The Use of Agent-Based Modeling and Data Farming for Planning System of Systems Tests in Joint Environments

Mary McDonald
Stephen Upton
Gary Horne
Operations Research Department
Naval Postgraduate School
Monterey, CA

SEED Center Mission: Advance the collaborative development and use of simulation experiments and efficient designs to provide decision makers with timely insights on complex systems and operations

http://harvest.nps.edu

76th MORSS
June 2008
The Use of Agent-Based Modeling and Data Farming for Planning System of Systems Tests in Joint Environments

Operations Research Department Naval Postgraduate School Monterey, CA

Mary McDonald, Gary Horne, Stephen Upton

Naval Postgraduate School

1411 Cunningham Rd
Monterey, CA 93943

Phone: (703) 655-8234
FAX: (703) 590-8802
Email: mlmcdona@nps.edu

Title of Presentation:
The Use of Agent-Based Modeling and Data Farming for Planning System of Systems Tests

This presentation is believed to be: ☑ UNCLASSIFIED and will be presented in:

☑ Special Session

The Releasing Official, with the understanding that MORS Symposia are supervised by the ONR N81, that all attendees have current security clearances of at least SECRET and that no foreign nationals will be present confirms that the overall classification of the presentation is:

☑ UNCLASSIFIED

Distribution statement A:
This presentation/paper is unclassified, approved for public release, distribution unlimited, and is exempt from U.S. export licensing and other export approvals under the International Traffic in Arms Regulations (22 CFR 120 et seq.)

Releasing Official’s title: ASSISTANT SECURITY MANAGER
Printed name: GUILLERMO COSTELLO
Organization: NAVAL POSTGRADUATE SCHOOL
Complete mailing address: Security Manager Code 261
Naval Postgraduate School
1411 Cunningham Rd, Rm B-13
Monterey, CA 93943-5015

Phone: 831-656-2450
FAX: 831-656-2350
Agenda

• SEED Center and Philosophy
• Data Farming
• Support to Joint Test and Evaluation Methodology (JTEM)
• Agent Based Modeling
• “TheTester” ABM
SEED Center in a nutshell…

Enable rapid and efficient computational experimentation and analysis to be readily available to those informing decision makers

• Harnessing Enabling Technologies
  – High-performance computation
  – New Design of Experiments (DOE)
  – (Emerging) models
  – Data mining and visualization

• Revolution in analysis capabilities
  – Quick turnaround…
  – Address uncertainties
  – Robust solutions
Resources: SEED Center for Data Farming

http://harvest.nps.edu

Check here for:

• lists of student theses (available online)
• spreadsheets & software
• pdf files for several of our publications, publication info for the rest
• links to other resources
• updates

All models are wrong, but some are useful—George Box
Data Farming: Iterative Loop of Loops

• Data Farming Loop
  – Scenario/Model Building Loop
    - Iterate model/simulation for experiment definition and analysis to support definition of hypothesis, and areas of interest
  – Possibility Space Development Loop
    - Iterate model/simulation using high-performance computing to refine analysis, study parameter sensitivity, drill-down into areas of interest, and confirmation or refutation of hypothesis
    - Data exploration, mining

and then

• Adjust-Synthesize (another loop)
  – Adjust model/simulation with knowledge/concepts/intuition from data farming… Repeat
Support to Joint Test & Evaluation Methodology

• **Overall Objective:** Determine if analytical techniques employing agent-based models and data farming can be applied to the following areas
  – Helping to select a limited number of test vignettes for accomplishment in an actual L/V/C joint mission environment
  – Determining overall joint mission effectiveness
  – Establishing the relationship between system or system-of-system performance and joint mission effectiveness

• **Previous Effort:**
  – Tested other agent-based models for applicability
  – Ran computational experiments within the SEED Center’s Data Farming environment
  – Developed custom-made agent-based modeling environment (“TheTester”)

Agent Based Modeling (ABM)

• What is an ABM?
  – Composed of (usually) relatively simple discrete autonomous entities making decisions based on interactions with other agents and their local environment
  – Are characteristically intuitive, transparent, transportable, repeatable, and farmable
  – Have been useful in studying complex adaptive systems in a number of domains

• Several have been developed specifically for military domain (ISAAC, MANA, Pythagoras, SEAS)

• Scenarios (usually) can be produced in a matter of hours/days vs weeks/months

“Any intelligent fool can make things bigger, more complex... It takes a touch of genius and a lot of courage to move in the opposite direction.”
“TheTester” ABM

• **Motivation:** To address some of the limitations encountered using more traditional agent-based models based on reactive agents, while retaining their strengths in farmability, ease of use, and fast run times

• **Primary Design Goal:** Focus on Systems of Systems testing, initially modeling one aspect (Joint Fires) of the C2 Joint Capability Area (JCA)
“TheTester”: Model structure

• Is written in JAVA, and uses the MASON multi-agent simulation toolkit for its underlying simulation infrastructure www.cs.gmu.edu/~eclab/projects/mason/

• Time-stepped

• Continuous 3D space, flat terrain

• Uses XML for input - working on an Automated Scenario Generator

• Selectable MOEs (CSV output)

• 3D visualization with probes
“TheTester”: Other Design Goals

- Composable
  allows users to build up or construct agents using software components specific to the domain

- Extensible
  allows users to develop their own software components to extend functionality provided by the basic framework

- Farmable
  enhances computational experiments with the model by allowing users to easily vary input parameters associated with the agents

- Fast-running
  analyses could be completed within a reasonably short period of time, commensurate with our experience with other agent-based models used for similar purposes
Agent Decision Making

- Each Agent has OODA loop
- “Observe” - depends on whether Agent has Effector for sensing
- Orient
  - Process Comm messages
  - Update Perceptions from other Perceivers
- Decide
  - Agent Decision Making is based on Deciders: these are composable object structures that base decisions on Perceptions - SimpleRuleBaseDecider currently implemented. Different agents can have different Deciders. SimpleRuleBaseDecider has a set of Rules that are a conjunction of Clauses (Perception Condition Value), with Actions as consequents
  - E.g., If NewEnemyDetected then SendMessageASR
- Act
  - Each Agent has a set of Actions that it can accomplish (based on what Effectors can do)
“TheTester”: Agent structure

Decider

Eff1
Eff2
Effn

Environment

Perceptions

Per1
Per2
... Perm

Eff1
Eff2
Effn

Action1
... Actionj

Environment
Examples (Implemented So Far)

• Observe-type Effectors
  – CookieCutterSensor

• Perceivers
  – SimpleThreatPerceiver
  – BasicMessageProcessor
  – MessageSentTracker
  – MemoryContactFilter

• Other Effector types
  – MoveAlongWaypoints
  – AgentCarrier / AgentCarried
  – BasicMessageSender
  – SingleMissionEffector
  – MultipleMissionEffector
  – FiresMissionTasker
  – BasicIndirectWeapon
Examples (cont.)

• Perceptions (concepts an agent “knows about”)
  – AgentPercept
  – LocationPercept
  – MessagePercept
  – Observation
  – RestrictedOperatingZone
  – SimplePercept
  – TargetPercept

• Deciders (used to choose an action, based on the current state of perceptions)
  – SimpleMoveDecider
  – RuleBaseDecider
Comm modeling

- **CommLinks**
  - Explicit communication links specified in input file
  - Reliability for the link
  - Range for the link

- **MessageData** – for each message class
  - messageClass for each message
  - probUnderstood
  - inProcessTime, inProcessTimeOffset
  - outProcessTime, outProcessTimeOffset
  - probability distribution used for times

- **MessageHandlers** - for inserting and extracting content

- **Implemented Message Handlers**
  - CallForFireMessageHandler
  - FiresMissionMessageHandler
  - GoToRequestMessageHandler
  - ThreatLocationMessageHandler
# FY07 Scenario Comm Matrix

<table>
<thead>
<tr>
<th></th>
<th>RSTA</th>
<th>BNFSE</th>
<th>BDEFSE</th>
<th>CAOC/JAOC/ASOC</th>
<th>AWACS</th>
<th>FIRE BN FSE</th>
<th>NLOS/FSPM</th>
<th>JSTARS</th>
<th>JTAC</th>
<th>CAS_AIRCRAFT</th>
</tr>
</thead>
<tbody>
<tr>
<td>RSTA</td>
<td>CFF</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BNFSE</td>
<td>RELAY-CFF</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BDEFSE</td>
<td></td>
<td>ACMREQ1-RFZ</td>
<td>ACMREQ1-COPY; ACMREQ2; RFZ</td>
<td>ACMREQ1-CF</td>
<td>ACMREQ1-APPROVAL</td>
<td>ACMREQ1-CF</td>
<td>ACMREQ1-APPROVAL</td>
<td>ACMREQ2-CF</td>
<td>ACMREQ2-APPROVAL</td>
<td>ACMREQ2-CF</td>
</tr>
</tbody>
</table>
Short Term Future Work on “TheTester” Will Include …

- Expert System / Fuzzy Logic Decider (JESS, Fuzzy JESS)
- Move to a Discrete Event Framework
- GUI / Automated Scenario Generator
QUESTIONS?