NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

THE PERFORMANCE OF EDGE ORGANIZATIONS IN A COLLABORATIVE TASK

by

Kok Meng Chang

December 2005

Thesis Advisor: Susan M. Sanchez
Co-Advisor: Paul J. Sanchez

Approved for public release; distribution is unlimited.

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			*Form Approved OMB No. 0704-0188*
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, 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, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.			
1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** December 2005	**3. REPORT TYPE AND DATES COVERED** Master's Thesis	
4. TITLE AND SUBTITLE: The Performance of Edge Organizations in a Collaborative Task			**5. FUNDING NUMBERS**
6. AUTHOR(S) Kok Meng Chang			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense, the U.S. Government, the Ministry of Defence of Singapore, or that of the Singapore Government.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited			**12b. DISTRIBUTION CODE**

13. ABSTRACT (maximum 200 words)

The rapidly changing and asymmetric threat environment that we are facing today has called into question the effectiveness of the traditional approach of hierarchical command and control (C2) structures. Edge organizations have been proposed as a more suitable alternative in the current information age.

Beside task-related factors, the characteristics and behavior of the people in an edge organization play an important role in determining the performance of the organization.

In this thesis, we look at how the various characteristics of agents influence the efficiency of an edge organization in an intelligence gathering task, using an agent-based simulation model developed in Java. We also look at the attributes of an agent that performs well in an organization, and whether a reward system that encourages individual success in an edge organization is detrimental to the organization's performance. Comparison between edge organizations with similar mean group attributes but different variability in agent characteristics, and comparison between an edge organization and a hierarchical organization are also performed.

14. SUBJECT TERMS Agent-based simulation, edge organization, hierarchical organization, intelligence gathering			**15. NUMBER OF PAGES** 99
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

THE PERFORMANCE OF EDGE ORGANIZATIONS IN A COLLABORATIVE TASK

Kok Meng Chang
Civilian, Singapore Defense Science and Technology Agency
B.Eng., Nanyang Technological University, 1996
M.Tech., National University of Singapore, 2004

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
December 2005

Author: Kok Meng Chang

Approved by: Susan M. Sanchez
Thesis Advisor

Paul J. Sanchez
Co-Advisor

James N. Eagle
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The rapidly changing and asymmetric threat environment that we are facing today has called into question the effectiveness of the traditional approach of hierarchical command and control (C2) structures. Edge organizations have been proposed as a more suitable alternative in the current information age.

Beside task-related factors, the characteristics and behavior of the people in an edge organization play an important role in determining the performance of the organization.

In this thesis, we investigate how the various characteristics of agents influence the efficiency of an edge organization in an intelligence gathering task, using an agent-based simulation model developed in Java. We also look at the attributes of an agent that performs well in an organization, and whether a reward system that encourages individual success in an edge organization is detrimental to the organization's performance. Comparison between edge organizations with similar mean group attributes but different variability in agent characteristics, and comparison between an edge organization and a hierarchical organization are also performed.

THIS PAGE INTENTIONALLY LEFT BLANK

THESIS DISCLAIMER

The reader is hereby cautioned that the computer programs and scenario files mentioned herein are developed solely for the purpose of this thesis research. While every practical effort has been made, within the time and resources available, to ensure that the programs and scenario files are free of computational and logic errors, they should not be considered validated in any way. Any application of these programs and scenario files without additional verification is at the risk of the user.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

THIS PAGE INTENTIONALLYLEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

I would like to express my sincere appreciation for Professor Susan Sanchez and Professor Paul Sanchez for their guidance and patience, which were instrumental in making this thesis a success. I would also like to thank Dr Richard Hayes, Dr Jimmie McEver and Ms Danielle Martin for their ideas and advice for the thesis.

Also, I would like to thank the Project Albert team for their support and comments on the thesis.

And most importantly, I owe a great deal of gratitude to my wife, Mui Ching, for her support, without which this thesis would not have been possible.

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

The rapidly changing and asymmetric threat environment that we are facing today has called into question the effectiveness of the traditional approach of hierarchical command and control (C2) structures. Edge organizations have been proposed as a more suitable alternative in the current information age. Edge organizations consist of a large number of components that can interact in complex manners without a centralized control that dictates the actions for each and every member. Edge organizations potentially have better performance due to their empowerment of members, better shared awareness, interoperability, agility, and adaptability to dynamic situations.

In this thesis, edge organization performance is studied via designed experiments conducted with agent-based simulations. A multi-agent system models an intelligence-gathering task undertaken by a group of agents organized in either an edge or a hierarchical structure.

The results indicate that the performance of an edge organization is not determined solely by the characteristics of the structure. It also depends on the nature of the task involved, the culture of the organization, the characteristics and behavior of the members, and how effectively these members collaborate and self synchronize to achieve a common goal.

The task completion time for the edge organization is improved if the people in the group are competent, work together as a whole, and are disinclined to hoard information. Group size also impacts performance: a balance must be struck between information overloading and information gain through increased sharing. Emergent leaders play an important role in aligning the goals in the edge organization, and the number of leaders emerging in the scenario affects the efficiency of the organization. The adaptability and robustness of the edge organization are also highlighted in the experiments.

For some task-related factors there is a threshold beyond which the performance of the edge organization is improved tremendously. Resources

such as technology and people can be used to mitigate the adverse effects of these task-related factors.

In our scenario, the hierarchical organization generally outperforms the edge organization, as expected due to the specialization of the structure. Comparing two edge organizations with identical mean characteristics for agents, the organization with the greater variability in agents' characteristics will generally perform better.

In these models, an agent that solves a problem is deemed a winner. Except for competency, the attributes of a winner differ from those that enhance the performance of the edge organization. This suggests that an incentive system which rewards individual winners is detrimental to the organization's objectives. However, an incentive system which encourages and rewards competency will, over time, improve the performance of an edge organization.

In summary, agent-based simulation provides a powerful tool for exploring the performance of complex organizational dynamics. When used in conjunction with efficient experimental designs, the simulation results yield insights regarding the interplay of task-related factors, agent characteristics, and structural form, in determining how quickly the edge organization can solve tasks.

I. INTRODUCTION

A. BACKGROUND

The rapidly changing and asymmetric threat environment that we are facing today has called into question the effectiveness of the traditional approach of a hierarchical command and control (C2) structure. An industrial age hierarchical C2 structure is based on the "principles of decomposition, specialization, hierarchy, optimization, deconfliction, centralized planning and decentralized execution" [Alberts & Hayes, 2003]. The edge organization has been proposed as a more suitable alternative C2 structure in the current information age, due to its empowerment of the edge members, better shared awareness among all the members in the organization, interoperability and most importantly, agility and adaptability to dynamic situations.

Edge organizations have the attributes to be agile. "While they may not be optimized to accomplish familiar tasks as hierarchical structures have evolved to do, edge organizations may be able to develop more innovative solutions to familiar problems over time" [Alberts & Hayes, 2003]. Given a specific familiar task, a hierarchical structure optimized to do the task will generally outperform an edge organization; but given a set of non-specific, dynamic tasks, an edge organization will usually perform better, i.e., the edge organization is generally more robust over a set of tasks but it is less optimized to a specific task.

The effectiveness and efficiency of an edge organization, however, do not solely depend on the characteristics of the structure. They also depend on the nature of the task involved, the culture of the organization, the characteristics and behavior of the members, and also how effectively these members collaborate and self-synchronize to achieve a common goal.

Information flow in an edge organization, for example, is influenced by the connectivity network of its members. A dense network may result in excessive information flow causing information overload; whereas the information flow will be minimal in a sparse network, [Perry & Moffat, 2004]. Both very sparse and very dense network models yield low plecticity, which is the ability of a connected set of actors to act synergistically via connectivity between them.

The behaviors and characteristics of the people are the most unpredictable sub-components of an edge organization. Members in an organization may vary in behavior, competency, attitude etc., which in turn influence the organization's behavior and performance as a whole. Members here could also refer to sub-organizations, which collectively constitute a larger edge organization. In this case, the cultures and protocols of individual member organizations will influence the behavior and performance of the overall edge organization. One assumption often made about edge organizations is that its members will always self-synchronize to achieve a common goal. This may not always be true. Even if the edge members do collaborate and self synchronize, the performance of the edge organization will vary with different compositions of member types.

An information age networked force can be described as a complex system. It consists of a large number of components that interact in complex manners without a centralized control that dictates the actions for each and every member. Ideally, it is self-organizing, capable of correlating of local effects, co-evolves continuously in a changing environment, and collectively generates group dynamics through a cascade of local effects [Atkinson & Moffat, 2005]. These complex organization dynamics can be suitably studied via agent-based simulation models.

In an agent-based simulation, the rules and behavior of individual entities are specified, as well as the rules governing their interactions. Results obtained from the simulation, through data farming, are often used to explore the consequences of specific individual level rules, factors and behaviors on the population as a whole. Agent-based simulation has the capability of generating complex and emergent properties - not so much from built-in rules of individual agent behavior, but from the complexity of the network of interactions among the agents [Srblijinovic & Skunca, 2003].

This thesis uses an agent-based simulation model to look into factors influencing the performance of an edge organization in an intelligence-gathering task.

B. PURPOSE

The purpose of this thesis is to examine the performance of an edge organization in an information gathering task, and specifically to gain insight about factors that might affect the efficiency of an edge organization. Understanding these factors will enable us to identify key enablers to an efficient edge organization performing a task that requires decision making and collaboration. Performance comparisons are made between the edge organization and a hierarchical structure suitable for this type of task, so that tradeoffs between agility and efficiency can be assessed. In addition, the research gives insights about factors and characteristics of an agent that "performs well" in an edge organization – defined as the person that solves the problem, or the "winner," in our scenario.

C. SCOPE

A multi-agent system was developed to model an intelligence-gathering task undertaken by a group of 12 agents organized in either an edge or a hierarchical structure. The system was developed using Java based on a discrete event simulation package, SIMpleKit [Sanchez, 2005]. Efficient experiment designs were used to conduct a series of experiments involving the multi-agent system. Results obtained were then analyzed with graphical and statistical tools.

This multi-agent system models the key characteristics of C2 processes in both the edge and hierarchical organizations. By abstracting real-world complex intelligence-gathering tasks to simple interactions between agents in the organization, control for variations in the members' capabilities and characteristics could be attained. This approach allows the analyst to focus on the fundamental issues of organizational design and isolate factors that might influence the efficiency of an organization. This approach also allows a large sample space of the factors influencing the edge organization to be explored and analyzed in a relatively short time.

The models were developed with reference to the physical experiment conducted by EBR Inc [EBR Preliminary Experimental Design Draft v41, 2005].

D. RESEARCH QUESTIONS

The research questions include, but are not limited to:

- To what extent will a hierarchical structure outperform an edge organization in the intelligence-gathering task?
- How robust is the edge organization?
- How do the following factors influence the performance of an edge organization?
    - Behavior of the agents
    - Different composition of agents in the organization
    - Formal or informal groupings of agents who share information
    - Negative information
    - Emergent leader
    - Information loading
    - Importance of agent competency vs. collaboration in the performance of the edge organization
    - Task-related factors
- What are the common characteristics of an agent that will “perform well” in an edge organization?

E. THESIS ORGANIZATION

This document is organized into six chapters. Chapter I provides an introduction and the purpose of this research work. Chapter II defines the problem scenario in detail, and presents the assumptions and measures of effectiveness for the scenario. Chapter III describes how the multi-agent simulation model was developed based on the scenario. Chapter IV introduces an efficient experiment design used to explore the scenario. Chapter V presents the results and detailed analysis of the experiment. Chapter VI concludes the research with a discussion of operational insights and recommendations for future work.

THIS PAGE INTENTIONALLY LEFT BLANK

II. SCENARIO DEFINITION AND ASSUMPTIONS

A. CHAPTER OVERVIEW

This chapter describes the problem scenario, assumptions, measures of effectiveness and key indicators used in the experiment. The physical experiment conducted by EBR Inc [EBR Preliminary Experimental Design Draft v41, 2005] serves as the basic outline for the scenario. Additional features are added because of the additional control possible in simulation settings.

B. SCENARIO DEFINITION

1. The Organization

An information gathering task is formulated for the experiment. Two groups of 12 agents are organized in either an edge or a hierarchical organization, as shown in Figure 1 and Figure 2.

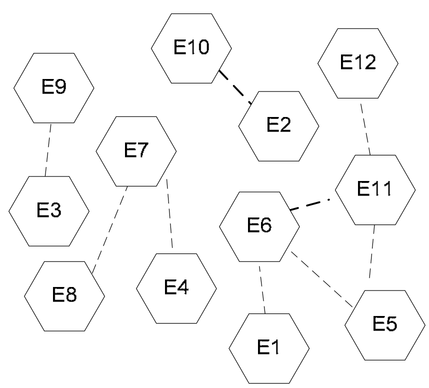


Figure 1. 12 Agents in an Edge Organization

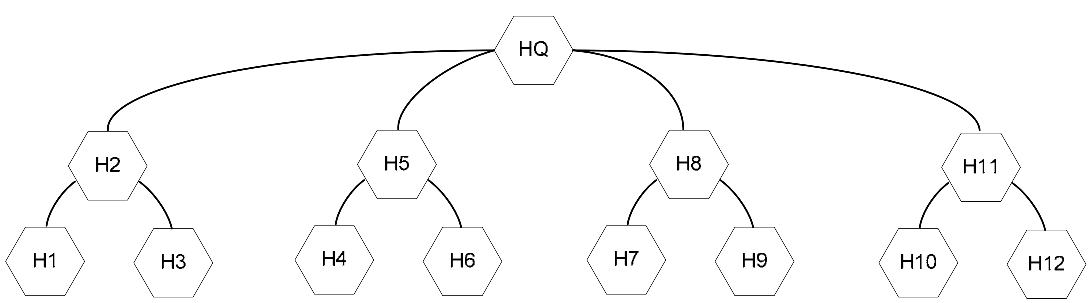


Figure 2. 12 Agents in a Hierarchical Organization

In the edge organization of Figure 1, the dashed lines indicate possible information links between the agents. Information tends to flow freely from one agent to another, as decided by the agent's behavior and grouping.

In the hierarchical organization of Figure 2, the solid lines indicate the fixed information links between the agents and information tends to flow upwards through the hierarchy. There are four specialized sub-groups of agents in the hierarchical structure, and agents H2, H5, H8 and H11 are the leaders in each of the sub-groups.

2. The Task

The agents in the two organizations are to identify the "who", "what", "when" and "where" of an adversary attack by discovering a set of information factoids. Like pieces of a puzzle, each factoid contains a piece of information for one of the four problem types. Collectively, these pieces form the solution to the problem. It is also possible to have factoids that are untrue, similar to false intelligence in an intelligence-gathering task.

In an edge organization, the agent may discover different types of factoids; whereas in the hierarchical organization, the agents discover only factoids according to their specialized type. For example, an agent who specialized in the "what" problem will only discover the "what" type factoids. This models the specialization and decomposition characteristics of a hierarchical structure.

The intelligence-gathering task can be characterized by five main attributes:

- total amount of information available for discovery
- total number of factoids available
- total amount of negative information
- total number of negative factoids
- discovery rate of the factoids

The objective for the agents is to discover and process a set of factoids sufficient to answer all four categories of questions, in order to correctly identify the attack. The agents may also be misled by negative intelligence they discover. The agents discover information by drawing from a pool of information that represents all available knowledge. That information is represented as a set of factoids. The available factoid set contains all the factoids that are available for discovery. The total information value of the available factoid set, which is sum of the individual factoid values, is the maximum amount of information that an agent can discover. Each agent acquires information according to an individual discovery rate.

Different tasks may have different characteristics even if they have the same total information value. Some tasks may just require a few significant factoids to solve while other may require numerous less significant factoids. A difficult task is defined as one with low discovery rate; whereas a less difficult task is defined as one with a higher discovery rate.

When the required information to solve the problem is much lower than the available information, we have a task with a lot of excess factoid information and the discovery rate of new information is generally constant throughout the information discovery process prior to solving the problem. In the case where almost all the available information is required to solve the problem, we have a task that has little excess factoid information. As more factoids are discovered, it will get more difficult to find the remaining factoids, i.e., discover new information rather than rediscover old information.

3. The Behavior

a. The Edge Organization

In the edge organization, there is a common portal where an agent can choose to post factoids he discovers. All agents in the organization can access the common portal to obtain factoid information posted by other agents.

This approach of information exchange can be thought of as a push and smart-pull approach inherent to a robustly networked environment [Alberts & Hayes, 2003], made possible by the advancement of information exchange technology. This portal is similar to the shared-awareness of the organization in the context of network centric warfare [Alberts, Garstka, Stein, 1999].

Upon discovering a factoid, an agent can decide to post it to all the members in the organization using the common portal, share the factoid with some of his selected peers, or completely hoard the information. There are factors that might influence the agent's propensities to post, share or hoard the information. For example, by hoarding the information the agent may end up being the first one to solve the problem. This may mean that an agent interested in becoming the "winner" might withhold necessary information and increase the time required to solve the problem. Other factors like individual agent characteristics, relationship with peers, the organization's reward policy, task criticality, peer pressure, organization culture, etc., will also influence the agent's decision.

In the edge organization, there is no fixed leader and agents are given a common goal to solve the problem. As the discovery process progresses and evolves, a leader may emerge. In this intelligence-gathering task, a leader is defined as one that posts the most significant factoids, i.e., factoids with the largest combined information value, in the common portal for all to share. The information posted must be significantly more important compared to the information posted by the rest of the agents, and the agent must be in this state for a period of time before he will emerge as a leader. The emergent leader will lose his status if at any point in time he fails to meet the above criteria. Another agent may subsequently emerge as a leader if he manages to post more significant factoids and meets the leader's criteria. This definition is analogous to that of an internet forum of special interests. Visitors to the forum for the first time are usually able to identify the "leader" of the forum by reading through some of the posts. The "leader" is usually the one that posts the more *important* messages and provide the *best* advice to the less well-versed.

When an agent decides to work in a group and share information with his selected peers, the receiving agent is more likely to reciprocate, provided that the receiving agent decides to share at all. This type of reciprocal sharing is common in our social and working relationships. An agent is also more willing to share his information with a leader, if there is one. The emergent leader assumptions in this scenario follow well with the hypothesis by Leavitt [Leavitt, 1951] that a centrally–located individual with the most access to information would emerge as a leader, in an organization with no designated individual as boss [Alberts & Hayes, 2003]. The emergent leader also acts as a source for synchronizing the behavior of all the agents towards a common goal, by generating an additional common information flow link (beside the direct link to the common portal) that redirects information from the rest of the agents to the common portal.

In an edge organization, the agent may form sub-groups in solving the problem. The groups can be of different sizes and types. A formal grouping is one where every agent knows who the group members are; an informal grouping is a virtual group where the member agents do not know the group members and each agent shares his information only with agents that share with him.

The competency of the agents in an organization also plays an important part in the intelligence gathering process. The competency of an agent affects how quickly the agent discovers intelligence, how fast the agent processes and interprets the information, and how the information and false information are interpreted. The more competent agent tends to gather intelligence faster, interpret the information faster, be less affected by the negative information, and require less information to identify the attack.

b. The Hierarchical Organization

In a hierarchical structure, there are also common portals agents can post factoids they discover. However, an agent can only access the portal belonging to his own specialization group.

Upon discovering a factoid, there are fewer incentives for an agent to hoard or share information, as the primary task of the agent is to discover factoids and pass the information to his leader. There are minimal interactions between agents of different sub-groups. The reward system of a hierarchical structure also encourages the agents to find as many factoids as possible and pass them up to their leaders.

The leaders in the hierarchical organization are fixed by position, and they are generally considered to be more competent than the group members. The definition of competency here is similar to that in the edge organization. It affects the discovery rate, message processing rate and information interpretation.

C. MEASURES OF EFFECTIVENESS (MOE) AND KEY INDICATORS

1. Solve Time

Solve time is the amount of time required to gain sufficient information for the four categories of information, so as to correctly identify the attack. This MOE gives insights on the factors affecting the efficiency of an organization structure.

2. "Winner" in an Edge Organization

The winner is defined as an agent that performs well in the edge organization, i.e., one that solves any of the four categories of problems. This indicator provides insights on the attributes that identify a winning agent in an edge organization. For example, a winner may be a leader, a hoarder, a sharer or just a normal agent in an edge organization.

3. Information Distribution

The distribution of information among the agents at the completion of the intelligence-gathering task gives insights on how robust the organization is to different agent characteristics and composition.

THIS PAGE INTENTIONALLY LEFT BLANK

III. BUILDING THE MODEL

A. CHAPTER OVERVIEW

Chapter III gives an overview on how the agent-based simulation was developed to model the intelligence-gathering scenario. The chapter describes the development of the main modules in the software, which include a scenario generator, a user interface, an edge model and a hierarchical model. This approach allows the modeling of specific organization behaviors, rapid and flexible scenario generation, and allows the large sample space of the factors to be explored and analyzed in a relatively short time.

B. CHOICE OF MODELING PLATFORM

The model is developed using Java, based on SIMpleKit [Sanchez, 2005] discrete event simulation package. Java provides the flexibility and the ease of adding specific rules, behaviors and interactions between agents, compared to other specific COTS multi-agent systems. Different type of scenarios can also be developed and configured quickly for the experiments.

SIMpleKit is a minimalist Java library which implements discrete event scheduling. It provides a basic and yet efficient discrete event simulation framework for implementing the simulation model [Sanchez, 2005].

The simulation model we developed consists of three main components: a scenario generator, the edge and hierarchical models and a debugging user interface.

C. SCENARIO GENERATOR

A design point is a spedific combination of levels for different factors to be used as inputs to the simulation. A single design point can be considered as one specific scenario. An experimental design is a carefully chosen set of design points that are used to explore factor combinations of interest and generate data for subsequent analysis.

A scenario generator module is developed to construct the scenario files for the models from a set of design points.

As Figure 3 shows, a design-point file is first developed from the design of experiment, and it defines all the desired design points for the experiment. The scenario generator reads the design-point file and generates a set of scenario files for each of the experiment runs. The scenario file defines the characteristics of every agent in a scenario.

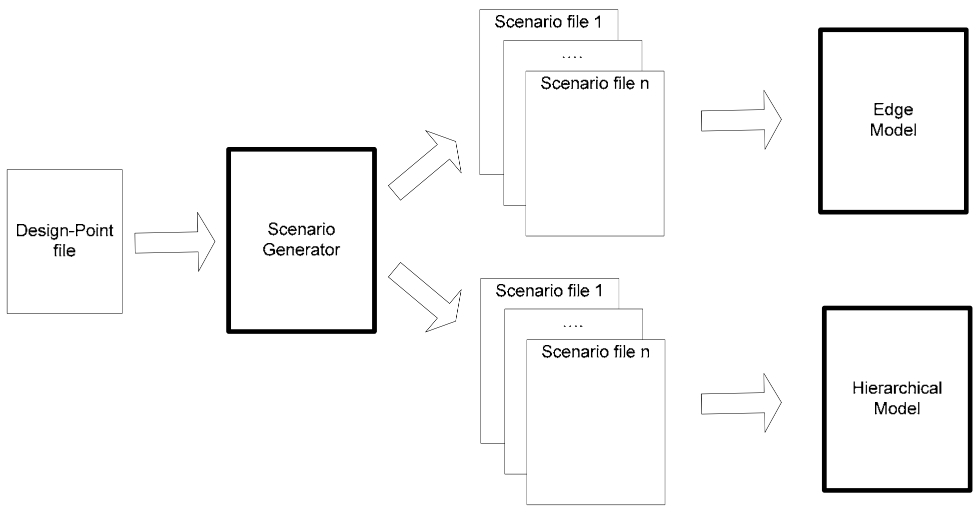


Figure 3. Scenario Generator and Scenario Files

D. BUILDING THE MODEL

1. The Factoids and Solutions

Each factoid is tagged with a number indicating the information value of the factoid. False intelligence is defined by factoids with negative information value. Examples of these types of factoids are provided in Figure 4.

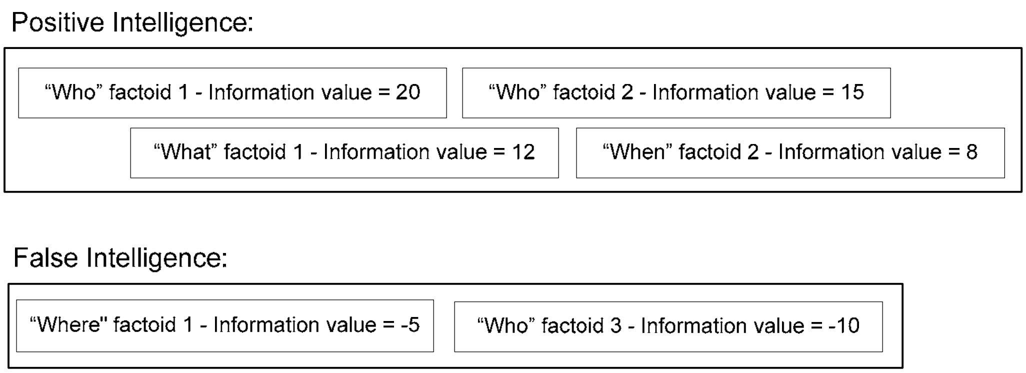


Figure 4. Example of Factoids

To solve a type of problem, an agent needs to collect a set of factoids with a combined information value that is more than the solution threshold for that type. The organization is deemed to have solved the problem when all four types of problems are solved by the agents.

Negative factoids discovered by an agent are added to the knowledge base of the agent, and these will lower the total information value derived by the agent. Therefore, in the presence of negative factoids, the agent will need to have a larger number of positive factoids in order to solve the problem. For example, an agent with competency of “1.0” that discovers three “who” factoids {who; 15}, {who; -10} and {who; 20}, will gain an information value of 25, i.e., 15 - 10 + 20. If the solution threshold for the “who” problem is 22, the agent is deemed to have sufficient information to solve the problem. If the solution threshold for the “who” problem is 30, the agent cannot solve the problem without additional information. In general, if the information value is more than the solution threshold of the “who” problem, the agent is deemed to have sufficient information to solve the problem.

Both the factoids and negative factoids are moderated by the competency level of the agent. The moderation of information value due to the agent’s competency will be discussed in the next section.

2. Agent and Task Specific Characteristics

The characteristics of an agent are defined by the parameters that follow. Our models allow for each agent to have distinct parameter values.

a. Post Probability

The post probability defines the probability that the agent posts a newly discovered factoid to the common portal for the rest of the agents to access.

b. Share Probability

The share probability defines the probability that the agent shares a newly discovered factoid with the members in his group. It also defines the probability that the agent shares the new factoid with the emergent leader if there is one.

c. Hoard Probability

The hoard probability defines the probability that the agent hoards a newly discovered factoid. The agent's actions in posting, sharing or hoarding a factoid are mutually exclusive and the sum of the post, share and hoard probability is one.

d. Group Number and Group Type

Each agent is tagged with a group number. If an agent decides to work in a group with selected peers, he shares his information with all his peers in the group based on the share probability. There are two types of grouping, formal and informal. In a formal grouping, upon receiving a new factoid from a peer, the agent will not re-share the new factoid with the rest of the group members since the agent knows they will also receive the same factoid. In an informal grouping, each agent only knows who he intends to share information with. Upon receiving a new factoid from a peer, the receiving agent may share the factoid with other members in his group list, if he decides to share.

e. Competency

Every agent has different competency levels, and competency is defined by a value with a maximum of 1.0. An agent with a competency of 0.8 is 20% less competent than one with competency of 1.0.

Competency affects three processes in the intelligence-gathering task: the factoid discovery rate, the message processing rate and the information interpretation.

- Given a normalized factoid discovery rate, *dr*, an agent with a competency level of 0.8 will have a discovery rate of 0.8 * *dr*.
- Similarly, given a normalized message processing rate, *mpr*, an agent with a competency level of 0.8 will have a message processing rate of 0.8 * *mpr*.
- An agent also extracts information from a factoid according to his competency level. For example, given a factoid with information value of *v*, an agent with a competency of 0.8 is able to extract 0.8 * *v* information value from the factoid. Given a factoid with information value of –*v*, an agent with competency of 0.8 is able to extract (1 - 0.8) * (–*v*) information value from the factoid. In essence, a competent agent discovers and processes information faster, extracts more information from a given factoid and is less affected by a negative factoid.

f. Normalized Discovery Rate

Discovery rate defines the rate at which the agent discovers factoids. The normalized discovery rate can be considered as a task-specific parameter. The actual discovery rate of each agent is the *normalized discovery rate* * *competency* level of the agent. The discovery time of an agent has an exponential distribution with mean 1/*DiscoveryRate*.

g. Message Processing Rate Factor

The agent's *message processing rate* is defined as the *message processing rate factor * discovery rate * competency* level of the agent. Upon receiving a factoid, an agent will interpret the factoid according to the message processing rate. The processing time has an exponential distribution with mean 1/*MessageProcessingRate.*

h. Posting Rate Factor

The agent's *posting check rate* is defined as the *posting check rate factor * discovery rate*. These events simulate agents accessing the common portal area for information. The inter-checking time at the common portal has an exponential distribution with mean 1/*PostCheckRate*.

i. Switch Task

If the *switch task* flag is set, when a solution of a particular category is found, the agent will no longer look for factoids for that particular category. Instead, the agent adapts and looks for information pertinent to a problem that is still unsolved. If the switch task flag is not set, the agents will still be looking at all categories of problems, even if some categories of the problem are solved.

3. The Edge Model

Event graphs are a way of graphically representing discrete-event simulation models. Their simplicity, together with their extensibility, make them an ideal tool for rapid construction and prototyping of simulation models [Buss, 2001]. The edge model can be described using the event graph in Figure 5.

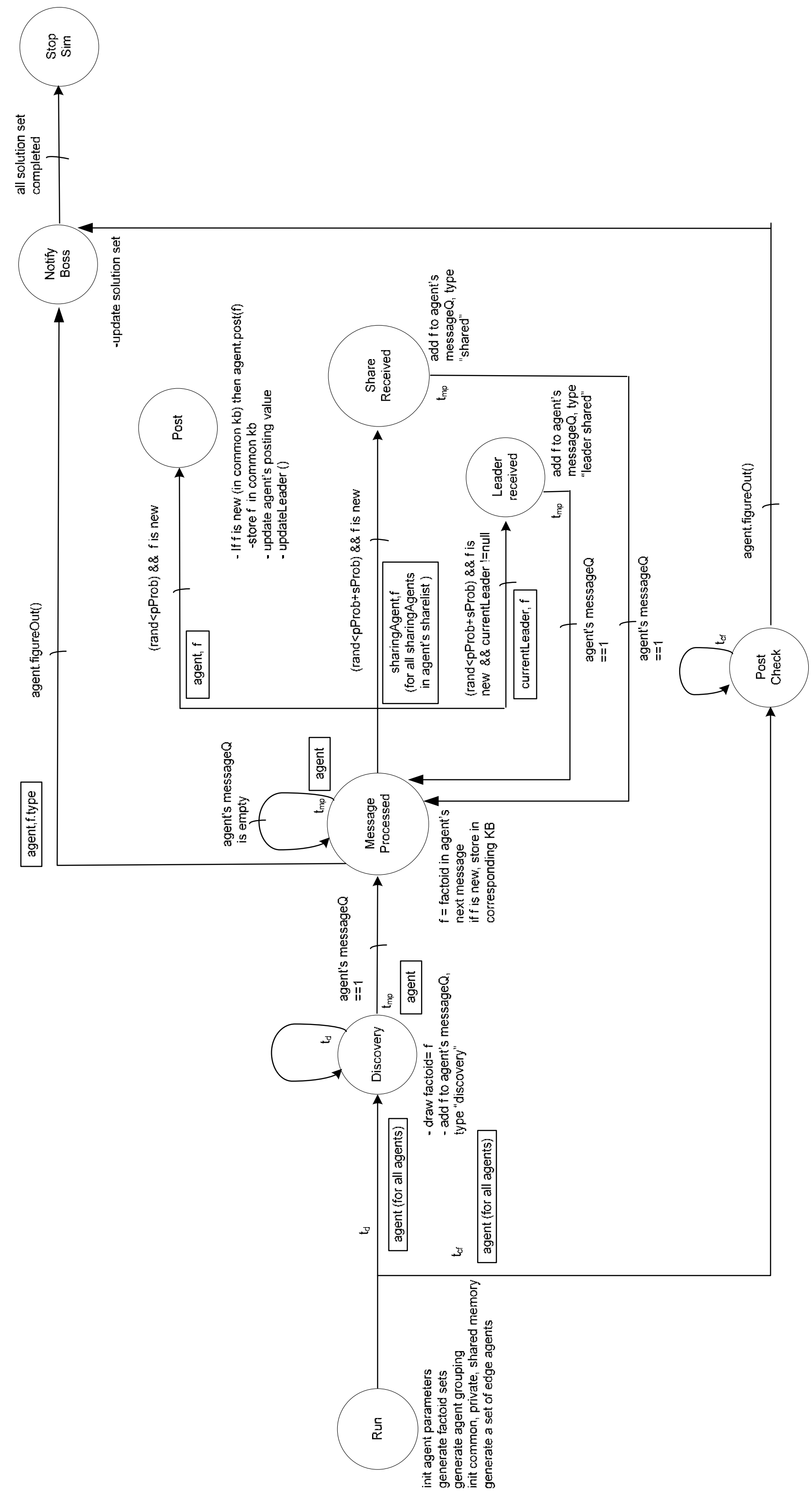


Figure 5. Event Graph of the Edge Model

a. Run Event

The *Run* event is the first event executed by the discrete event simulation package. The event initializes a set of 12 agents with parameters from an input scenario file. Agent parameters include the post probability, share probability, hoard probability, competency level, factoid discovery rate, message processing rate, posting check rate, group members and type of groupings. Scenario parameters like the number of negative factoids and total negative information value are also initialized.

Initialization of other objects, like the available factoid sets and the agent's private, shared and common memory spaces, are also performed in this event. A private memory space stores the factoids that the agent discovers; a shared memory space stores the factoids he received from his peers; a shared-leader memory space stores the factoids he received by virtue that he is a leader; and a common memory space stores the factoids posted by all the members of the organization.

After the initialization, the model schedules an information *Discovery* event and a *Post Check* event for each agent, according to the agent's discovery rate and posting check rate.

b. Discovery Event

At the *Discovery* event, the agent randomly draws a factoid from the available "who", "what", "when" and "where" factoid sets. The factoid is then added to the agent's message queue. If the agent's message queue has only one message, a *Message Processed* event is then scheduled according to the agent's message processing rate. If the *switch task* flag is set and some categories of the problem are solved, the agents will only draw factoids from the categories that are yet to be solved.

Lastly, the event schedules the next agent's *Discovery* event according to the agent's discovery rate.

c. Post Check Event

At the *Post Check* event, the agent accesses the common posting area and consolidates all the factoids he currently has. If he figures out the solution, i.e., the total information is more than the solution threshold, a *Notify Boss* event is scheduled.

Lastly, the event schedules the agent's next *Post Check* event according to the agent's post check rate.

d. Message Processed Event

At the *Message Processed* event, the agent removes the next factoid in his message queue. If the factoid is new, i.e., a factoid not known to him previously, the agent will store the factoid in the private, shared or leader-shared memory space, depending on whether the factoid is obtained through discovery, from a peer, or by virtue of him being a leader. After the factoid is stored, the agent will figure out whether he has solved the problem based on the current factoids that he knows. If the total information is more than the solution threshold, he has figured out the solution and a *Notify Boss* event is scheduled.

Next, the agent will decide whether to post, share with his peers and leader (if there is one) or hoard the factoid. The decision will depend on the post, share and hoard probabilities of the agent. If the agent decides to post the information, a *Post* event with the factoid is immediately scheduled. If the agent decides to share the information only with peers in his group, a series of *Share Received* events for the receiving agents in his group are scheduled. If the agent is in a formal grouping, he will not share the factoids received from a peer with other peers in his group. If a leader exists, and if the agent decides to share with him, a *Leader Received* event for the current leader is scheduled.

After processing the factoid, if the agent's message queue is not empty, a new *Message Processing* event for the agent is then scheduled according to his message processing rate.

e. Post Event

At the *Post* event, if the factoid to be posted is new, i.e., not available in the common posting area, it will be added to the common area for all agents to access. The posting score of the agent is then updated and an *Update Leader* method is invoked.

The *Update Leader* method determines whether the current leader, if one exists, still meets the leader's criteria after an agent has updated his posting score. If there is no current leader, the method will determine whether the posting agent satisfies the leader's criteria. The leader criteria defines a leader as an agent having the (i) the highest posting score, (ii) a posting score at least 20% larger than that of the agent with the second highest posting score, and iii) the above two criteria must be satisfied in at least in two consecutive postings. More stringent criteria could be set by increasing the 20% threshold and number of consecutive postings required.

f. Share Received Event

At the *Share Received* event, the received factoid is added to the agent's message queue. If the agent's message queue contains only this message, then a *Message Processed* event is scheduled according to the agent's message processing rate.

g. Leader Received Event

At the *Leader Received* event, the received factoid is added to the agent's message queue. If the agent's message queue contains only this message, then a *Message Processed* event is scheduled according to the agent's message processing rate.

h. Notify Boss Event

At the *Notify Boss* event, the solution set is updated. If all four categories of problems are solved, the simulation will stop and results will be output.

4. Information Flow Diagram for the Edge Model

The information flow diagram in Figure 6 depicts the information flow between agents in the edge model.

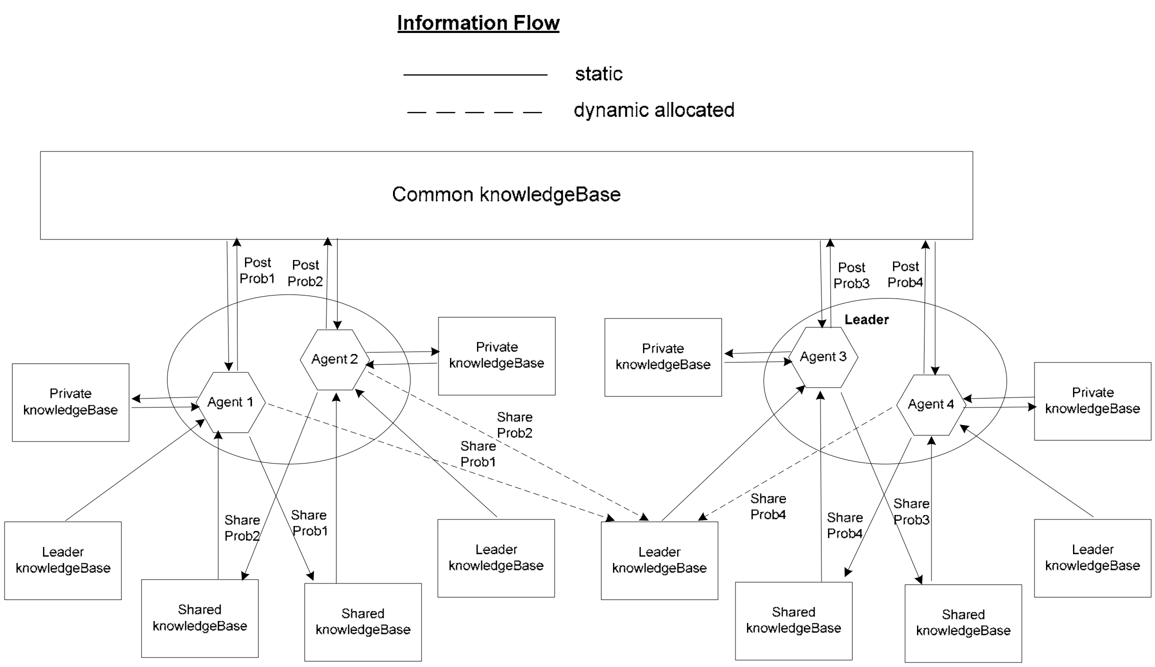


Figure 6. Information Flow Diagram in the Edge Model

Each agent has a private knowledge base (memory space), shared knowledge base, and leader knowledge base to store the factoids he discovers, receives from his peers, and receives by virtue of being a leader, respectively. In addition, all agents in the organization share a common knowledge base that stores the factoids posted by the agents.

The solid lines depict the fixed information flow between the agents and the knowledge bases. The dash lines depict information flow between the agents and the knowledge bases that may change as the simulation progresses.

5. The Hierarchical Model

The hierarchical model is similar to the edge model except for the following differences:

a. In the hierarchical model, the 12 agents are grouped into four groups of three agents. Each group of agents is tasked to solve one of the four categories of the problem, simulating the specialization in a hierarchical structure.

b. Agents of a category type will only look for information of that type during the discovery process. For example, agents working on the "who" problem, will only draw factoids from the available "who" factoid set. After a group of agents solve their category of problem, they will not switch tasks to look for other types of information due to their specialization.

c. Leader agents are pre-defined in the scenario file. They are fixed by their position and their roles do not change during the simulation. Since leader agents are assumed to have higher competency than the normal agents, the competency of the leaders is set at 1.0 (the maximum possible value). There are no emergent leaders.

d. Each group of agents has their own common posting area that is only accessible by the group members.

e. There is essentially no hoarding or sharing between members in the hierarchical organization, as the main objective of the agents is to gather as much information as possible and pass it to their leader. These characteristics are defined by setting the appropriate agent parameters in the scenario files for the experiments.

E. THE USER INTERFACE

A user interface for the edge model facilitates debugging. A screen shot appears in Figure 7. The user interface has 12 green columns displaying the current private knowledge (type and information value) of each agent, including the information he discovered, received from peers and received by virtue of being a leader. The blue column displays the current information in the common portal posted by all the agents. The event information column tracks all the events and actions of the agents and time of these events. The solution information column shows the solve time of each problem category.

When the debugging user interface is turned on, the simulation process thread is put to sleep for a short time at every major event. This slows down the simulation process for the purpose of debugging and displaying the information during the simulation. In the actual experiment, the debugging screen is turned off for more efficient simulation runs.

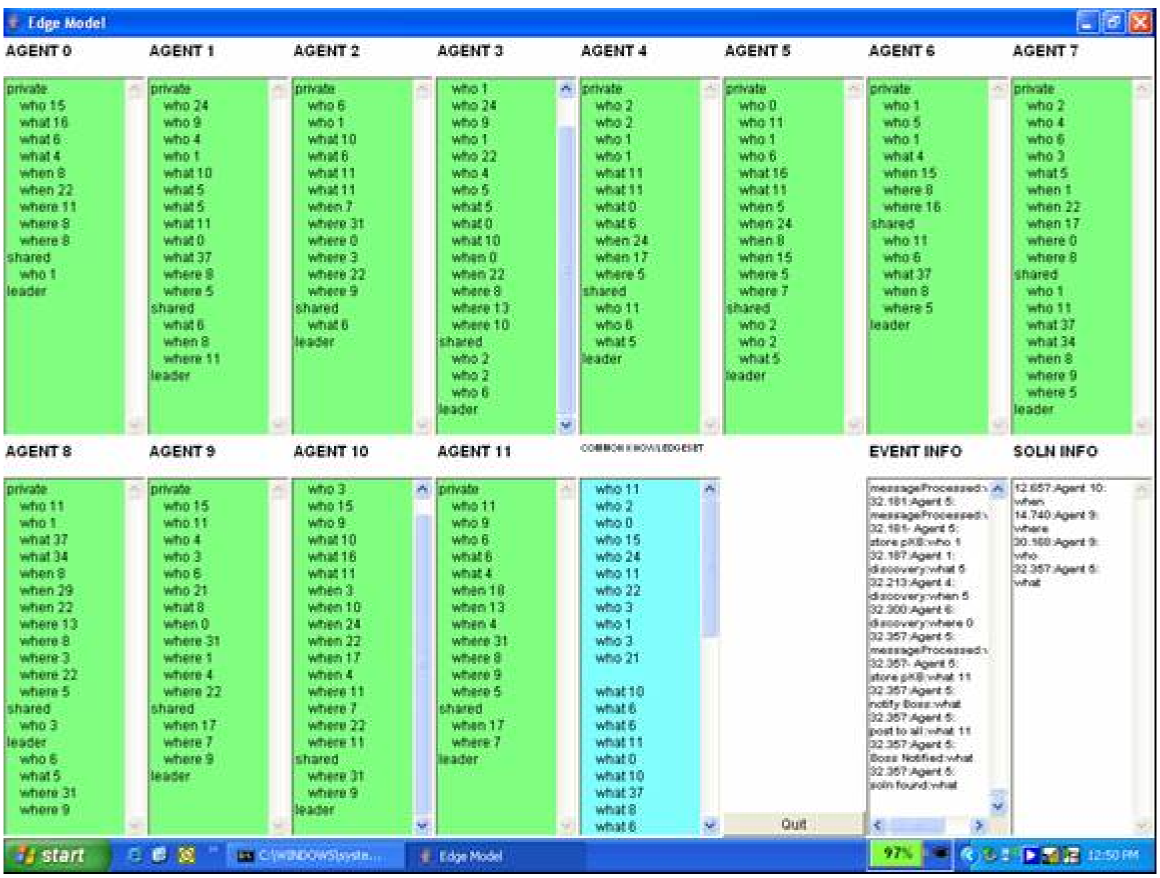


Figure 7. Edge Model Debugging User Interface Screenshot

A screen shot of the user interface for the hierarchical model appears in Figure 8. Here, agents who belong to the same group type (e.g., the “who” group) will have the same color for their private information columns. This makes it easy to identify the different groups. The private information column consists of the information the agent discovered and received from his peers. There are four common portals for the four different group types, “who”, “what”, “when” and “where”, which are only accessible by the agents belonging to that group type. The event information column and the solution column show the event and solution status of the simulation, similar to the edge model.

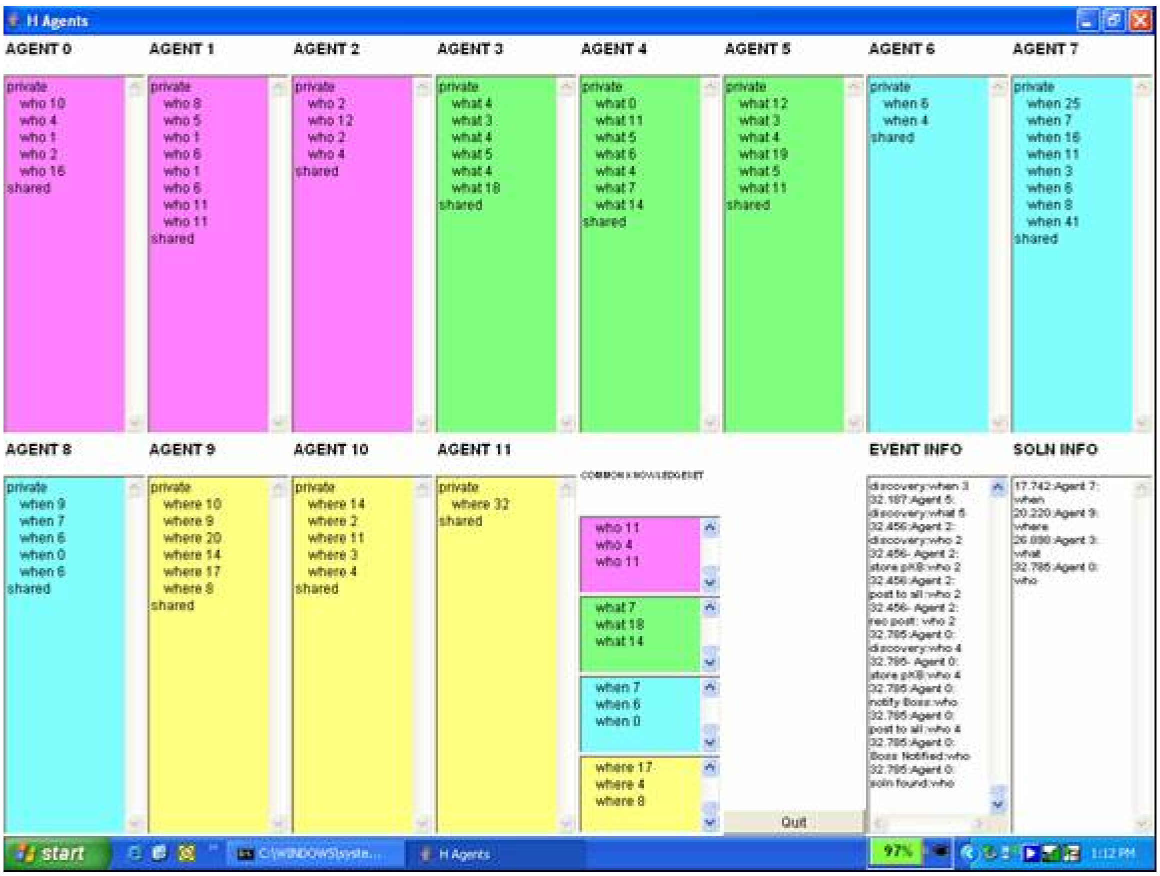


Figure 8. Hierarchical Model Debugging User Interface Screenshot

F. DATA OUTPUT

At the end of an experiment run of either the edge or the hierarchical model, the simulation outputs are sent to a text file for subsequent analysis. The output data and their descriptions are provided in Table 1. Note that the input parameter values also appear to facilitate analysis of the data.

S/N	Data	Description
1.	designPtID	Design point ID of the run
2.	ReplicationID	Replication ID of the run
3.	postProb	Average group post probability
4.	shareProb	Average group share probability
5.	hoardProb	Average group hoard probability
6.	Competency	Average group competency
7.	GroupSize	Group size of the agents
8.	Disc Rate	Average normalized discovery rate
9.	PostChkRateFac	Posting check rate factor
10.	MsgProcrateFac	Message processing rate factor
11.	FormalGrouping	Formal grouping flag
12.	solutionThreshold	Solution threshold
13.	NumNegFac	Number of negative factoids
14.	NegFacVal	Total negative factoid value
15.	switchTask	Switch task flag
16.	CompleteTime	Solve time
17.	WhoFactoids	Number of “who” factoids
18.	WhatFactoid	Number of “what” factoids
19.	WhenFactoid	Number of “when” factoids
20.	WhereFactoid	Number of “where” factoids
21.	WhoSolverID	ID of agent that solves the “who” problem
22.	WhatSolverID	ID of agent that solves the “what” problem
23.	WhenSolverID	ID of agent that solves the “when” problem
24.	WhereSolverID	ID of agent that solves the “where” problem
25.	PostProb(0-11)	Post probability of agent 0 to 11
26.	ShareProb(0-11)	Share probability of agent 0 to 11
27.	HoardProb(0-11)	Share probability of agent 0 to 11
28.	Competency(0-11)	Competency of agent 0 to 11

29.	Type(0-11)	Type of agent 0 to 11. Defines leader, normal agent and specialization group type in the hierarchical model
30.	DiscRate(0-11)	Normalized discovery rate of agent 0 to 11
31.	PostCheckRate(0-11)	Posting check rate of agent 0 to 11
32.	MsgProcRate(0-11)	Message processing rate of agent 0 to 11
33.	HasBeenLeader(0-11)	Flag indicating whether agent 0 to 11 has been leader
34.	WHOSolveTime	Time when the “who” problem is solved
35.	WHOnumWho(0-11)	Number of “who” factoids agent 0 to 11 has, when “who” problem is solved
36.	WHOwhoPoint(0-11)	Information value of “who” factoids agent 0 to 11 has, when ”who” problem is solved
37.	WHOnumWhat(0-11)	Number of “what” factoids agent 0 to 11 has, when “who” problem is solved
38.	WHOwhatPoint(0-11)	Information value of “what” factoids agent 0 to 11 has, when ”who” problem is solved
39.	WHOnumWhen(0-11)	Number of “when” factoids agent 0 to11 has, when “who” problem is solved
40.	WHOwhenPoint(0-11)	Information value of “when” factoids agent 0 to 11 has, when ”who” problem is solved
41.	WHOnumWhere(0-11)	Number of “where” factoids agent 0 to11 has, when “who” problem is solved
42.	WHOwherePoint(0-11)	Information value of “where” factoids agent 0 to 11 has, when ”who” problem is solved
43.	WHATSolveTime	Time when “what” problem is solved
44.	WHATnumWho(0-11)	Number of “who” factoids agent 0 to 11 has, when “what” problem is solved
45.	WHATwhoPoint(0-11)	Information value of “who” factoids agent 0 to 11 has, when ”what” problem is solved
46.	WHATnumWhat(0-11)	Number of “what” factoids agent 0 to11 has, when “what” problem is solved
47.	WHATwhatPoint(0-11)	Information value of “what” factoids agent 0 to 11, has when ”what” problem is solved
48.	WHATnumWhen(0-11)	Number of “when” factoids agent 0 to 11, has when “what” problem is solved
49.	WHATwhenPoint(0-11)	Information value of “when” factoids agent 0 to 11 has, when ”what” problem is solved
50.	WHATnumWhere(0-11)	Number of “where” factoids agent 0 to 11 has, when “what” problem is solved
51.	WHATwherePoint(0-11)	Information value of “where” factoids agent 0 to 11 has, when ”what” problem is solved
52.	WHENSolveTime	Time when the “when” problem is solved
53.	WHENnumWho(0-11)	Number of “who” factoids agent 0 to 11 has, when “when” problem is solved
54.	WHENwhoPoint(0-11)	Information value of “who” factoids agent 0 to 11 has, when ”when” problem is solved
55.	WHENnumWhat(0-11)	Number of “what” factoids agent 0 to 11 has, when “when” problem is solved
56.	WHENwhatPoint(0-11)	Information value of “what” factoids agent 0to 11, has when ”when” problem is solved
57.	WHENnumWhen(0-11)	Number of “when” factoids agent 0 to 11 has, when “when” problem is solved
58.	WHENwhenPoint(0-11)	Information value of “when” factoids agent 0 to 11 has, when ”when” problem is solved
59.	WHENnumWhere(0-11)	Number of “where” factoids agent 0 to 11 has, when “when” problem is solved

60.	WHENwherePoint(0-11)	Information value of “where” factoids agent 0 to 11 has, when ”when” problem is solved
61.	WHERESolveTime	Time when the “where” problem is solved
62.	WHEREnumWho(0-11)	Number of “who” factoids agent 0 to 11, has when “where” problem is solved
63.	WHEREwhoPoint(0-11)	Information value of “who” factoids agent 0 to 11 has, when ”where” problem is solved
64.	WHEREnumWhat(0-11)	Number of “what” factoids agent 0 to11 has, when “where” problem is solved
65.	WHEREwhatPoint(0-11)	Information value of “what” factoids agent 0 to 11 has, when ”where” problem is solved
66.	WHEREnumWhen(0-11)	Number of “when” factoids agent 0 to 11 has, when “where” problem is solved
67.	WHEREwhenPoint(0-11)	Information value of “when” factoids agent 0 to 11 has, when ”where” problem is solved
68.	WHEREnumWhere(0-11)	Number of “where” factoids agent 0 to 11, has when “where” problem is solved
69.	WHEREwherePoint(0-11)	Information value of “where” factoids agent 0 to11 has, when ”where” problem is solved

Table 1. Output Data

THIS PAGE INTENTIONALLY LEFT BLANK

IV. THE EXPERIMENT

A. CHAPTER OVERVIEW

Chapter IV describes how the simulation experiments are set up. The factors under investigation are listed for three models: Edge Model 1, Edge Model 2 and Hierarchical Model. The basic data collection plan for all experiments is an efficient experiment design called a Nearly Orthogonal Latin Hypercube design (NOLH) [Cioppa, 2002]. For a detailed discussion of the design and analysis of simulation experiments, see Kleijnen et al. [2005].

B. DESIGN OF EXPERIMENTS

There are several factors that the experiment seeks to study. These factors can be broadly classified into two main types: primary factors and task-related factors (which are considered as noise factors in the experiment). The task factors consist of:

- Discovery rate
- Normalized message processing rate factor
- Posting check rate factor (factor of discovery rate)
- Number of negative factoids in the available factoids
- Total negative factoid values in the available factoids
- Solution threshold

The primary factors include:

- Average post probability of the organization
- Average share probability of the organization
- Average hoard probability of the organization
- Average competency of the organization
- Formal grouping (yes or no)

- Group size of the agents
- Switch task flag

The following factors are set as constants in the experiment

- Number of agents in each organization (12)
- Number of categories to be solved (4, "who", "what", "when", and "where")
- Total available information value per category (1800)
- Number of available factoids (uniform distribution: 100 to 140)

1. Design for the Edge Model 1

Table 2 shows the factor descriptions and ranges for the first edge model. A NOLH experimental design is suitable since all design factors and agent parameters are continuous or take on several potential values.

Design Factors	Edge Model	Agent Parameters	Description
shareProb	Group Mean = 0.10-0.40	Uniform distribution (+- 0.1) from the mean	Probability that the agent shares (note probability of agent posts = 1- shareProb - hoardProb)
hoardProb	Group Mean = 0.10-0.40	Uniform distribution (+- 0.1) from the mean	Probability that the agent hoards (note probability of agent posts = 1- shareProb - hoardProb)
Competency	Group Mean = 0.60-0.90	Uniform distribution (+- 0.1) from the mean	Competency of the agent (factor moderates discovery, message processing and factoid interpretation)
groupSize	1-6	1-6	Size of the sharing group
discRate	Group Mean = 0.20-1.00	Uniform distribution (+- 0.1) from the mean	Normalized discovery rate of the agent
postChkRateFac	Group Mean = 0.20-1.00	Uniform distribution (+- 0.1) from the mean	Agent's common portal check rate (factor of discovery rate)
msgProcRate Fac	Group Mean = 0.50-10.00	Uniform distribution (+- 0.1) from the mean	Agent's Message processing rate (a factor of discovery rate)
NumNegFac	18-36	18-36	Number of factoids
NegFacVal	36-360	36-360	Total value of negative factoids

Table 2. Factors and Ranges of Exploration for NOLH Factors in Edge Model 1

The factors in Table 2 are then crossed with the factors in Table 3 so as to obtain an overall design with low correlations among all factor settings.

Design Factors	**Edge Model**	**Agent Parameters**	**Description**
Solution Threshold	500, 600, 700	500, 600, 700	Minimum solution value to solve the problem
Formal grouping	0,1	0,1	Infomral / Formal grouping
Switch Task	0,1	0,1	Once a task is completed, the agent will concentrate on other unsolved tasks

Table 3. Factors and Ranges of Exploration for Crossed Factors in Edge Model 1

The final crossed design yields the correlation matrix shown in Figure 9.

	FormalGrouping	**switchTask**	**solutionThreshold**	**shareProb**	**hoardProb**	**Competency**	**GroupSize**	**Disc Rate**	**PostChkRateFac**	**MsgProcrateFac**	**NumNegFac**	**NegFacVal**
FormalGrouping	1											
switchTask	0	1										
solutionThreshold	0	0	1									
shareProb	-2E-18	-1.6E-18	0	1								
hoardProb	0	0	7.89E-19	-3E-06	1							
Competency	0	0	3.16E-19	-3E-06	-3E-06	1						
GroupSize	0	0	7.29E-20	0.011939	0.048351	-0.0074	1					
Disc Rate	-9E-19	-9.1E-19	2.85E-18	0.000882	0.000214	-0.00431	-0.0236	1				
PostChkRateFac	-1E-18	-1.2E-18	1.11E-18	-0.00099	0.010247	0.00867	0.06201	-0.0084	1			
MsgProcrateFac	2E-19	2.04E-19	0	-0.00061	0.005785	-0.00205	0.01919	-0.00771	-0.00651	1		
NumNegFac	0	0	6.77E-19	0.014189	-0.00166	-0.01783	0.01999	0.001352	-0.00628	-0.00745	1	
NegFacVal	0	0	6.01E-19	0.013984	-0.00905	0.006689	0.02391	-0.00071	-0.00828	-0.01782	-0.00295	1

Figure 9. Correlation Matrix of the Factors for Edge Model 1

The largest correlation coefficient is 0.06, between the groupSize and the PostCheckRateFac factors. Since this is small, the design has good orthogonal properties, which is a desirable criterion for evaluating designs as it simplifies computation. As the inputs factors are essentially uncorrelated, it is easier to determine whether to include them in a metamodel (e.g., a regression model) and to separate their contributions to the overall metamodel fit. This in turn simplifies interpretation of results [Kleijinen et al., 2005].

The agents' parameters in each experiment run are defined in the scenario files, which are generated by the scenario generator module with the design point as inputs. The agents have characteristics with uniform ± 0.1

distribution from the group mean, simulating a group of agents having similar (but not identical) characteristics and traits.

2. Design for the Edge Model 2

Another set of agent parameters was generated with less variability among agents. As Table 4 shows, each agent's parameters are set to the same values as the group's mean values. This will be known as the edge model 2 in the experiments.

Design Factors	Edge Model	Agent Parameters	Description
shareProb	Group Mean = 0.10-0.40	Same as group mean	Probability that the agent shares (note probability of agent posts = 1- shareProb - hoardProb)
horadProb	Group Mean = 0.10-0.40	Same as group mean	Probability that the agent hoards (note probability of agent posts = 1- shareProb - hoardProb)
Competency	Group Mean = 0.60-0.90	Same as group mean	Competency of the agent (factor moderates discovery, message processing and factoid interpretation)
groupSize	1-6	1-6	Size of the sharing group
discRate	Group Mean = 0.20-1.00	Uniform distribution (+- 0.1) from the mean	Normalized discovery rate of the agent
postChkRateFac	Group Mean = 0.20-1.00	Uniform distribution (+- 0.1) from the mean	Agent's common portal check rate (factor of discovery rate)
msgProcRate Fac	Group Mean = 0.50-10.00	Uniform distribution (+- 0.1) from the mean	Agent's Message processing rate (a factor of discovery rate)
NumNegFac	18-36	18-36	Number of factoids
NegFacVal	36-360	36-360	Total value of negative factoids

Table 4. Factors and Ranges of Exploration for NOLH Factors in Edge Model 2

Both edge model 1 and edge model 2 use a 65 point NOLH design [Sanchez, 2005] crossed with the three factors, which generates a total of 780 (65 x 2 x 2 x 3) design points. In the experiment, 30 replications were performed for each design point yielding a total of 23,400 runs for each of the edge models.

3. Design for the Hierarchical Model

In order to model some of the characteristics of a hierarchical model, some of the design factors were set to constant values. Table 5 shows the factors and ranges used to investigate the performance of the hierarchical model.

Design Factors	**Hierarchical Model**	**Agent Parameters**	**Description**
shareProb	Group Mean = 0	0	Probability that the agent shares (note probability of agent posts = 1- shareProb - hoardProb)
horadProb	Group Mean = 0.10-0.20	Uniform distribution (+- 0.1) from the mean	Probability that the agent hoards (note probability of agent posts = 1- shareProb - hoardProb)
Competency	Group Mean = 0.60-0.90	Uniform distribution (+- 0.1) from the mean (Leader =1)	Competency of the agent (factor moderates discovery, message processing and factoid interpretation)
groupSize	1	1	Size of the sharing group
discRate	Group Mean = 0.20-1.00	Uniform distribution (+- 0.1) from the mean	Normalized discovery rate of the agent
postChkRateFac	Group Mean = 0.20-1.00	Uniform distribution (+- 0.1) from the mean	Agent's common portal check rate (factor to discovery rate)
msgProcRate Fac	Group Mean = 0.50-10.00	Uniform distribution (+- 0.1) from the mean	Agent's Message processing rate (a factor to discovery rate)
NumNegFac	18-36	18-36	Number of factoids
NegFacVal	36-360	36-360	Total value of negative factoid

Table 5. Factors and Ranges of Exploration for NOLH Factors in Hierarchical Model

The above factors were then crossed with the solution threshold factor to obtain a design with low pair-wise correlation between the parameters (Table 6). The formal grouping and switch task factors have no meaning in the context of a hierarchical model, so their values are set to zero for all runs.

Design Factors	Edge Model	Agent Parameters	Description
Solution Threshold	500, 600, 700	500, 600, 700	Solution value to solve the problem
Formal grouping	0	0	Agents know/do not know all the members of the group
Switch Task	0	0	Once a task is completed, the agent will concentrate on other unsolved task

Table 6. Factors and Ranges of Exploration for Crossed Factors Hierarchical Model

The final crossed design yields the correlation matrix in Figure 10. The largest correlation coefficient is 0.03, between the hoardProb and the NumNegFac factors. The design has good orthogonal properties.

	solution Threshold	*hoardProb*	*Competency*	*Disc Rate*	*PostChkRateFac*	*MsgProcrateFac*	*NumNegFac*	*NegFacVal*
solution Threshold	1							
hoardProb	3.7577E-18	1						
Competency	1.2629E-18	0.0293179	1					
Disc Rate	0	-0.019685	-0.0057248	1				
PostChkRateFac	-8.209E-18	-0.014086	-3.033E-06	0.0065973	1			
MsgProcrateFac	0	0.0062333	0.0093679	-0.00257	-0.0029268	1		
NumNegFac	0	0.0333805	-0.0019857	0.0008548	-0.0267435	-0.00194803	1	
NegFacVal	-2.405E-18	-0.003601	-0.0142598	0.0063065	0.00773386	0.00262174	-0.008503	1

Figure 10. Correlation Matrix of the Factors

Similar to the edge model, the agents' parameters in each experiment run are defined in the scenario files, which are generated by the scenario generator module with the design point as inputs. The agents have characteristics with uniform variation over +0.1 from the group mean, simulating a group of agents having similar (but not identical) characteristics and traits.

Using the 65 point NOLH design crossed with the solution threshold factor, we have a total of 195 (65 x 3) design points. In the experiment, 30 replications were performed for each design point yielding a total of 5,850 runs for the hierarchical model.

V. EXPERIMENT RESULTS AND DISCUSSION

A. CHAPTER OVERVIEW

Chapter V discusses the data collection process and results obtained from the experiments.

B. DATA COLLECTION

The data collected from these experiments are analyzed using Excel and the statistical software package JMP. Regression models and statistical plots are developed to identify significant factors and gain insights to the scenario. In the analysis, data obtained from the same design point but different replications are grouped together, and the mean of the completion time of the replications is used as a measure of effectiveness. In tools like the contour plots, which consider only two factors at any one time, the data are further grouped together according to the input factors we are investigating, and the average of the mean completion time is used as the measure of effectiveness. This is because when considering only a few input factors in a stacked design, it is possible to have identical factors across two design points.

C. RESULTS AND DISCUSSION

1. Edge Model 1 - Information Distribution at Time of Completion

At the time of completion of any category of a problem, all the agents in the organization possess part of the information pertinent to that problem. By taking the average over the remaining 11 agents, the percentage difference between the average information and the winner's information can be obtained. The distribution of the percentage difference is obtained and plotted in Figure 11.

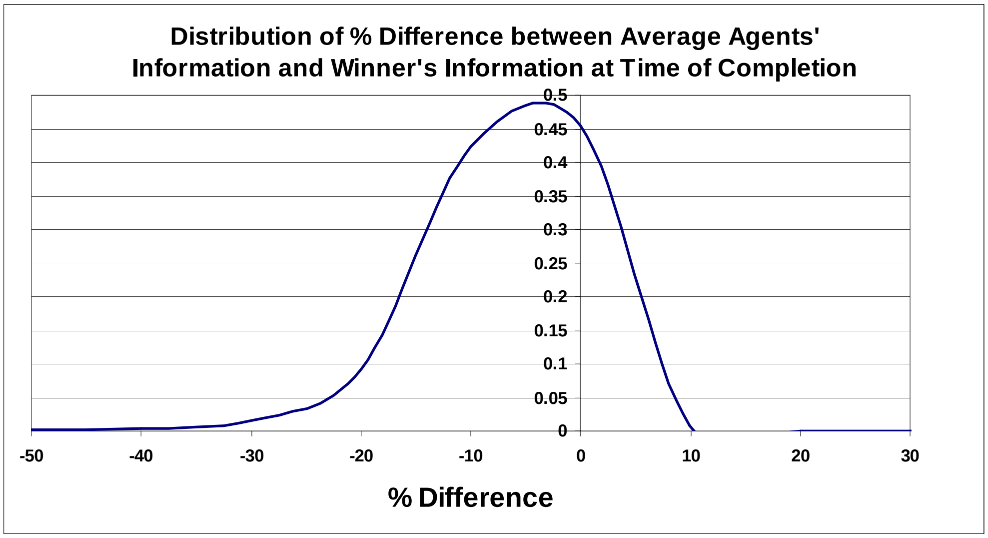


Figure 11. Distribution of Percentage Difference between Average Agents' Information and Winner's Information at Time of Completion

In 12.2% of the cases, the average agents' information value is slightly greater than the winner's at the time of completion. Although the winning agent has slightly lower information value than the average information of the agents, he is able to solve the problem, possibly due to his competency.

In 61.3% of the cases, the average agents' information value is between 0-10% less than the winner's information. For 21.6% of the time, the average agents' information value is between 10-20% lesser than the winner's information.

That is to say, in about 95% of the cases, the average agents' information value is either slightly higher or within 20% of the winner's information, at the solution completion time. This shows that there is high average information among the agents at the time of completion compared to the winner's information value. The edge organization is very robust in this sense.

If the winner is to leave the organization or can no longer participate in solving the problem, the problem can still be solved by the remaining agents without much delay, since the average information value of all the agents is high. Therefore, the efficiency of the edge organization is usually not adversely affected by departure of key personnel due to its robustness.

2. Edge Model 1 – Group Size vs Share Probability Contour Graph

In the contour graph, the various contour lines show paths of equal density or equal completion time in our scenario. The density for each point on the grid is estimated by taking a weighted average of the points in the neighborhood, where the weights decline with distance [JMP User Manual, 2005]. Figure 12 shows a plot of the group average share probability vs the group size. The response is the mean of average completion of the 30 replications.

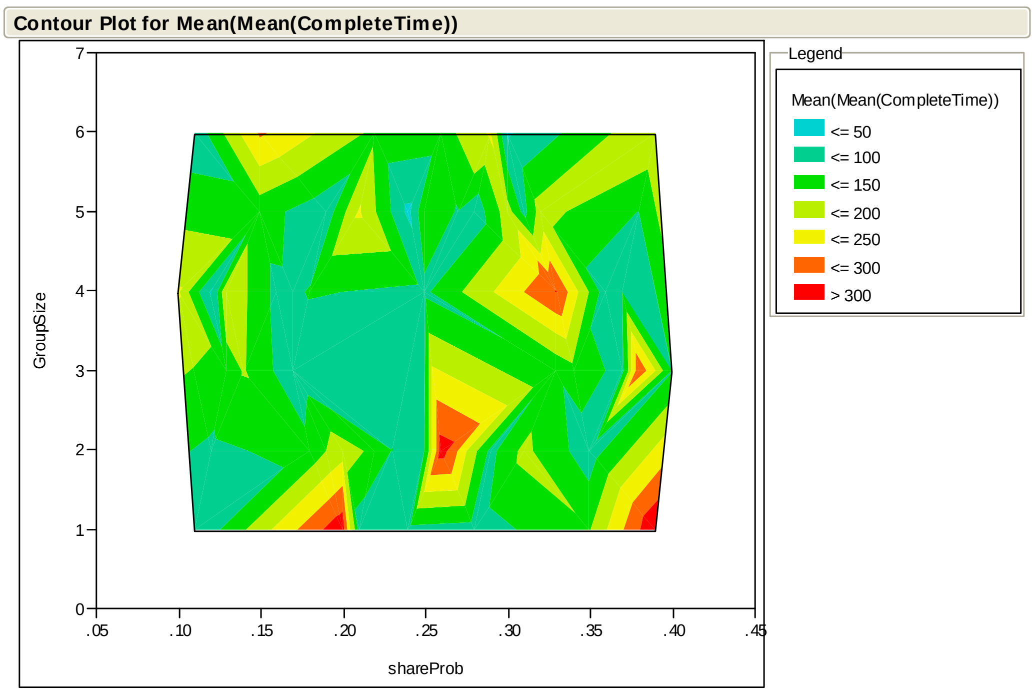


Figure 12. Group Size vs Share Probability Contour Graph

Pockets of long completion time occur at the lower right triangular region in the graph. A small group size with high group share probability leads to long completion time. With a larger group size, high share probability has less of an effect on the completion.

With a small group size and high sharing probability, information loading may occur which, in turn, causes the increase in completion time. With a larger group size and higher share probability, information loading will still occur, but the information gain in sharing the information among the group members is more substantial and it reduces the overall completion time.

There seems to be an optimal group size for sharing given a group average share probability, which occurs in the upper left triangular region in the graph. In an efficient edge organization, if the members tend to work in groups rather than together as an organization (as in posting), there is an optimal operating group size that achieves a balance between information value gained through sharing and information loading due to sharing.

3. Edge Model 1 – Competency vs Hoard Probability Contour Graph

At low to mid group's competency level, an increase in hoarding probability generally increases the completion time significantly (Figure 13). If the group of agents has high competency level, an increase in hoard probability has less of an adverse effect on the completion time.

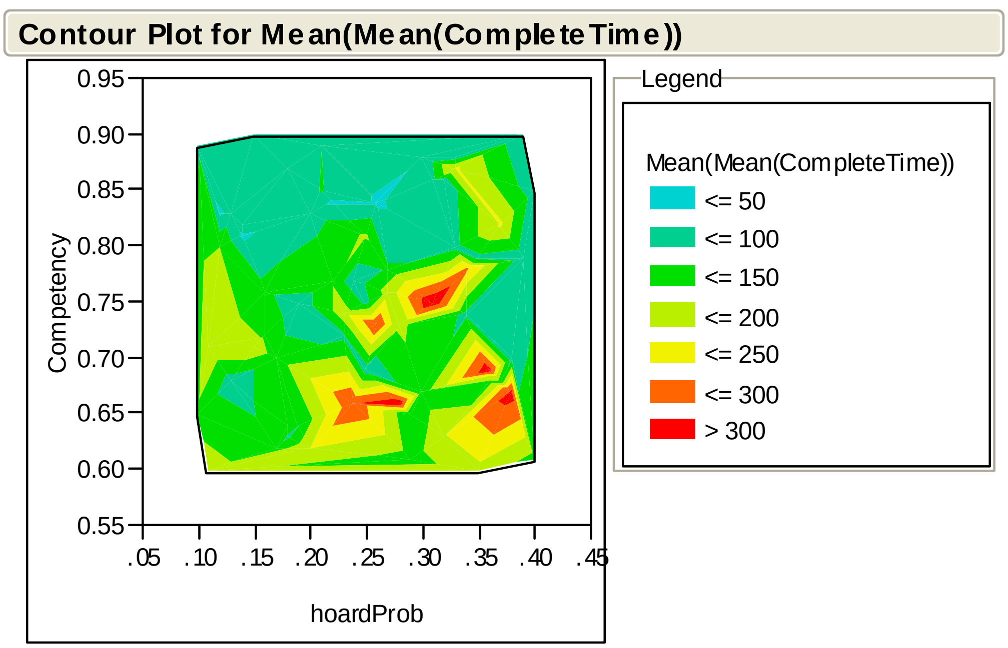


Figure 13. Competency vs Hoard Probability Contour Graph

The competency level of the group is a significant factor in determining the efficiency of the edge organization. If the group has low competency, more sharing among the agents is able to improve the efficiency of the organization significantly.

4. Edge Model 1 – Hoard Probability vs Share Probability Contour Graph

As the hoard probability increases, the completion time generally increases (Figure 14). The rate at which the completion time deteriorates tends to increase as the hoard probability increases.

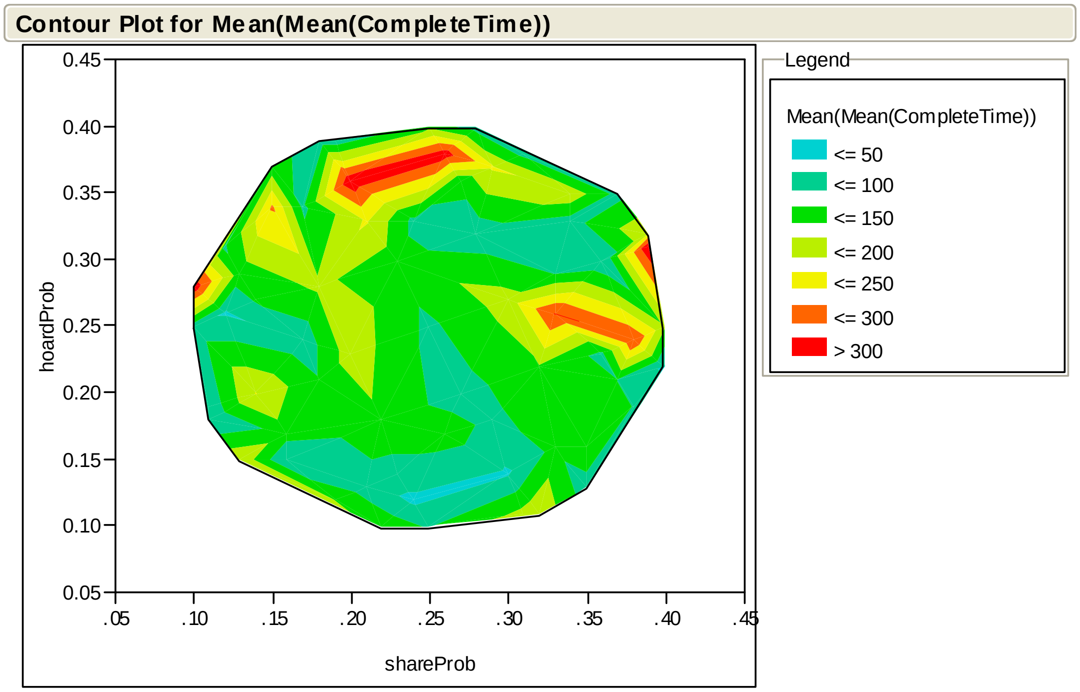


Figure 14. Hoard Probability vs Share Probability Contour Graph

The effect of share probability on the completion time varies differently with different hoard probabilities, and the effect is not obvious. Given any share probability, an edge organization that posts more and hoards less will generally perform better.

5. Edge Model 1 – Message Processing Rate Factor vs Discovery Rate Contour Graph

For the task-related factors, message processing rate and discovery rate, it is obvious that low discovery rates or low message processing rates lead to high completion times. Figure 15 shows this relationship.

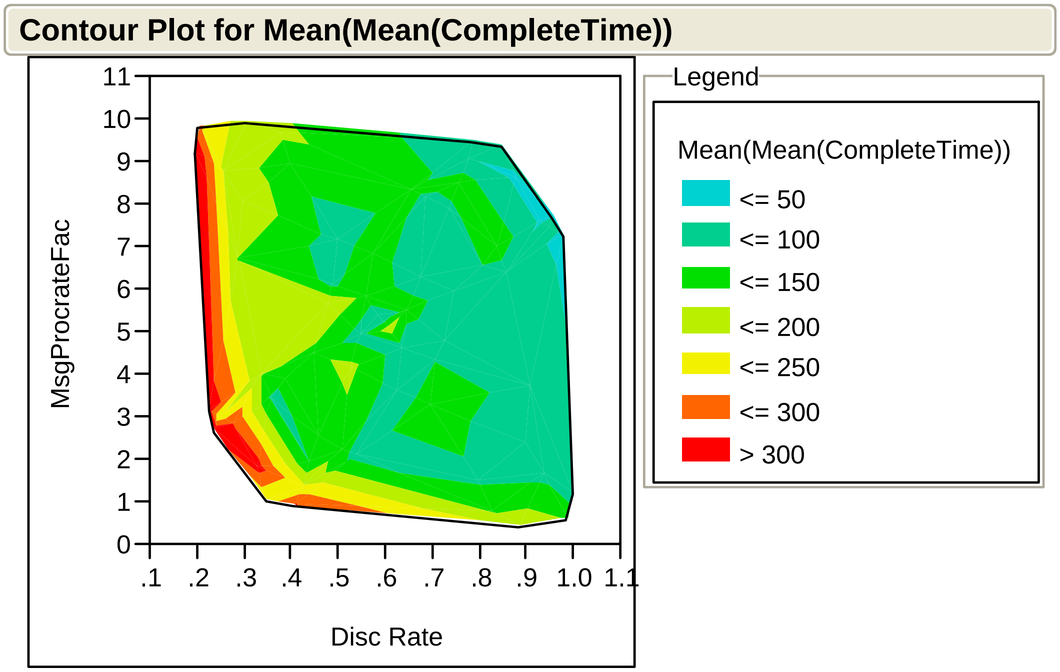


Figure 15. Message Processing Rate vs Discovery Rate Contour Graph

There seems to be a threshold for both the discovery rate and the message processing rate, beyond which there is a significant improvement in the completion time. For an edge organization to be efficient at solving a task, it is desirable to have sufficient resources (people, technology etc.) to improve the discovery rate and message processing rate beyond these threshold values.

6. Edge Model 1 – Negative Factoid Value vs Number of Negative Factoids Contour Graph

An increase in the total negative factoid value generally increases the completion time, though there are some pockets of low completion time at high negative factoid values (Figure 16). The effect of the number of negative factoids on the completion is neither significant nor obvious.

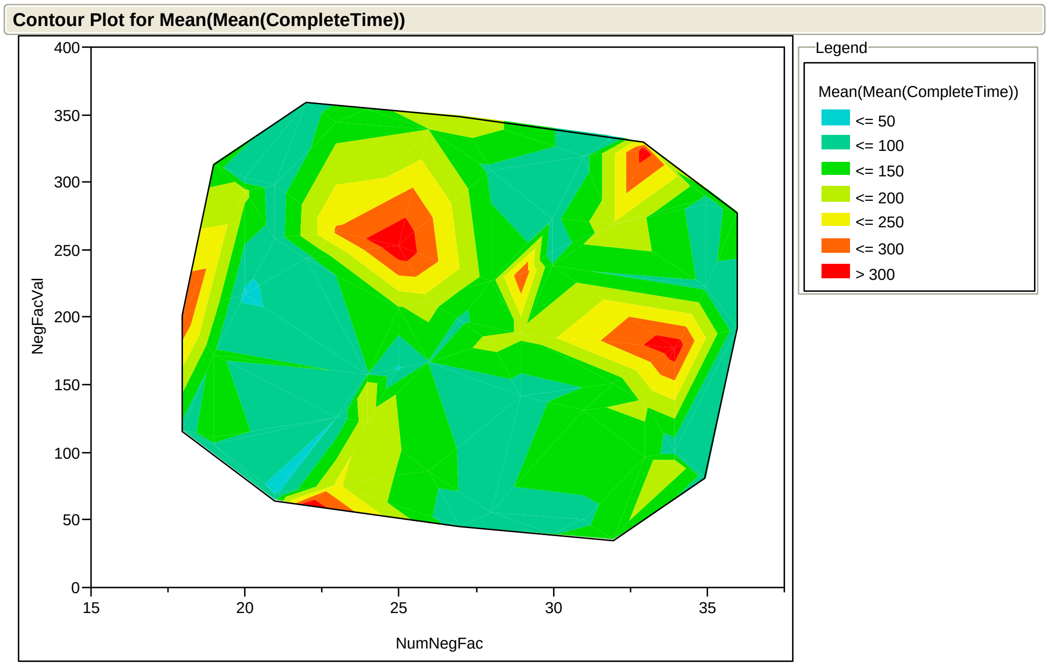


Figure 16. Negative Factoid value vs Number of Negative Factoids Contour Graph

7. Edge Model 1 – Competency vs Total Has-Been-Leader Contour Graph

One of the outputs obtained from the experiment is the number of has-been-leader agents that occurred in each of the simulation runs. The number of has-been-leader occurrences is a stochastic process that essentially depends on the leader criteria rules employed and the characteristics of the agents in the scenario. Given the same initial scenario, a more stringent leader definition will limit the number of leaders that emerge in the simulation.

The has-been-leader count data is evaluated as a predictor of the performance of the edge organization. The contour plot in Figure 17 shows the completion time of the group, given the group's competency and the number of has-been leader occurrences in the simulation run.

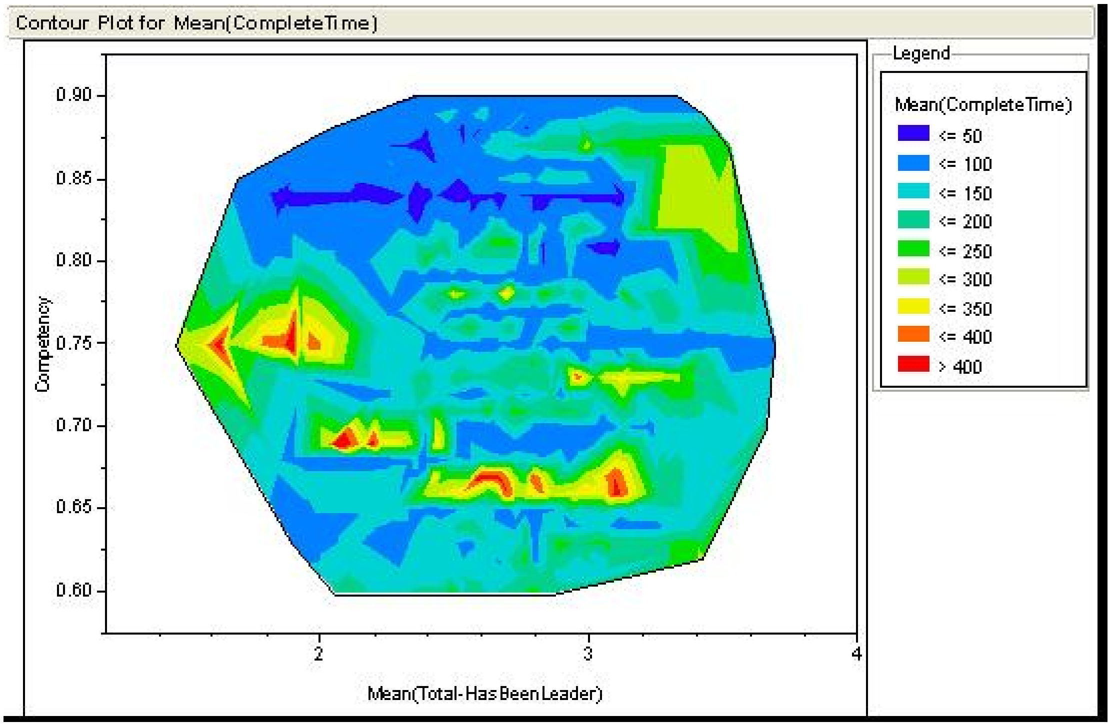


Figure 17. Competency vs Total-Has-Been-Leader Contour Graph

The results show two distinct regions, one with high competency, and another with low competency. In a high competency group, if the number of emergent leaders is high, the completion time is also high; when fewer leaders emerged, the completion time is low.

When a leader emerges, he will receive a stream of factoids from the rest of the agents. In a high competency group, the probability of information overloading will be low, and the leader is able to redistribute the factoids to the rest of the agents effectively through the common area. This redistribution of factoids reduces the completion time.

When many leaders emerge, there will be a longer period of time when there is no leader, since there is a lead time before a leader emerges. With that, the effectiveness of the redistribution of information is reduced which results in a longer completion time.

At mid to low group competency levels, the effect of the number of emergent leaders on the completion time is less obvious. In fact, there are pockets of long completion time in regions where few leaders emerge.

One possible reason is that with many factoids received from other agents the leader agent is not able to process the information quickly enough as information overloads occurs due to lesser competency. There is less effective redistribution of factoids facilitated by the leader.

When more leaders emerge, the factoids sent by other agents are distributed among the emergent leaders and these leaders experience less information overloading. More effective redistribution is possible, which results in the reduction of completion time.

A leader in our scenario aligns the goal for the rest of the agents in the edge organization by having them share and contribute more to the organization. This is done by redistributing the factoids to the common portal.

8. Edge Model 1 – Mean and Box Plot by Group Size

In a mean diamond plot of Figure 18, the center lines of the diamonds are the group means. The top and the bottom of the diamonds form the 95% confidence intervals of the means. If the confidence intervals shown by the means plot on the right hand side of the figure do not overlap, the difference between groups are statistically significant.

In the box plot, the rectangle represents the middle 50% of the data and the range is known as inter-quartile range. The middle line in the rectangle is the median. Each end line extending from the box, known as a whisker, indicates the range of data which fall within a distance equal to 1.5 * inter-quartile range of the upper and lower quartiles. Individual dots beyond the whiskers are outliers.

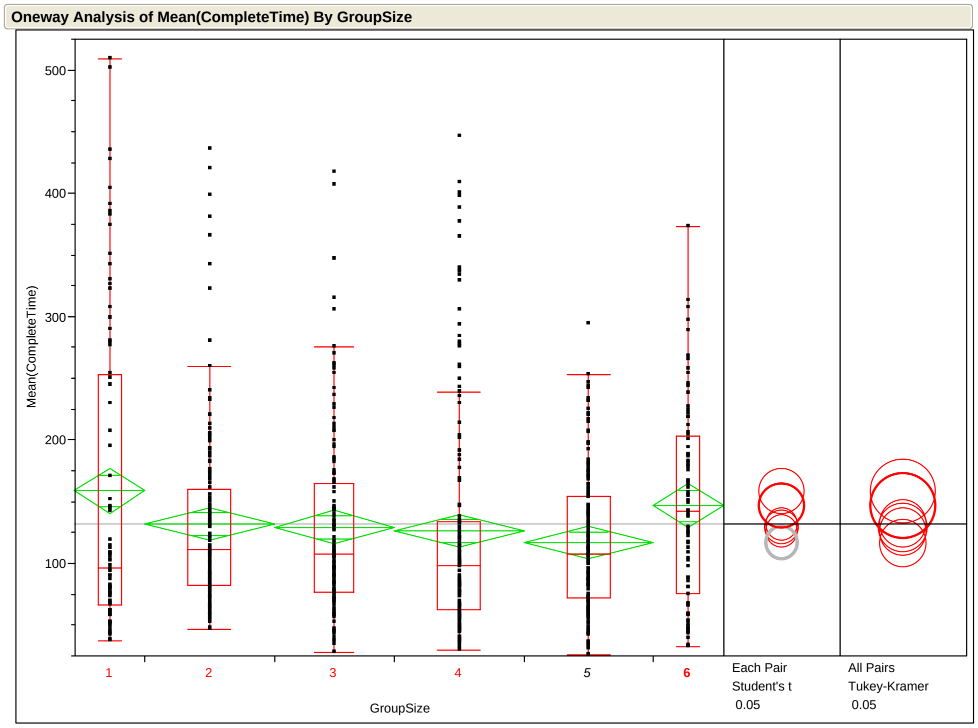


Figure 18. Mean and Box Plot of Completion Time by Group Size

In the simulation, we have group sizes of one (no sharing), two (six groups of two), three (four groups of three), four (three groups of four), five (two groups of five, one group of two) and six (two groups of six). Group one is not considered here, as it represent no sharing, and the interpretation differs from the rest of the group sizes.

Performing each paired t-test on all the group sizes, there is only a significant difference in the mean completion time between groups of size five and six. Using the Tukey-Kramer comparison test, which tends to reduce falsely declaring significance, there is no significant difference between groups of size five and six. There are also no significant differences in the mean completion times between groups of size two, three, four, and six for both tests. Note that for a group size of five, the group structure is slightly different from the rest of the group sizes, as it consists of groups with different sizes.

The distribution of the completion time is generally skewed, i.e., the median is generally less than the mean, especially for the smaller group size. There are more cases with low completion time, and few cases with extremely large completion time. For group size of six, the distribution is less skewed, i.e., the completion time distribution is more symmetric.

Looking at the completion time standard deviations for the different group sizes in Table 7, a group size of five has the smallest standard deviation and a group size of four has the largest standard deviation. It appears that group size five, which consists of groups with different sizes, performs better and is more robust. The effect of different group sizes, however, is not considered in this thesis and is recommended for further study.

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
1	84	159.219	127.079	13.865	131.64	186.80
2	156	132.666	73.424	5.879	121.05	144.28
3	144	129.698	75.427	6.286	117.27	142.12
4	156	126.612	95.001	7.606	111.59	141.64
5	156	116.995	56.483	4.522	108.06	125.93
6	84	147.235	79.218	8.643	130.04	164.43

Table 7. Means and Standard Deviation of Completion Time by Group Size

9. Edge Model 1 – Mean and Box Plot by Switch Task

A mean and box plot of completion time by the switch task flag is shown in Figure 19. Both t-tests and Tukey-Kramer tests show that difference between the switch task factor is significant.

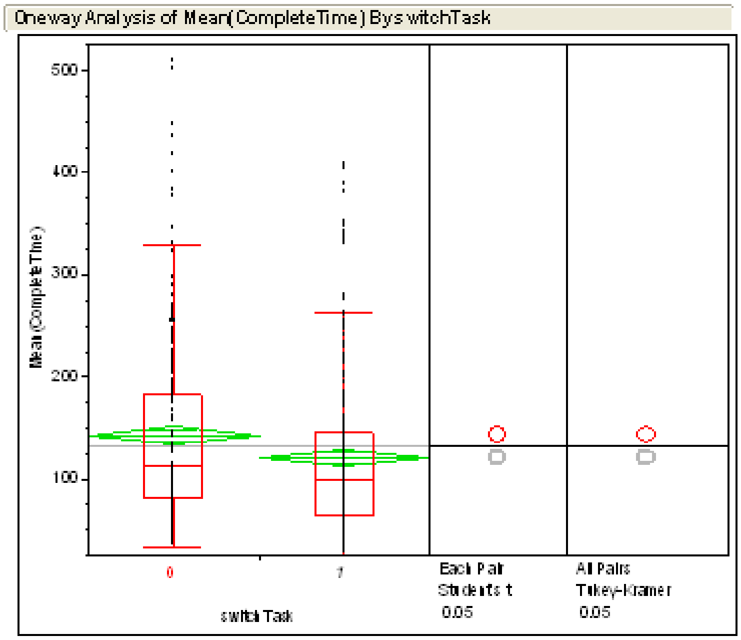


Figure 19. Mean and Box Plot of Completion Time by the Switch Task Factor

Upon completion of a task, if agents focus their effort in other unsolved tasks, the completion time will be reduced. This reinforces the advantage of adaptability in an edge organization vs the specialization of the hierarchical organization.

10. Edge Model 1 – Mean and Box Plot by Solution Threshold

Figure 20 shows a mean and box plot of completion time by the solution threshold. Both t-tests and Tukey-Kramer tests show that there are significant differences between the different solution thresholds.

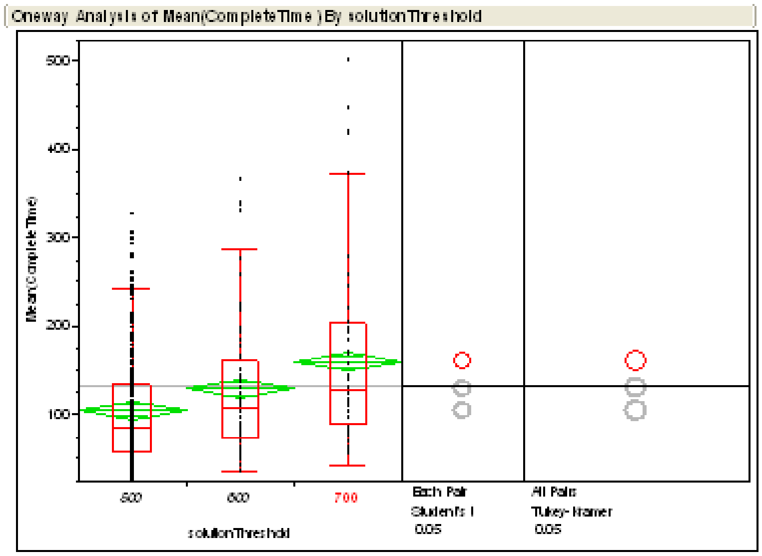


Figure 20. Mean and Box Plot of Completion Time by Solution Threshold Task Factor

From the mean plots, the difference in means between the solution threshold 600 and 700 is larger than the difference between solution threshold of 500 and 600. This is due to the fact that in problems with higher solution thresholds, there are fewer excess factoids and therefore it will take a longer time to solve the problem.

11. Edge Model 1 – Mean and Box Plot of FORMAL GROUPING

A mean and box plot of completion time by formal grouping appears in Figure 21. Both t-tests and Tukey-Kramer tests show that there is no significant difference between the formal grouping and informal grouping.

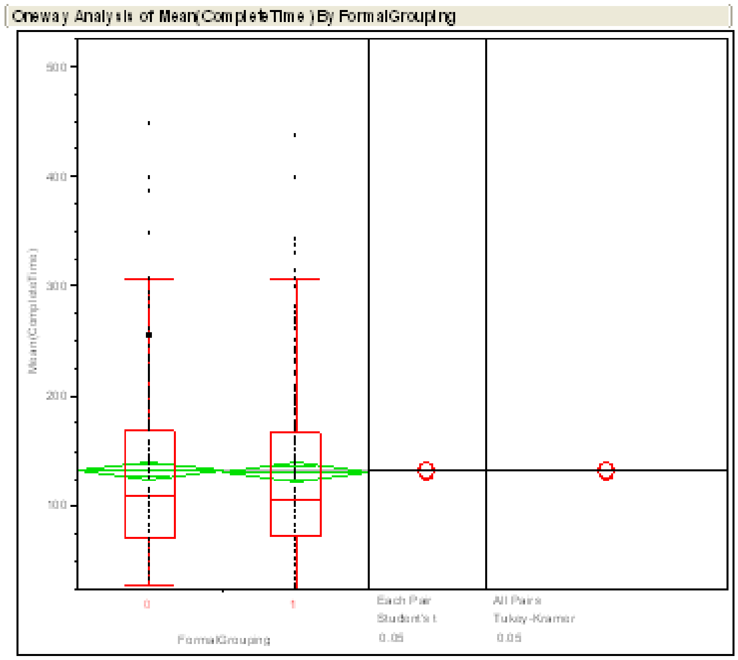


Figure 21. Mean and Box Plot of Completion Time by Formal Grouping

12. Edge Model 1 – Parallel Plots of the primary factors

To gain insight into what are the characteristics of an efficient organization, we look at the parallel plots of the longest and shortest completion time plotted with only the primary factors (Figures 22 and 23).

In the case of the longest completion time, the organization has low post probability, high hoard probability (since it has a group size of 1), low competency and a low number of emergent leaders.

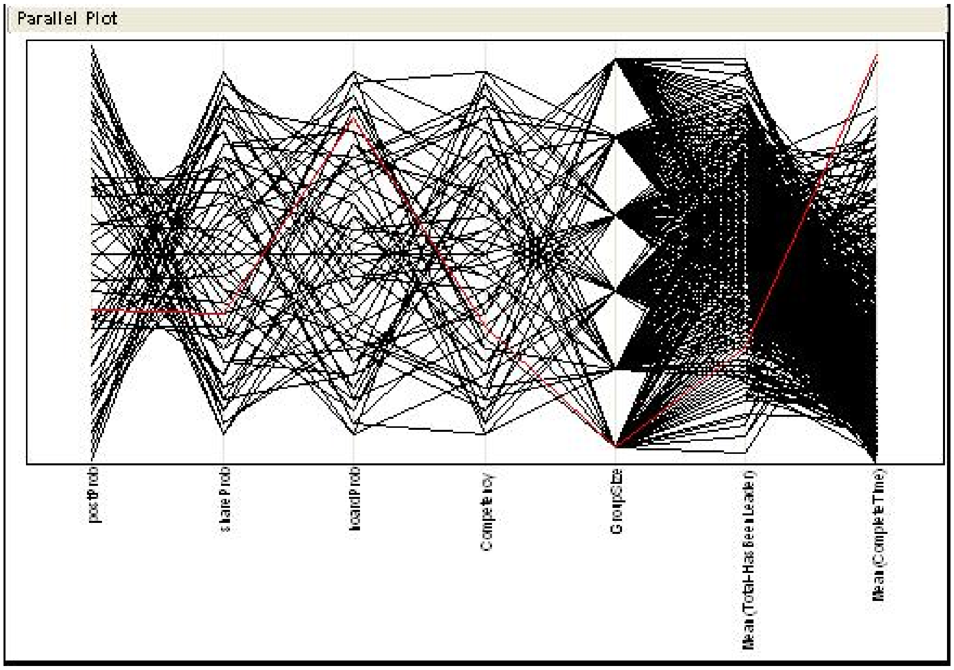


Figure 22. Parallel Plot for the Longest Solve Time

In the case of the shortest completion time, the organization has high post probability, mid sharing probability with group size of five, low hoarding probability, high competency and mid-low number of leaders emerged.

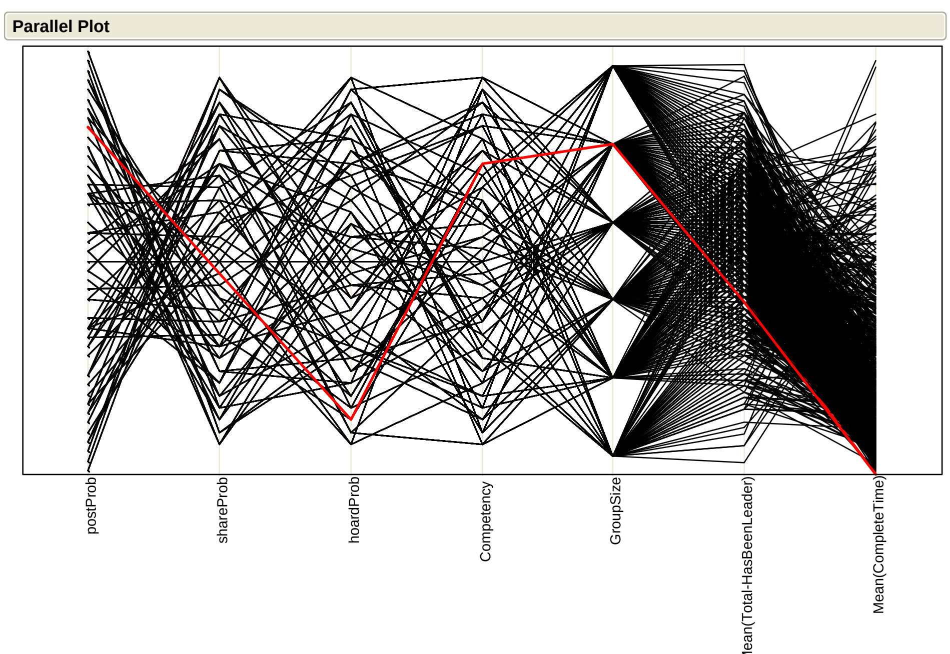
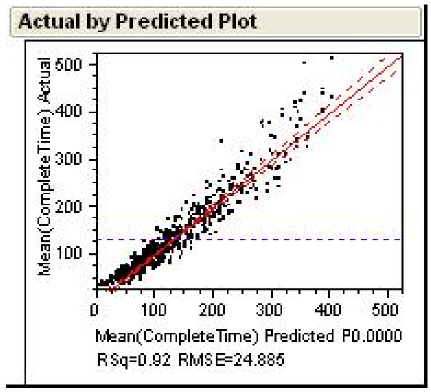


Figure 23. Parallel Plot for the Shortest Solve Time

13. Edge Model 1 – Regression Model

A stepwise regression is performed on the data to filter out the less significant factors using the automatic model-fitting procedure in JMP. From the resulting stepwise regression model, more factors are removed manually so as to have a more parsimonious model. The model in Figure 24 has an R^2 of 0.916, i.e., it is able to explain about 91.6% of the variability in the completion times.



Summary of Fit

RSquare	0.915503
RSquare Adj	0.912701
Root Mean Square Error	24.88516
Mean of Response	132.2017
Observations (or Sum Wgts)	780

Parameter Estimates

Term	Estimate	Std Error	t Ratio	Prob>|t|
Intercept	609.44225	14.81508	41.14	<.0001
postProb	-63.64575	11.39945	-5.58	<.0001
hoardProb	124.63945	15.51112	8.04	<.0001
Competency	-432.0317	14.64537	-29.50	<.0001
GroupSize{1-2&3&4&5&6}	16.769068	1.601691	10.47	<.0001
GroupSize{2&3&4&5-6}	12.336321	1.612984	7.65	<.0001
GroupSize{2&3-4}	-7.686066	1.332009	-5.77	<.0001
GroupSize{2-3}	7.1207016	1.535853	4.64	<.0001
Disc Rate	-257.9682	5.418509	-47.61	<.0001
MsgProcrateFac	-5.796566	0.321869	-18.01	<.0001
solutionThreshold{500&600-700}	-21.34974	0.945082	-22.59	<.0001
solutionThreshold{500-600}	-12.62709	1.091287	-11.57	<.0001
NegFacVal	0.0455251	0.00955	4.77	<.0001
switchTask[1-0]	-21.72998	1.782064	-12.19	<.0001
(postProb-0.49969)*(GroupSize{2-3}-0.01538)	65.316882	16.28214	4.01	<.0001
(hoardProb-0.25015)*(Disc Rate-0.60123)	-485.7386	49.66303	-9.78	<.0001
(Competency -0.75015)*(Disc Rate-0.60123)	672.66969	55.71988	12.07	<.0001
(Competency -0.75015)*(solutionThreshold{500&600-700}-0.33333)	73.38663	10.69782	6.86	<.0001
(Competency -0.75015)*solutionThreshold{500-600}	36.017431	12.35275	2.92	0.0037
(Competency -0.75015)*switchTask[1-0]	64.986003	20.17215	3.22	0.0013
(GroupSize{2-3}-0.01538)*(MsgProcrateFac -5.25015)	-4.338613	0.541043	-8.02	<.0001
(Disc Rate-0.60123)*switchTask[1-0]	36.747359	7.596158	4.84	<.0001
(MsgProcrateFac -5.25015)*(solutionThreshold{500&600-700}-0.33333)	0.9418559	0.339385	2.78	0.0057
(Competency -0.75015)*(Competency -0.75015)	889.95004	151.2293	5.88	<.0001
(Disc Rate-0.60123)*(Disc Rate-0.60123)	429.19169	23.32882	18.40	<.0001
(MsgProcrateFac -5.25015)*(MsgProcrateFac -5.25015)	2.0169431	0.161366	12.50	<.0001

Figure 24. Regression Model

In this model, all factors are significant at a 95% confidence level in the presence of the rest of the factors. Looking at the main factors alone, the post probability, competency, discovery rate, message processing rate factor and switch task factor have negative coefficients for the completion time, i.e., they reduce completion time. The hoard probability, solution threshold and total negative factoid values all have positive coefficients, i.e., they increase completion time. Group size has different effects on the completion time. Share probability, post check rate, number of negative factoids, and formal grouping are dropped from the model. Share probability is not important in the model, in the presence of post and hoard probability.

Figure 25. Interaction Effects

Of the interactions that appear in the model, the more substantial interactions occur between discovery rate and competency level, and between discovery rate and hoard probability, shown in Figure 25. With a low discovery rate, low hoard probability increases the completion time substantially; whereas with high discovery rate, the effect of hoard probability has little effect on the completion time. Similarly with a low discovery rate, low competency substantially increases the completion time; whereas with a high discovery rate, competency has little effect on the completion time.

14. Edge Model 1 – Partition Regression MODEL

Considering only primary factors of interest (post probability, share probability, hoard probability, competency level, number of has-been-leader, group size, number of negative factoids, negative factoid values), a partition regression tree was generated. Using these factors alone, the partition tree shown in Figure 26 achieved an R^2 of 0.72. The first levels of the tree indicate that competency, number of has-been-leader and hoard probability are the most significant factors in explaining the variations in average completion time.

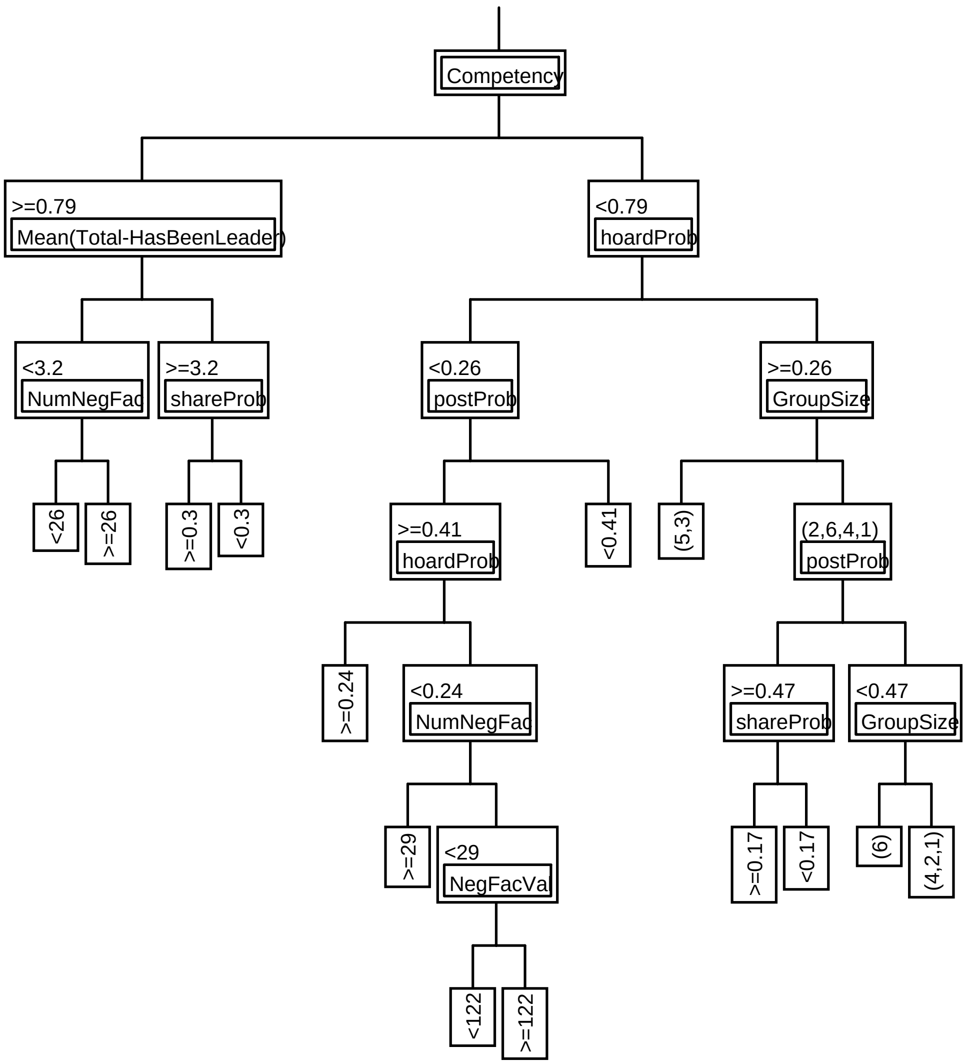


Figure 26. Partition Small Tree view

Using the partition tree model, prediction rules are generated so as to gain some insights about the factors and rules that determine the efficiency of the edge organization. These prediction rules are provided in Figure 27.

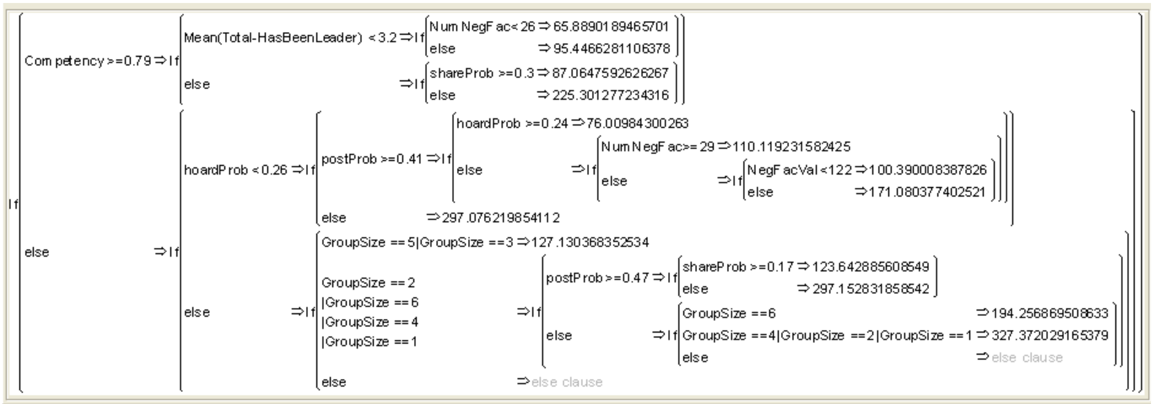


Figure 27. Prediction Rules From Partition Model

A few of the “leaves” on this regression tree are of particular interest. From the partition model, the shortest mean completion has the following rules:

- competency level must be more than 0.79
- number of leaders emerged is less then 3.2
- number of negative factoids is less than 26.

Another rule for an efficient organization

- competency level must be more than 0.79
- number of leaders emerged is more then 3.2
- share probability is more than 0.3.

Long completion times resulted when:

- competency level is less than 0.79
- hoard probability is more than 0.26
- post probability is less than 0.47
- group size of 1, 2 or 4,

or

- competency level is less than 0.79
- hoard probability is less than 0.26
- post probability is less than 0.41.

The last rule is probably due to information overloading. The agents are less competent, share less as an organization, hoard less, and they prefer to work in groups. With low competency level, the agents need more time to process the information they receive from their peers, and not all information is new to the receiving agent. Since the share probability is high, information overloading may occur and this reduces the overall completion time of the organization.

The partition tree also show that the governing rules between competent and not so competent groups are quite different. For example, with high competency groups, the number of leaders which emerge will have an effect on the completion time; whereas in low competency groups, hoard probability has a more significant effect on the completion time.

15. Edge Model 1 – Winner Analysis

A winner in the model is defined as the agent that solves a category of the problem. So in a simulation run it is possible to have from one to four winners, since there are four categories of problems.

In the model, agent characteristics are varied uniformly (+- 0.1) from the mean, modeling similar but not identical characteristics and traits among the people in an organization. It is unlikely to find an agent that has an extreme behavior within an organization. For example, it is unlikely to find an agent who hoards information completely in an organization where everybody else shares, because of organization culture, peer pressure, implicit rules of the organization etc. Other factors such as competency differences among the agents should also not vary too much, since agents in the same organization working on similar jobs should have similar (or minimum) competency levels.

The distribution of the difference between the winner's hoard probability and the actual average hoard probability of the remaining agents is plotted in Figure 28.

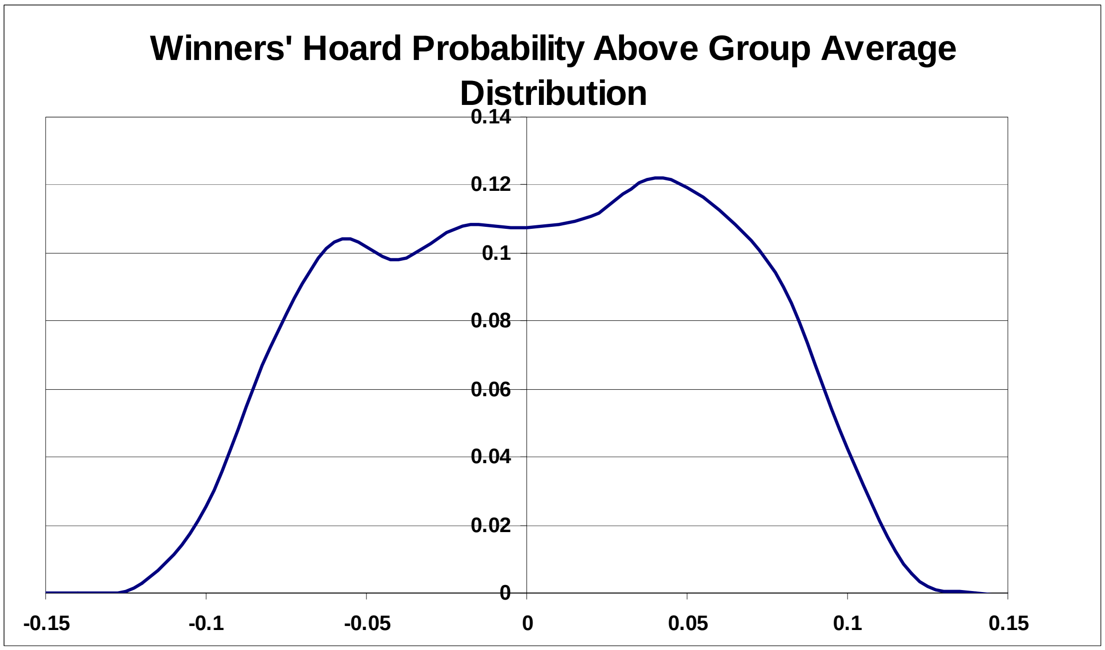


Figure 28. Winner's Hoard Probability above Group Average Distribution

It is observed that about 53.9% of the winning agents have hoard probability above the group average of the remaining agents. This is a small but noticeable difference in a group where agents vary only slightly. Performing a hypothesis test on the winner's hoard probability above the group average, we have

$$
\begin{aligned}
&\bar{x} = 0.0048; s = 0.0547 \\
&H_0 : \mu = \mu_0 = 0 \\
&H_a : \mu \neq \mu_0 \\
&Z = \frac{\bar{x} - \mu_0}{s / \sqrt{n}} = \frac{0.0048 - 0}{0.0547 / \sqrt{93600}} = 26.8 > Z_{0.025} = 1.96 \\
&\Rightarrow reject \quad H_0
\end{aligned}
$$

Because of the large number of data points and above average sample mean, we obtained a test statistic of 26.8. The null hypothesis is easily rejected, indicating that the winner's hoard probability is generally higher than the group average.

The distribution of the difference between the winner's share probability and the actual average share probability of the remaining agents is plotted in Figure 29.

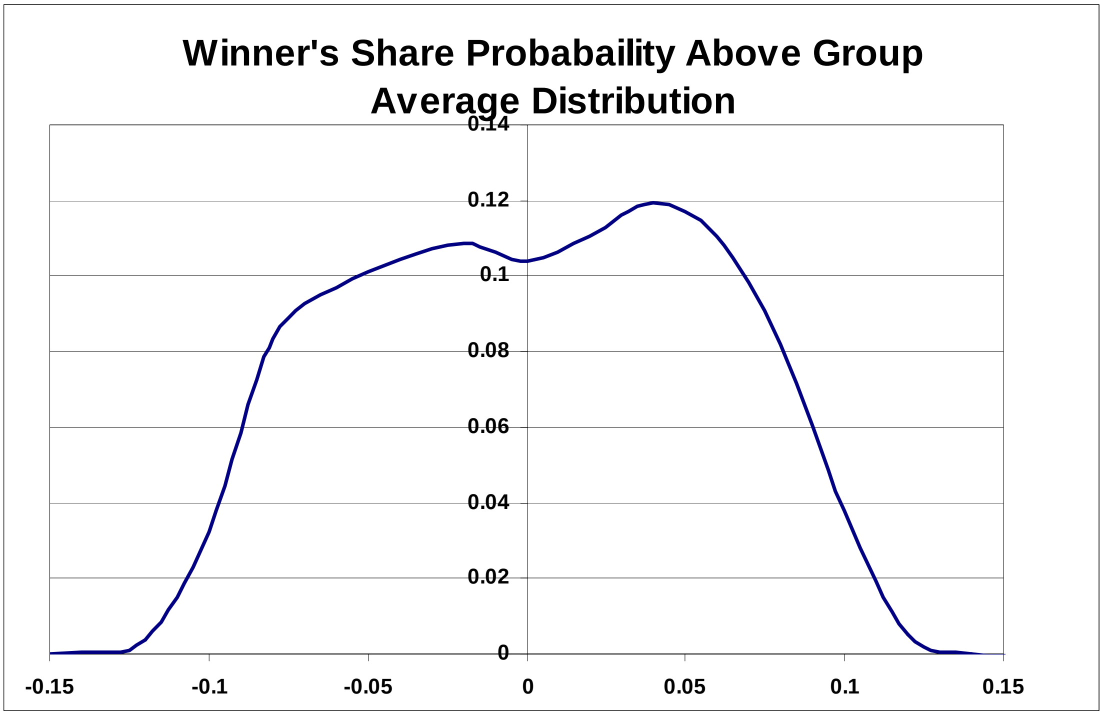


Figure 29. Winner's Share Probability above Group Average Distribution

It is observed that about 51.6% of the winning agents have share probability above the group average. This is a small difference. Performing a hypothesis test on the winner's share probability above the group average, we have

$$
\begin{aligned}
&\bar{x} = 0.0019; s = 0.0553 \\
&H_0 : \mu = \mu_0 = 0 \\
&H_a : \mu \neq \mu_0 \\
&Z = \frac{\bar{x} - \mu_0}{s / \sqrt{n}} = \frac{0.0019 - 0}{0.0553 / \sqrt{93600}} = 10.5 > Z_{0.025} = 1.96 \\
&\Rightarrow reject \quad H_0
\end{aligned}
$$

Because of the large number of data points and above group average sample mean, we obtained a test statistic of 10.5. The null hypothesis is rejected, indicating that the winner's share probability is generally higher than the group average.

The distribution of the difference between the winner's post probability and the actual average post probability of the remaining agents is plotted in Figure 30.

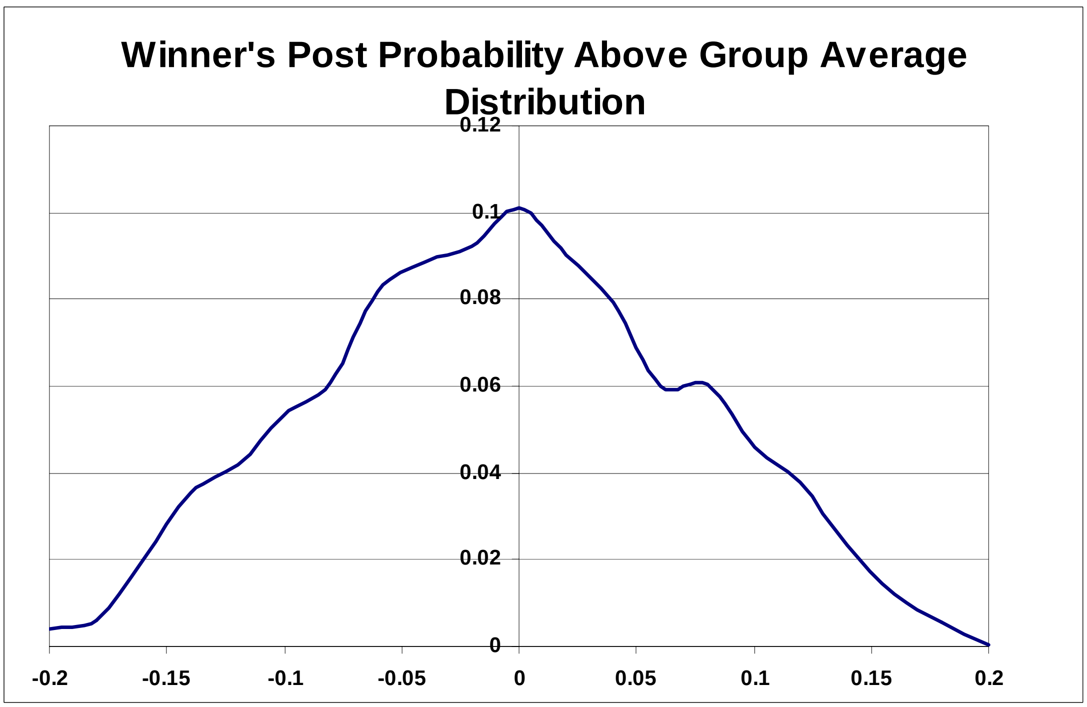


Figure 30. Winner's Hoard Probability above Group Average Distribution

It is observed that about 53.6% of the winning agents have post probability below the group average. This is a small but noticeable difference. Performing a hypothesis test on the winner's hoard probability above the group average, we have

$$\bar{x} = -0.0068; s = 0.0789$$
$$H_0 : \mu = \mu_0 = 0$$
$$H_a : \mu \neq \mu_0$$
$$Z = \frac{\bar{x} - \mu_0}{\sigma / \sqrt{n}} = \frac{-0.0068 - 0}{0.0789 / \sqrt{93600}} = -26.4 < -Z_{0.025} = -1.96$$
$$=> reject \quad H_0$$

Because of the large number of data points and below group average sample mean, we obtained a test statistic of -26.4. The null hypothesis is rejected, indicating that the winner's post probability is generally lower than the group average.

The distribution of the difference between the winner's competency and the actual average competency of the remaining agents is plotted in Figure 31.

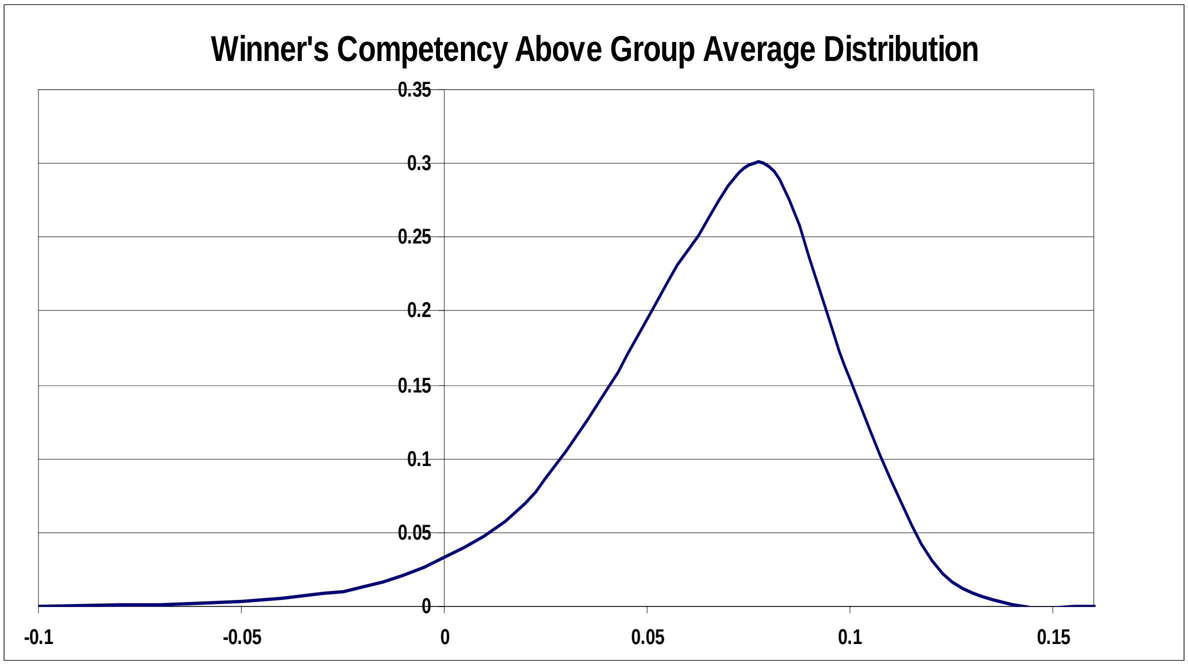


Figure 31. Winner's Competency above Group Average Distribution

It is observed that about 96.2% of the agents have competency above the group average. This is a big difference. Performing a hypothesis test on the winner's competency above the group average, we have

$$\bar{x} = 0.0642; s = 0.0307$$
$$H_0 : \mu = \mu_0 = 0$$
$$H_a : \mu \neq \mu_0$$
$$Z = \frac{\bar{x} - \mu_0}{\sigma / \sqrt{n}} = \frac{0.0642 - 0}{0.0307 / \sqrt{93600}} = 639.8 > Z_{0.025} = 1.96$$
$$\Rightarrow reject \quad H_0$$

Because of the large number of data points and above average sample mean, we obtained a test statistic of 639.8. The null hypothesis is easily rejected, indicating that the winner's capability is higher than the group average.

In essence, the winning agents tend to be more competent, hoard more, and post less than the group average.

16. Edge Model 1 – Leader and Winner Analysis

The total number of has-been-leaders in all simulation runs (including all replications) is 62,880, giving an average of 2.7 leaders per simulation run or per task.

The total agents in all tasks is (780 design points * 30 replications) * 12 agents = 280,800. Therefore, about 22.4% of agents are leaders in all the tasks. There are 17,773 leaders who are winners in all tasks, i.e., 6.3% of the agents are both has-been-leaders and winners. Therefore, the probability an agent is a winner, given that he is a leader, is

$$P(winner \mid leader)$$
$$= \frac{P(winner \,\&\, leader)}{P(leader)}$$
$$= \frac{0.063}{0.224}$$
$$= 0.28$$

Assuming that any agent has equal chance to become one of the four winners per task, then the probability that an agent is a winner is $1-\left(\frac{11}{12}\right)^4 = 0.29$. Therefore, it seems that the probability of becoming a winner is not enhanced when the agent is a leader.

17. Edge Model 1 vs Edge Model 2

Edge model 2 has the same design points as edge model 1, i.e., the group means of the agent characteristics in the two models are the same. The difference is that in the edge model 2, all the agents have exactly the same post probability, share probability, and hoard probability, which are equal to the respective group means; whereas in edge model 1, the agents' characteristics have a uniform distribution of $\pm$0.1 around the group means.

Comparing some of the contour plots obtained from the two edge models, we obtained two sets of similar plots with the same shapes and structures (Figure 32). The main difference is that the edge model 2 seems to have larger areas of longer completion time than the edge model 1, i.e., edge model 1 generally performs better than edge model 2 at the same design points.

Given two edge organizations which have the same group means in terms of agents' characteristics, the group with more variations in the agents' characteristics tends to perform better. The group performance seems to be strongly influenced by the performance of the best members of the group. Allowing variability, therefore, improves the overall performance of the group.

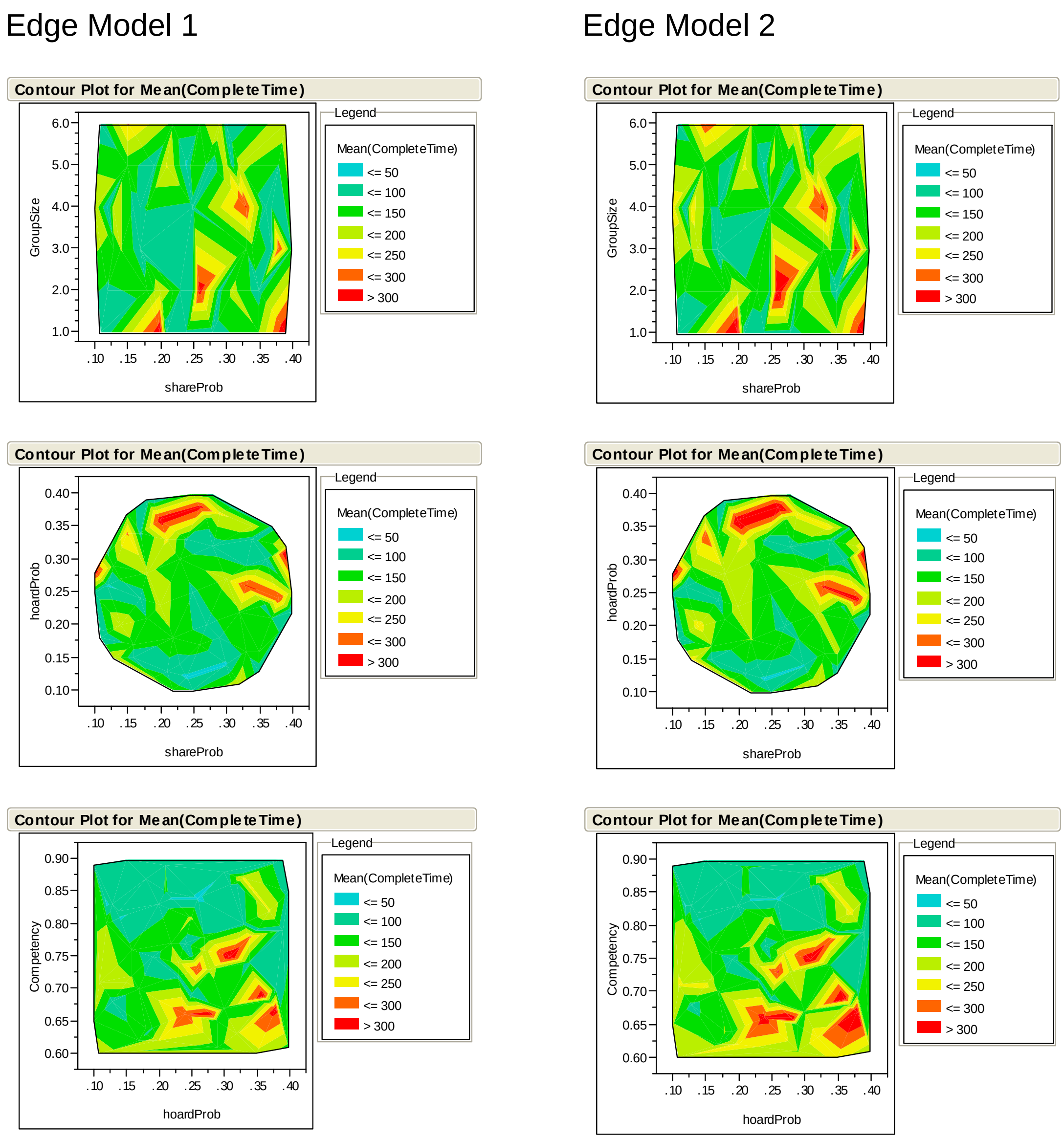


Figure 32. Comparison of Contour Plots between Edge Model 1 and Edge Model 2

18. Edge Model 1 vs Hierarchical Model

In the hierarchical model, the task-related factors (normalized discovery rate, posting check rate factor, message processing rate factor, number of negative factoids, total negative value) and competency of normal agents are set at similar ranges to those in the edge model. The share probability is set to zero, hoard probability to a smaller range, and competency of the fixed leader to one.

The purpose of the changes is to model some of the inherent characteristics of the hierarchical structure.

Edge Model 1 Hierarchical Model

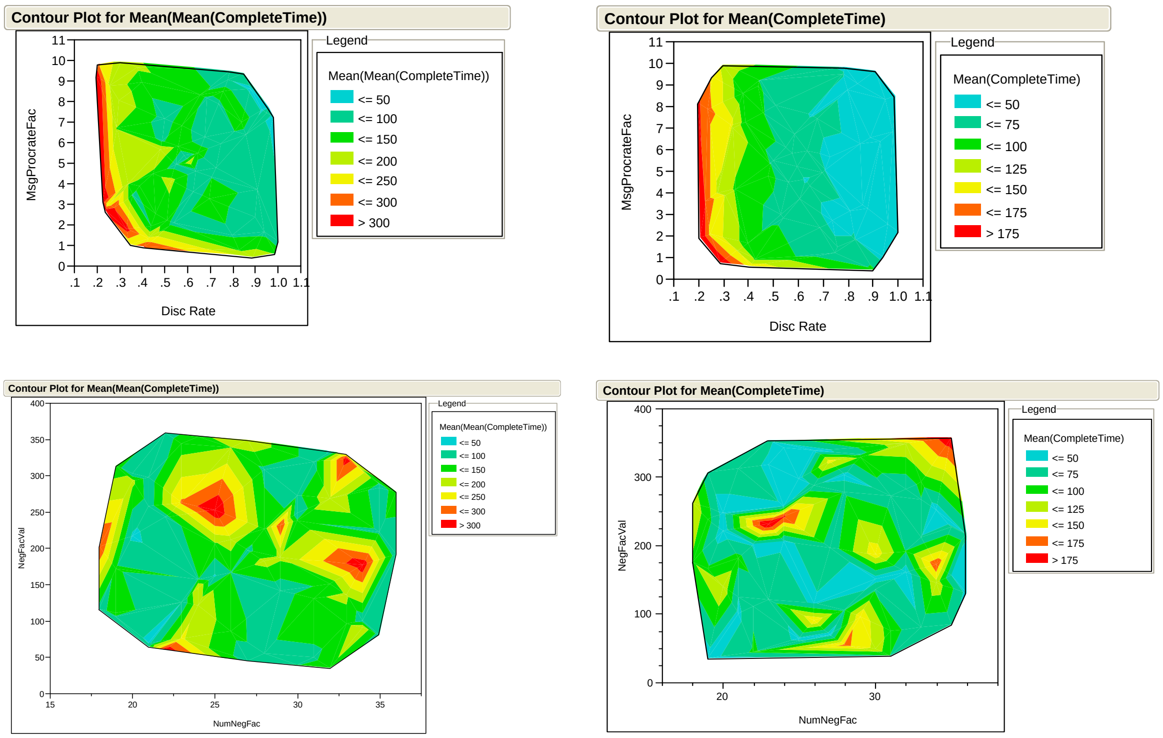


Figure 33. Comparison of Contour Plots between Edge Model 1 and Hierarchical Model

The two set of plots in Figure 33 do not generally have similar structures or shapes. The hierarchical model generally performs better than the edge model, and in some cases is almost twice as fast at solving the problem. Looking at the message processing rate vs discovery rate plots, the completion time in the hierarchical model varies more smoothly with changes in the two factors than the edge model. This is because the hierarchical organization is more systematic and less dynamic, and the changes in these factors affect the completion time in a more proportional and orderly way.

VI. OPERATIONAL INSIGHTS

A. CHAPTER OVERVIEW

Chapter VI gives a summary of results obtained and discusses some operational insights.

B. FACTORS DRIVING PERFORMANCE

Looking at the primary factors, the performance of an edge organization is improved if the people in the group are competent, work together as a whole (in our scenario, post more) and hoard less. Competency is especially important, as it affects discovery rate, message processing rate, and information interpretation. If the organization has high competency, the efficiency of the organization is less affected when people hoard information. If the organization has low competency, hoarding will have a significant adverse effect on the efficiency of the organization.

When people prefer to work in groups, i.e., sharing with peers in our scenario, the group size has some effects on the overall efficiency of the organization with respect to the share probability. There seems to be an optimal size which balances between information overloading, which increases with sharing and group size, and information gain through more sharing. The density of the network, or the group size in our scenario, affects the plecticity of a network [Perry & Moffat, 2004]. In addition, the traffic in the network, determined by the share probabilities of the agents and how fast the agents process the information, has a significant effect on the performance of the organization.

When considering the variability across all the factors in our experiments, group size has no significant effect on the performance of the edge organization. However, group size five, which consists of groups with different sizes, seems to perform slightly better and has less variability.

For task-related factors such as the normalized discovery rate and message processing rate, there is a threshold beyond which the performance of the edge organization is improved tremendously. Resources such as technology and people can be used to mitigate the adverse effects of task-related factors. An increase in the total negative value of factoids generally increases the completion time of the problem, whereas the effect of the number of negative factoids on the completion time is less obvious.

In an edge organization with high competency level, the performance of the organization is improved when there are fewer emergent leaders during the discovery process. In an organization with mid-low competency level, completion time is reduced if there are more emergent leaders. This is essentially a balance between having information overloading with fewer emergent leaders versus longer periods of time without a leader when there are more emergent leaders. Competency, which reduces the probability of information overloading, therefore influences the effects of emergent leaders on the completion time (see Chapter V for a more detailed discussion). If the message processing rate is not a concern or competency is high, it is beneficial to have a single emergent leader to align the goals for the group. This also highlights the importance and contribution of a leader in an edge organization.

As the agents in an edge organization have no specialization, the ability of the agents to switch between tasks, especially when a particular task is completed, makes the edge organization more robust and efficient. This highlights the advantage of adaptability of the edge organization and is a feature usually not found in hierarchical organizations.

With fewer total available factoids, more time is required to solve the problem because the effective discovery rate of the factoids is reduced as more factoids are discovered by the agents. Encountering such a task, completion time could be reduced with better resources (e.g., better technology, more people, etc.), that will eventually improve the task-related factors.

For our scenarios, there is no significant difference between assigning the agents to formal groups, or allowing them to form informal groups if they decide to work with selected peers. This may be due to the way groups are modeled.

From the partition regression model on the primary factors, an efficient edge organization is one that has high average group competency, few emergent leaders, and a task which has low number of negative factoids. With high competency but more emergent leaders, a high sharing probability also produces a more efficient organization. Low competency, high hoard probability and low post probability are common characteristics of an inefficient edge organization.

Note that not all the rules for achieving high performance in edge organizations are controllable. These include rules involving the number of negative factoids available and number of emergent leaders. However, the characteristics of low performance edge organizations are factors that could be set, or at least influenced, by organizational policies.

C. HIERARCHICAL MODEL

Performing the same information gathering task, a hierarchical organization performs better than the edge organization, as the model assumes that agents in the hierarchical organization are trained and specialized to perform the specific tasks. The task is also considered as stable and routine. In some cases, the hierarchical organization outperforms the edge organization by as much as two times.

Furthermore, with the advancement of information technology, some disadvantages inherent to the hierarchical organization are also reduced. For example, a tendency for information to tend to stay at the top of the hierarchical structure may not necessarily be true now, since information exchange technology has facilitated the information flow process in the hierarchical organization as well.

Despite the greater flexibility of edge organizations, hierarchical organizations should not be written off, as they have a distinct advantage of efficiency and better control in specialized tasks. This is greatly enhanced by the advancement of information technology.

D. ROBUSTNESS

In an edge organization, the average information an agent possesses at time of completion is very high. Therefore, if the winner agent is not available, e.g., leaves the organization or takes leave of absence, the efficiency of the organization is not severely affected. This shows the robustness of the organization and highlights the fact that in edge organizations, everybody (as a group) is important but nobody (as individual) is important.

E. VARIABILITY IN AGENTS' CHARACTERISTICS

Given two groups of agents with the same group means in terms of agents' characteristics, the group with a larger variability will generally perform better. This highlights the fact that, when agents with better traits (in terms of solving the problem) work with agents that are not as good, better performance could be obtained than from a group that have average traits working together. The group performance seems to be strongly influenced by the performance of the best members of the group. Allowing variability, therefore, improves the overall performance of the group.

F. REWARD SYSTEMS

In a group where all agents are similar in terms of behavior characteristics, competency etc., an agent which hoards a bit more and posts a bit less information than the rest has a higher probability of becoming a winner. Winners are also significantly more competent than non-winners.

If the reward system of the edge organization encourages winners, this will cause the agents to hoard more and post less information in order to have a higher probability of becoming a winner. An increase in hoard probability and a reduction in post probability, on the other hand, reduce the performance of the edge organization. This results in a vicious cycle that reduces the overall efficiency of the organization, as shown in Figure 34.

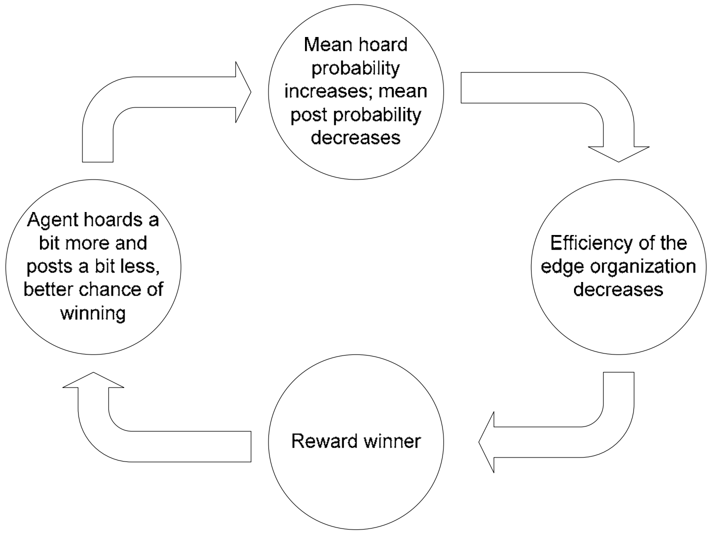


Figure 34. Inefficient Reward System

A better reward system is one that encourages sharing with all in the group, i.e., post more and hoard less information. There should be little or no incentives and headlines for winners. Instead, rewards and incentives given to a successful group will encourage information sharing. This will lead to a virtuous cycle which results in an improved efficiency for the organization, as illustrated in Figure 35.

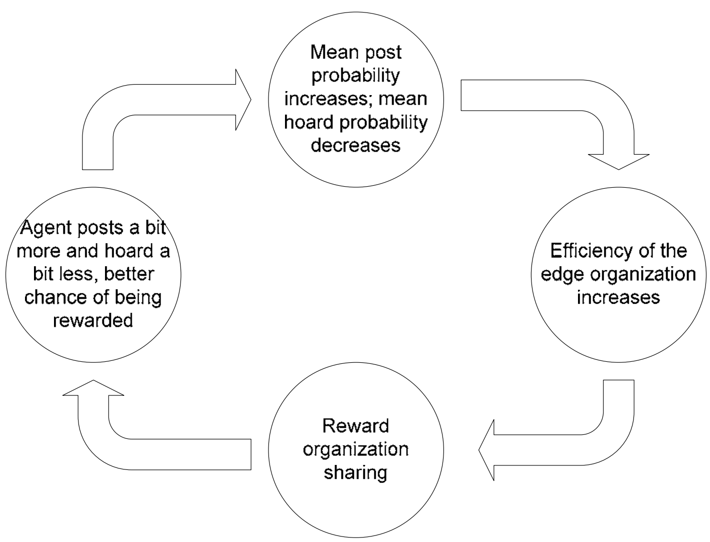


Figure 35. Efficient Reward System

Competency is the only factor that enhances both the chance of becoming a winner and the efficiency of the organization (Figure 36). The reward system should also reward people of high competency. Rewards could also be given to the winner not for winning, but for his competency. This is difficult to achieve in practice. However, it suggests that organizational efforts in both hiring and training practices may be very worthwhile.

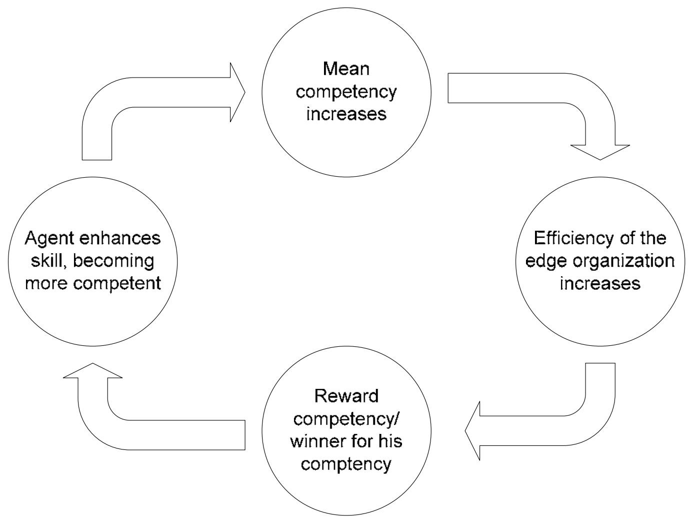


Figure 36. Reward System for Competency

The edge organization should also inculcate organization culture that values education and training, and also places emphasis on having people that regard the organizational goals as their primary goals. The emergent leader also plays an important role in the organization, as he tends to align the organizational goals and achieve a better overall efficiency for the organization. He should be rewarded for his contribution.

In the real world, it is not easy to identify an individual's competency. It may also be difficult to identify an emergent leader in an autonomous edge organization, as there is no "I" in "team". The amount of training a person has, the amount of knowledge shared, and peer appraisal by all the agents in the organization may be suitable surrogate measures.

G. DIRECTIONS FOR FUTURE RESEARCH

The effects of having agents belong to different groups, and the effect of different group size on the completion time, could be looked into in more detail. In an organization where sharing is prominent, the characteristics of the person that belongs to many sub-groups may have significant effect on the completion time.

A single factoid that is composed of different categorical types could be used instead of the one-factoid-one-type information used in our experiments. Rules where agents can decide only to share some parts of the factoids received could be explored. More specific posting and sharing rules could also be modeled, for different specific scenarios.

A reward system that rewards winners and/or organizations could be modeled and implemented, to explore how the various reward systems will influence the performance of the organization over a series of tasks.

Finally, the effect of absence of personnel at random times could also be modeled to give better insights about the robustness of edge organizations.

H. SUMMARY

This thesis shows that agent-based simulation provides a powerful tool for exploring the performance of complex organizational dynamics. When used in conjunction with efficient experimental designs, the simulation results yield insights regarding the interplay of task-related factors, agent characteristics, and structural form, in determining how quickly the edge organization can solve tasks.

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

Alberts, David S., Garstka, John J., Stein, Frederick P., Network Centric Warfare, 1999

Alberts, David S., Hayes, Richard E., Power to the Edge, 2003

Atkinson, Simon R., Moffat, James, The Agile Organization, From Informal Networks To Complex Effects And Agility, 2005

Buss, Arnold, Technical Notes On Basic Event Graph Modeling, 2001

Cioppa, Thomas M., Efficient Nearly Orthogonal and Space-filling Experimental Designs for High-dimensional Complex Models, Ph.D. Dissertation, Department of Operations Research, Naval Postgraduate School, September 2002.

Evidence Based Research, Inc, Preliminary Experimental Design Draft v41, 2005

Kleijnen, Jack P. C., Sanchez, Susan M., Lucas, Thomas W., Cioppa Thomas M., A User's Guide To The Brave New World Of Designing Simulation Experiments, INFORMS Journal on Computing, Vol. 17 No. 3, pp. 263-289, 2005

Leavitt, Harold J., Some Effect Of Certain Communication Patterns On Group Performance, 1951

Perry, Walter L., Moffat, James, Information Sharing Among Military Headquarters, 2004

Sall, John, Creighton, Lee, Lehman, Ann, Jmp Start Statistics 3rd ed, 2005

Sanchez, Paul J., SIMpleKit, (C) 2004, 2005

Srbljinovic, Armano, Skunca, Ognjen, An Introduction To Agent Based Modeling And Simulation Of Social Process, 2003

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Professor Susan M. Sanchez
   Department of Operations Research
   Naval Postgraduate School
   Monterey, California

4. Professor Paul J. Sanchez
   Department of Operations Research
   Naval Postgraduate School
   Monterey, California

5. Professor Arnold Buss
   MOVES
   Naval Postgraduate School
   Monterey, California

6. Kok Meng Chang
   Naval Postgraduate School
   Monterey, California