several different possible terms, in increasing accuracy.

If we choose the name, "Woods", as most important, a simple lookup returns `WoodedArea`. The comment associated with `WoodedArea` is "A specialization of `GeographicalRegion`. Each `WoodedArea` is a place with a lot of trees." If we choose the visual representation of "Woods" as most important, a deeper search starting from `WoodedArea` returns `ConiferForest-C4` as a candidate mapping term. OpenCyc describes `ConiferForest-C4` as "A specialization of `ConiferForest`. Each `ConiferForest-C4` is a `GeographicalRegion` that is 75-100% covered with coniferous trees." The good news from this search is that `ConiferForest-C4` is a specialization of `WoodedArea`, so all attributes that apply to `WoodedArea` also apply to `ConiferForest-C4`.

Ultimately, it appears that the most reasonable term in OpenCyc for "Woods" is "`ConiferForest-C3`" (a less dense version of `ConiferForest-C4`). It matches "Woods" on a semantic and visual level. Additionally, `ConiferForest-C3` generalizes to `CanopyClosure-Dense`, `ConcealmentFromAerialDetection-Good`, and `CoverFromDirectFire-Good` (descriptions which closely match the cover and concealment properties of "Woods"). The effect of slowing agents is not quite covered, but the proportion of slowing (50%-75%) is at least similar to the density of the trees. Despite a rather exhaustive search of OpenCyc, our term is still not quite perfect.

The simple lookup mapping for "Woods" is "`WoodedArea`", while the in-depth exploration mapping is "`ConiferForest-C3`". There was substantial work in determining the **single best** OpenCyc term for "Woods"; for a discussion of whether or not it was worth it, see section 4.

### 3.2 Pruning the Hierarchy

We have created software written in Common Lisp that communicates with OpenCyc through an ASCII API. Once a collection of domain-specific terms have been mapped to OpenCyc terms, we can extract the hierarchical structure from OpenCyc. This structure is a multiple-

inheritance tree with a root at "Thing" (the most general OpenCyc term) and leaves for each of the supplied terms.

Once the web of terms has been extracted from OpenCyc, some amount of pruning can be done; the level of pruning (or possibly expansion) depends highly on the intended use of the web. For instance, a web pruned from the root down to the most specific parent term (Lowest Common Genl or LCG) is a useful way to get an overall sense of the complexity and structure of OpenCyc. Pruning to just the key terms (terms that contain more than one child term) results in significant pruning and is probably the best, most compact way to visualize the relationships of the terms to each other. The resulting web can also be pruned to a single-inheritance tree. The single-inheritance tree may be the most useful for mapping to ACT-R since it matches the single-inheritance mechanism in ACT-R. Visualizations of each method of pruning are shown in the next section, "3.3 Visualizing the Hierarchy".

Our current pruning methods involve selecting nodes and roots for pruning based on the count of leaves reachable from each node. Roots which have child nodes with the same count are removed as a method of automatically finding the LCG. Nodes which have no increased count compared to child nodes are removed as a method of simplifying the branches of the hierarchy. When creating the single-inheritance tree, parents with lower counts are retained; the object is to get the deepest, skinniest tree possible which would correspond to the richest discrimination tree in representation space.

Because the pruning and visualization of the OpenCyc structure is quick and automated, we recommend exploring all versions of pruning and use the resulting visualization to determine the structure that is most useful for the desired task.

### 3.3 Visualizing the Hierarchy

We have come to the realization that understanding terms and their relationships is nearly as hard a problem as determining a relationship in the first place. Thus, we have invested considerable effort in developing methods for quickly visualizing any mapping of ontology to cognitive architecture.

Our software incorporates the open-source graph visualization software, GraphViz ([www.graphviz.org](http://www.graphviz.org)). We translate the OpenCyc structure into a GraphViz-compatible representation of nodes and edges; GraphViz automatically handles the layout and visualization of the structure.

The following figures are representations of different pruning methods applied to the same structure; the OpenCyc structure connecting all of the terrain objects from dTank. For all figures, the green boxes represent the initial list of terms that was used to generate the structure (the user-determined OpenCyc terms). The ellipses

represent the top of the object hierarchy (LCG), and white boxes represent intermediate terms that were extracted due to their connection to both the LCG and the atomic terms. Note that we do not include the completely unpruned structure up to “Thing” as the image is only readable when poster-sized. Indeed, the unpruned structure up to the LCG (GeographicalThing) is barely readable.

Figure 1 is the extracted web of all seven dTank terrain concepts up to their LCG: GeographicalThing. The labels are intentionally unreadable; the point of including the figure is to illustrate the scope and complexity of the hierarchy up to the LCG. Figure 2 is the same structure, but pruned to only the key terms. Only the terms that

directly decompose into more than one term are considered key terms. Notice that the entire left half of Figure 1 is pruned down to GeographicalRegion, OutdoorLocation, and CoverFromDirectFire-Good in Figure 2. Also note that Figure 2 provides just as much information as Figure 1 about the similarities between terms.

Figure 3 is the web from Figure 1 pruned to just single-inheritance. Figure 4 is a fully-pruned version of Figure 3, where only key terms are included. Note that there is very little difference between the two fully-pruned figures; Figure 2 has only one more term than Figure 4, which is CoverFromDirectFire-Good.

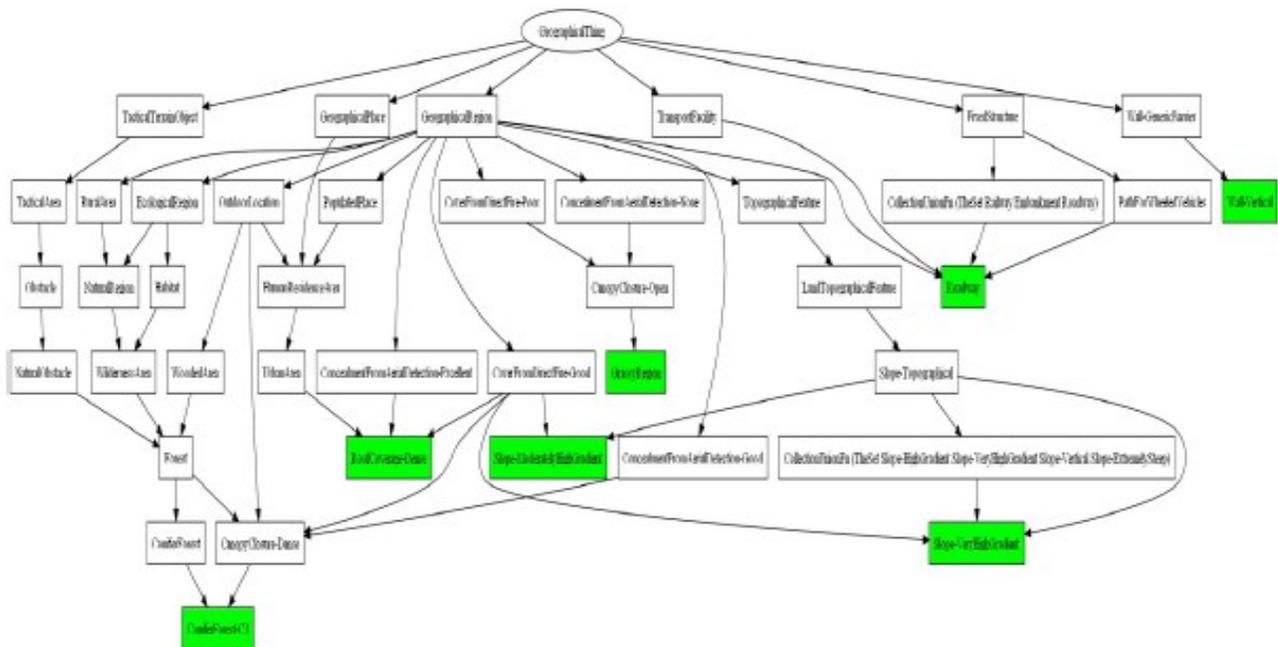


Figure 1: Full Hierarchy of dTank Terrain Terms up to the Lowest Common Genl. Node labels are deliberately unreadable; the same structure (parsed) is presented clearly in the following figures.

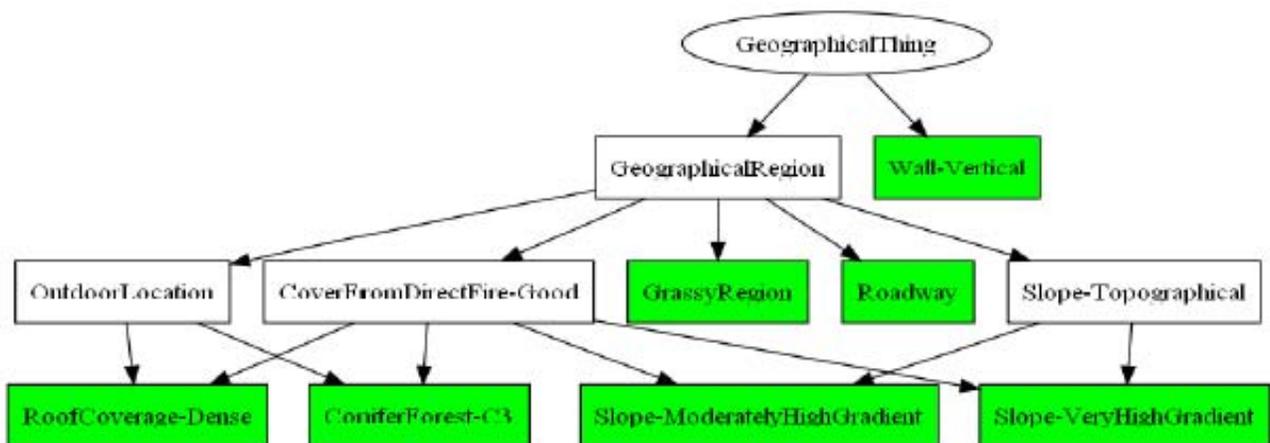


Figure 2: Pruned Hierarchy of dTank Terrain Terms up to the Lowest Common Genl

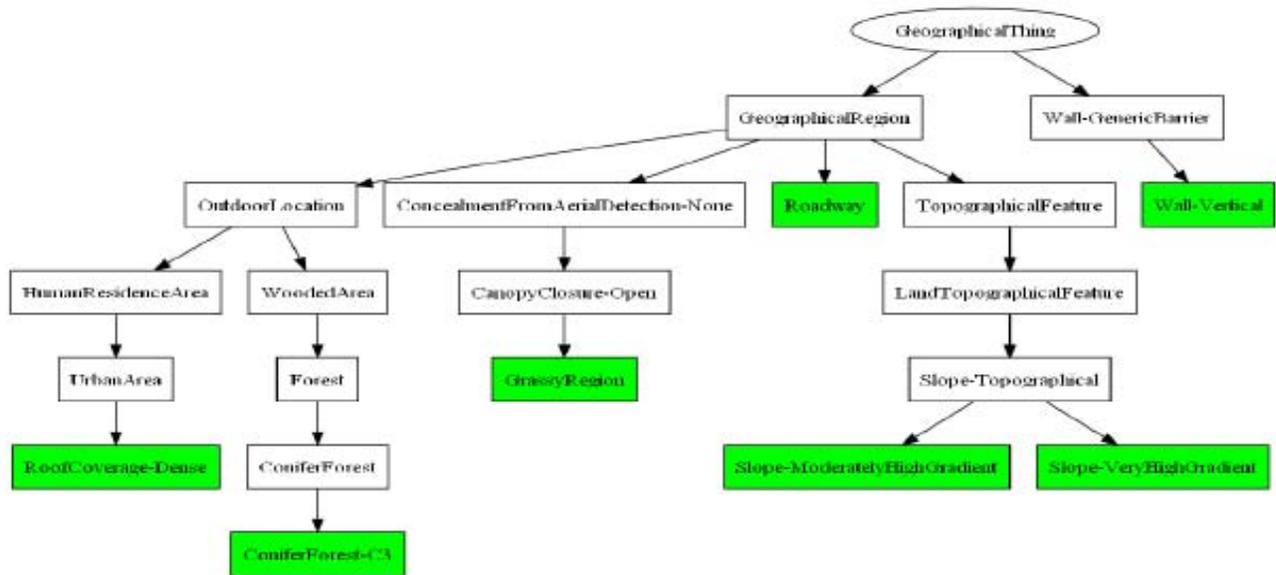


Figure 3: Single-Inheritance Hierarchy of dTank Terrain Terms up to the Lowest Common Genl

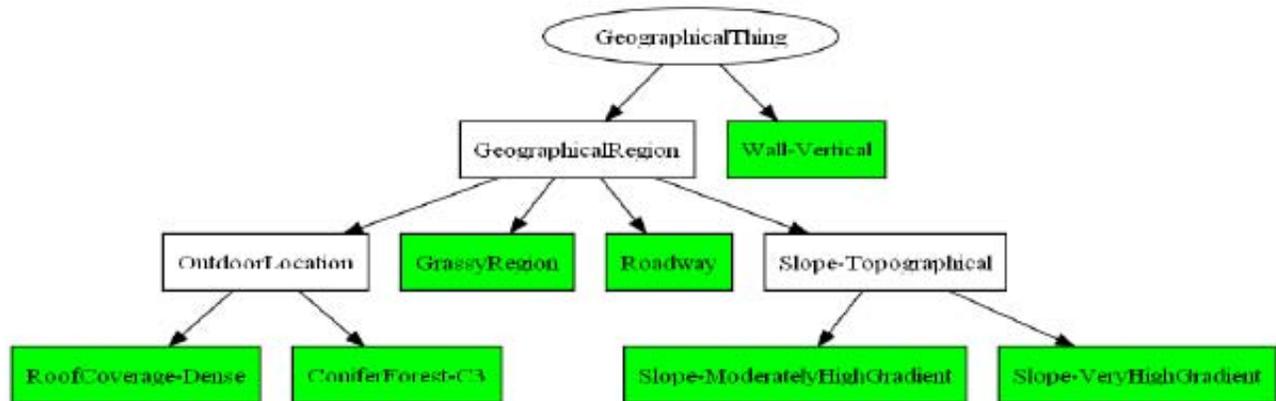


Figure 4: Single-Inheritance Pruned Hierarchy of dTank Terrain Terms up to the Lowest Common Genl

In a single inheritance hierarchy derived from Figure 4, this term would be represented as a relation, *CoverFromDirectFire*, between various object types and their value, *Good*. This same relation with different objects types, e.g. *GrassyRegion*, and values, e.g. *Poor*, could also be used to represent related terms such as *CoverFromDirectFire-Poor*. This approach born out of the necessity of leveraging a cognitive architecture with a limited single-inheritance mechanism thus has a number of advantages. First, it makes explicit the semantic relation between apparently unrelated terms *CoverFromDirectFire-Good* and *CoverFromDirectFire-Poor* (and *CoverFromDirectFire-Excellent*, etc.) and thus provides a unification of those terms. Second, it also introduces a distinction between terms in the hierarchy that correspond to fundamental distinctions (e.g., a human-built structure vs. a natural feature), and thus are mapped to the type hierarchy in the cognitive architecture, and those that correspond to superficial, potentially changing features (e.g., a forest provides good cover from fire unless it is sprayed with defoliant) that are mapped to relations binding objects to properties and their values.

However, as previously mentioned, the needs of the user should determine which of the four representations is most useful.

#### 4. Considerations for the Mapping Process

We have chosen to pursue a limited static mapping of terms to the cognitive architecture, largely for performance reasons. Ontologies are logic-based formalisms that often make unreasonable runtime demands upon the systems operating upon them (e.g. rule-based inference). However, embedded agents in real-time environments (as is the case of most of our target environments) are under severe time constraints to produce effective behavior. Moreover, cognitive architectures impose additional constraints upon the space of acceptable processing mechanisms, ruling out some (e.g., logical inference) in favor of others (e.g., subsymbolic mechanisms of activation spreading and matching, adaptive learning processes, etc). These considerations have been extremely important in providing a set of constraints for designing a feasible interaction between OpenCyc and a cognitive architecture.

In the previous section, we presented two vastly different methods for determining a translation from domain-specific terms and their OpenCyc counterparts. All previous figures showed the structure obtained by the in-depth exploration method. The quick lookup mapping of the same terms created a similar, but not identical structure (structure not shown). However, it is unclear as to whether the differences between the two structures present any problems to the main goal, which is model reuse and portability.

It would seem that the time saved in the mapping process (on the order of 15 minutes **per term**) presents a strong case for using the simple lookup procedure. The terms obtained from this procedure are not quite accurate, however, when it comes to describing the behavior of the terms in the specific domain. In fact, one runs the risk of creating a mapping that is still domain-specific, despite the use of generic vocabulary. If the attributes related to the terms are idiosyncratic, then the term cannot be reused in a different domain. The time required to rectify the situation is likely orders of magnitude less than the time needed to create a new model from scratch. Ultimately, the proposed abstraction to domain-independent vocabulary could present a substantial step towards model reuse and portability.

## 5. Discussion

The choice of whether to perform the representational mapping between OpenCyc and a cognitive architecture statically or dynamically has significant implications. While dynamic access to the ontology and knowledge base is more general, static mapping requires less meta-cognitive management on the part of the architecture and is easier to manage. However, given the size of ontologies such as OC, it would impose significant capacity commitments upon the architecture. The solution we have employed here is a combination of a static mapping of key representational terms with dynamic access to additional knowledge (e.g. inference) as needed. A full static mapping is not, as of this date, feasible within the ACT-R cognitive architecture, but this is a practical limitation rather than a theoretical one and may be overcome as the architecture is applied to larger-scale problems and domain-specific models are integrated into increasingly complex assemblies converging to the knowledge of a human individual (or collective).

Another area where the current paper has been somewhat silent is that of inter-agent communication. The ontological approach taken here might be used to provide a solution not only to the acquisition of information from, and the expression of actions upon, the environment but also to the communication between entities operating in that environment. For instance, plans of action might be expressed using the same terms with appropriate augmentations, potentially allowing even agents developed using different formalisms to communicate

with each other. This use would correspond to the more recent purpose of ontologies, which is to facilitate and integrate communication across electronic media.

## 6. References

- Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review* 111, (4). 1036-1060.
- Anderson, J. R., & Lebiere, C. (1998). *The Atomic Components of Thought*. Mahwah, NJ: Lawrence Erlbaum Associates.
- Ball, J., Rodgers, S., & Gluck, K. (2004). Integrating ACT-R and Cyc in a large-scale model of language comprehension for use in intelligent agents. In *Papers from the AAI Workshop*, Technical Report WS-04-07, pp. 19-25. Menlo Park, CA: AAAI Press.
- Best, B. J. & Lebiere, C. (2006). Cognitive agents interacting in real and virtual worlds. In R. Sun (ed.), *Cognition and Multi-Agent Interaction: From Cognitive Modeling to Social Simulation*. Cambridge University Press; New York, NY, 186-218.
- Cycorp (2002). *OpenCyc Tutorial : An Introductory Walk Through Ontological Engineering*. Retrieved from [http://www.cyc.com/cycdoc/walkthroughs/oeintro\\_cats\\_frames\\_long.html](http://www.cyc.com/cycdoc/walkthroughs/oeintro_cats_frames_long.html).
- Euzenat, J. & Shvaiko, P. (2007). *Ontology Matching*. Springer.
- Langley, P. & Choi, D. (2006). A unified cognitive architecture for physical agents. *Proceedings of the Twenty-First National Conference on Artificial Intelligence*. Boston: AAAI Press.
- Lenat, Douglas. Hal's Legacy: 2001's Computer as *Dream and Reality*. "From 2001 to 2001: Common Sense and the Mind of Hal"
- Matuszek, C., Cabral, J., Witbrock, M., and DeOliveira, J (2006). An introduction to the syntax and content of Cyc. In *AAAI Spring Symposium*.
- Morgan, G. P., Ritter, F. E., Stevenson, W. E., Schenck, I. N., & Cohen, M. A. (2005). dTank: An environment for architectural comparisons of competitive agents. In *Proceedings of the 14<sup>th</sup> Conference on Behavior Representation in Modeling and Simulation*, 133-140. 105-BRIMS-044. Orlando, FL.
- Newell, A. (1990). *Unified theories of cognition*. Cambridge, MA: Harvard Univ. Press.

## Author Biographies

**BRAD BEST** is a Principal Scientist at Adaptive Cognitive Systems, LLC, in Boulder, CO., where he focuses on cognitive modeling of adaptive behavior in complex environments, especially those that have significant spatial and temporal aspects. His current research interests include integrating perception with decision making in robotic and virtual agents and the development of methods for analyzing, understanding and visualizing model behavior in these environments.

**NATHAN GERHART** is a Research Programmer at Adaptive Cognitive Systems, LLC, in Boulder, CO. He is the resident subject matter expert for performance in virtual environments and is currently writing software to assist with the exploration and optimization of parameterized models as well as the OpenCyc interaction software discussed in this paper.

**CHRISTIAN LEBIERE** is a research faculty in the Psychology Department at Carnegie Mellon University. His main research interest is computational cognitive architectures and their applications to psychology, artificial intelligence, human-computer interaction, decision-making, intelligent agents, robotics and neuromorphic engineering.

# Plan, replan and plan to replan

## Algorithms for robust courses of action under strategic uncertainty

Maciej Łatek and Seyed M. Mussavi Rizi  
 Department of Computational Social Science  
 George Mason University  
 4400 University Drive, Fairfax, VA 22030  
 mlatek, smussavi@gmu.edu

January 6, 2010

**Abstract** In this paper we present an efficient computational implementation of non-myopic  $n$ -th order rationality using multi-agent recursive simulation in which simulated decision makers use simulation to inform their own decision making. An agent is  $n$ -th order rational if it determines its best response assuming that other agents are  $(n - 1)$ -th order rational with zeroth-order agents behaving according to a specified, non-strategic rule. We describe how to combine these two techniques with a replanning heuristic to create a decision rule, called REplanning  $N$ -th order RAtionality (RENORA), allow an agent to strategize for more than one move forward in a tractable manner. Our approach addresses (a) randomness of the environment, (b) strategic uncertainty arising when an opponent has more than one equally good courses of action to choose from and (c) failures in plan execution caused by either the environment or the opponent interference. To demonstrate the properties of RENORA, we introduce a model of a dynamic environment that encompasses both competition and cooperation between two agents, trace the relative performance of agents as a function of RENORA parametrization, and outline in detail the steps RENORA agents go through as they reason about the environment and other agents.

**Keywords** Recursive Agent-based Models, Multiagent Learning and Decision-making, Cognitive Architectures, Robust Replanning

## 1 Introduction

The departure point for this paper is  $n$ -th order rational agents. An agent is first-order rational if it calculates the best response to his beliefs about the strategies of zeroth-order agents and the state of the world. An agent is  $n$ -th order rational if it determines its best response assuming that the other agents are  $(n - 1)$ -th order rational.  $n$ -th level rationality models have few degrees of freedom, often only the rationality levels of all strategic agents that can be calibrated from data. If this is accomplished, such models can perform *descriptive* and *normative* roles of a decision framework that guides agents on their courses of action (COA) in a multiagent setting. Combined with easy sensitivity analysis of results and an efficient multiagent formulation that can be solved even for complex environments,  $n$ -th order rationality is a convenient heuristic for reasoning in multiagent environments.

In this paper, we first integrate myopic  $n$ -th order rational with multiagent models. We then show how to extend such a framework to make them robust and tractable for non-myopic agents with long planning horizons. Finally, we introduce a

dynamic multiagent environment and use it to outline in detail the steps that endogeneously replanning  $n$ -th order rational agents go through as they reason about the environment and other agents. Finally, we perform sensitivity analysis of the extended  $n$ -th order rationality formulation.

## 2 Multiagent Recursive Simulation

Assume a model of reality  $\Psi$  either as a multiagent model that describes strategic interactions among  $K$  agents or as a statistical model that simply predicts some macro variables. Before we show how to introduce planning agents into  $\Psi$ , let us describe what questions it answers.  $\Psi$

**What can happen** Defines the space of feasible COA for each agent and all possible sequences of interactions among agents.

**What has happened** Contains a library of historical trajectories of interactions among agents called historical behaviors library (HBL). If no actual information is

available, HBL is either empty or filled with hypothetical expert-designed interaction scenarios.

**How agents value the world** Codes every agent’s payoffs for any trajectory of interactions among agents based on the agent’s implicit or explicit preferences or utility function.

Latek et al. (2009) show that  $\Psi$  (a) can be decomposed into a state of the world  $C_t$  and agents’ current COA  $\mathbf{p}_t = (p_t^1 \ p_t^2 \ \dots \ p_t^K)$  where  $p_t^i$  stands for agent  $i$  COA at time  $t$  and (b) maps  $C_t$  and  $\mathbf{p}_t$  into a realization of the future state  $C_{t+1}$  and current agents’ payoffs  $\mathbf{r}_t = (r_t^1 \ r_t^2 \ \dots \ r_t^K)$  where  $r_t^i$  is the current payoff for agent  $i$  :  $\langle \mathbf{r}_t \ C_{t+1} \rangle = \Psi(\mathbf{p}_t, C_t)$ .

Agent  $i$  can use heuristics or statistical procedures to compute the probability distribution of payoffs for each COA it picks; then pick a COA that is in some sense “suitable”. Alternatively, it can clone  $\Psi$ , simulate the world forward; derive the probability distribution of payoffs for each available COA by simulation and pick a suitable COA. When applied to multiagent models this recursive approach to decisionmaking amounts to having simulated decisionmakers use simulation to choose a COA (Gilmer, 2003). Note that agents perceive  $\Psi$  with varying degrees of accuracy and have different computational capabilities to clone  $\Psi$ . So agents need not produce a clone of  $\Psi$  that is isomorphic to  $\Psi$  itself; however, in this paper we assume they do. We call this technology multiagent recursive simulation (MARS).

### 3 $n$ -th Order Rational Agents

Agents in any  $\Psi$  pick COA that achieve a goal, for example maximizing the stream of expected payoffs for the planning horizon of  $h$  periods forward. If a  $\Psi$  contains strategic agents whose payoffs depend on the choices of other agents, such agents must have access to plausible mechanisms to compute optimum COA.  $n$ -th order rationality is one such mechanism. An  $n$ -th order rational agent (NORA) assumes that other agents in  $\Psi$  are  $(n - 1)$ -th order rational and best responds to them. A zeroth-order rational agent acts according to a non-strategic heuristic such as randomly drawing a COA from the HBL or continuing the current COA. A first-order rational agent assumes that all other agents in  $\Psi$  are zeroth-order rational and best responds to them. A second-order rational agent assumes that all other agents in  $\Psi$  are first-order rational and best responds to them. NORA have inconsistent beliefs about the level of rationality each has. For example, observe that if the assumption of a second-order rational agent about other agents in  $\Psi$  is correct; they must assume that the second-order rational agent is zeroth-order rational agent.

NORA offer the following advantages in a multiagent settings: (a) Models can be heterogeneous in NORA level of ra-

tionality. (b) NORA do not require any learning phase to satisfy Hannan consistency: they converge to the best response to the other agents’ COA at every stage. (c) NORA can be efficiently implemented even for complex  $\Psi$  by using MARS. In the next section we introduce a structural design that introduces uses MARS to solve planning and replanning for NORA.

## 4 Robust Planning with MARS-NORA

### 4.1 Myopic Planning

To describe the algorithm that introduces myopic NORA into a  $\Psi$ , we denote the level of rationality for an NORA with  $d = 0, 1, 2, \dots$ . We label the NORA corresponding to level of rationality  $d$  as  $A_d$  and a set of its possible COA as  $\ell_d$ :

- $d = 0$  A zeroth-order rational agent  $A_0$  chooses COA in  $\ell_0$ ;
- $d = 1$  A first-order rational agent  $A_1$  chooses COA in  $\ell_1$

and so forth. Now we can show how myopic NORA use MARS to plan COA.

$\ell_0$  contains **non-strategic** COA that are not conditioned on  $A_0$  expectations of what other agents will do. Without assuming that other agents optimize,  $A_0$  arrives at  $\ell_0$  by using non-strategic heuristics like expert advice, drawing a COA from a probability distribution over the COA space or sampling the HBL for a COA. Example 1 shows possible choices for  $\ell_0$  used by an  $A_0$  stock trader.

A trader holds a stock that has lost 15% value. He can sell the stock, hold it, or buy more:

1. If the industry stock value has shrunk less than 15%, sell. Else, hold.
2. With probability 0.1, sell or buy more. Else hold.
3. If in the last year the stock has not rebounded 90% of times within 2 weeks of a 15% devaluation, sell. Else hold.

**Example 1:** Rule-driven  $\ell_0$  for an  $A_0$  stock trader.

Recall that an  $A_1$  agent forms  $\ell_1$  by best responding to  $\ell_0$  adopted by another agent in  $\Psi$  whom it assumes to be  $A_0$ . If  $A_1$  assumption is true, the other agent does not assign a level of rationality to the  $A_1$  agent. So  $A_1$  finds a strategy that **on average** performs best when the  $A_0$  agent adopts any COA in its  $\ell_0$ , integrating out the stochasticity of the  $\Psi$ .  $A_1$  can sample its opponent  $\ell_0$  uniformly or according to the opponent empirical frequency of adopting each COA. Algorithm 1 shows this process.

**Input:** Set  $\ell_0$  for  $A_0$ ; COA space for  $A_1$ ; number of samples  $K$

**Output:** Set  $\ell_1$  of optimal COA for  $A_1$

```

foreach COA  $a_1$  available in  $A_1$  do
    foreach  $a_0 \in \ell_0$  do
        foreach  $i \leq K$  do
             $s = \text{cloned } \Psi$ ;
            query  $s(a_0, a_1) = A_1$  payoff;
        end
    end
    Compute average  $\bar{s}(a_1)$  over  $K$  samples;
end
Eliminate all dominated COA, arriving at  $\ell_1$ ;
Return  $\ell_1$ ; choose a single COA for  $A_1$  from  $\ell_1$ .
    
```

**Algorithm 1:** NORA( $A_1, 1$ )

Best response formation for  $A_2$  follows a similar vein: An  $A_2$  agent best responds to another agent who it assumes to be  $A_1$ . Therefore,  $A_2$  assumes that the other agent *assumes* that the  $A_2$  agent is indeed  $A_0$ .  $A_2$  finds a strategy that **on average** performs best when the  $A_1$  agent adopts **any** of its  $\ell_1$  COA. In order to accomplish this, the  $A_2$  agent first computes a set of  $\ell_1$  for  $A_1$ ; then it best responds to the  $\ell_1$  it has computed. Algorithm 2 shows this process.

**Input:** Set  $\ell_1 = \text{NORA}(A_1, 1)$ ; COA space for  $A_2$ ; number of samples  $K$

**Output:** Set  $\ell_2$  of optimal COA for  $A_2$

```

foreach COA  $a_2$  available to  $A_2$  do
    foreach  $a_1 \in \ell_1$  do
        foreach  $i \leq K$  do
             $s = \text{cloned } \Psi$ ;
            Calculate  $s(a_1, a_2) = A_2$  payoff;
        end
    end
    Compute  $\bar{s}(a_2)$ ;
end
    
```

Eliminate all dominated  $A_2$  COA, arriving at  $\ell_2$ ;  
Return  $\ell_2$ ; choose a single COA for  $A_2$  from  $\ell_2$ .

**Algorithm 2:** NORA( $A_2, 2$ )

## 4.2 Non-myopic Planning

Algorithms 1 and 2 use MARS to solve the myopic planning problem for NORA. How can  $A_d$  derive optimum COA if (a) it wishes to plan for more than one step; (b) takes random lengths of time to execute a COA or aborts COA execution mid course, and (c) interacts asynchronously with other NORA. To address these issues, we introduce the notion of planning horizon  $h$ . While no classic solution to problems (b) and (c) exists, the classic method of addressing (a), finding the optimum of  $h \times$  number of COA, leads to exponential explosion. The

following algorithm called RENORA solves (a), (b) and (c) simultaneously:

**Input:** COA space for  $A_d$ ;  $\ell_{d-1}$ ;  $d$ ;  $h$ ; number of samples  $K$

**Output:** Set  $\ell_d$  of optimal COA for  $A_d$

```

foreach COA  $a_d$  available to  $A_d$  do
     $s = \text{cloned } \Psi$ ;
    Assign initial COA to all agents  $\in s$ ;
    foreach  $a_{d-1} \in \ell_{d-1}$  do
        while  $s.\text{time}() < h$  do
            if  $a_d$  is not executing then
                RENORA( $A_d, d, h - s.\text{time}()$ )
            end
        end
        Accumulate  $A_d$  payoff  $\ += s(a_{d-1}, a_d)$ ;
    end
    Compute  $\bar{s}(a_d)$ ;
end
    
```

Eliminate dominated COA, return  $\ell_d$ .

**Algorithm 3:** RENORA( $A_d, d, h$ )

## 5 Experiments

### 5.1 Environment

To demonstrate the properties of RENORA, we use a multi-agent environment we call PushGame, a two-player stochastic game with 5 states  $A$  to  $E$  shown in Figure 1. Formally, a general-sum, two player, stochastic-game  $M$  on states  $S = \{1, \dots, N\}$ , and actions  $A = \{a_1, \dots, a_k\}$  consists of:

- **Stage Games:** Each state  $s \in S$  is associated with a two-player, fixed-sum game in strategic form, where the action set of each player is  $A$ . We use  $R^i$  to denote the payoff matrix associated with stage-game  $i$ .
- **Probabilistic Transition Function:**  $P_M(s, t, a, a')$  is the probability of a transition from state  $s$  to state  $t$  given that the first agent plays  $a$  and the second agent plays  $a'$ .

In PushGame, each agent has to choose one of the two actions at each state: agent 1 has actions  $U$  and  $D$  and agent 2 actions  $L$  and  $R$ . A  $2 \times 2$  matrix associated with each state codes payoffs  $p_1$  for agent 1 and  $p_2$  for agent 2 depending on the state, the agent and its opponent actions. Additionally, certain combinations of agent actions may cause states to change. For example, if agent 1 plays  $D$  and agent 2 plays  $L$  in state  $A$ , both agents receive payoff 0, but the state will change to  $B$ . States are grouped into three categories. State  $A$  does not favor any agent and requires coordination between agents to ensure payoff 1. If one of the agents deviates in order to secure a payoff higher than 1, it may break the symmetry of the game. States  $B$  and  $C$  favor agent 1 who receives a constant

payoff of 2 at the expense of agent 2 who receives either 0 or  $-1$ . States  $D$  and  $E$  favor player 2.

At each asymmetric state, the stronger agent is predictable: agent 1 in states  $B$  and  $C$  always plays  $U$ ; agent 2 in states  $D$  and  $E$  always plays  $R$ . Suppose in state  $A$  agent 1 deviates and forces transition to state  $B$ . The weaker agent 2 has two choices: it can either avoid payoff  $-1$  and coordinate with the stronger agent 1 to receive 0 or accept the punishment of  $-1$  in order to return to the symmetric state  $A$ . Return to symmetry requires the weaker agent to accept a short-term loss in the hope of long-term gain. This deterministic setup for PushGame allows us to test the influences of agent rationality levels and planning horizons without the obfuscating effect of inherent randomness in the environment or strategic uncertainty.

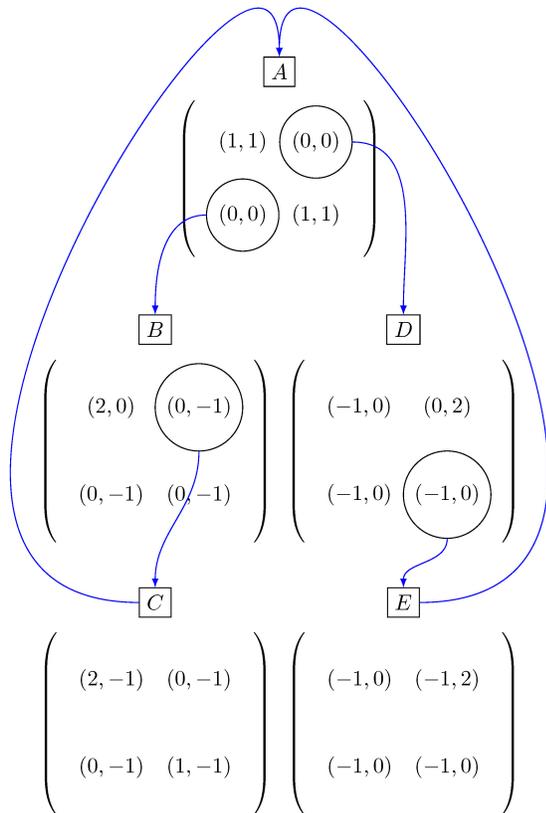


Figure 1: PushGame: A 5-state stochastic game used as a testbed to demonstrate the properties of RENORA. Possible transitions among states are denoted with  $\rightarrow$  and happen with probability 1 if agents play a proper combination of actions. If no transition is drawn, the state does not change from iteration to iteration.

## 5.2 Simulation traces

Figure demonstrates the mechanics of simulation cloning and replanning. We lay out the trace of a single call to

RENORA( $A_1, 3, 3$ ) on Figure 2(a). Six outgoing paths appear on each reoptimization node: 3 of which are blue, corresponding to simulations cloned by agent 1; 3 are red, corresponding to simulations cloned by agent 2. Each bundle of 3 same colored paths corresponds to a single call to RENORA with one subtree shorter than the remaining two. The shorter subtree corresponds to the first instruction of RENORA where an  $A_d$  is figuring out the initial step by its  $A_{d-1}$  opponent. The remaining two subtrees evaluate the fitness of each of the two available actions available in each state of PushGame. Assuming that each agent reoptimizes after completing an action, every call to RENORA leads to other calls to RENORA with smaller  $d$ , shorter  $h$  or both. In the process of solving the replanning problem each agent uses cloned simulations to optimize over its COA and to predict the steps its opponents would take and the evolution of the states of PushGame. Repeated interactions between the two agents generate traces shown on Figure 2(b).

## 5.3 Influence of $d$ and $h$

In order to assess the influence of  $d$  and  $h$  on the performance of a PushGame agent, we performed a simple parameter sweep outlined in Table 1, the results of which are summarized in Table where absolute and relative performance of agent 1 is averaged out and presented as a function of  $h_1 - h_2$  and  $d_1 - d_2$ . Additionally, we enumerate the frequency with which cooperative state  $A$  is visited. We divide  $(h_1 - h_2) \times (d_1 - d_2)$  into three regions:

$|h_1 - h_2| \geq 3 \wedge |d_1 - d_2| \geq 3$  One agent has a very short planning horizon and a low rationality level whereas the other has a long planning horizon and high rationality level. Cooperation is sustained and the more rational agent ensures fast return to state  $A$ . If agent 1 is the rational agent, it makes sure that the return to symmetry happens through a branch of PushGame that favors him;

$h_1 - h_2 \leq -2 \wedge d_1 - d_2 \geq 3$  Agent 1 has a higher level of rationality, but a much more shorter planning horizon than agent 2. Agent 1 is unable to make short-time tradeoffs and gets locked in an asymmetric branch that does not favor him. His absolute and relative performance is minimized;

$(h_1 - h_2) + (d_1 - d_2) \approx 0$  Both agents have similar cognitive capacities, cooperate often maximizing their absolute payoffs. If agent 1 has a higher planning horizon, it may also maximize its relative payoff.

Table presents the projection of a 4-dimensional parameter space into 2 dimensions; therefore, it should be interpreted with caution. Nevertheless, it proves that the RENORA algorithm allows an agent to make strategic decisions in a dynamic environment.

Parameter	Scenario value	Meaning
$h$	$\{1, \dots, 5\}$	Planning horizon. Each agent has its own $h$ and $d$ .
$d$	$\{0, \dots, 4\}$	Level of rationality. For $d = 0$ , $\ell_0$ is assumed to be uniform randomization over actionspace regardless of planning horizon.
numSamples	1	Number of samples taken to control for the randomness of the environment. PushGame is deterministic.
forwardLookingSamples	1	Number of samples taken to control for strategic uncertainty.
backwardLookingSamples	0	Number of historical COA that agents include in $\ell_0$ .
maxT	50	Maximal time for an individual simulation run.
numRep	20	Number of repetitions per combination of $h$ and $d$ .

Table 1: Simulation parameters used in experiments.

## 6 Summary

In this paper, we introduced a context-independent multi-agent implementation of  $n$ -th order rationality for replanning agents with arbitrary planning horizons and demonstrated its functionality on a test cases. We presented algorithms that enable us to introduce  $n$ -th order rational agents into any multi-agent model and demonstrated that  $n$ -th order rational agents are model-consistent. We also showed how an  $n$ -th order rationality model deviates systematically from equilibrium predictions as agents are engaged in a multi-tiered game of out-guessing each others' responses to the current state of world.

## References

Gilmer, J. (2003). The use of recursive simulation to support decisionmaking. In S. Chick, P. J. Sanchez, D. Ferrin, & D. Morrice (Eds.) *Proceedings of the 2003 Winter Simulation Conference*.

URL <http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:THE+USE+OF+RECURSIVE+SIMULATION+TO+SUPPORT+DECISIONMAKING#0>

Latek, M., Axtell, R., & Kaminski, B. (2009). Bounded ratio-

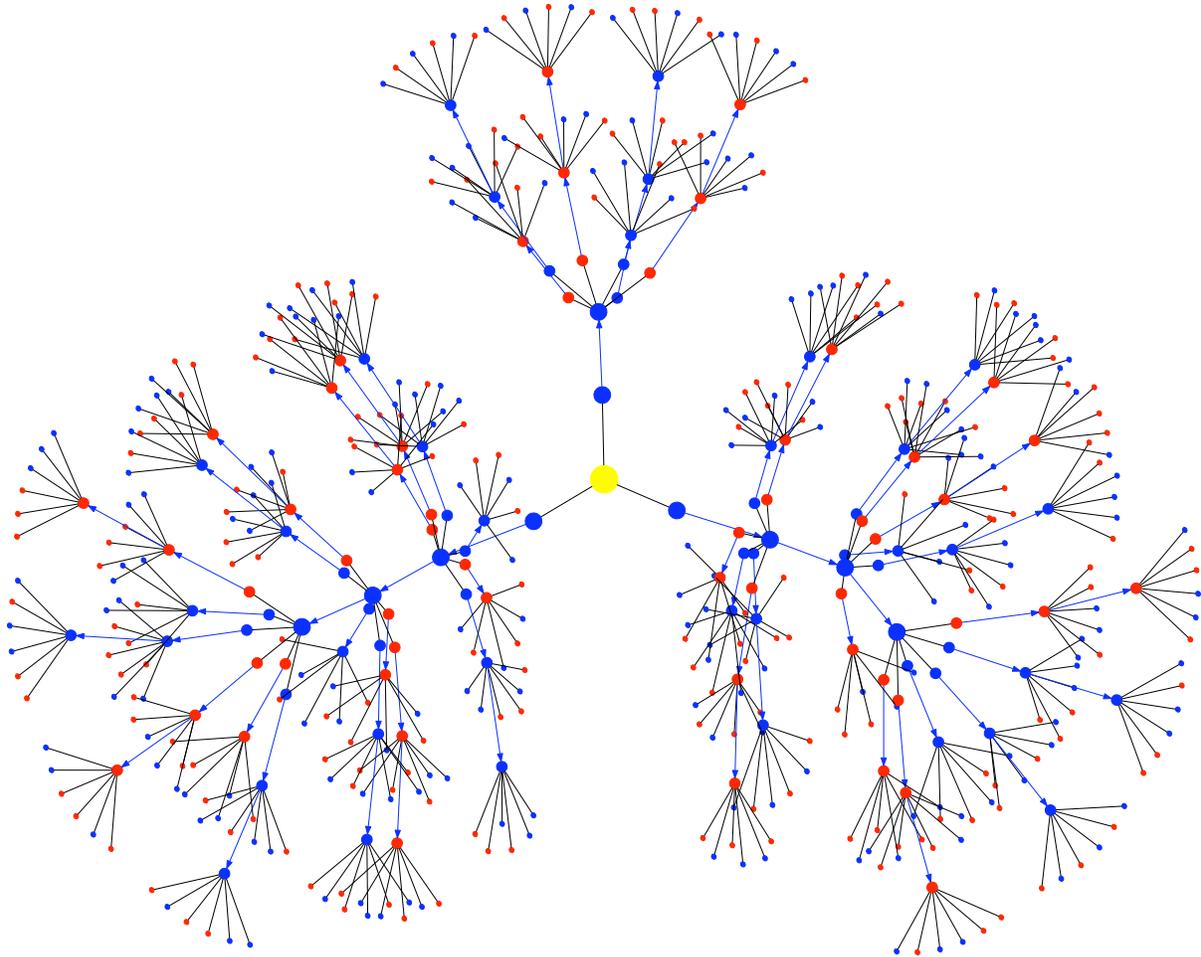
nality via recursion. *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*, (p. 457-464).

URL <http://portal.acm.org/citation.cfm?id=1558013.1558076>

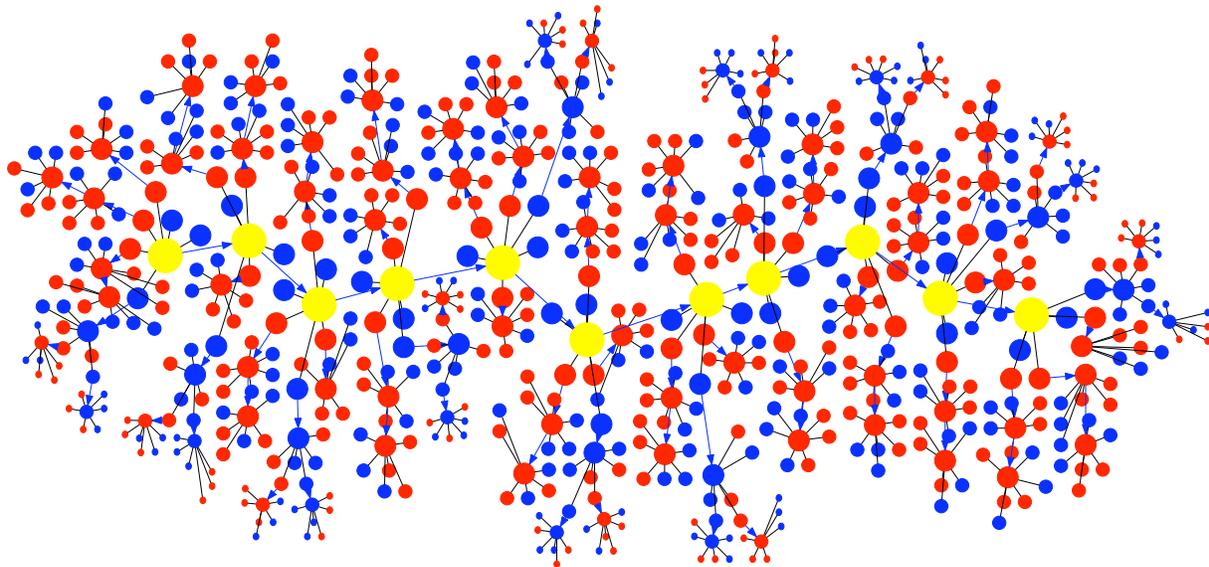
## 7 Biographies

**Maciej M. Latek** is a doctoral candidate in Computational Social Science at George Mason University. He holds a graduate degree in Quantitative Modeling from the Warsaw School of Economics. An operations researcher and data miner, Mr. Latek has worked with strategic interaction environment using a number of modeling approaches such as game theory, multi-agent modeling and data mining.

**Seyed M. Mussavi Rizi** is a doctoral candidate in Computational Social Science at George Mason University, developing multiagent models of conflict and political economy, protocols for model verification and validation and HSCB data requirements of large-scale agent-based models. He holds graduate degrees in economics, specializing in econometrics, from Tufts University, and in International Relations, specializing in international security, from the Fletcher School.



(a) A sample invocation of  $\text{RENORA}(A_1, 3, 3)$  for agent 1. The small subtree in the middle corresponds to solutions of  $\text{RENORA}(A_2, 2, 3)$  and  $\text{RENORA}(A_1, 2, 3)$  used by agent 1 to obtain predictions of  $\ell_2$ .



(b) 10 iterations of PushGame with two  $\text{RENORA}(2, 2)$  agents.

Figure 2: Mechanics of RENORA. Legend: ● the top-level universe, ● ● observations of cloned simulations, — cloning process,  $\rightarrow$  observations of the same universe at different times. Blue instances are simulation cloned by agent 1, red by agent 2.

Difference of p1-p2		h1-h2								
		-4	-3	-2	-1	0	1	2	3	4
d1-d2	-4	0.02	-0.12	0.07	0.37	0.15	0.19	0.68	-0.02	0.30
	-3	0.15	-0.44	0.03	0.34	0.15	0.33	0.41	0.57	0.32
	-2	-0.35	-0.44	-0.38	0.14	0.07	0.25	0.33	0.62	0.14
	-1	-0.47	-0.54	-0.20	-0.04	0.09	0.30	0.50	0.49	0.50
	0	-0.54	-0.53	-0.43	-0.24	0.08	0.25	0.43	0.50	0.60
	1	-0.66	-0.51	-0.45	-0.33	-0.09	-0.02	-0.02	0.23	0.42
	2	-0.70	-0.50	-0.42	-0.27	-0.01	0.08	0.15	0.49	0.57
	3	-0.18	-0.40	-0.19	-0.26	-0.15	-0.15	0.15	0.37	-0.53
	4	0.03	-0.83	-0.27	-0.27	-0.06	0.07	-0.09	0.06	0.26

Absolute payoff p1		h1-h2								
		-4	-3	-2	-1	0	1	2	3	4
d1-d2	-4	0.97	0.66	0.69	0.85	0.44	0.60	0.72	0.05	0.31
	-3	1.05	0.73	0.90	0.87	0.79	0.79	0.69	0.70	0.30
	-2	0.28	0.50	0.52	0.77	0.61	0.73	0.75	0.72	0.30
	-1	0.46	0.45	0.73	0.83	0.88	0.94	1.00	0.81	0.72
	0	0.26	0.30	0.48	0.58	0.77	0.87	0.85	0.89	0.77
	1	0.15	0.42	0.53	0.66	0.76	0.87	0.85	0.86	0.80
	2	0.01	0.19	0.34	0.52	0.69	0.83	0.82	1.02	0.91
	3	0.19	0.16	0.35	0.41	0.55	0.64	0.91	1.14	0.57
	4	0.01	0.00	0.11	0.31	0.44	0.42	0.66	0.85	0.60

Frequency of state A		h1-h2								
		-4	-3	-2	-1	0	1	2	3	4
d1-d2	-4	0.98	0.85	0.72	0.61	0.47	0.57	0.31	0.35	0.36
	-3	0.77	0.70	0.66	0.56	0.52	0.48	0.35	0.34	0.35
	-2	0.59	0.61	0.66	0.59	0.57	0.54	0.52	0.37	0.37
	-1	0.55	0.56	0.51	0.62	0.58	0.53	0.55	0.42	0.47
	0	0.56	0.53	0.58	0.57	0.67	0.58	0.54	0.59	0.49
	1	0.41	0.46	0.54	0.56	0.56	0.69	0.54	0.49	0.49
	2	0.36	0.43	0.43	0.55	0.60	0.59	0.78	0.67	0.58
	3	0.36	0.36	0.45	0.52	0.51	0.64	0.57	0.60	0.60
	4	0.35	0.29	0.34	0.42	0.54	0.50	0.73	0.89	0.73

Figure 3: The first two tables show averages of absolute and relative payoffs of agent  $A_1$  as a function of differences  $d_1 - d_2$  and  $h_1 - h_2$ . The last table enumerates the frequency with which the cooperative state  $A$  is visited.

# Data-Driven Coherence Models

*Peter Danenberg*

Computer Science Department  
University of Southern California  
941 Bloom Walk  
Los Angeles, CA 90089-0781  
peterchd@usc.edu

*Stacy Marsella*

Institute for Creative Technologies  
13274 Fiji Way  
Marina del Rey, CA 90292  
marsella@ict.usc.edu

Keywords: Thagard, coherence, Iraq, data-driven

## Abstract

How groups maintain and revise their beliefs and attitudes in the face of new information is a basic research question in human social behavior and communications, as well as having a range of applications in crafting effective communications in such areas as health interventions, political campaigns and advertising. In this paper, we argue for a coherence based approach for modeling group belief revision processes and as a framework for studying belief and attitude change. Coherence models have a rich history of applicability in the psychological sciences where they have been used to explain a range of belief maintenance processes in the individual. Given that processes of social comparison and pressure can homogenize a cohesive group's beliefs, we argue in this paper to extend the application of coherence models to modeling group belief systems. Additionally, we address challenges in constructing and using such belief models. Typically, creating accurate models of either an individual or group's beliefs requires the painstaking engagement of domain experts. We present and demonstrate a method for producing them from data and exploring potential vectors of attitude change in their subpopulations.

## 1 Introduction

### 1.1 Thagard's coherence

Coherence has been proposed as a general cognitive mechanism by which, for instance, a person forms explanations (Bonjour, 1976), integrates information to form impressions of others (Rawls, 1974-5); and resolves cognitive dissonances between beliefs and behavior (Festinger, 1954).

Coherence is part of a rich history of philosophical debate. Bosanquet argues (Bosanquet, 1912, p. 340) that it stretches back to Plato's theory of forms; where a set of  $N$  manifestations asymptotically coheres towards its universal form; and even in Hegel's dialectic, where disparate or indeed antithetical elements cohere in the process of "sublation."

By contrast, Aristotle's critique of Platonic realism lays the foundations of empiricism; and the Platonic-Aristotelean breach eventually leads to the foundationalism vs. coherentism debate (BonJour, 1985; Moser, 1988a; BonJour, 1988; Moser, 1988b). Whereas the former argues that epistemological justification "requires a non-propositional basis in the contents of experience;" (Moser, 1988c) the latter maintains that "beliefs are justified by being inferentially related to other beliefs in the overall context of a coherent system." (Bonjour, 1976)

Thagard establishes his system of "coherence as constraint satisfaction", we argue, by drawing from coherentist and foundationalist models of justification. One determines, for instance, the justification of a belief *vis-à-vis* its explanatory corroboration by other beliefs in its system with which it's associated (Thagard and Verbeugt, 1998, p. 155); but surprisingly, perhaps, Thagard gives priority to beliefs from observation (Thagard and Verbeugt, 1998, p. 157). Bonjour calls this 'weak foundationalism,' whereby the "initial modicum of justification [for empirical beliefs] must be augmented by a further appeal to coherence before knowledge is achieved." (Bonjour, 1976, p. 284)

### 1.2 Attitude change

We'd like to address the problem of attitude change, proposing a practical method for identifying potential vectors of

communicative hegemony; of interest in health intervention, political campaigns and marketing propaganda.

Finding the right communication to persuade someone, however, often hinges on tailoring it to their attitudes and beliefs. This suggests a dual-pronged approach of exploring alternative messages even while tailoring them to potentially receptive subgroups. Such a dual-pronged approach requires searching two spaces simultaneously: the space of possible message contents and the space of possible subgroups to which the message will be conveyed. To avoid searching the Cartesian product of message-subgroups, we can identify subgroups based on whether they share a common coherence model that is amenable to change and then use that model to suggest approaches to attitude change (see section 4.1, “Perturbation”). We take for granted, however, that coherence mechanisms provide a way to optimize messages for a given subgroup.

Coherence models in psychology, however, have largely been seen as cognitive mechanisms operating within the individual. The strong view of our approach is to argue that the coherence mechanisms also operate in group attitude change; nevertheless, a weaker view may be sufficient: e.g. finding a stereotypical, average individual of a group for which the message works.

The argument for extending coherence to modeling groups follows from several classic theories in social psychology. Most notably, Festinger’s work on social comparison theory (Festinger, 1954, p. 125) that argues that individuals have a need to assess their beliefs by comparison with others. Festinger’s work suggests that groups strive for a quiescent homogenization of opinion; and to that end tend to exclude discrepant members, pressure non-discrepant ones towards uniformity. As a result, groups evince a principle of spontaneous self-cohesion not unlike the reduction of cognitive dissonance in individuals. Similar views can be seen in more recent theories as social appraisal theory.

We argue, therefore, that persuasive messages targeted at groups will demonstrate a similar attitude-mutating effect across its members.

Thagard’s doctrine of coherence as constraint satisfaction provides our point of departure (Thagard and Verbeugt, 1998); whose models, however, are laboriously forged by domain experts relying on intuition. Our counterproposal, therefore, is a data-driven approach whose process is three-fold:

- inducing structural models from survey data;
- “drilling down” into the beliefs of subgroups exposed

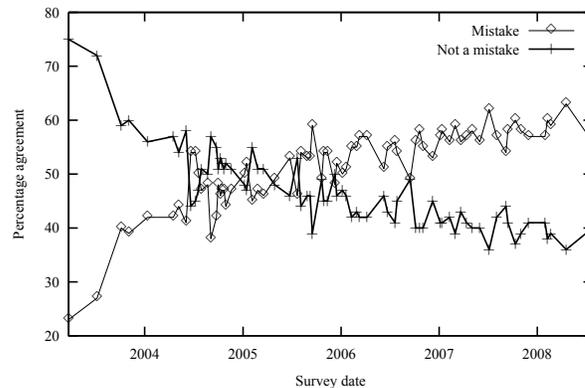


Figure 1: Gallup survey, “Do you think the United States made a mistake in sending troops to Iraq?”

by the data;

- perturbation of the subgroup-models to expose mutable attitudes as potential targets of persuasion.

By way of case study, we apply our method to public opinion around the Iraq War.

## 2 Motivating Example: Iraq

The Iraq war was a highly polarizing event. A January 2007 poll showed that roughly three-quarters of the world’s population disapproved of how the U.S. policy on Iraq (BBC World Service, 2007); American opinion had a relatively constant, even bipartition from 2004 until 2006 (Gallup, Inc., 2008), when opposition to the Iraq War began to increase by a widening margin (figure 1).

The Iraq War struck us as a potentially fertile ground for studying attitude change, given the volatile and strong, even radicalizing, nature of people’s opinion on the matter; and, indeed, motivating people to provide data was relatively simple (see section 5).

## 3 Coherence Model

Our working model of attitude stability and mutation is based on Thagard’s formalization of coherence as constraint satisfaction (Thagard and Verbeugt, 1998): *videlicet*, the partitioning of a system of propositions  $E$  into disjoint

subsets  $A$  and  $R$ ; corresponding to accepted and rejected propositions, respectively. The propositions themselves,

$$\{e_1, e_2, \dots, e_n\}$$

are subject to the weighted constraints

$$\{(e_{i_1}, e_{j_1}), (e_{i_2}, e_{j_2}), \dots, (e_{i_n}, e_{j_n})\}$$

such that

$$\begin{aligned} ((e_i, e_j) \in C+ &\rightarrow e_i \in A \leftrightarrow e_j \in A) \wedge \\ ((e_i, e_j) \in C- &\rightarrow e_i \in A \leftrightarrow e_j \in R) \end{aligned}$$

where  $C+$  and  $C-$  are sets of positive and negative constraints. The coherence problem, then, becomes the maximization of  $W$ ; *id est*, the sum of all satisfied constraints' weights.

Although the coherence problem is NP-complete (Thagard and Verbeugt, 1998, page 2), there exist a number of approximating algorithms; from which we chose the connectionist for its natural affinity to coherence problems (see section 3.1) and general applicability.

The connectionist model has been variously described as:

- minimizing the “energy” of a system though gradient descent (Sejnowski, 1986; Hopfield, 1982);
- maximizing the “harmony” of a system (Smolenksy, 1986);
- maximizing the “goodness-of-fit” of a system’s constraints, such that

$$G(t) = \sum_i \sum_j w_{ij} a_i(t) a_j(t) + \sum_i input_i(t) a_i(t)$$

where  $w$  corresponds to the weight of a constraint,  $a$  to a node’s activation, and  $input$  to an imposed bias (Rumelhart and McClelland, 1986).

### 3.1 Goodness-of-fit

Thagard characterizes coherence as constraint satisfaction by abstracting upon Rumelhart’s goodness-of-fit (Thagard and Verbeugt, 1998, page 10); and generalizes away, in particular, the latter’s adherence to neural networks. Armed with his abstracting coherence, Thagard is able to reformulate classic problems across several areas of research, including:

**psychology:** cognitive dissonance (Schultz and Lepper, 1996), interpersonal relations (Read and Marcus-Newhall, 1993);

**politics:** deliberate democracy (Arrow, 1963; Black, 1998);

**ethics:** reflective equilibrium (Daniels, 1979; Reuzel et al., 2001).

Spellman, et al. (Spellman et al., 1993) adapt Thagard’s coherence model to simulate attitudinal shifts during the First Gulf War; which adaptation they characterize as “dissonance reduction.” Proceeding from a hand-crafted network of attitudinal relations, they capture the maintenance of cognitive consistency across attitude-shifting events; which corroborates survey data they gathered and independently analysed.

Going beyond Thagard, we’ve developed a technique of perturbation (*vide* section 4.1) or *subjunctive* constraint satisfaction; whereby we determine, for any given target node, its prime hegemons.

Coherence models are typically hand-crafted by researchers and other domain experts (Thagard, 2003); requiring not only extensive knowledge but also subject to gaps in knowledge and biases. What follows is a method to create coherence models directly from data.

## 4 Data-driven Model Construction

Spirtes *et al.* developed a search algorithm for discovering causal structures from data, which they called the “PC algorithm” (Spirtes et al., 2000, p. 84). It starts by forming a complete undirected graph (whose vertices correspond to random variables), deleting conditional independencies and orienting the remaining links according to Pearl’s IC algorithm (Pearl, 2000, page 50).

Assuming that the functions  $Adjacent(G, i, j)$  and  $Adjacencies(G, i)$  have been defined, which return whether  $i$  and  $j$  are adjacent in graph  $G$ , and all the vertices adjacent to  $i$  in  $G$ , respectively.

The SGS algorithm, predecessor to PC, had an expected running time of  $\Omega(k^n)$ ; which PC has improved to  $O(n^k)$  by testing fewer d-separations in the case of sparse DAGs. (That a given DAG be sparse is often a reasonable assumption (Kalisch and Bühlmann, 2007, page 2).) PC works, namely, by incrementally removing conditional independencies of order  $0 \leq k \leq n$ ; where  $n$  is the cardinality

of the largest set  $k$  d-separating some nodes  $i$  and  $j$ . Its performance is therefore inversely proportional to the connectedness of a given graph.

### 4.1 Perturbation

The skeleton  $C$  returned by PC-Algorithm is a coherence-like model suitable for exploration by perturbation.

For a given target node  $t$  among nodes  $\{v_1, v_2, \dots, v_n\} = V$  in a coherence network, perturbation individually sets the activation of  $v_i \in V$  to min-activation or max-activation, runs the connectionist algorithm, notes the divergence of  $t$ 's activation, and performs a partial ordering of  $V$  for each  $t$  by  $\max(|\Delta_{ti\min}|, |\Delta_{ti\max}|)$  into non-, weak- and strong-hegemons.

## 4.2 Method

### 4.2.1 pcalg

Data is collected, stored and imported into R; the `pcalg` (Kalisch and Maechler) package is then used to create an apposite skeletal UDAG, and specialize this UDAG into one of an equivalence class of underlying DAGs.

### 4.2.2 Influence

The underlying DAGs are then imported into Influence, a reimplement of Thagard's ECHO by Danenberg, et al.; via one of two methods:

1. a Scheme-to-Java bridge implemented in SISC;
2. a custom R server on an arbitrary machine.

Once in Influence, one can create arbitrary cross-sections of the data by subsetting on demographics or response; and from this cross-section, recreate the graph structure (including node activations and internodal relationships).

Next, the graphs of sufficiently interesting subpopulations can be perturbed and compared; and their structural differences reasoned upon (see section 5).

## 5 Experiment

For the survey instrument, we assembled twenty-eight items on a five-point Likert scale; with a demographic section covering education, ethnicity, income and party affiliation. As of this writing, the survey is still available on-line (Danenberg, 2007).

We solicited for subjects on Google AdWords (Google, Inc., 2008) from March 27–29, 2007 under the slogan: “We need your opinion on Iraq. Take our Iraq War survey!” The cost of the campaign was \$1451.29; and of the 473, 685 ad impressions, we had 627 visits; of those visits, 442 surveys were submitted; of those surveys, 98 were rejected for incompleteness: leaving 344 valid responses.

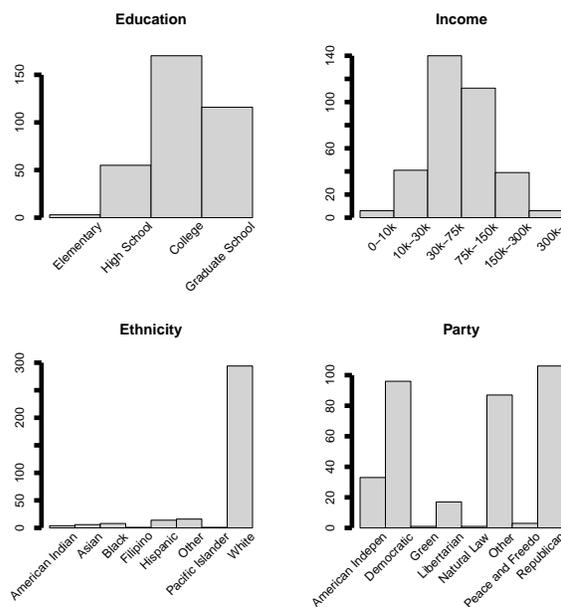


Figure 2: Histogram of respondents over education, ethnicity, income and party

Figure 2 summarizes the demographic data. Although education,  $\chi^2(2, N = 341) = 468, p < 0.001$  (Stoops, 2004); ethnicity,  $\chi^2(2, N = 322) = 76.4, p < 0.001$  (Survey, 2006); and income,  $\chi^2(2, N = 187) = 95.3, p < 0.001$  (U.S. Census Bureau, 2006) defied the census; party affiliation,  $\chi^2(2, N = 344) = 7.62, p < 0.05$  compared favorably with the latest Pew statistics (Pew Research Center, 2008), but that fewer Democrats filled out the survey than expected (table 1).

	Republicans	Democrats	Others	Total
Observed	106	96	142	344
Expected	96.3	120.4	127.3	344
Residuals	0.986	-2.224	1.305	0.67

Table 1: Observed vs. expected party affiliation

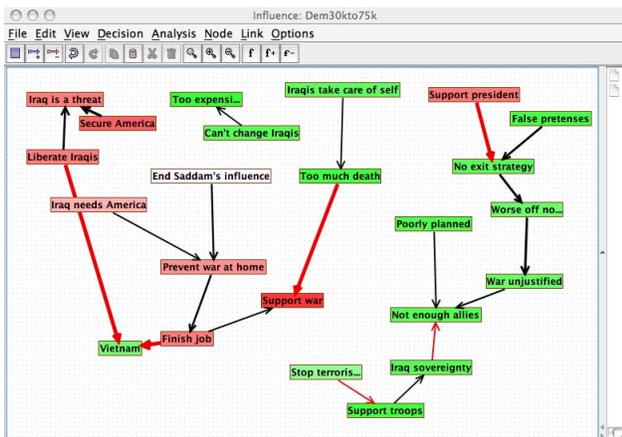


Figure 3: Poor Democrats

### 5.1 Model

Figures 3, 4, 5, 6, show the models gleaned from subsetting the data by income (poor/rich) and party (Democrat/Republican) after analysis; whose relative sparseness compared to the full model is proportional to their data density.

### 5.2 Analysis

Table 2 summarizes the perturbation results on all four subgroups; a striking observation whereof is how class runs thicker than party: rich Democrats and Republicans are repulsed by the war’s cost (“Too expensive”), while poor Democrats and Republicans are repulsed by its inhumanity (“Vietnam”).

Almost universally, however (with the exception of rich Democrats, for whom we lack data), “Support the president” positively correlates with “Support war” (figures 3, 5, 6); even though Democrats and Republicans differ across party lines.

Amongst poor Democrats and poor Republicans (figures 3, 5), “Vietnam” appears to be associated with “Too much death;” we speculate the cause being that American casual-

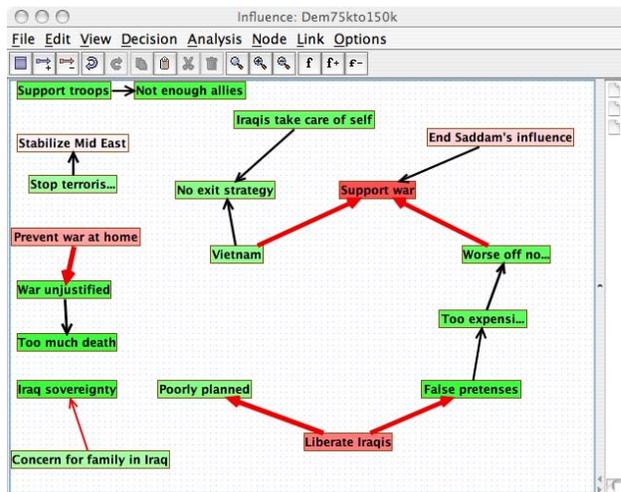


Figure 4: Rich Democrats

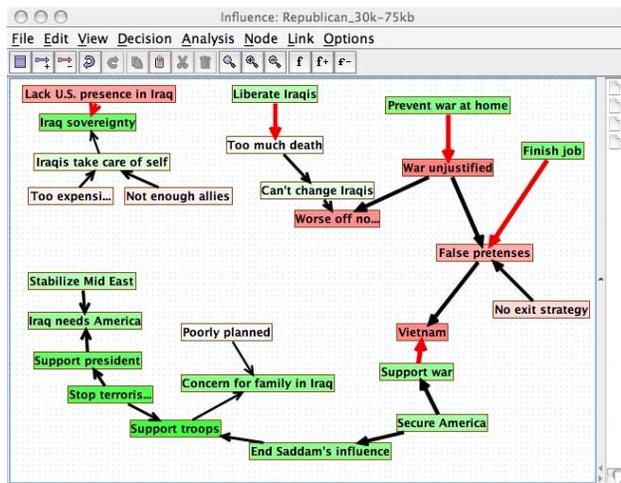


Figure 5: Poor Republicans

ties in Iraq are predominantly poor (Scotland, 2008).

Rich Democrats and rich Republicans (figures 4, 6), on the other hand, demonstrate correlation between “No exit strategy” and “Too expensive;” could it be that they foresaw asymmetrical taxes on this liability (Montopoli, 2009)?

## 6 Conclusion

Creating coherence models by hand is an error-prone activity which beggars, furthermore, one’s ingenuity; we present a method for creating models from data and identifying potential vectors of attitude change through perturbation.

Influence	Republican		Democrat	
	Poor	Rich	Poor	Rich
<i>Positive</i>				
Strong	Iraq needs America	n.d. <sup>a</sup>	n.d.	n.d.
Moderate	n.d.	Liberate Iraqis Stabilize Mid East Iraq needs America Concern for family in Iraq Prevent war at home Finish job Secure America	Secure America Finish job Stabilize Mid East Prevent war at home	n.d.
Weak	n.d.	n.d.	n.d.	n.d.
<i>Negative</i>				
Strong	Vietnam	n.d.	n.d.	n.d.
Moderate	n.d.	Worse off now War unjustified Poorly planned Vietnam No exit strategy Can't change Iraqis Not enough allies Too expensive Iraqis take care of self	Vietnam False pretenses No exit strategy	Too expensive Worse off now Not enough allies Can't change Iraqis
Weak	n.d.	n.d.	n.d.	n.d.

<sup>a</sup>No data

Table 2: Perturbation on “Support war” for poor/rich Democrats/Republicans

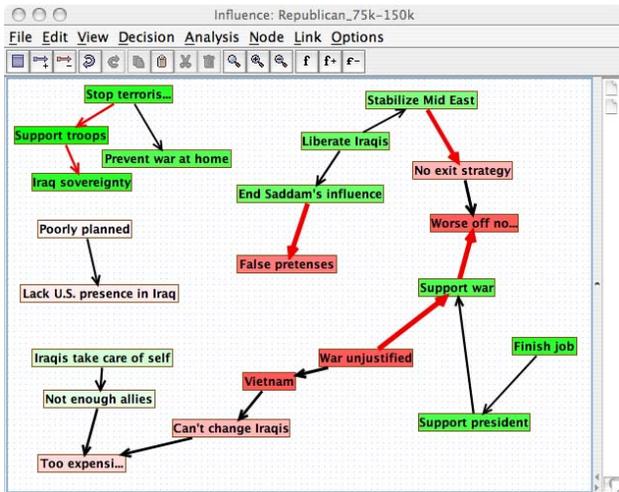


Figure 6: Rich Republicans

We'd like to test mutating craft of the thus prescribed vectors in a follow-up study, wherein appropriate or inappropriate messages preface the administration of the instrument and attitude deviation is tested against the null hypothesis.

We're also evaluating the utility of the method for marketing and political campaigns.

## References

K. J. Arrow. *Social choice and individual values*. Wiley, 1963.

BBC World Service. World view of us role goes from bad to worse. [http://news.bbc.co.uk/1/shared/bsp/hi/pdfs/23\\_01\\_07\\_us\\_poll.pdf](http://news.bbc.co.uk/1/shared/bsp/hi/pdfs/23_01_07_us_poll.pdf), January 2007.

Duncan Black. *The Theory of Committees and Elections*. Kluwer, 1998.

Laurence Bonjour. The coherence theory of empirical knowledge. *Philosophical Studies*, 1976.

Laurence BonJour. *The structure of empirical knowledge*. Harvard, 1985.

Laurence BonJour. Reply to moser. *Analysis*, 48(4):164–5, October 1988.

Bernard Bosanquet. *The Principle of Individuality and Value*. Macmillan, 1912.

Peter Danenberg. Iraq war survey. <http://iraq-war-survey.com>, March 2007.

N. Daniels. Wide reflective equilibrium and theory acceptance in ethics. *Journal of Philosophy*, 76:256–82, 1979.

Leon Festinger. A theory of social comparison processes. *Human Relations*, 7:117–40, 1954.

Gallup, Inc. Gallup's pulse of democracy: The war in iraq. <http://www.gallup.com/poll/1633/Iraq.aspx>, 2008.

Google, Inc. Adwords. <https://adwords.google.com>, 2008.

J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences of the USA*, 9:2554, 1982.

Markus Kalisch and Peter Bühlmann. Estimating high-dimensional directed acyclic graphs with the pc-algorithm. *Journal of Machine Learning Research*, 8: 616–36, 2007.

Markus Kalisch and Martin Maechler. pcalg: Estimating the skeleton and equivalence class of a dag. <http://cran.r-project.org/web/packages/pcalg/index.html>.

Brian Montopoli. David obey calls for war tax on wealthy. <http://www.cbsnews.com/blogs/2009/11/23/politics/politicalhotsheet/entry5748813.shtml>, November 2009.

Paul K. Moser. Internalism and coherentism: a dilemma. *Analysis*, 48(4):161–3, October 1988a.

Paul K. Moser. How not to be a coherentist. *Analysis*, 48(4):166–7, October 1988b.

Paul K. Moser. The foundation of epistemological probability. *Erkenntnis*, 28(2):231–51, 1988c.

Judea Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2000.

Pew Research Center. Fewer voters identify as republicans. <http://pewresearch.org/pubs/773/fewer-voters-identify-as-republicans>, 2008.

John Rawls. The independence of moral theory. *Proceedings and Addresses of the American Philosophical Association*, 48:5–22, 1974-5.

S. Read and A. Marcus-Newhall. The role of explanatory coherence in the construction of social explanations. *Journal of Personality and Social Psychology*, 65:429–47, 1993.

- R.P.B. Reuzel, G.J. van der Wilt, H.A.M.J. de Heer, and P.F. de Vries Robbé. Interactive technology assessment and wide reflective equilibrium. *Journal of Medicine and Philosophy*, 26(3):245–61, 2001.
- David E. Rumelhart and James L. McClelland, editors. *Parallel distributed processing: explorations in the microstructure of cognition*. MIT Press, 1986.
- T. R. Schultz and M. R. Lepper. Cognitive dissonance reduction as constraint satisfaction. *Psychological Review*, 103:219–40, 1996.
- Herald Scotland. Us death toll in iraq is ‘mostly white and poor’. <http://www.heraldscotland.com/us-death-toll-in-iraq-is-mostly-white-and-poor-1.877356>, March 2008.
- T. J. Sejnowski. *Parallel distributed processing: explorations in the microstructure of cognition*, volume 2, chapter 6, “Parallel distributed processing: explorations in the microstructure of cognition”, pages 372–89. MIT Press, 1986.
- P. Smolensky. *Parallel distributed processing: explorations in the microstructure of cognition*, volume 2, chapter 7, “Neural and conceptual interpretation of PDP models”. MIT Press, 1986.
- Barbara A. Spellman, Jodie B. Ullman, and Keith J. Holyoak. A coherence model of cognitive consistency: Dynamics of attitude change during the persian gulf war. *Journal of Social Issues*, 1993.
- Peter Spirtes, Clark Glymour, and Richard Scheines. *Causation, Prediction, and Search*. MIT Press, 2000.
- Nicole Stoops. Educational attainment in the united states: 2003. Technical report, U.S. Census Bureau, June 2004.
- American Community Survey. B02001. race - universe: Total population. Technical report, U.S. Census Bureau, 2006. URL {[http://factfinder.census.gov/servlet/DTable?\\_bm=y&-context=dt&-ds\\_name=ACS\\_2006\\_EST\\_G00\\_&-mt\\_name=ACS\\_2006\\_EST\\_G2000\\_B02001&-CONTEXT=dt&-tree\\_id=306&-redoLog=false&-all\\_geo\\_types=N&-geo\\_id=01000US&-currentselections=ACS\\_2006\\_EST\\_G2000\\_B02001&-search\\_results=01000US&-format=&-\\_lang=en](http://factfinder.census.gov/servlet/DTable?_bm=y&-context=dt&-ds_name=ACS_2006_EST_G00_&-mt_name=ACS_2006_EST_G2000_B02001&-CONTEXT=dt&-tree_id=306&-redoLog=false&-all_geo_types=N&-geo_id=01000US&-currentselections=ACS_2006_EST_G2000_B02001&-search_results=01000US&-format=&-_lang=en)}.
- Paul Thagard. Why wasn’t O.J. convicted? emotional coherence in legal inference. *Cognition and Emotion*, 17(3):361–83, 2003.
- Paul Thagard and Karsten Verbeugt. Coherence as constraint satisfaction. *Cognitive Science*, 1998.
- U.S. Census Bureau. Current population survey, 2006 annual social and economic supplement. [http://pubdb3.census.gov/macro/032006/perinc/new01\\_001.htm](http://pubdb3.census.gov/macro/032006/perinc/new01_001.htm), 2006.

## Author Biographies

**PETER DANENBERG** is a humanist-turned-hacker and graduate student at USC, specializing in cognitive modelling.

**STACY MARSELLA** is a research associate professor at the University of Southern California and an Associate Director of Research for Social Simulation at the Institute for Creative Technologies. He has worked for a number of years in the areas of planning, cognitive science, computer-assisted learning and simulation. He received his Ph.D. from Rutgers University.

# Adapting the Taxon-Task-Taxon Methodology to Model the Impacts of Chemical Protective Gear

Shane T. Mueller  
George Anno  
Corey Fallon  
Gene E. McClellan  
Owen Price

Applied Research Associates, Inc.  
1750 Commerce Center Blvd  
Fairborn, OH 45324  
smueller@ara.com

Keywords:

Performance prediction, cognitive moderators

**ABSTRACT:** *The Taxon-Task-Taxon method (Anno et al., 1996) is a statistical modeling approach to predict performance decrements in response to various stressors. Our research is extending this approach to accommodate new more acute stressors associated with chemical protective gear, and new tasks with greater involvement of cognitive, perceptual, and motor function. In this paper, we describe the basics of the T3 method and our approach to adapting it, and give a illustrative example that shows how the method can be used to account for performance decrements associated with wearing protective gloves. This illustration provides a substantive way in which the current T3 method can be augmented to account for performance decrements in a new subdomain, but also provides lessons for extending the method to new stressors and performance domains.*

## 1. Background

Many cognitive and behavioral models aim to predict performance under new conditions, such as predicting performance for new tasks based on a measured set, or predicting performance on yet-to-be-built systems based on current performance, or predicting performance on a current task in response to new stressors. Our research program aims to understand the cognitive and behavioral performance decrements of chemical protective gear (i.e., Mission-Oriented Protective Posture; or MOPP) worn by U. S. warfighters in response to the threat or presence of chemical or biological agents. The intent of our models is to understand how new equipment may impact performance across a wide range of tasks to provide guidance for future suit design. Thus, we aim to predict performance decrements on a much wider range of tasks than can be effectively measured, under equipment conditions that have not yet been developed, and for novel combinations of new stressor.

### 1.1 Taxon-Task-Taxon (T3) Methodology

Our approach to simulating performance decrements in novel tasks under novel stress conditions is based on the Task-Taxon-Task methodology (T3; Anno, Dore, and Roth, 1996). The method works by assuming that performance degradation is mediated through a set of skill taxons (based on pioneering work by Fleishman,

1975). Any task is assumed to use these taxons to different extents, and each stressor is assumed to slow processes related to each taxon by different amounts. A predicted performance decrement for a particular stressor on a particular task can be computed by essentially computing the sum of the taxon-related decrements from the stressor, weighted by the relative importance of each taxon for the task. This statistical modeling approach is substantially less detailed than many agent-based modeling systems, but has advantages to the extent that it can be tied fairly closely to data, and that the effort for modeling new tasks or systems is fairly minimal (essentially a process of performing task analysis in order to develop ratings across skill taxa). This is important for our goal, because a single suit design will eventually be used across most branches and specialties of the U.S. military, and so a crude model that can predict across many tasks is preferred over a detailed model that can only predict a small range of tasks.

To use the method, a task  $T_i$  may be represented as a set of weights (e.g., between 0 and 5) relating to the relative importance over five taxa (attention, perception, physical, psychomotor, cognitive):

$$T_i = [0, 1, 3, 0, 1]$$

And similarly a stressor may be represented as a set of decrements across taxa (with 0 representing no impact, and values smaller than 0 representing the increase in  $\log(RT1/RT0)$  ratio)

$$S_j = [-.05, -.01, -.2, -.05, -.1]$$

Here,  $T_i$  would represent a task with moderate physical requirements, and low requirements on other taxa. If  $T_i$  is assumed to take on unit of time, then the T3 model would assume that under stressor  $S_j$ ,  $\log(1/RT)$  of the task would be impacted by a factor of  $(0(-.05) + 1(-.01) + 3(-.2) + 0(-.05) + 1(-.1)) = -.71$ , which is a factor of 2.03. Thus, the large decrement high importance of the physical taxon, coupled with the large impact of the stressor on physical abilities would essentially double the time taken to perform the task.

The benefit of this method is that once careful assessment of the taxonomic weights are provided for a set of tasks, the impact of a particular stressor can be assessed using standard regression techniques (assuming a wide enough range of input tasks is available). Thus, the data fitting is a statistical process, although the decrements obtained could be used in other types of models. For example, along with its original use in predicting hypothesized impacts of chemical agents on soldier performance (e.g., Anno et al., 1996), this same method forms the basis for how the IMPRINT tool predicts performance decrements (Allender et al., 1997) for a number of stressors (MOPP, heat, cold, noise, and sleeplessness), although IMPRINT uses a set of nine taxons.

The T3 method was originally designed to predict behavioral decrements from toxic chemicals, based on a set of mediating symptomology. Such stressors have large-scale effects that may be well captured by global skill taxons. However, we are extending this method to account for the physical and especially cognitive stressors associated with chemical protective gear. Such stressors can have a much more acute impact on task performance. For example, one part of the MOPP suit is the gas mask and goggles, which have a well-understood impact on peripheral vision. Another component is butyl-rubber gloves, which impact a number of dexterous behaviors across specialties (see Mueller, et al., 2008a, 2008b). For such stressors, global taxons such as 'psychomotor' or 'perceptual' may no longer be sufficient to make useful predictions about performance decrements.

Along with the need to augment or change the current skill taxonomy, another problem for the T3 method is that as tasks become more complex and the stressors more acute, one may need better representations of tasks to make useful predictions about performance

decrements. Next, we will describe our approach to representing tasks.

## 2. Task-Goal-Operator-Taxon Analysis

One limitation of the original T3 method is it represents any task as a weighting across skill taxons. This may be appropriate for gross prediction of blunt stressors on highly constrained tasks, but it may be inappropriate for understanding the acute stressors of MOPP gear on detailed cognitive work. We have developed a task analysis method based on earlier GOMS methodologies (John & Kieras, 1994, Gray et al., 1993) by which we take a task and represent it as a critical path in a subgoal network (see Schweickert, Fisher, & Proctor, 2003) where each subgoal is accomplished by an operator, and each operator has a set of weights across relevant taxa (see Mueller et al., 2009a, for more detail). TGOT is similar to GOMS (Goal-Operator-Method-Selection rules) analysis in that is based on logical analysis of goals and subgoals which are traced to a set of operators. However, it differs because it uses a set of bottom-level operators that are tied to the task context, rather than low-level operators tied to an architecture. The point of TGOT analysis is to get to a level at which a task can be described in terms of its taxa, such that a stressor will have a linear impact on its time-to-perform. Thus, for GOMS, an operator is like a molecule: it can not be broken down further without changing its essence. For TGOT, an operator is like a mineral sample: any further subdivision will lead to identical parts in terms of the taxon distribution.

The use of a task network to represent tasks is important because of the ways in which we have hypothesized that protective gear may slow task performance. A partial list includes: First, the additional mass may simply make motor movement slower. Second, limited range-of-motion or perception may require taking new sets of actions (e.g., moving head to see in periphery). Third, reduced precision may lead to more errors which need to be corrected (e.g., mistaken key entry on keyboard). Fourth, wearing gear may place the wearer into a 'novice' performance mode as they grow accustomed to doing work under new conditions; eliminating automaticity gains. Fifth, gear may represent an attentional draw stemming from discomfort or additional self-monitoring required. Sixth, biophysical metabolic processes (heat, oxygen, bloodflow CO2 maintenance, etc) may produce neurophysiological inefficiency or physical fatigue that impacts task performance. Seventh, the wearer intentionally and strategically slow down to avoid costly immediate error correction or long-term fatigue.

Although some of these sources may be well-captured by describing a high-level task as a set of operators, others are not. For example, intentional strategic

slowing may work to even out performance over a long period of time, rather than having fast performance initially and very slow performance later. So, one may observe slowing on a task in response to wearing MOPP gear, but the source of that slowing is strategic rather than physical. More critically, strategic shifts in task performance may also stem from limited mobility or limited sensory input. This type of shift may change the operators associated with performing a task, and may change the critical path in task performance. So, a stressor may not only change how long it takes to perform each step of a task, but it may also change the number of steps. An example of this in the context of manual dexterity will be shown in Section 3. Finally, stressors that impact accuracy may produce highly non-linear effects on certain aspects of a task, because slowing could stem primarily from error correction rather than slowed operation. Some type of task network analysis is necessary to understand whether that type of impact will have a large impact on overall task performance.

### 3. Example: Impact of Protective Gear on Human Dexterity

As an illustration, we will examine how the T3 method can be deployed to model human dexterity data. The original method included only one taxon (psychomotor) that can reasonably be used to describe performance in dexterity tasks. Imprint incorporates two taxa (fine motor discrete and fine motor continuous), and assumes that only discrete action is impacted by protective gear. Such an example raises several questions. First, is a single taxon sufficient to capture the performance degradation on manual tasks associated with protective gear; and second, are there ways to know, *a priori* the extent to which a dexterity task will be impacted by a stressor?

As a first step, we present in Table 1 a set of proportional decrements for various motor dexterity tasks. In this Table, the performance decrement represents (Time with gloves)/(time in bare hands), so that a value of 1.0 would indicate no slowing from gloves, and larger values indicate larger impacts.

What can be said about the skill taxa necessary to capture these decrements? First, the one relevant taxon used previously (psychomotor) is probably insufficient. Certainly, one could assume that those tasks with greater decrements simply have higher psychomotor loadings. However, this is probably at odds with the ratings one would give *a priori*, and so is not very useful. For instance, it is probably unrealistic to say that those manual tasks which see little or no impact from protective gloves do not require psychomotor skill, and it would be difficult to predict *a priori* which types of tasks will have greater or lesser decrements,

especially when the decrements for similar tasks can vary so much.

Table 1: Performance decrements of various dexterity tasks.

Test	Perf. Decr.	Grasp	Touch	Pred
O'Connor Finger Test <sup>12456</sup>	1.14-1.72	5	1	1.29
Purdue Pegboard <sup>126</sup>	2.4-3.4	5	5	<b>1.6</b>
Minnesota Dexterity 1 hand <sup>3</sup>	1.17	2	3	1.27
Minnesota Dexterity-2 hand <sup>4</sup>	1.2-1.37	3	3	1.33
Manual Pursuit Rotor <sup>1</sup>	1.05	1	1	1.09
M16A1 Dis-Assembly <sup>5</sup>	1.24	3	3	1.33
M16A1 Assembly <sup>5</sup>	1.24	3	3	1.33
Find page in book <sup>3</sup>	1.25	3	3	1.33
1-5 number keypad entry <sup>3</sup>	1.09	1	1	1.1
Hunt-and-peck word typing <sup>3</sup>	1.22	1	3	1.23
Touch word-typing <sup>3</sup>	2.07	1	5	<b>1.37</b>
Typing response <sup>3</sup>	1.70	1	5	1.37
Mouse tracking <sup>3</sup>	1.15	1	3	1.23
Mouse—aimed movement <sup>3</sup>	1.01	1	1	1.1
Cord & Cylinder <sup>2,4</sup>	1.5-1.76	5	3	1.44
Bennet Dexterity test <sup>4</sup>	1.0-1.09	1	1	1.1
Pick up cylinder (20 mm+) <sup>3</sup>	1.05	1	1	1.1
Pick up cylinder (1 to 20 mm) <sup>3</sup>	1.25	3	3	1.3

<sup>1</sup>Bensel et al., 1987; <sup>2</sup>Taxiera et al., 1990; <sup>3</sup>Unpublished data by present authors; <sup>4</sup>McGinnis, Bensel, & Lockhart, 1973; <sup>5</sup>Garrett et al. 2006; <sup>6</sup>Johnson & Kobrick, 1997.

Note: Model fit excluded Purdue pegboard and touch-typing, which we assumed would have strategy shifts in response to protective gloves.

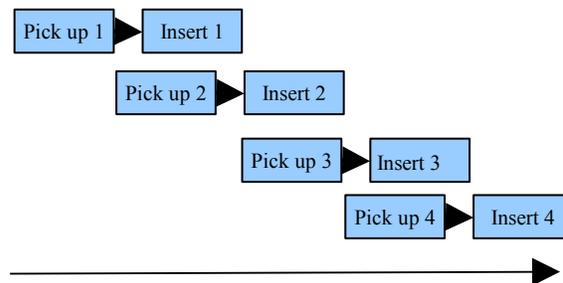
The two taxa used by IMPRINT are somewhat better, but they simply assume that ‘continuous’ tasks do not slowing, which could capture the small effects on the pursuit rotor and mouse aimed movement, but would miss the mouse tracking impact. As a first hypothesis, we propose that a way to capture these impacts would be to hypothesize two taxa: one related to grasping, and one related to the sense of touch. Initial ratings on the task for these taxa are provided in Table 1.

The Grasping taxon is important because picking up small objects has a moderate impact (25%) on performance, and this is a component that is present to in many of the tasks in Table 1. Loss of touch-sense could have a large impact depending on the context, because it may require costly error correction or strategy shifts. We hypothesize that this is partly responsible for the large decrements seen in typing (and indirectly, the Purdue task). Here, loss of touch sense is devastating. It can prevent touch-typing, which means that the errors one makes are not seen until it is very costly to correct. A typist must choose to either type, check for errors, and then correct errors, or slow down to a degree such that errors are not made (perhaps relying on visual and auditory feedback instead of touch sense). Either way, performance will slow substantially. The smallest impact seen on typing tasks was for number keypad entry: these were done hunt-and-peck style in both conditions, and the spacing of the number pad is big enough to avoid many mistakes. In essence, number-keypad entry would depend little on touch sense, whereas touch-typing relies heavily on it to know whether ones fingers are on the correct keys.

The Purdue test is interesting because it contains many of the same components measured in other tests, such as picking up small cylinders and placing them in holes or posts, which we showed to have a performance decrement of about only 25%. Yet the Purdue test had a substantial decrement at least ten times larger than these. What then can account for the difference? To answer this,, we need to understand better what the task involves.

The basic Purdue task involves four consecutive operations: 1. pick up and insert post; 2. pick up and insert washer; 3. pick up and insert sleeve; 4. pick up and insert second washer. Each consecutive step is performed by a different hand, so performance may be able to overlap substantially: Figure 1 illustrates how these four tasks may overlap because they use different hands.

Figure 1: Hypothesized subgoals to perform Purdue Pegboard task.



Total time to perform this task could be modeled as the sum (with p indicating pick-up time and i indicating insert time) of roughly  $p1 + i1 + i2 + i3 + i4$ .

However, for performance like this to occur, one needs to assume that these two tasks can be easily overlapped. Without protective gloves, the 'pick up' subtask might be thought of as performed by two operators, such as: move hand to tray; grasp object by feel. If we were to make a prediction about the performance decrement based on these operators using standard T3 methodology, we would find that overall task decrement should be driven by individual decrement for either the insert or pick up task (whichever requires more time). If we assume these operators have decrements of about 25%, the time to perform the overall sequence would increase by about 25%. This of course does not match the empirical finding that performance is slowed by a factor of 2 to 3.

However, task overlapping may not be possible with protective gloves, because limited sensory input will prevent the tasks from being overlapped. Thus, slowing in this task may stem from a shift to a non-overlapping performance strategy necessitated by reduced sensory impact. The sequence would be stretched out, as shown in Figure 2.



Figure 2: Hypothesized sequence of goals to perform Purdue Pegboard task with protective gloves.

Now, each pick up/insert subgoal must be achieved serially, and each of those subcomponents may slow as well. A reasonable estimate for the slowing would be that the task time would double, plush each component should increase by 25%, producing an estimated performance impact of 2.5, (instead of the 1.25 estimated from each individual operation).

To assess the extent to which the two dexterity taxa can account for performance decrements, we applied the T3 method as described by Anno et al. (1996). To estimate the impact  $I$  for each task,  $\log(1/I)$  was computed, ensuring that all decrements would be negative numbers. Next, a linear regression model was fit to predict  $\log(1/I)$  based on the two performance taxa ("grasp" and "touch"), excluding the Purdue and touch typing tasks because they were thought to involve strategy shifts. The intercept of the model was set to 0, as an intercept would simply amount to a generic decrement for all tasks. This regression was reliable ( $F(2,14)=55$ ,  $p<.01$ ) with an adjusted  $R^2=.87$ . The two predictors were reliable  $p<.05$  (grasp=  $-0.04$ ,  $t(14)=-2.8$ ,  $p=.01$ ; touch=  $-.054$ ,  $t(14)=3.9$ ,  $p<.01$ ). These coefficient values indicate that each rating unit of the taxon reduces log-inverse-proportional performance by about .04-.05. Because for small values of  $p$ ,  $\exp(-p)$  approximates  $1+p$ , this means that each level of the rating scale slows performance by about 5%. Predicted performance values for each task are also printed in Table 1, along with the predictions for the two excluded task (shown in bold).

It should be noted that this method tends to underestimate the impact of those stressors with large decrements. The performance model described has a limited upper level, with log-inverse-proportion having a maximum decrement of about .45 (or 1.6). Most likely, to accommodate larger impacts, one must incorporate simple notions of strategy shifts (such as we argued for in the Purdue task), or costly error-recovery processes that are outside the linear model used in the T3 process. As a rough guide, in order to predict a decrement of 3.0, the Purdue task would need a touch value of about 22, which is well beyond the end of our scale.

#### 4. Discussion

The T3 method offers a simple statistical method for predicting coarse decrements across tasks in response to a number of stressors. Although predictions needing finer precision may require agent-based modeling with systems such as EPIC (e.g., Meyer et al., 2001, in the context of age-related stressors), we are developing ways to adapt the process to enable prediction for acute stressors related to MOPP gear, and involved with more perceptual, motor, and cognitive tasks. These adaptations take two forms. First, we are beginning to hypothesize new performance taxa that can be used to understand whether some task will see large decrements from protective gear. Second, we hypothesize that a more detailed task representation needs to be used, which can at least help identify whether a stressor will induce strategic shifts or costly error recovery processes.

We illustrated how these additional factors are important for extending the T3 method to the relatively simple domain of manual dexterity. In future and ongoing work, we are extending the method to tasks with stronger cognitive and perceptual components, which we believe will require similar additions to the T3 process.

#### 5. References

- Allender, Salvi, & Promisel (1997). Evaluation of Human Performance under Diverse Conditions.
- Anno, G. H., Dore, M. A., and Roth, T. J. (1996). Taxonomic model for performance degradation in combat tasks. (DNA-TR-95-115) Alexandria, VA: Defense Nuclear Agency.
- Bensel, C. K., Teixeira, R. A., & Kaplan, D. B. (1987). The Effects of US Army Chemical Protective Clothing on Speech Intelligibility, Visual Field, Body Mobility, and Psychomotor Coordination of Men (TR-87-037). Natick, MA: United States Army Natick Research, Development, and Engineering Center.
- Fleishman, E. A. (1975). Toward a Taxonomy of Human Performance. *American Psychologist*, 30(12), 1127-1149.
- Garrett, L., Jarboe, N., Patton, D. J., & Mullins, L. L. (2006). The Effects of Encapsulation on Dismounted Warrior Performance. Retrieved from <http://stinet.dtic.mil/oai/oai?&verb=getRecord&metadataPrefix=html&identifier=ADA448610>
- Gray, W. D., John, B. E., & Atwood, M. E. (1993). Project Ernestine: Validating a GOMS analysis for predicting and explaining real-world task performance. *Human-Computer Interaction*, 8(3), 237-309.
- John, B. E., & Kieras, D. E. (1994). The GOMS family of analysis techniques: Tools for design and evaluation. *Rappot technique CMUCHI-94*, 106.
- Johnson, R., & Kobrick, J. (1997). Effects of Wearing Chemical Protective Clothing on Rifle Marksmanship and on Sensory and Psychomotor Tasks. *Military Psychology*, 9(4), 301-314. doi: 10.1207/s15327876mp0904\_3
- McGinnis, J. S., Bensel, C. K., & Lockhar, J. M. (1973). Dexterity afforded by CB protective gloves. US Army Natick Laboratories, Natick, MA, Report.
- Meyer, D. E., Glass, J. M., Mueller, S. T., Seymour, T. L., & Kieras, D. E. (2001). Executive-process interactive control: A unified computational theory for answering 20 questions (and more) about cognitive ageing. *European Journal of Cognitive Psychology*, 13(1), 123-164.
- Mueller, S. T., Zimmerman, L. A., McClellan, G. E., Crary, D. J., & Cheng, K. (2009). Cognitive Performance Prediction with the T3 Methodology.

- Mueller S. T., Zimmerman, L. Cheng, K., Crary, D., Simpkins, B., Gabbard, S., and McClellan, G. (2009). Establishing a set of cognitive tests to predict the effect of novel protection suit design concepts. Chemical Biological Defense (CBD) Science and Technology Conference, Dallas, TX, November 2009.
- Schweickert, R., Fisher, D. L., & Proctor, R. W. (2003). Steps toward building mathematical and computer models from cognitive task analyses. *Human factors*, 45(1), 77.
- Teixeira, R. A., & Bense, C. K. (1990). The effects of chemical protective gloves and glove liners on manual dexterity (U). US Army Natick Research, Development and Engineering Center, Natick, Massachusetts, NATICK/TR/91/002.

## Author Biographies

**DR. SHANE MUELLER** is Senior Research Scientist at Applied Research Associates in Fairborn, OH. He specializes in measuring and modeling human behavior, with emphasis in decision making, human performance, and memory. He is the developer of the PEBL test battery for measuring psychology performance.

**GEORGE ANNO** is an independent consultant on this research effort. He is the originator of the T3 method, and specializes in modeling the impacts of chemical and biological stressors on human performance.

**COREY FALLON** is a Staff Research Scientist at Applied Research Associates in Fairborn, OH. He specializes in applied human factors, with an emphasis on understanding the impact of new technology on individual and team workflow.

**GEORGE ANNO** is an independent consultant on this research effort. He is the originator of the T3 method, and specializes in modeling the impacts of chemical and biological stressors on human performance.

**DR. GENE E. MCCLELLAN** is Principal Research Scientist at Applied Research Associates in Arlington, VA. He is Director of the Health Effects/Medical Response group, and has led efforts for estimating battle casualties from NBC attacks in support of U.S. and NATO medical defense planners, and was Program Manager for the development of the medical NBC Casualty and Resource Estimation Support Tool (NBC CREST) for medical planning.

**OWEN PRICE** is Senior Research Scientist at Applied Research Associates in Raleigh, NC, where he develops the Multiple-Path Particle Dosimetry (MPPD) model, and assists with modeling and software development for other projects in the Health Effects and Medical Response group.

## Prediction Intervals for Future Performance

Kelly M. Addis<sup>a</sup>  
Michael Krusmark<sup>a</sup>  
Tiffany S. Jastrzembski<sup>b</sup>  
Kevin A. Gluck<sup>b</sup>

<sup>a</sup>L-3 Communications at Air Force Research Laboratory

<sup>b</sup>Air Force Research Laboratory

6030 S. Kent St.

Mesa, AZ 85212

480-988-6561

[kelly.addis@mesa.afmc.af.mil](mailto:kelly.addis@mesa.afmc.af.mil), [michael.krusmark@mesa.afmc.af.mil](mailto:michael.krusmark@mesa.afmc.af.mil), [tiffany.jastrzembski@mesa.afmc.af.mil](mailto:tiffany.jastrzembski@mesa.afmc.af.mil),  
[kevin.gluck@mesa.afmc.af.mil](mailto:kevin.gluck@mesa.afmc.af.mil)

Stuart Rodgers  
AGS TechNet  
10887 Miriam Lane  
Dayton, OH 45458  
937-903-0558  
[stu@agstech.net](mailto:stu@agstech.net)

Keywords: mathematical modeling, prediction intervals, performance, learning

### 1. The Predictive Performance Optimizer

Building on more than a century of research on human memory and performance, the Predictive Performance Optimizer (PPO) is a state-of-the-art cognitive tool to help decision-makers, instructors, and learners of all types to assess current performance and predict future performance by capturing the dynamics of human learning with basic cognitive science principles.

The PPO is a user-friendly software tool that can track performance over the course of a learner's training history for virtually any quantitative measure of performance. It generates performance predictions at specified future points in time, and allows users to visually and graphically assess and compare the impact of potential future training regimens. The PPO accomplishes this by utilizing a mathematical model for performance prediction (shown in Equation 1.1 below) inspired by the General Performance Equation (Anderson & Schunn, 2000).

$$\text{Performance} = S * St * N^c * T^d \quad (\text{Equation 1.1}).$$

It comprises three main parts: the power law of learning ( $N^c$ ), the power law of forgetting ( $T^d$ ), and a stability term ( $St$ ) which captures the effects of practice and retention as they are spaced over time. The combination of these terms, along with a scaling factor ( $S$ ), produces point predictions of future performance based on mathematical regularities in the learner's historical performance (for additional details, see Jastrzembski et al., 2009).

A major intended use of the PPO is to provide instructors and trainers with principled guidance concerning the

readiness of their trainees. We will now frame PPO's practical relevance into a "just-in-time" training refresher scenario. Consider a training manager attempting to gauge how much training a warfighter must receive to ensure performance at or above a specified level of effectiveness before he may be deployed. The training manager may load the warfighter's unique training history into PPO to generate point predictions of future performance. The training manager can then assess whether adjustments must be made to the future training routine to meet the desired training goals.

Given the variability in human performance, generation of pure point predictions is insufficient in helping training managers make critical training decisions. One can imagine a scenario where a point prediction is at or very close to the effectiveness standard. Should the training regimen be deemed sufficient in that case? Is additional training heeded? Can we be confident that the performer will achieve that level of effectiveness at all? It is therefore necessary to provide training managers with scientifically-principled estimates of risk around the model's point predictions, to better guide decisions that have an impact in the real-world. We now turn to a discussion concerning how best to compute a prediction interval (PI) around the model's point predictions.

### 2. Prediction Interval Calculations

Rather than discrete point predictions, PIs provide a range of possible values of future performance, and thus offer the trainer a more complete picture of what outcomes future training regimens may possess. Identifying a method to compute a principled PI for our needs, however, is far from straightforward.

One issue we face is that we must balance two interacting effects: on one hand, human performance generally becomes less variable with increased practice; on the other hand, model predictions generally become less certain with longer lead times. A second issue is the limited existing data with which to validate the model's extended predictions. Related fields, such as economics, typically possess data spanning months or years, but few psychological studies examine data across time scales longer than a few days. A third problem is that there is little in the psychological literature which focuses on predicting performance at future times, and within that research, the incorporation of PIs on future performance is almost entirely absent. Thus, we lack sufficient exemplars to directly apply any one methodology to our situation, and have turned to other disciplines (e.g., econometrics and biostatistics), whose application to our situation is less straightforward, for guidance as a result. A final hurdle is maintaining the generality of the model. The model is intended to be used for predicting performance in a wide range of areas, and thus a large range of dependent variables. Accordingly, any methodology to compute PIs must not make mathematical assumptions that cannot be met with most measures of performance.

One method commonly used to generate PIs is the incorporation of a noise parameter into one or more parts of the model. In a computational model, this can be relatively straightforward, and the ACT-R framework has several extant noise parameters that can be utilized in a variety of situations. In our mathematical implementation, however, it is less obvious how to add in a noise parameter. As such, we are evaluating which terms in our mathematical model have a strong theoretical motivation to vary, and how these terms might interact with one another. For example, the learning rate and/or the forgetting rate might vary from one training session to the next based on fluctuations in the attentiveness of the warfighter or variability in the quality of the information in the briefing before the training session begins. However, one still has to determine the form and magnitude of the distribution from which to sample the noise. For this, we are investigating measures of variability in model fits to observed data that may be used to estimate the variability expected in future data.

The resulting PIs from this method, or any similar method, on predicted future performance provide an important tool for trainers and decision-makers by presenting a range of likely values for future performance. In our warfighter scenario, the training manager may decide to adopt a conservative criterion and use the worst likely performance shown by the PIs as a guide to impact future training needs. Such a criterion would ensure that the warfighter is most likely to

actually perform at or above the desired level of effectiveness.

### 3. Summary

The question of how to properly calculate PIs for a mathematical model of performance and learning is a challenging one. The existing psychological literature offers little insight. We are, however, investigating a number of promising methods from related fields. Specifically, implementing noise in the model to generate variability is one of several promising possibilities. The development of an elegant method for calculating PIs for psychological performance data would hopefully encourage widespread use of such intervals as opposed to simple point predictions which inherently have unspecified certainty in their precise value. Our poster will present results from our ongoing explorations of these methods.

### 4. References

- Anderson, J.R., & Schunn, C. D. (2000). Implications of the ACT-R learning theory: No magic bullets. In R. Glaser (Ed.), *Advances in instructional psychology: Educational design and cognitive science (Vol. 5)*, (pp. 1-34). Mahwah, NJ: Erlbaum.
- Jastrzemski, T.S., Gluck, K.A., & Rodgers, S. (2009). Improving Military Readiness: A State-of-the-Art Cognitive Tool to Predict Performance and Optimize Training Effectiveness. In *Proceedings of IITSEC 2009*. Orlando, FL: NTSA.

### Author Biographies

**KELLY M. ADDIS** is a Research Scientist with L-3 Communications at the Air Force Research Laboratory with a background in modeling of human memory and learning, and collaborator on the PPO project.

**MICHAEL KRUSMARK** is a Research Scientist with L-3 Communications at the Air Force Research Laboratory and collaborator on the PPO project.

**TIFFANY S. JASTRZEMBSKI** is a Research Psychologist at the Air Force Research Laboratory focused on developing the mathematical model in PPO to capture the dynamics of human memory.

**KEVIN A. GLUCK** is a Senior Research Psychologist at the Air Force Research Laboratory and enthusiastic collaborator on the PPO project.

**STUART RODGERS** is a computer scientist focused on implementing cognitive models of human performance and other adaptive, reactive, and autonomous software systems. He is Director at AGS TechNet, Dayton, OH.

# Subjective Logic for Composing Utility Functions from Maslow Models

Nathan T. Denny  
21<sup>st</sup> Century Systems, Inc.  
6825 Pine St.  
Omaha, NE 68105  
402-502-8439  
nathan.denny@21csi.com

## Keywords:

social science, utility functions, graphical models

**ABSTRACT:** *Researchers in the social sciences often collaborate with software developers to create agent-based simulations that are increasingly used in the study of sociology, political science, economics, etc. Maslow is a nascent, graphical (network or connectionist) modeling language that aims to make the modeling of motivation more intuitive to the social scientist and facilitate the translation of simulation specifications into executable code. This paper builds upon the Maslow language, illustrating how subjective logic can be used as a means to represent influence between elements in a Maslow model. So constructed, an acyclic Maslow model can be expressed as a subjective logic expression which in turn can be compiled into executable code. The result is a model that can represent motivations with arbitrary detail that is also computationally efficient. The detail and scalability of this approach may be of particular interest in multi-agent simulations of large groups, where a good degree of modeling fidelity can be achieved with relatively little impact in computing performance.*

## 1 Introduction

Agent-based simulations (ABS) have broad applicability and can be applied to modeling teams of robots, the spread of infectious diseases, and even entire ecosystems. ABS has found increasing use in the study of sociology and economics where researchers can simulate organizational behavior, market exchanges, and other social interactions to study the emergence of macro characteristics from micro entities. In the present context, these micro entities are behavioral models that are proxies for real human behavior.

As has been noted (see. Iba, 2004; da Silva and de Melo, 2008; Rixon, Moglia, and Burn, 2005), ABS simulations are not always easy to develop. Available simulation platforms typically require some degree of technical ability in order to implement simulations using what is often (e.g. Java) a general purpose programming language. Social scientists must either acquire the necessary technical skills themselves or collaborate with software developers that already possess the technical know-how. Both options can be prohibitive and costly.

For those social scientists that do their own software development, re-use of previous models is enticing (Newell, 1990). Indeed, the software engineering community seems to be able to deliver, to some degree, on its long promise of object and component reuse. However, this has only come about after many years of incremental accumulation of intellectual capital, accreting into software libraries and frameworks. By

comparison, ABS simulations are too new and too few to have built up enough intellectual property and most ABS studies build their models from scratch with highly-domain specific agents.

The division of effort between social scientist and software developer is an efficient use of resources, but is not without difficulties. In particular, describing a behavioral model at a granularity that is easily understood by both the social scientist and the software developer may not be trivial. Furthermore, the description should outlive the lifetime of the study, thereby promoting model re-use in later studies.

Maslow is a nascent, graphical (network or connectionist) modeling language that aims to make the modeling of motivation more intuitive to the social scientist and facilitate the translation of simulation specifications into executable code. This paper builds upon the Maslow language, illustrating how subjective logic can be used as a means to represent influence between elements in a Maslow model. So constructed, an acyclic Maslow model can be expressed as a subjective logic expression which in turn can be compiled into executable code. The result is a model that can represent motivations with arbitrary detail that is also computationally efficient. The detail and scalability of this approach may be of particular interest in multi-agent simulations of large groups, where a good degree of modeling fidelity can be achieved with relatively little impact in computing performance.

## 2 Maslow

There are certain elements of the human experience which seem to be common. For instance, at the most basic level, all humans need air, water, and food. However, the common aspects of human experience seem to extend far beyond individual subsistence. Many psychological theories have been advanced which aim to capture common human values, ambitions, and actions. Maslow's Hierarchy (Figure 2.1) (Maslow, 1943) is a classic example of such theories (and the inspiration for the name of the language presented here). Alderfer's Existence, Relatedness, and Growth (ERG) (Alderfer, 1972) builds on Maslow's earlier work and replaces the original hierarchy with a parallel relationship between the three dimensions he identifies.

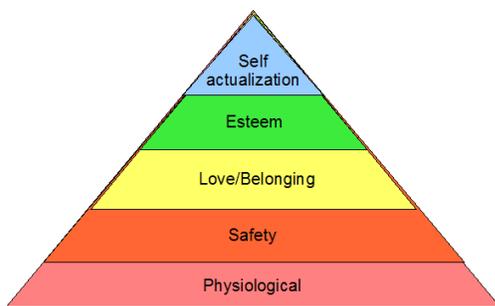


Figure 2.1 Maslow's Hierarchy

Whereas Maslow and Alderfer have advanced psychological models, sociology has also attempted to advance theories of human motivation. For instance, the Fundamental Human Needs identified by Max-Neef, et al (1989) propose that human motivation is described across nine dimensions: subsistence, protection, affection, understanding, participation, leisure, creation, identity, and freedom. Similar in some respects is the work of Nussbaum and Sen (1993) where human welfare (and motivation) is described in terms of capabilities and the ability to move from capability towards actuality. Recent work by The World Bank (Alkire, 2002) considers the possibility of unifying the sociologically inspired theories into a usable metric of human welfare

The human brain has a nearly universal structure, with the location of specialized functions found in more-or-less the same relative locations across individuals. This lends support to the concept of a universal cognitive architecture that can model human cognition. A consequence of both universal structure and universal cognitive architecture is the existence of a universal architecture of human utility functions. Although Abraham Maslow did not describe his work as such, his eponymous hierarchy reflects such a universal architecture of human utility.

*Maslow* (Denny, 2009) is a simple, graphical language which is intended to model human motivation in much the same way that the Unified Modeling Language (Rumbaugh, Jacobson, and Booch, 1999) describes software architecture. The *Maslow* graphical language is composed of four elements (Figure 2.2) which are called welfare, aspect, stimulus, and action. Each model must have one and only one welfare (Figure 2.2-a) node. This node represents the overall utility state of the agent. Welfare nodes are a special case of the more general aspect nodes. An aspect node (Figure 2.2-b) represents some component of the overall welfare and can be arbitrarily decomposed. Stimulus nodes (Figure 2.2-c) embody conditions and procedures that influence an aspect of an agent's welfare. Action nodes (Figure 2.2-d) represent alternative courses of action that will positively affect the associated aspect. In building a model, each instantiated element is given a short name and a sufficient description to convey the function of the instantiated node.

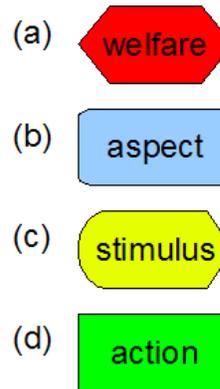


Figure 2.2 Maslow elements

In general, stimulus nodes decrease utility and executed actions increase utility. Note that a planning arc represents a belief on the part of the agent that executing the associated action will in some way improve the condition of the associated aspects. *Maslow* makes no assumptions about the actual outcome of the action and implementations of the action are not constrained to producing positive results.

The grammar of directed influential connections is straightforward. Decomposition arcs denote aggregation or subsumption and can connect an aspect node to one or more aspect nodes or to the root welfare node. Affecting arcs connect a stimulus node to one or more aspect nodes. Planning arcs are placed in order to denote an association between an action node and one or more welfare nodes.

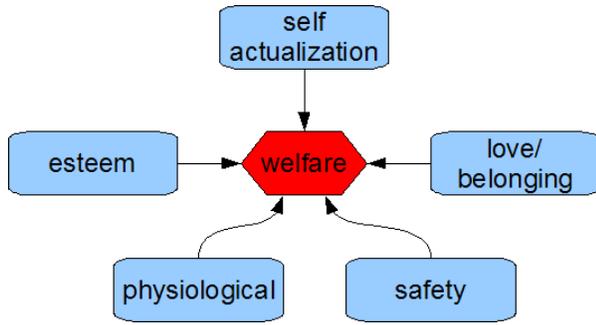


Figure 2.3 Aspects of Maslow's Hierarchy

Figure 2.3 shows the top level welfare and aspect nodes used to represent the components of Maslow's hierarchy. There is an explicit ordering in the hierarchy which implies a utility function over the satisfaction of the components of the hierarchy. *Maslow* does not explicitly represent such a utility function but utility functions are implicit to the specific fusion algorithms used in the aspect nodes and the heuristics implemented to select potential actions. An agent's overall utility function is an emergent phenomenon resulting from the interactions between the states of stimulus processes and aspect fusion.

### 3 Overview of Subjective Logic

Subjective Logic (Josang, 1997, 2009) is a type of probabilistic logic that is often used in evidential reasoning (e.g. Lindahl and Petrov, 2007 and Lindahl and Zhu, 2007) where belief, disbelief, and uncertainty must be explicitly and simultaneously accounted. Before discussing the method by which Subjective Logic can be used to compose utility functions, a brief introduction to Subjective Logic and a summary of the relevant algebraic operations is in order.

In contrast to systems described by Boolean Logic, for those systems described by Subjective Logic the basic object is an opinion rather than a fact. An *opinion*  $\omega^A(x)$  about some proposition "x" held by source "A" is a 4-tuple of the belief ( $b_x^A$ ), disbelief ( $d_x^A$ ), uncertainty ( $u_x^A$ ), and relative atomicity ( $a_x^A$ ). (Atomicity is the base-rate of the proposition.) Note that  $b_x + d_x + u_x = 1$ , so while it is not necessary to specify all three of the values, it is convenient when performing certain calculations.

The Subjective Logic algebra provides an array of operations that manipulate opinions. These operators have many applications in evidential reasoning and data fusion. For the present purpose, only the consensus and discount operators are of interest.

The consensus operator (written as  $\oplus$ ) is used for belief fusion, providing the capability to fuse possibly conflicting opinions while still forming coherent,

summary judgments. The underlying calculations on the belief tuple elements are given in Figure 3.1.

$$\begin{aligned}
 K &= u_x^A + u_x^B - u_x^A u_x^B \\
 b_x^{A,B} &= \frac{b_x^A u_x^B + b_x^B u_x^A}{K} \\
 d_x^{A,B} &= \frac{d_x^A u_x^B + d_x^B u_x^A}{K} \\
 u_x^{A,B} &= \frac{u_x^A u_x^B}{K} \\
 a_x^{A,B} &= \frac{a_x^A u_x^B + a_x^B u_x^A - (a_x^A + a_x^B) u_x^A u_x^B}{K - u_x^A u_x^B}
 \end{aligned}$$

Figure 3.1: Subjective Logic Consensus Operation

Subjective logic also provides a well developed "discount" operation (written as  $\otimes$ ) that can be used for modifying the contribution of evidence based upon a subjective measure of confidence in the source of the evidence. The discount operator thus provides a rather general means of describing degrees influence and can be used to represent semantic similarity, relevance, trust, etc. The calculations for implementing a discount operator over belief tuples is shown in Figure 3.2.

$$\begin{aligned}
 b_x^{A,B} &= b_B^A b_x^B \\
 d_x^{A,B} &= b_B^A d_x^B \\
 u_x^{A,B} &= d_B^A + u_B^A + b_B^A u_x^B \\
 a_x^{A,B} &= a_x^B
 \end{aligned}$$

Figure 3.2: Subjective Logic Discount Operation

### 4 Composing Utility Functions

As a modeling tool, Maslow is predicated upon Rational Choice Theory (see Allingham, 2002). That is, agents have a utility function and reason and act to maximize the utility function. Although Rational Choice Theory is sometimes derided as too simple a model of human behavior, most of the criticisms of simplicity are well addressed by Bounded Rationality (e.g. Simon, 1957).

The welfare, aspect, and stimulus nodes of an executing *Maslow* model are essentially the component variables of a utility function. The welfare node is the ultimate dependent variable and contains the present, summarized utility state of the agent. Stimulus nodes contain the state of external stimuli. Aspect nodes are intermediate variables that are calculated as a function of other aspects and affecting stimuli. Both decomposition arcs (between aspect nodes) and affecting arcs (from stimulus to aspect) carry a measure of influence that is defined over the range [0,1.0].

An example model is shown in Figure 4.1 where a subgraph of a Maslow model focuses on the influence of claustrophobia on welfare. The color fill in the boxes next to each arc represent the degree of influence that propagates along the arc. The agent in Figure X is highly sensitive to claustrophobia. The same structure is re-used in Figure 4.2, with another agent that is relatively insensitive to claustrophobia.

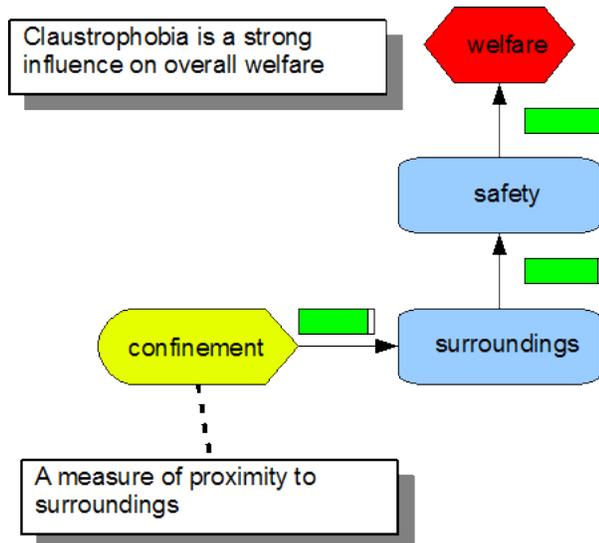


Figure 4.1. Claustrophobic agent

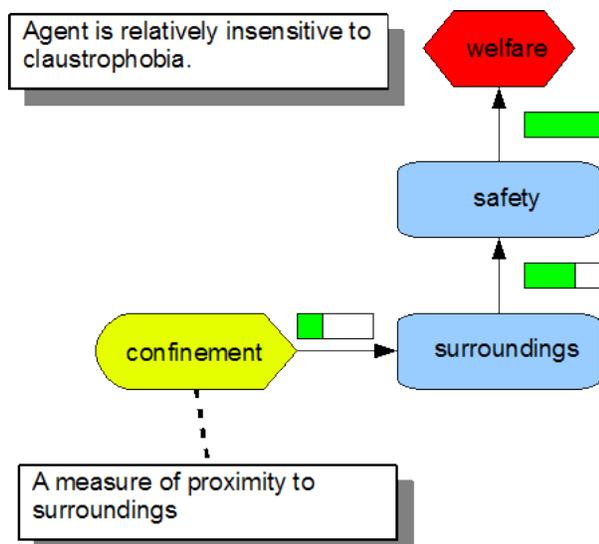


Figure 4.2. Agent is little affected by claustrophobia

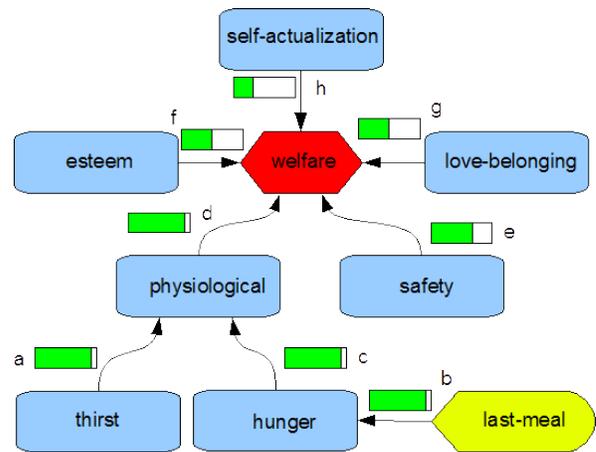


Figure 4.3. Propagating influence

The process of generating a computable utility function from a Maslow model is relatively straightforward: aspect and stimulus nodes are treated as opinions while decomposition and affecting arcs act as discounts on propagated influence. The compiler would then traverse the model in topological order (working from the exterior nodes to the interior nodes) and generate an infix expression of the graph. For example, the physiological contribution of the model shown in Figure 4.3 can be represented algebraically as:

$$(\oplus (\otimes (\oplus (\otimes \text{last-meal } b) \text{hunger } c) (\otimes \text{thirst } a))$$

Before executing the model, the infix expression would first be compiled to byte-code or machine code for efficient evaluation. This latter characteristic is of particular importance when running simulations of large groups where demands on computing resources can be severe.

When the model is executed at run-time, aspect and stimulus nodes are stateful and hold the default vacuous opinion where all belief mass is allocated to uncertainty. As stimuli act on the model, the influence from the stimuli propagates through the network of aspect nodes, changing their state and ultimately influencing whatever reasoning engine is employed for the agent. As Subjective Logic is not yet widely supported in reasoning engines, the Subjective Logic expectation function is a simple and convenient function for mapping from a 4-tuple belief vectors to the more common representation of belief as a scalar in the range of [0, 1.0]. (The expectation function loses information and should only be used on the result taken from the welfare node.)

## 5 Conclusions and Future Work

To date, Maslow has remained ambiguous on how influence was to be propagated from stimulus through aspects to the overall welfare of the agent. Although Subjective Logic was developed for evidential reasoning, there is an intuitive similarity between

evidence and influence and the algebra of Subjective Logic lends itself for use in composing functions from relatively distinct components. Given an acyclic Maslow model, the model can be assembled into an infix expression in subjective logic and can then be further compiled into byte-code or machine-code that can be efficiently executed at run-time.

Maslow is still in its infancy and undergoing gradual improvement. Maslow remains agnostic to the reasoning mechanism, but this may need to be changed given commitments that the model is now assuming. Furthermore, the method of composing utility functions that has been described here represents only the instantaneous utility. For a higher-fidelity model, the language and framework must be amended to include something akin to the inertia that individuals often have in their emotional (the surface manifestation of welfare) states.

## 6 References

- Alderfer, C. (1972) *Existence, Relatedness and Growth*. The Free Press: New York.
- Alkire, S. (2002) Dimensions of Human Development, *World Development* 30(2) pp. 181-205, 2002.
- Allingham, M. (2002) *Choice Theory: A Very Short Introduction*, Oxford University Press.
- Denny, N.T. (2009) Maslow: A Graphical Framework for Communicating Models of Human Behavior. In *Proceedings of the 19<sup>th</sup> Conference on Behavior Representation in Modeling and Simulation (BRIMS)*.
- Iba, T. (2004) A Framework and Tools for Modeling and Simulating Societies as Evolutionary Complex Systems, In *Proceedings of the 2<sup>nd</sup> International Conference of the European Social Simulation Association*.
- Josang, A. (1997) Artificial Reasoning with Subjective Logic. In *Proceedings of the 2<sup>nd</sup> Australian Workshop on Commonsense Reasoning*.
- Josang, A. (2009) Conditional Reasoning with Subjective Logic. *Journal of Multiple-Valued Logic and Soft Computing*, 15(1).
- Lindahl, E.R. and Petrov, P. (2007) Situation Awareness and Dynamic Group Formation of Agents and Humans Using Jabber. In *Proceedings of the 12<sup>th</sup> International Command and Control Research and Technology Symposium (12<sup>th</sup> ICCRTS)*, 2007.
- Lindahl, E.R. and Zhu, Q. Epistemic Belief Frames in Distributed Effects-based Reasoning. In *Proceedings of the International Conference on Integration of Knowledge Intensive Multi-Agent Systems (KIMAS 2007)*, 2007.
- Maslow, A.H. (1943) A Theory of Human Motivation, *Psychological Review* 50 pp.370-396.
- Max-Neef, M. et al. (1989) Human Scale Development: An Option for the Future, *Development Dialogue*, 5-81.
- Newell, A. (1990) *Unified Theories of Cognition*. Cambridge, MA: Harvard University Press.
- Nussbaum, M. and Sen, A. (1993) *The Quality of Life*, Oxford: Clarendon Press.
- Rixon, A., Moglia, M., and Burn, S. (2005) Bottom-up Approaches to Building Agent-based Models: Discussing the Need for a Platform, In *Proceedings of the Joint Conference on Multi-agent Modeling for Environmental Management*.
- Rumbaugh, J., Jacobson, I., and Booch, G. (1999) *The Unified Modeling Language Reference Manual*. Addison-Wesley.
- Simon, H. (1957) A Behavioral Model of Rational Choice, in *Models of Man, Social and Rational: Mathematical Essays on Rational Human Behavior in a Social Setting*. New York: Wiley.
- da Silva, P.S. and De Melo, A.C.V. (2008) Reusing Models in Multi-Agent Simulation with Software Components, In *Proceedings of the 7<sup>th</sup> International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*.

## Author Biography

**NATHAN T. DENNY** (*Scientist, 21<sup>st</sup> Century Systems, Inc.*) received his Master of Science in Computer Science from Southern Illinois University in 1998 and his Bachelors of Science in Computer Science and Mathematics from Southern Illinois University in 1997. Mr. Denny is also pursuing a PhD in Computer Engineering from the University of Arizona. He has published in diverse fields such as design for testability (DFT) of very large scale integrated circuits (VLSI), information re-use in case-based reasoning, Internet spam control, automation in agricultural irrigation, peer-to-peer networking and knowledge management in global software development. His current research interests include distributed computing, artificial intelligence, human-machine interfaces, cognitive science, knowledge management, and agile software development in the 24-Hour Knowledge Factory. Mr. Denny assists with research and development at the Omaha office of 21<sup>st</sup> Century Systems, Inc.

## Human Behavior Modeling in Network Science

### Panel Chair

*Sibel Adali*  
110 8<sup>th</sup> Street, Troy, NY  
Rensselaer Polytechnic Institute  
518-276-8047  
sibel@cs.rpi.edu

### Panel Chair

*Jeffrey T. Hansberger*  
115 Lake View Parkway, Suffolk, VA  
Army Research Laboratory  
757-203-3431  
jeff.hansberger@us.army.mil

The U.S. Army Research Laboratory (ARL) has begun a 5-10 year research program with the Network Science Collaborative Technology Alliance (NS CTA) in Network Science bringing three distinct research areas together, communication networks, information networks, and social/cognitive networks. The NS CTA is an alliance across a wide range of academic and industry researchers working collaboratively with ARL and the Department of Defense researchers.

A critical part of the social/cognitive network effort is the modeling of human behavior. The modeling efforts range from organizational behavior to social cognitive trust to explore and refine the theoretical and applied network relationships between and among the human, information, and artifacts used.

The participants are:

**William Wallace** – Rensselaer Poly. Inst.  
**Wayne Gray** – Rensselaer Poly. Inst.  
**Ching-Yung Lim** – IBM  
**David Hachen** – Notre Dame University

The participants will describe ongoing research in how information is transmitted along trusted paths both in case of emergency warnings and in an organizational setting, patterns of reciprocity in social communications and cognitive components of human behavior in social interactions.

### **Emergency Warnings: A Case of Diffusion of Information on Dynamic Networks, W.A. Wallace**

This presentation will discuss ongoing research concerned with warning messages in evacuation situations. We propose a model for studying the diffusion of evacuation warning messages through a population where the network dynamics are a function of the information flow. In evacuation situations, individuals in the network leave the network when they decide to evacuate, causing disruptions to the flow of information as warnings are still being diffused through the network. Propagation of the messages is based upon the interaction of agents in the network and includes consideration of the trust between them. When individual nodes receive a warning message, they often do not immediately take the prescribed action. Instead,

they will seek information, converge with others, and try to make a decision. Individual nodes can fall in to one of several states, depending on their perception of the information they have received. Depending on their state, the individual nodes will perform certain actions, such as spread information or evacuate and leave the network. We use the model to examine how social group structure, distribution of trust, and existence of weak ties affect the spread of evacuation warnings. Preliminary results from simulation experiments show that effectiveness of the diffusion process depends upon trust and social groups, and the structure of the network.

### **Markovian Information Propagation Behavior Modeling in Dynamic and Probabilistic Social Networks, Ching-Yung Lim**

While most existing social network research focus on finding and modeling the structure of social network graph topologies, we consider the dynamic topology of a network obtained from observation, instead of being modeled as a random graph. Because of the well-known small world phenomenon, small changes in edges can significantly alter the network topology, information propagation speed, etc. We consider the exact modeling of the behavior of each actor nodes as well as the relationships. We propose a novel Behavioral Information Flow (BIF) model which can be used to predict how information is propagated through a complex social network. We consider both the dynamic and probabilistic characteristics of human behavior in receiving and redirecting information. A significant difference between this model to the traditional random walk-based propagation model is that information is considered duplicable at nodes and thus the way information propagation does not really follow the entity-based 'walks' behavior. We first modeled Dynamic Probabilistic Social Network as a combination of the state probabilities of user nodes and connection edges and two transition functions that are dependent on the network topology and user properties. Then, we propose to model user transitions as Susceptible-Active-Informed (SAI) states and edge transitions as a Markov Model with Susceptible-Dormant-Active-Removed (SDAR) stages. Based on these modeling methods, we can then predict information flows in a social network. We have

deployed a real system in a big organization to collect 20 million of emails and instant messages from 10,000 users to examine this network-based behavior predictability issue.

### **The Evolution of Dyadic Reciprocity in Social Networks, David Hachen**

Dyadic reciprocity is an important dimension of social networks that is in all likelihood related to trust. Reciprocity is conceptualized as the degree to which the directional flows of social interaction (including information flows) between two nodes are more or less balanced. We expect that most new social ties begin in a more non-reciprocal (unbalanced) state, with one agent initiating interaction more often than the other agent. We also expect that if the tie is to persist, then the dyadic relationship will have to become more balanced. The central research question then concerns what factors predict which new non-reciprocal ties are more likely to become reciprocal over time and, therefore, persist. We test two different hypothesis about the evolution of reciprocity. According to the Social Distance Hypothesis, the more similar the nodes in a dyad are, the more balanced the dyad will become over time. Nodal similarity/difference can be measured in numerous ways: sex, age, social status, physical distance, nodal degree. The Embeddedness Hypothesis expects that the more neighbors two nodes have in common, the more balanced the dyad will become. Using cell phone network data on the calling patterns of over 9 million subscribers of a cellular telephone company we identify who communicates with who within a given time period and among those dyads calculate how often each node initiates communication. Then we measure whether the tie persists in subsequent time periods and if so the extent to which both the level of interaction and reciprocity between the nodes changes. Hazard rate and machine learning models are used to predict tie persistence,

while growth models are used to test hypothesis about the factors associated with increases in reciprocity.

### **Reductionism, Constructivism, Networks, and Cognitive Science, Wayne Gray**

In his 1971 *Science* article, *More is Different*, the Nobel Laureate physicist, Phillip W. Anderson maintained that the generally accepted reductionist hypothesis does not imply a constructionist one. That is, “the ability to reduce everything to simple fundamental laws does not imply the ability to start from those laws and reconstruct the universe”. For example, the elementary entities of cognitive science obey the laws of neuroscience but cognitive science has its own laws that cannot be “constructed” out of neuroscience. Likewise, the elementary entities of social psychology obey the laws of cognitive science but social science has its own laws that cannot be constructed out of the laws of cognitive science. Behavior at each level is an emergent function of the structure of the network and the behavior of its component parts. To make all of this more difficult, the network’s structure is dynamic in that it changes as a function of the behavior of its elements and the elements in the network are dynamic in that their behaviors also change as a function of the network’s structure. The good news is that the new science of networks promises to provide formal mechanisms by which to study this complex process. It also suggests a new paradigm for behavioral and social science in that research focused on one level must be informed by knowledge of the lower and higher levels. For example, basic research on the laws of cognitive science requires an understanding of the range in performance exhibited by individual cognitive components as parts of a network that produces social interactions, but also requires an understanding of the behavior of the neurocognitive elements underlying each cognitive component.

# Evaluating Behavior Modeling Toolsets

*Jennifer Pagan*  
Kaegan Corporation  
12350 Research Parkway  
Orlando, FL 32826  
jennifer.pagan1.ctr@navy.mil

*Beth F. Wheeler Atkinson*  
*Melissa M. Walwanis Nelson*  
Naval Air Warfare Center Training Systems Division (NAWCTSD)  
12350 Research Parkway  
Orlando, FL 32826  
beth.atkinson@navy.mil, melissa.walwanis@navy.mil

## 1. Introduction

Simulation-based training is increasingly important in Navy training. However, replicating real-world environments has inherent challenges such as the necessity to provide realistic human behaviors in the simulated environment. One solution is to use human role-players for friendly and enemy forces. However, using role-players is costly in terms of money used to hire outside contractors, operational time foregone by volunteer role-players, and the added equipment for role-players. Semi-Automated Forces (SAFs) provide a less costly alternative to replicating friendly, enemy, and neutral platforms in the virtual environment. They are controlled and monitored by a human that pre-scripts command processes (Department of Defense, 1998). Although SAFs decrease the costs associated with using human-role players, the pre-scripted nature of their behaviors presents some inherent challenges. This paper provides an overview of the current state-of-the-art in human behavior modeling and outlines remaining challenges. The authors then provide a practical framework for evaluating rapid human behavioral modeling toolsets to overcome the presented challenges.

## 2. Challenges of Pre-scripted Behaviors

While SAF behavior significantly contributes to the realism of training scenarios, limited behaviors provide an unrealistic situation that may hinder training transfer (Gelenbe, Hussain, & Kaptan, 2005). This lack of realism is often because SAFs must be scripted prior to the training event. For this reason, many mission variations are preprogrammed to facilitate realistic tactical behaviors. Further, some training scenarios require thousands of SAF entities that must be pre-scripted to successfully execute training. However, pre-scripting this many entities with several behavioral variations is

impractical due to time constraints and increased manpower requirements (Cox & Fu, 2005).

Even when SAF entities are scripted with few behavior variations, scripting large numbers of SAFs in short periods of time also presents challenges. There is often an increase in manpower to support scenario generation, (albeit, less than using live role-players) and instructors work long hours to ensure that training events are kept on schedule. Increased work hours contribute to cognitive fatigue and thus could limit the quality of training provided by an instructor (Whelan, Loftus, Perme, & Baldwin, 2002). Finally, as large scale simulation-based training events become more common and increase in scale, additional instructors are required to monitor SAF behaviors, causing training costs to increase (Furness & Tyler, 2001).

## 3. Behavior Modeling Evaluation

The previously mentioned challenges to SAFs limit fidelity and increase costs, showing a need to practically evaluate current human behavior modeling toolsets in a manner that can help overcome these challenges. A review of current behavior modeling technologies indicates two prominent technical approaches for creating more realistic SAFs: algorithms and hierarchies. While algorithmic approaches use behavioral instances to capture demonstrated behaviors, hierarchal approaches decompose high level tasks or goals into primitives to elicit behaviors. Both approaches of behavior modeling have shown to be effective methods of producing more realistic behaviors (Banks & Stytz, 2003). While these approaches are effective means of modeling realistic behavior, toolsets using these approaches should be evaluated on several criteria to practically increase Return on Investment and drive future scientific inquiry.

We have developed a behavior modeling toolset evaluation framework which can be divided into three categories: cost, schedule, and performance. Each category has its own set of evaluation criteria.

### 3.1 Cost

The cost category is broken into three criteria thought to reduce the cost of implementing a behavior modeling toolset. The three evaluation criteria are:

- 1) *Domain Independence*. Can entities be reused in a variety of training scenarios and simulations regardless of developmental domain?
- 2) *Technology Readiness Level (TRL)*. What is the level of maturity of the technology?
- 3) *Resource Requirements*. How much funding is required to increase product maturity?

### 3.2 Schedule

The time category consists of one criterion:

- 1) *Rapid Scripting Capabilities*. Can the toolset rapidly script entity behaviors?

### 3.3 Performance

The performance category is focused on the actual performance of the entity or toolset, and consists of two components:

- 1) *Autonomy*. Does the toolset reduce the manpower required to monitor entities?
- 2) *Communication Capability*. Does the toolset support more realistic interaction with entities?

## 4. Benefits

There are numerous anticipated benefits of evaluating toolsets using this framework. First, training fidelity and transfer are expected to increase, as rapid scripting reduces the time necessary to produce more behavior variations than current SAFs provide. Communication capabilities can also enhance realism by allowing the trainee to simulate communication with entities (Furness & Tyler, 2001). Next, manpower requirements are expected to decrease as the reuse of behavior models in various training scenarios and simulations reduces scenario generation time. The production of autonomous entities is expected to further reduce manpower costs by reducing monitoring requirements. Costs are further reduced by selecting toolsets that have higher TRLs and fewer resource requirements. Finally, reduction in scenario generation time and monitoring requirements can also alleviate the cognitive strain placed on instructors allowing them to focus on other aspects of the training scenario, such as performance measurement.

**Authors' Note.** The views expressed herein are those of the authors and do not necessarily reflect the official position of the organizations with which they are affiliated.

## 5. References

- Banks, S., & Stytz, M. (2003). Progress and prospects for the development of computer-generated actors for military simulation: Part 2 – Reasoning system architectures and human behavior modeling. *Presence*, 12(4), 422-436.
- Cox, C., & Fu, D. (2005). AI for automated combatants in a training application. *Proceedings of the second Australasian conference on Interactive entertainment*.
- Department of Defense (1998). *Department of Defense Modeling and Simulation Glossary*, Manual No. 5000.59-M., Washington DC.
- Furness, Z., & Tyler, J. (2001). Fully automated simulation forces (FAFs): A grand challenge for military training. *Proceedings of the 2001 Fall Simulation Interoperability Workshop*.
- Gelenbe, E., Hussain, K., & Kaptan, V. (2005). Simulating autonomous agents in augmented reality. *The Journal of Stems and Software*, 74, 255-268.
- Whelan, M., Loftus, J., Perme, D., & Baldwin, R. (2002). Reducing training costs through integration of simulations, C4I systems, and expert systems. *Proceedings of the 2002 Winter Simulation Conference*.

## Authors' Biographies

**Jennifer Pagan** is a Research Psychology Assistant at Kaegan Corporation, Orlando FL. She has an M.S. degree in Industrial/Organizational (I/O) Psychology from the University of Central Florida.

**Beth F. Wheeler Atkinson** is a Research Psychologist at NAWCTSD. Her research interests include human-computer interaction, instructional tools, and distributed mission training. She holds a M.A. in General Psychology from the University of West Florida.

**Melissa M. Walwanis Nelson** is a Senior Research Psychologist at NAWCTSD. She has a M.S. degree in I/O Psychology from the University of Central Florida. Her research interests include distributed mission training, leadership development, simulator instructional tools, and network centric warfare concepts for coalitions.

## Policy Analysis using $Q$ -learning

Ceyhan Eksin

University of Pennsylvania

Ackoff Collaboratory for Advancement of the Systems Approach(ACASA)

Department of Electrical and Systems Engineering

220 South 33rd Street

Philadelphia, PA 19104-6315

ceksin@seas.upenn.edu

**ABSTRACT:** *Building of complex white box models brings the need to build tools that guide validation, verification and analysis processes. The goal of this study is to develop an automated tool for policy analysis. The tool utilizes an approximate reinforcement algorithm to improve the behavior of the simulation according to predefined objectives. The Stochastic and complex nature of the model makes approximate learning algorithms a good fit for the problem. The approximation technique requires a summary of information about the model that the user finds essential. This information is subjective. Hence, depending on the run results, user may verify whether user's understanding of the model overlaps with the model's representation of the system. Therefore, the tool merges a policy analysis phase with the verification and validation phases.*

### 1. Introduction

Recently, there has been a strong initiative in many fields such as economics, decision sciences, and psychology etc. to have descriptive white box approach to modeling socio-economic systems. Specifically, agent-based simulation has been one of the popular methodologies to model social phenomena. Combined with the white box approach, agent-based models focus on descriptive representations of human behavior which further introduce complexity to socio-economic models. Complexity in the models comes deliberately from the desire to capture and explain the dynamics of systems. It is often impossible even for an expert familiar with the model to interpret and analyze the results such complex computational models. This is the main reason why these models did not meet the expectations of many scholars (Richiardi et al., 2006).

According to Richiardi et al. (2006), the underlying problem of complex white box models is the lack of evolved, automated and standardized analysis tools that help verify, validate, fine tune, and design policies. Like Richiardi et. al., there are papers that call for formal methodological guidelines to model building process i.e. verification, validation, calibration and/or sensitivity analysis specifically for agent based models (Windrum et al., 2007). These papers spot the reasons for the need of rigorous methodology and either raise questions or provide suggestions on how to proceed. There are also papers that provide theoretical guidelines to validation and/or analysis processes (Gonenc and van Daalen, 2009, Glenn et al., 2004). These papers elaborate on the questions they raise and provide theoretical answers but they "do not provide precise prescriptions" (Gonenc and van Daalen, 2009). There are supplementary papers that provide tools along some theoretical guidelines (Moss, 2008,

Kase and Ritter, 2009, Schreiber and Carley, 2007). General consensus in the literature is that verification, validation and policy analysis are essential phases of model building and require structured protocols and guided tools.

This paper proposes a tool that can be useful in model validation, scenario and policy analysis. Current application is on policy analysis. A policy analysis is the process of designing applicable policies that improve performance according to predefined objectives. Given certain objectives in the simulation world, our aim is to guarantee some improvement compared to benchmark runs using a learning algorithm. Our particular goal is to have a tool that can guarantee reasonable improvement in expected performance for a stochastic model without having to simulate the model many times with multiple steps. The main reason for trying to minimize the number of runs is concern for computation time. For this purpose, we use  $Q$ -learning algorithm which requires single training run. The learning algorithm replaces the decision making mechanism of a particular agent. Hence, the application looks for plausible policies for that agent. The application is on a complex agent based model of a country developed using PMFServ (Silverman et al., 2006), a software for building agent-based models with socio-cognitive agents.

The paper is organized as follows. First, we introduce Approximate  $Q$ -learning algorithm. The following section goes over PMFServ and the country model. Application section will define the model specific properties of the algorithm and discuss the results. The final section concludes with a discussion of reflections of the tool to model validation.

## 2. Approximate $Q$ -Learning Algorithm

In  $Q$ -learning (Watkins and Dayan, 1992), the algorithm learns which action is profitable for a given state.  $Q$ -learning algorithm requires a single run for training. Then trained function is used for performance improvement. Given a state, an agent can switch to another state by taking an action,  $u \in U$ . Each state has a certain cost associated with it. The goal of the agent is to minimize the total cost.

$Q$ -learning algorithm is a function that maps combination of state space,  $\mathbf{S}$  and action space to real space,  $Q : \mathbf{S} \times \mathbf{U} \mapsto \mathcal{R}$ .

$$Q(i, u) = \sum_{j=1}^n p_{ij}(u)(c(i, u, j) + \alpha J_{\mu}(j)) \quad (1)$$

where  $c(i, u, j)$  is the cost of transition from  $i$  to  $j$  by taking action  $u$  (Bertsekas, 2005a).  $J_{\mu}(j)$  is the cost-to-go with policy  $\mu$  and generally can be defined as below:

$$J_{\mu}(i) = E_w \left\{ c_k(i, \mu(i), w) + \alpha J_{\mu}(f(i, \mu(i), w)) \right\} \\ \text{for all } i \in \mathbf{S}$$

where  $w$  is a random variable and  $f$  is the function that represents the model i.e.  $i + 1 = f_k(i, \mu(i), w)$  (Bertsekas, 2005b). Notice that the models we are interested in do not have mathematical representations for  $f$ . Additionally, we are not given transition probabilities for computing the expectation as done in equation 1. Hence, we introduce a parametric architecture for approximation of  $Q$ ,  $\tilde{Q}(i, u, r)$ . We have  $\tilde{Q}(i, u, r) \approx Q(i, u)$  in linear form,

$$\tilde{Q}(i, u, r) = \phi(i, u)'r \quad (2)$$

where  $r = (r_1, \dots, r_m)$  is the parameter vector.  $\phi(i, u)$  is called features vector, a vector with known scalars,  $\phi_k(i, u)$ , that depend on state  $i$  and action  $u$ . This type of approximation is called feature extraction. It is a process that maps the state  $i$  and action  $u$  into some other vector  $\phi(i, u)$ .

$$\phi(i, u)' = (\phi_1(i, u), \dots, \phi_m(i, u))$$

These features are handcrafted based on insight and experience on the model. They are meant to capture the most important aspects of the current state. For example, in chess where the state is the current position of the pieces on the board, appropriate features can be balance of pieces, their mobility, king safety, etc (Shannon, 1950). Eventhough approximation is linear, we can capture nonlinearities in the model by crafting features well (Bertsekas, 2005a).

Once approximate  $Q$ -factors are obtained, we can use the minimization

$$\bar{\mu}(i) = \arg \min_{u \in \mathbf{U}(i)} \tilde{Q}(i, u, r) \quad (3)$$

to obtain the optimal policy.

The algorithm is very similar to the optimistic approximate policy iteration methods based on temporal difference(TD). The only difference is it uses approximate values of  $Q$ . The

pseudocode for the algorithm is given as such (Bertsekas, 2005b):

At the beginning of iteration  $k$ , simulation is at some state  $i_k$ , agent has chosen a  $u_k$ , and we have the current parameter vector  $r_k$ . Then:

We simulate the next transition  $(i_k, i_{k+1})$ . We generate the action  $u_{k+1}$  by using the minimization

$$u_{k+1} = \arg \min_{u \in \mathbf{U}(i)} \tilde{Q}(i, u, r)$$

We calculate the TD

$$d_k = c(i_k, u_k, i_{k+1}) + \alpha \tilde{Q}(i_{k+1}, u_{k+1}, r_k) - \tilde{Q}(i_k, u_k, r_k)$$

Then parameter vector is updated using

$$r_{k+1} = r_k + \gamma_k d_k \nabla \tilde{Q}(i_k, u_k, r_k)$$

where  $\gamma_k > 0$  stands for the step size. Then the process is repeated after replacing  $r_k$ ,  $i_k$ , and  $u_k$  with  $r_{k+1}$ ,  $i_{k+1}$ , and  $u_{k+1}$ , respectively (Bertsekas, 2005b).

We say that the algorithm has converged when  $d_k$  approaches 0. When  $d_k$  reaches zero, we can say that parameter vector,  $r$  is learned and we can use  $\tilde{Q}(i, u, r)$  for policy analysis. Literature has varying suggestions for choice of algorithm specific variables such as discount factor,  $\alpha$ , and step sizew,  $\gamma$ . Through out the study, we have them as constants where  $\alpha$  is 0.9 and  $\gamma$  is 0.1.

## 3. PMFServ and Model Definition

PMFServ is a human behavior emulator that drives agents in simulated gameworlds. This software was developed over the past 11 years at the University of Pennsylvania as a "model of models" architecture to synthesize many best available models and best practice theories of human behavior modeling (Silverman et al., 2006). PMFServ models profile the traits, cognitions, and reasoning of agents to capture the cognitive-affective state and reasoning abilities of agents. PMFServ agents can play the roles of leaders, follower archetypes, and institutional ministers that allocate services to others based on cultural norms, corruption, and other inefficiencies.

The country model (Silverman et al., 2009) is built using agents in PMFServ. The agents in the country base their decisions solely on the current state of the world. Each agent's action has a certain impact on determining the next state of the world. The next state of the world only depends on the actions taken in the previous step. Each agent perceives the state of the world, and other agents around. The agents are socio-cognitive i.e. they are aware of the agents around them, and have feelings of their own and toward other agents. They develop emotions based on their profile (traits, norms, relations etc.) and the actions of their own and others. For further discussion and mathematical underpinnings of profiling leaders and followers refer to (Silverman and Bharathy, 2005) and (Silverman et al., 2007a).

The country model includes all the important political and

ethnic groups in the region. There are two types of agents within a group: Follower and Leader. Each group can have multiple followers but only one leader. Similarly, leader agent can only lead one group but a follower agent can be a member of multiple groups. Groups have relations with other groups corresponding to socio-economic, political and ethnic conflicts which has role in determining the action space of agents. For example, an action to attack is not available if you are perceiving your friend. Leaders are the agents that take action on behalf of the group. Leader manages the *resources* (Security, Politics, Economy etc.), and in and out group *relations*. In-group relations stand for relations between leader and followers. Followers show their support for the group leader via a property called *membership*. Additionally, welfare of the followers is important for the leader if she wants the support of the followers. Leader also incorporates *Capital* i.e. economic situation of the group into her decision mechanism. Out-group relations are the actions based on group's relations and *relative power* to the target group(s). Leader's decision making is affected by whether the leader feels *superior* and whether the leader's group has higher relative power. Additionally, all agents have an aggregate variable called *VID* (Vulnerability, Injustice, Distrust). VID is a directed metric i.e. an agent has VID against all groups which shows whether she feels vulnerable toward that group or she is treated unjustly by that group or she trusts the group or not. The values for this parameter are negative meaning the more negative they are, she feels less vulnerable toward that group. Further descriptions and mathematical representations of leader/follower modeling can be found in (Silverman et al., 2007b), (Silverman et al., 2008) and (Silverman et al., 2009). All of these parameters mentioned above create the context the agent is in. Context can be considered as the circumstances or the state that the agent is in. We are specifically interested in the circumstances right at the time of the agent's decision. Given the context, the agent decides based on maximizing her subjective expected utility, **SEU**, that depends on her personality. The word subjective comes from the fact that each agent has different traits and norms which are reflected as the weights of the utility function. For example, given the same context two agents would decide to act differently because of the difference in their profile. Each action satisfies these norms and traits with certain probability; therefore, we consider expected utility.

Additionally, the model is stochastic. Stochastic nature of the model comes from the randomness in the result and effects of the actions. Hence, an action such as Give Economy (Economic Aid) might fail under certain circumstances with a given probability.

## 4. Application

This section explains the application of the algorithm to the country model. First, we will parametrize the model information discussed in Model Definition section and then define features using them. Second, we will define the cost function i.e. the objectives for policy analysis. Final subsection will provide the computational results and discuss methodological ideas based on computational results.

### 4.1. Defining Features

The set of features ( $\phi$ ) was based on majority of the variables discussed in the model description. Notice that these variables are already aggregated variables that summarize certain parts of the state. These variables do not exhaust the variables that make up the state space. They also do not exhaustively cover the information that can be extracted from the model. They were chosen so that they contain the sufficient information for the algorithm to converge and provide good policies. Choice of features depends on the researcher and is limited with his available insight and experience. Hence, there is no correct set of features but there is set of features that work.

We start by properly parametrizing state variables of interest to be able to define features.  $g \in \mathcal{G}$  denotes a group.  $x \in \mathcal{A}$  denotes an agent.  $\mathbf{VID}_k(x, g) \in (0, 1)$  is the *vulnerability*, *injustice* and *distrusted* at time  $k$  of  $x \in \mathcal{A}$  directed towards group,  $g$ .  $\mathbf{R}_k(g_1, g_2) \in (-1, 1)$  is the *relationship* between  $g_1 \in \mathcal{G}$  and  $g_2 \in \mathcal{G}$ .  $\mathbf{RP}_k(g_1, g_2) \in (-1, 0)$  is the *relative power* of  $g_1$  over  $g_2$ . The negative number indicates a stronger  $g_1$  than  $g_2$ .  $\mathbf{GP}_k(g) \in (0, \infty)$  stands for amount of "good" properties which is a sum of group's *capital* divided by 52 (each step is a week and there are 52 steps in a year) and group's property *economic output*. In other words, it is another economic indicator. Leader cannot take certain actions if they have insufficient capital.  $\mathbf{RS}_k(g) \in (0, \infty)$  stands for the total resources of group  $g$  at step  $k$ .  $\mathbf{S}_k(x, g) \in (-1, 1)$  stands for how *superior* the agent  $x$  feels over the group  $g$ . This is a summary of agent's emotions toward groups.  $\mathbf{FVID}_k(f, g) \in (0, 1)$  basically stands for the same thing as  $\mathbf{VID}_k(x, g)$ .  $\mathbf{f}$  stands for the follower agents of the group,  $\mathbf{f} \in \mathcal{F}$  where  $\mathcal{F} \cup \mathcal{L} = \mathcal{A}$  and  $\mathcal{F} \cap \mathcal{L} = \emptyset$ .  $\mathbf{FM}_k(f, g) \in (0, 1)$  looks at a follower agent's *membership level* toward a group,  $g$  at time  $k$ .  $\mathbf{FW}_k(f) \in (0, 1)$  denotes the *welfare* of a follower agent,  $f \in \mathcal{F}$ . Notice that this is not directed to any group as it represents the current situation of the follower. The parameters that sum up to  $\mathbf{FW}_k(f) \in (0, 1)$  are *BasicNeedsLevel*, *Capital*, *EducationLevel*, *SuppressionLevel*, *HealthLevel*, *JobsLevel*, *LawLevel*. These are the properties of the follower which the leader have direct influence on.  $\mathbf{SEU}_k(u) \in (0, 1)$  is the subjective expected utility associated with the decision at time  $k$ . As mentioned in the previous section, each agent differs in her utility function from others based on her profile.

These are the aggregate elements that summarize the huge state space,  $x$ . Hence, we can think of a function  $\Psi : \mathbf{X} \mapsto \varphi$

where

$$\begin{aligned}
 \varphi(1) &= \sum_{g \in \mathcal{G}} \frac{\text{VID}_k(x, g)}{\text{card}\mathcal{G}} \\
 \varphi(2) &= \sum_{g_2 \in \mathcal{G}} \frac{\text{R}_k(g_1, g_2)}{\text{card}\mathcal{G}} \\
 \varphi(3) &= \sum_{g_2 \in \mathcal{E}} \frac{\text{RP}_k(g_1, g_2)}{\text{card}\mathcal{E}} \text{ where } \mathcal{E} \subset \mathcal{G} \text{ are enemies of } g_1 \\
 \varphi(4) &= \frac{\text{RS}_k(g)}{\sum_{k=1}^t \text{GP}_k(g)/t} \\
 \varphi(5) &= \frac{\text{GP}_k(g)}{\sum_{k=1}^t \text{GP}_k(g)/t} \\
 \varphi(6) &= \sum_{g \in \mathcal{G}} \frac{\text{S}_k(x, g)}{\text{card}\mathcal{G}} \\
 \varphi(7) &= \sum_{f \in \mathcal{F}} \frac{\text{FVID}_k(f, g)}{\text{card}\mathcal{F}} \\
 \varphi(8) &= \sum_{f \in \mathcal{F}} \frac{\text{FM}_k(f, g)}{\text{card}\mathcal{F}} \\
 \varphi(9) &= \sum_{f \in \mathcal{F}} \frac{\text{FW}_k(f, g)}{\text{card}\mathcal{F}} \\
 \varphi(10) &= \text{SEU}_k(u)
 \end{aligned}$$

And we can summarize,

$$\phi(i, u)' = (f_1(\varphi(1), u), \dots, f_{10}(\varphi(10), u)) \quad (4)$$

Next, each action was divided into positive diplomacy ( $\mathcal{U}_{DP}$ ), positive economy ( $\mathcal{U}_{EP}$ ), positive military ( $\mathcal{U}_{MP}$ ), negative diplomacy ( $\mathcal{U}_{DN}$ ), negative economy ( $\mathcal{U}_{EN}$ ), and negative military ( $\mathcal{U}_{MN}$ ). Then, using insight about the model, we tried to spot certain situations where taking actions from a certain set would be advantageous. Functions  $f_1, f_2, \dots, f_{10}$  map the conditions,  $\varphi(\cdot)$ , and actions,  $u$  to real numbers. For example, if the leader agent feels superior and powerful with respect to her enemies and has follower support then she might be inclined to take risky aggressive actions for the purpose of increasing her group's resources. Hence, features vector is a 10 by 1 vector where value of each  $\varphi(\cdot)$  would define a context in which a certain action is favorable. I denote this features vector  $\phi(i, u)_C$  standing for a features vector.

## 4.2. Defining Cost Function

Before going into analysis, we need to define cost function,  $c(i, u, j)$ . It is defined as the total sum of the resources at the current step for the chosen leader agent,  $l$ , plus some penalty ( $f_c$ ) related to leader's actions;

$$c(i, u, j) = -\text{RS}_k(g) + f_c(u) \quad (5)$$

where  $f_c(u) = -\text{SEU}_k(u) + p(u)$ . And finally  $p(u)$  depends on the action set that  $u$  belongs to. Specifically,

$$f(n) = \begin{cases} 0 & \text{if}(u \in \mathcal{U}_{DP}) \\ 0.2 & \text{if}(u \in \mathcal{U}_{DN}) \\ 0.2 & \text{if}(u \in \mathcal{U}_{EP}) \\ 0.4 & \text{if}(u \in \mathcal{U}_{EN}) \\ 0.4 & \text{if}(u \in \mathcal{U}_{MP}) \\ 0.6 & \text{o/w} \end{cases}$$

This way more peaceful and diplomatic actions are preferable than negative or military actions unless they really have a high utility. Notice that the cost function does not depend on the following state,  $j$ . The cost function is designed so that leader takes actions to increase her resources.

## 4.3. Results

This section summarizes and discusses the results obtained from the experiments with feature vectors. The results are preliminary and they require further investigation. The plots of parameter vector  $r$ , and  $d_k$  are provided. There are 3 training runs made for 52 steps. Elements of the parameter vector seem to converge to same point (Figure 1). This shows us that only a single training run is enough to obtain the parameter vectors. Additionally, we see that  $d_k$ s converge to zero for all training runs (Figure 2).

Figure 3 summarizes the results of benchmark, training and trained runs. Benchmark runs constitute of runs of the model without the training algorithm i.e. the agent of interest acts according to the same decision making mechanism as the other agents. The benchmark decision making mechanism is based on picking the action that maximizes SEU. Notice that SEU maximization has no direct relation to maximization of resource levels. Maximization of SEU represents the action that fits best with the leader's views and norms. In training runs, the Q-learning algorithm replaces the subjective expected utility maximization for the agent of interest. For the trained run, the agent acts according to the action that minimizes  $\tilde{Q}$ . Since the aim is to obtain a policy that will increase the total resource level of the chosen agent, the performance measure is the Total Resource Level.

Finally, looking at the resources, training runs obtain higher resource values than the benchmark run (Figure 3). Of course to guarantee an improvement in performance and develop trust on the policy, we need to look at multiple benchmark runs since the model is stochastic. We have done 3 benchmark runs and observed that in all of these runs resources

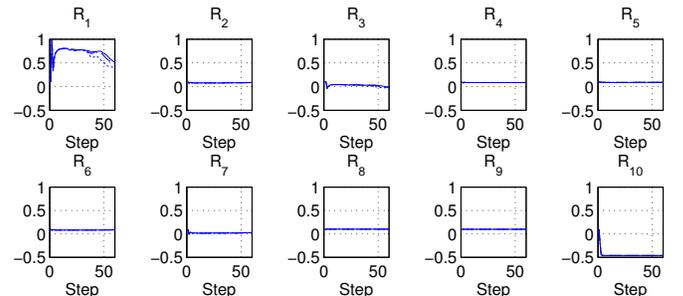


Figure 1. Three runs of the algorithm for  $\phi(i, u)_C$ , plots the 10 elements of the parameter vector,  $r$

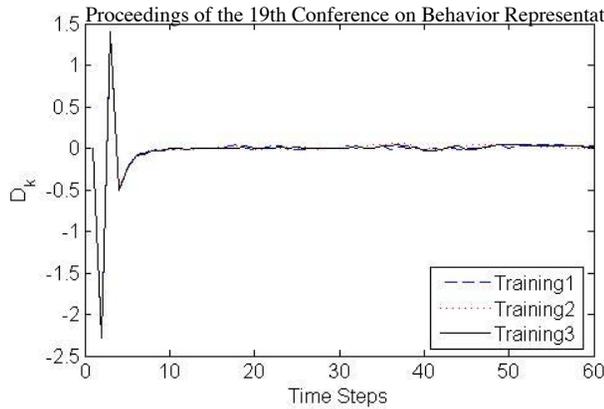


Figure 2. Three runs of the algorithm for  $\phi(i, u)_C$ , plots  $d_k$

are strictly less than resources in training runs. The run with trained parameter vectors obtain a reasonable improvement. Trained  $\tilde{Q}(i, u, r)$  is the resulting policy function that will dictate the actions to take. We need further runs with the trained  $\tilde{Q}(i, u, r)$  to show that the algorithm did not converge prematurely. We also need additional runs to show that the trained  $\tilde{Q}(i, u, r)$  works for different initial conditions. Current results suffice to say that there has been a reasonable improvement in resource levels when the leader adheres to algorithm's decisions.

So far, we discussed computational results. However, these are not the most interesting parts of the results. More interesting results come from the nature of the approximate  $Q$ -learning algorithm. Specifically, the way we define features vector reflect our insight on the model (recall the chess example). They correspond to which information we feel is important to take actions towards reaching the desired objectives. Looking at Figure 1, we observe that  $R_1$  and  $R_{10}$  corresponding to functions (see Equation 4) that depend on **VID** and **SEU** are the most influential in the policy. This means next time we develop features vector for the same model and objective, we might consider a simpler parameter vector that consider these two variables and a combination of the others. Moreover, when the algorithm

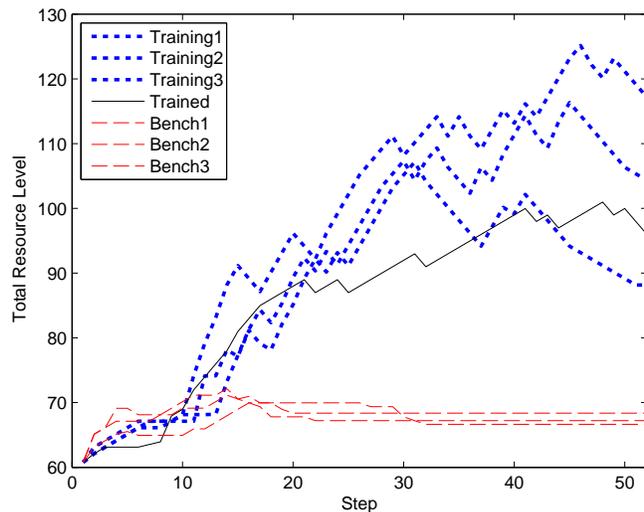


Figure 3. Total Resource Level for three benchmark runs(dashed), three training runs of the algorithm for  $\phi(i, u)_C$  (dotted) and single run with trained  $\tilde{Q}$ (solid)

converges and the results show improvement, that reassures our understanding. This can be considered a sanity check for the model. Furthermore, we need to look at how the leader agent's actions differ from the benchmark runs in trained runs. This corresponds to validating the means to achieve goals. If the actions taken to minimize cost do not make sense then we can infer that there is something wrong with the model. This sanity check is a way to poke structural representation (representation of underlying mechanisms) of our model. Valid structural representation makes sure that the goal of the model is not "just replication but also explanation" (Gonenc and van Daalen, 2009).

## 5. Discussion and Conclusion

We implemented a reinforcement learning algorithm to achieve certain goals in the model by letting the algorithm decide for the agent. One generalization is to make the algorithm control multiple agents. In that case, the algorithm returns a vector of actions that has cardinality equal to the number of agents. Although the implementation seems easy, it will be harder to define features.

Throughout the paper, we have avoided the case where convergence fails. This is simply because convergence is achieved in this study. However, if the results do not converge, then we might have to reconsider our understanding of the model and/or the structural representation of the model. The worst case scenario for convergence is when the model has a lot of volatility and the state space is huge. In that case, training runs might take infinite steps for convergence. This might fool us to question our understanding of the model i.e. features selection. This would be a false rejection of our correct understanding and representation of the system.

The tool is proposed for policy analysis. Yet, we see that both success or failure to achieve convergence can leave us with valuable information about the model. The design of the algorithm gives room to the experimenter to reflect her insight about the model. Although this might sometimes be cumbersome, it enforces the experimenter (usually the model builder) to reflect and summarize her ideas once more and cross check them with the model during policy analysis. Hence, policy analysis is added to the iterative loop of model verification and validation.

## References

Bertsekas, D. (2005a). *Dynamic Programming and Optimal Control*, volume 1. Athena Scientific, 3. edition.  
 Bertsekas, D. (2005b). *Dynamic Programming and Optimal Control*, volume 2. Athena Scientific, 3. edition.  
 Glenn, F., Neville, K., Stokes, J., and Ryder, J. (2004). Validation and calibration of human performance models to support simulation-based acquisition. In *Proceedings of the 2004 Winter Simulation Conference*, pages 1533–1540.  
 Gonenc, Y. and van Daalen, E. (2009). An objective-based perspective on assessment of model-supported policy processes. *Journal of Artificial Societies and Social Simulation*, 12(4):3.

- Kase, S. and Ritter, F. (2009). A High Performance Approach to Model Calibration and Validation. In *Proceedings of the Eighteenth Conference on Behavior Representation in Modeling and Simulation (BRIMS)*, pages 07–BRIMS–10.
- Moss, S. (2008). Alternative approaches to the empirical validation of agent-based models. *Journal of Artificial Societies and Social Simulation*, 11(1):5.
- Richiardi, M., Leombruni, R., Saam, N. J., and Sonnessa, M. (2006). A common protocol for agent-based social simulation. *Journal of Artificial Societies and Social Simulation*, 9(1):15.
- Schreiber, C. and Carley, C. (2007). Agent Interactions in Construct: An Empirical Validation using Calibrated Grounding. In *Proceedings of the Sixteenth Conference on Behavior Representation in Modeling and Simulation (BRIMS)*, pages 07–BRIMS–54. Orlando, FL: Simulation Interoperability Standards Organization.
- Shannon, C. (1950). Programming a digital computer for playing chess. *Phil. Mag.*, 41:356–375.
- Silverman, B. and Bharathy, G. (2005). Modeling the Personality and Cognition of Leaders. In *Proceedings of the Fourteenth Conference on Behavior Representation in Modeling and Simulation (BRIMS)*.
- Silverman, B., Bharathy, G., and Kim, G. (2009). The New Frontier of Agent-Based Modeling and Simulation of Social Systems with Country Databases, Newsfeeds, and Expert Surveys. In Uhrmacher, A. and Weyns, D., editors, *Agents, Simulation and Applications*. Taylor and Francis.
- Silverman, B., Bharathy, G., and Nye, B. (2007a). Personality Profiling is Politically Correct. In *IITSEC*.
- Silverman, B., Bharathy, G., Nye, B., and Eidelson, R. (2007b). Modeling factions for 'effects based operations': Part i - leader and follower behaviors. *Journal Computational and Mathematical Organization Theory*, 13(4):379–406.
- Silverman, B., Johns, M., Cornwell, J., and O'Brien, K. (2006). Human behavior models for agents in simulators and games: Part i - enabling science with pmfserv. *Presence*, 15(2):139–162.
- Silverman, B., Johns, M., Cornwell, J., and O'Brien, K. (2008). Human behavior models for agents in simulators and games: Part i - enabling science with pmfserv. *Presence*, 14(2):120–155.
- Watkins, C. and Dayan, P. (1992). Q-learning. *Machine Learning*, 8:279–292.
- Windrum, P., Fagiolo, G., and Moneta, A. (2007). Empirical validation of agent-based models: Alternatives and prospects. *Journal of Artificial Societies and Social Simulation*, 10(2):8.

## Author Biographies

**CEYHUN EKSIN** is a Ph.D. candidate in the Department of Electrical and Systems Engineering at University of Pennsylvania. He is affiliated with Ackoff Collaboratory for Advancement of the Systems Approach(ACASA).