7-11-2015

An Agent Interaction Mechanism based on Near-Term Analysis

ANIVESH REDDY MINIPURI
University of Windsor

An Agent Interaction Mechanism based on Near-Term Analysis Framework

By

ANIVESH REDDY MINIPURI

A Thesis

Submitted to the Faculty of Graduate Studies

Through the School of Computer Science In

Partial Fulfillment of the Requirements for the

Degree of Master of Science at the

University of Windsor

Windsor, Ontario, Canada

2015

An Agent Interaction Mechanism based on Near-Term Analysis Framework

by

Anivesh Reddy Minipuri

APPROVED BY:

Mohammed Khalid, External Reader
Department of Electrical and Computer Engineering

Dan Wu, Internal Reader
School of Computer Science

Ziad Kobti, Advisor
School of Computer Science

May 25, 2015

DECLARATION OF ORIGINALITY

I hereby certify that I am the sole author of this thesis and that no part of this thesis has been published or submitted for publication.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis and have included copies of such copyright clearances to my appendix.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

ABSTRACT

To estimate the changes of a particular organization under uncertainty is essential. “What-if” some employees leave the company suddenly? “What-if” some of the officers are hurt in a military action? To answer these type of questions, “Near-term analysis” (NTA) framework was previously introduced. It simulates the social dynamic within an organization, isolates the particular agents in it, and calculates the output with a degree called knowledge diffusion However, the drawback for such a tool is that it cannot produce agent interactions for a group size larger than 100. We propose a modified framework that can handle the group size of more than 100 agents in an organization and also produces valid interactions. Our main objective is to make the proposed framework suitable to estimate changes in a large organization and in uncertain situations. The experimental results confirm that our proposed framework outperforms the near-term analysis when it comes to large organizations.

DEDICATION

I dedicate my work to my parents, my sister, my grandparents and my friends who have supported me in every aspect of my life and have consistently shown faith in me which has got me this far in my life. I am grateful to almighty God for His blessings.

ACKNOWLEDGEMENTS

I would like to thank my supervisor, Dr. Ziad Kobti, for giving me the opportunity to get an exposure of the research field. It has been a great experience working under his guidance. His constant motivation, support and faith guided me in successful completion of my thesis. I would also like to appreciate him for all the financial support he has given, which helped me focus on my work and took care of my expenses.

I would like to thank my internal reader Dr. Dan Wu for his support and positive attitude towards my approach. I am also thankful to my external reader Dr. Mohammed Khalid for his suggestions and interest in the approach.

My sincere gratitude goes to Mrs. Gloria Mensah, secretary to the director, who has always helped set up meetings with my supervisor and ensured that he always meets me, despite his busy schedule. In addition, my sincere thanks to Mrs Karen Bourdeau who has always been there to take care of other issues related to my master's degree

Finally, I would like to thank my parents, who have gone through all the ups and downs I faced during my research, and were along with me.

TABLE OF CONTENTS

LIST OF TABLES

FIGURES

CHAPTER 1: INTRODUCTION

1.1 Introduction

1.1.1 What-if in an organization:

In many organizations, situations will change dynamically. As a result, “what-if” analysis is a critical methodology that we need to prepare for future. What-if analysis is particularly important in organizations such as hospitals, corporate offices, intelligence agencies, and the military, among others. Each organization has different threat scenarios. For instance, if we assume a hospital as an organization, What-if some of the physicians in a team are not available for a particular procedure or surgical procedure? We require to recognize whether that team can perform successfully or not. This is one of the threat scenarios in a hospital’s organization. Alternately, an important question for office managers is what-if a company was suddenly affected by a financial crisis. They may be required to lay off some of the employees provided that the performance level of organizations should not negatively impacted because of that activity. A company might also consider what would happen if some of the employees suddenly left the company. Managers would want to know if the performance and organizational structure of the company would be negatively impacted due to the departure. Similarly, if military officers died in a military action senior officers would need to know if the unit would still be dependable. These are some of the possible “what-if” situations in an organization.

In order to answer these types of questions the best method is to replicate the organization structure in real world and test the threat scenarios with replicated experiments in a given environment. We can answer this by two methods: one is using human participants, and another is running the simulations with multi-agent based systems.

1.1.2 Multi-Agent Simulation:

A multi-agent simulation (MAS) is an artificial simulation model comprising several artificial agents. These agents tend to react in a given artificial environment. According to Ferber (2004), an agent can be a physical and virtual entity in any artificial environment communicating with other agents. Such agents interact together to form a multi-agent based system (MABS). This MABS contains an environment, an object and agents. The relation between these entities depends upon the environment scientists choose to study. The agents in multi-agent based systems have relations between various entities and also perform a set of actions. A MABS is used to create an artificial model that could be used to study complex systems. One of the most significant uses of the MAS is to study the organization structure by replicating the original one. Deploying simulations with different parameters on the replicated organization structure gives the opportunity to predict the performance of the organization.

MABS have a number of benefits. It presents a detailed and accurate analogy to human organizations and actors. It can also be used to run the multiple experiments with low cost. MABS are also now being used for theory building and to study organization structures.

1. 2 Current Research Motivation:

To check the performance of an organization, two methods can be used: one involves human participants, and the other involves running simulations using MABS.

When using the first method, researchers need to investigate in the real world to collect the experimental data and perform laboratory experiments. However, these are expensive and it is also impossible compared to simulation. Another main drawback is that there are many real world cases that can be replicated and run the experiments using human participants.

MABS is another method that can be used to check the performance of the organization. As discussed earlier, MABS can be used to run multiple experiments with low cost. There are some tools which using MABS to check the performance of an organization. Some of them are Organizational Risk Analyzer (ORA 2006) and construct (2010). By using Near Term Analysis in ORA, we can check the performance of the organization.

The primary motivation behind this research is that there are many large group organizations in the real world. Although the above tool performs well in terms of producing agent interactions, the drawback for such tool is that it cannot handle the large group size. The authors assumed that the people in an organization have a shared understanding of other agents. This assumption seemed to be reasonable, but when the same assumption applied to large organization, its desirably failed. So we propose a framework that produces agent interactions for large group size and also able to check the performance of organizations for large group size or large organizations.

1.3 Thesis Contribution:

This model predicts the performance of an organization over time for large group organizations. This study assumes that by deploying different parameters and running the experiment several times, one can predict the performance of an organization. The general hypothesis behind this research is that the above mentioned agent-interaction mechanism tools store the information about other agents, thereby forming a transitive memory that makes things complicated for large group organization. This model doesn't have transitive memory and can easily predict the performance of an organization. To confirm our hypothesis, an algorithm is developed that produces agent interactions and also predicts the performance of a large organization.

1.4 Thesis Outline:

At the beginning of the thesis, a problem statement was presented that spoke about predicting the performance of an organization for large group size. In the first part of an introduction, a brief explanation was given to why we need to predict the changes of an organization. Identification of such aspects and implementation of the model using multi-agent based simulation is presented. In chapter 1, an introduction to the research work is presented that explains the relation to our study and computer science. In chapter 2 a brief literature review on entire topic is covered. Chapter 3 contains a detailed description of the near term analysis framework. Chapter 4 presents the proposed approach with detailed description of input to the model, Model description and output. Chapter 5 discuss the experiments that we carried out using our solution framework. In this chapter, we discuss the various experiments with different configurations that we carried out using large datasets. Results and discussions on how the performance of an organization change in large organizations are discussed in chapter 6, followed by concluding remarks on entire research and brief explanation of real-world use of this model.

CHAPTER-2: LITERATURE REVIEW

2. Literature Review:

This section tries to define the work done by other researchers to predict performance of an organization. The "what-if" analysis of an organization can be managed by two methods one including social experiments, including human participants and other is by simulating the organizational structure by applying multi-agent based system.

2.1 Social Experiments:

Weber et al 2004 claims that organization structure and code will have great impact on the performance of an organization. They performed some experiments by altering representation of the organization structure. They represented two organization structures one is centralized firm and other is decentralized firm. "*They performed some experiments by varying software firm structure, from centralized to decentralized and measured the performance of two different kinds"* (Weber et al 2004). The results indicate that centralized firms develop more code rapidly and have more impact on performance measure than that of decentralized organization. However the main drawback is that is very hard to create the centralized organization and perform experiments. It is very expensive and time taking process.

Bought and Meher (2001) studied how an organization structure measures and predicts the performance of an organization with a large set of public organizations. Their main focus is on large organizations. They used span of control as their output measure. This study mainly focused on the effects of span of control in an organization. According to them the impact of organizational structures on performance varies with task difficulty. By using the span of control as our structural variable they found that the structure had a very little impact in improving performance of an organization. The drawback of this method is that it did not focus on "what-if" analysis rather it analyzes social phenomenon in an organization.

Jin and Levis' 1990 also performed some social experiments in order to analyze organizational structure. They experimented on "*how two organization structures in different organizations perform. The two different organization structures are one is parallel one and other is hierarchical one. The performance of each organization was measured in terms of decision maker's time and accurately. The experimental result shows that individual differences between the decision makers have more influence on performance in parallel organizations. According to the author the performance in a parallel organization restricted the choice of decision makers and coupled with individual decisions with decision of other members in an organization*".

Another researcher Graham et al 2005 focused on organizational structure and its performance in a military organization. The output that they used here is shared situation awareness (SSA). The authors did "*regression analysis along with the physical distance and social network distance. They performed their experiments, mainly on three variables, physical distance, social network distance and background similarity. Then they proposed a statistical way to calculate the SSA by using those three variables. This research work shows the important aspects of the use of performance measures*".

However, there are some drawbacks associated with these social experiments. These experiments cannot replicate same as the original one and could not able to represent real world scenarios. Another main drawback is that these experiments cannot be repeated. All these social experiments analyzed social phenomenon in an organization, but none these can able to generate “what-if” scenarios. Therefore the best way is to use simulations by using multi-agent based systems. The advantage of these social experiments is that we can adopt a performance measure and experiment scenario generation and can use them during simulations.

2.2 Multi-Agent Models:

This section tries to define the work done by researchers which measures the performance of an organization by using computer modeling techniques.

Snider et al 2005 “ *simulates the co-evolution of an organization and also its member’s behavior. They used this model for understanding the relation between individual behavior and actions in the social structural organization*”. To simulate an organization's social network was used. The social network measures such as degree centrality, dyadic covariance, etc. Were used in this model. To represent human behavior measures like utilized tendency attributed related similarity, etc. Was also used. They conducted a survey of “*teenage students of a school Cohort in Glasgow about their friendship networks and self-reported smoke and alcohol consumption. The experiments performed through their model indicate that they found network dynamics and homophily tendencies in an organization*” (Snijder et al 2005). The main drawback of this model is that it can explain only interactions between the members of an organization. It cannot able to predict performance of an organization. Another drawback is that use of performance measures. The better use of performance measures yields better results.

There are some MAS models which generate what-if scenarios Jin et al 1998 developed a model which *"aims at developing computational tools to analyze decision making and communication behavior to support organization structure"*. Their model includes a total effort to do the predicted time to complete a project and it measures the process quality, etc. They also found that they can tune up their model to predict the performance of an organization. Furthermore, their model can predict the activity by using "what-if" scenario and can also predict project duration time, etc. By using this model.

Another research team Lin and Carley 1997 designed MAS model to predict organizational performance. In this paper they *"set up computer modeling of an organization's performance based on information processing and resource dependency"*. By using this model they compared some attributes of performance, such as "*time pressure training, organizational complexity, environmental complexity, etc"*. After the comparison they found out that the above mentioned factors are very important which they affect the performance of an organization.

According to Lin and Carley 1997 the importance of an organizational structure and their design will have great impact on organizational performance. Their *" presents the role of organizational design in affecting organizational performance. They designed a computer model called CORP to examine the organizational structure and its performance under test scenarios such as operating in optimal conditions, operating under internal/ external stress. These examples indicate that how MAS can be used to determine which factors and what-if scenarios are important in predicting the performance of an organization" (*Lin and Carley 1997).

Carley et al 2004 designed a framework which collects the data from real world organization and "what-if" analysis with their model. Later they developed this model called an organizational risk analyzer (ORA) by deploying threat scenarios. By using near term analysis tab in an ORA tool we can simulate the organization structure. The authors built a framework that utilizes the existing Multi-agent based model (MAS). "*The MAS that they used are Dynet and their program creates threat scenarios such as isolation of agents and assesses the impacts of the scenarios automatically. The Near-term Analysis simulate the social dynamics within an organization based on the organizational meta-matrix and expected isolation event of agents and it generates its estimation about the degree of knowledge diffusion from the simulation over the simulated time period. This framework is capable of detecting the vulnerabilities of various organizations at various levels*".

This paper demonstrates *"how we can use bridge multi-agent simulation and social network analysis. It also shows the value of data-farming environments by successfully generating and testing multiple what-if scenarios. This framework can be used to predict the impact of corporate personnel movements, removal of terrorists from their networks, etc"* (Carley et al 2006). The detailed information about Near-term analysis was discussed in chapter-3.

Carley et al 2012 designed an extended framework called construct. Construct is a multi-agent network model for the co-evolution of agents and social-cultural environments. It is designed to capture different cultural and technological configurations and also to capture dynamic behaviors in organizations. Construct models groups and organizations as complex systems and captures the variability in human and organizational factors through heterogeneity in information processing capabilities, knowledge and resource. In constructing agents are decision-making units and can represent various levels of analysis, such as individuals, groups or organizations. Construct can produce agent interactions that are representative of communication networks in real-world organizations.

The figure 1 below explains the agent life cycle in construt. In this cycle agent choose interaction partners, communicate, learn knowledge, change their belief about the world, and adopt their networks based on their updated understanding. At the end of cycle agent perform tasks based on their current understanding.

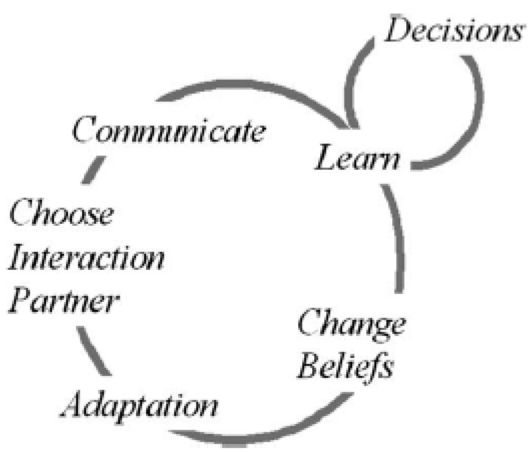

Figure 1 Agent Life Cycle in Construct (Schreiber et al 2013)

An obvious limitation of this construct tool is that the number of large groups represented. There are only two large groups are represented and both are of same context. Another is that there is an uneven distribution of organizational representations of group size. This warrants caution about concluding the usefulness of organizational representation to produce valid interactions using this construct.

Although the above tools perform well in terms of producing agent interactions, the drawback of such tools is that they cannot handle the large group size. The authors assumed that the people in an organization have shared understanding of other agents. This assumption seemed to be reasonable, but when the same assumption applied to large organizations it failed to produce interactions.

CHAPTER-3 NEAR-TERM ANALYSIS

3.1 Near Term Analysis:

To generate "what-if" scenario a multi-agent model called "Near term Analysis" was introduced. This framework generates "what-if" situations by takin the input with the social network analysis method. *"To perform "what-if" analysis of an organization under different possible threat scenarios are done by using Multi-Agent system (MAS) called Dynet"* (Carley et al 2006). This framework puts "*Dynet in data framing environment so that a large number of simulations can be run with different possible threat scenarios*".

3.2 Working of Near Term Analysis:

The figure 2 below clearly explains the working of near term analysis. It takes a Meta - matrix as input and then the agent interaction mechanism takes place. The isolating of agents is the threat scenario that is employed here. Finally, the knowledge diffusion is calculated as an output measure to collect the impact caused by isolations.

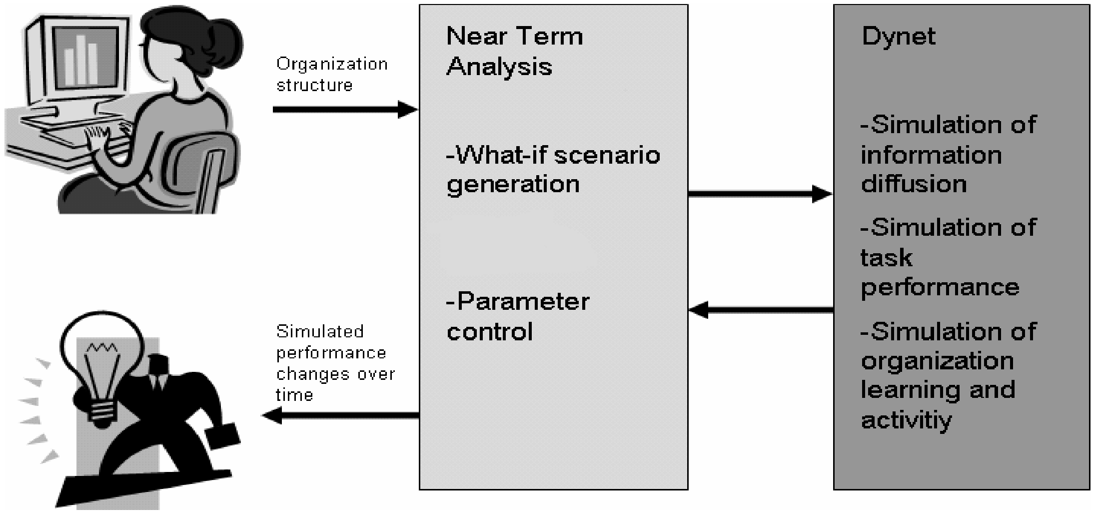

Figure 2 Working Of Near-Term Analysis (Carley et al 2006)

3.2.1 Input:

The traditional network analysis has several limitations. They do not handle multi-mode, multiplex and also social networks that are changing dynamically. This traditional analysis method does not able to represent agent, knowledge at the same time. In order to avoid this limitation meta-matrix was introduced. This meta-matrix can be used for complex systems. The Meta - matrix is defined as adjacency matrix of a network. From an organizational task perceptive there are four basic types of nodes location, belief, event, organization can be included. The relation among those who interacted with whom, who knows what, what has what knowledge can be observed with some level of uncertainty.

The table 1 explains the structure of Meta-matrix. It contains various kinds of nodes and Internode type links. This network has sub-network such as agent-agent network, agent-knowledge network. By including these networks the interactions among the agents can be simulated. An illustrative example of the Meta - matrix network is shown below.

Table 1 Meta-matrix representation (Carley et al 2006)

	Agents	Knowledge	Tasks
Agents	Social Network	Knowledge Network	Assignment Network
Knowledge		Information Network	Needs Network
Tasks			Precedence Ordering

Agent-Knowledge Network:

The knowledge network is "who knows what" in the organization. Knowledge is defined as different categories that are relevant to a particular organization. For example, if we are collecting the data about an organizational simulation group we may have the information like software development, organizational theory and statistics. The knowledge network is simply who possess what level of expertise in that particular field.

Table 2 Agent-Knowledge Matrix

	k1	k2	k3	k4
A1	1	0	0	1

A2	1	1	0	1
A3	1	0	0	1
A4	0	1	1	0

The table 2 shows an illustrative example for Agent-knowledge matrix.

“0” indicates that particular agent has no knowledge about the particular knowledge bit and “1” indicates otherwise. From the above table, we can say that A1 has knowledge about knowledge bit k1 and similarly A2 has no knowledge about the knowledge string k3.

3.2.2 Agent-interaction mechanism:

The agents in this model have the opportunity to interact with others for each time period. They select an agent to interact with them based on the probability of interaction. It is the weighted sum of two factors relative similarities or relative expertise.

After choosing an agent to interact the two agents will exchange knowledge piece. For each exchanged knowledge piece a number will be drawn ranging from 0 to 1. If the number is under the learning rate for that agent, the receiving agent will have a new line to the communicated knowledge piece on the agent knowledge network.

Relative similarity:

It is the ratio of reflecting the similarity in knowledge of choosing an agent and chosen agent. This is based on sociological principles of homophily. Homophily is defined as a person is likely to interact with another person sharing the same knowledge

$$RS_{ij} = \frac{\sum_{k=0}^{K}(S_{ij} * S_{jk})}{\sum_{j=0}^{I}\sum_{k=0}^{K}(S_{ik} * S_{jk})} \qquad (1)$$

The equation 1 shows the calculation of probability that the agent I interact with agent j based on the relative similarity of the knowledge. K refers to a set of knowledge bits and S refers to an agent knowing a specific bit of knowledge within the set. For example If *sik* is binary and *Sik*=1 then agent i have knowledge of knowledge bit ok. If *Sik*=0 then agent i doesn't have the knowledge about knowledge bit k. If agent i and agent j have more number of knowledge bits in common than their relative similarity will be high.

Relative Expertise:

It is the ratio of reflecting the amount of knowledge that chosen agent have and the chooser agent doesn't have.

$$RE_{ij} = \frac{\sum_{k=0}^{K}(X_{jk})}{\sum_{j=0}^{I}\sum_{k=0}^{K}(X_{jk})} \tag{2}$$

The equation 2 helps in calculating the probability of agent i interact with agent j based on the relative expertise of knowledge. X refers to specific bits of knowledge that agent j knows which agent I does not know. If agent j knows the number of bits that agent I does not know then we can say agent i is relative expertise to agent j. The output of either relative similarity or relative expertise is a matrix consisting of interaction probabilities for every pair of agents.

Probability of Interaction:

The agents select another agent to interact with them based on the parameter called probability of interaction. It is the weighted sum of two factors relative similarity and relative expertise.

$$P_{ij} = RS_{ij} + RE_{ij} \tag{3}$$

This probability of interaction can be calculated for any two pair of agents in a given network by using equation 3. The probability value never exceeds one and if such case happens it automatically reset to one as the probability values cannot be more than one. An example for probability of interaction matrix can be seen below.

Table 3 Probability of Interaction Matrix

	A1	A2	A3	A4
A 1	.00	.18	.05	.09
A2	.10	.00	.06	.01
A3	.04	.08	.00	.19
A 4	.11	.02	.15	.00

The table 3 indiacates an illustrative example for probability of the interaction matrix for relative similarity. It is a partial matrix and thus it does not show all the probabilities. In full matrix, the probabilities associated with agent i would sum over to all the other agents.

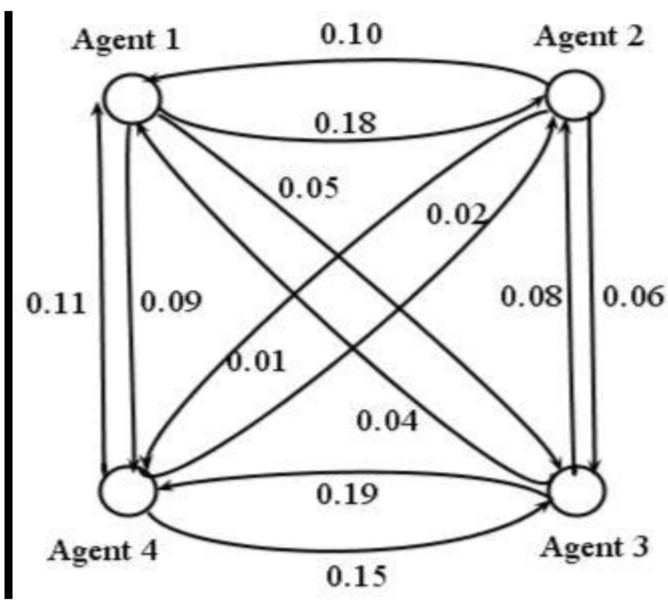

Figure 3 Probability of interaction between two agents

The figure 3 indicates the probability values between two agents which they interact. The relative probabilities between pair of agents are not symmetric. Communication can be initiated from one pairwise direction more frequently than another due to relative asymmetries.

Isolation strategies:

The threat scenarios that are employed here are isolating a group of agents from an organization structure. The output measure called knowledge diffusion is calculated after removing agent from a network after particular timestamp during the simulation. The selection of agents is the key for generating threat scenario. *"The social network analysis has been developed metrics to identify key players in a network*". There are six measures that used to calculate key players in a network and it is done by using dynamic network analysis (DNA) tool. The six measures are

- Cognitive demand
- Total degree centrality
- Clique count
- Eigenvector centrality
- Betweenness centrality
- Task/knowledge exclusivity

Cognitive demand:

It ***"****measures the total amount of effort expended by each agent to do its tasks"*. Individuals who are high in cognitive demand value are emergent leaders. Removal of these individuals tends to be quite disruptive to networks.

Total degree centrality:

It tells us the relative number of direct connections a WHO might have in a network; the higher the score the more likely a WHO might be likely to receive and potentially pass on critical information that flows through the organization.

Clique Count:

It is delineated as a group of three or more players that hold many links to each other and relatively few connections to those in other groups. Individuals or organizations who are high in number of cliques are those that belong to a large number of distinct cliques.

Eigenvector centrality:

It reflects one's connections to other well-connected people. A person connected to many isolated people in an organization will have a much lower score in this measure than those that are connected to people that have many connections themselves.

Betweenness centrality:

It tells us which node is the most connected to other parts of a network. For example, Betweenness can tell us which person in a network is the most central to the network as a whole. Betweenness measures the number of times that connections must pass through a single individual to be connected.

Task/knowledge exclusivity: It detects the *"agents who exclusively perform tasks or have singular knowledge"*.

By applying the above measures we can find key players in a given network and we can isolate set of agents in a network. The performance of the organization can be calculated after isolating the set of agents.

3.2.3 Output Measures:

After isolating the agents we need to calculate the impact caused by isolations. Knowledge diffusion is the degree that is used here.

diffusion:

Knowledge diffusion stands for how much the agents in an organization exchange knowledge during the interaction phase. It is calculated by using the below formula by using equation 4.

$$KD = \frac{\sum_{i=0}^{k} \sum_{j=0}^{n} AK_{ij}}{kn} \tag{4}$$

n=(number of agents in a network at the time)

k=(number of knowledge bits)

AK=(Adjacency matrix of Agent-Knowledge)

Limitations:

Although the above tool performs well in terms of producing agent interactions, the drawback for such tool is that it cannot handle the large group size. The authors assumed that the people in an organization have shared understanding of other agents. This assumption seemed to be reasonable, but when the same assumption applied to large organization it desirably failed.

CHAPTER-4 SOLUTION FRAMEWORK

The near term analysis framework estimates the performance of the organization over time. But the limitation with the tool is that it cannot handle the large group size. In order to avoid this limitation, we propose a framework based on "Near-term analysis". The agents in this framework simulate each agent and agent interaction with others. The agents interact and learn their behavior will eventually change that intern changes the performance of the organization. Selecting the agent to interact with, which is defined probabilistically. After selecting the agent to interact they exchange knowledge pieces and updates their own knowledge strings.

Unlike the "Near-term analysis" method this framework does not have a transitive memory that shares the information of others. After they exchange knowledge we calculate the amount of knowledge diffused from one agent to another agent during agent interaction mechanism. In this framework the agents interact and learn their behavior will eventually change the organizational structure and performance.

This framework follows certain process. Firstly, we need to create a dataset of size over 100. This dataset may contain Agent-Agent network, Agent-Knowledge network, which certainly called as a meta - network. The next step is to calculate the probability of interaction matrix between two agents in a given dataset. This probability of the interaction matrix decides which agent interacts with whom. After that we need to apply measures with which we can decide key players in a network. This can be done by using a social network analysis (SNA) tool called the Organizational Risk Analyzer (ORA). After we get to know about the key agents in an organization, we will isolate the agents and checks the performance of the organization. We can remove key agents or any other agent in any time during the simulation. The agent interaction mechanism takes place between the agents after the agents are removed. Finally, knowledge diffusion is the degree that is used to calculate the impact caused by isolations. It calculates the amount of knowledge diffused from one agent to the other during the interaction between them.

4.1 Meta-matrix:

An organization structure is the input for this model. The information on knowledge who knows what is used here. The network contains a different set of nodes With agents and knowledge. The assumption is that if there is an interaction between the two agents then there is a link is established between those two agents. Similarly, if an agent possesses a knowledge piece, then the agent node is linked to knowledge node. The table 4 indicates the meta-network that we used. In our dataset we used 104 agents and 80 knowledge bits.

Table 4 Input Meta-matrix structure

	Agent	Knowledge
Agent	Social network	Knowledge network
Knowledge	----------------	Information network

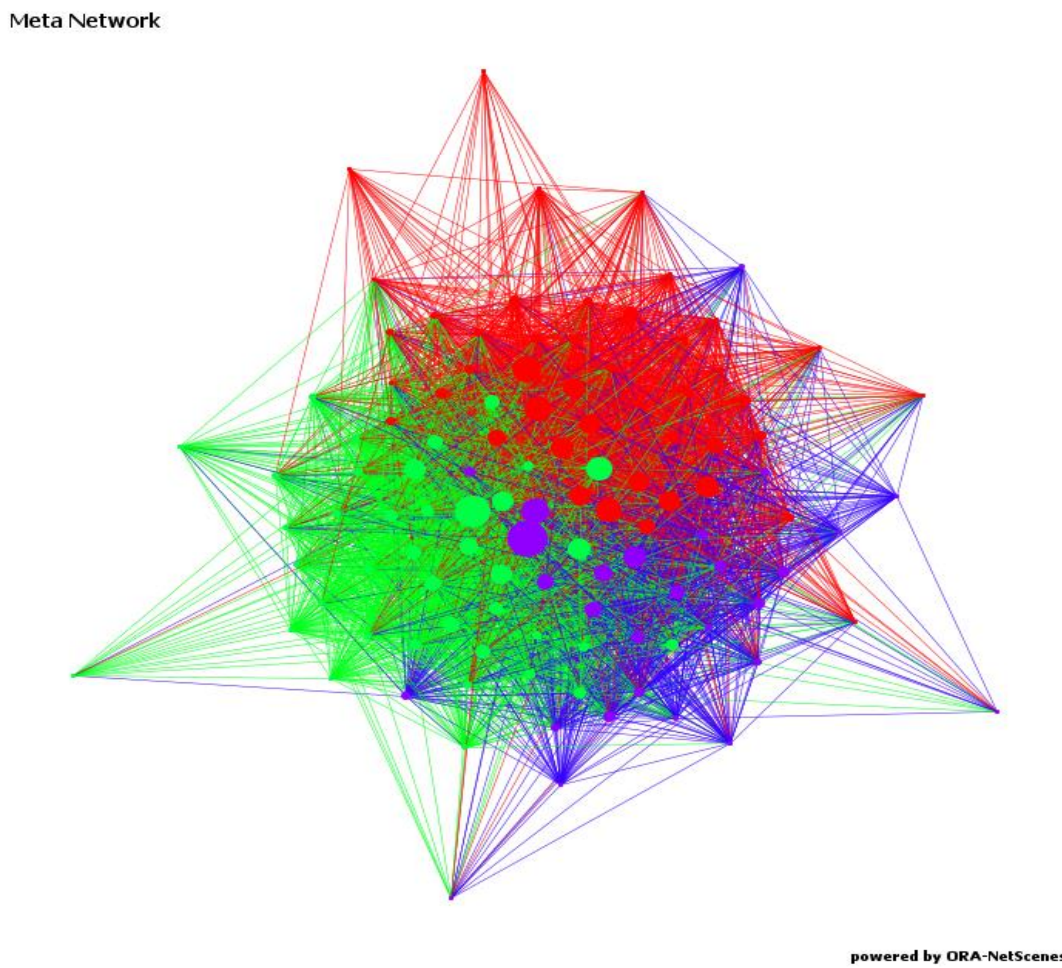

Figure 4 Visualization of network. (ORA tool) (Carley et al 2004)

4.2 Agent Interaction Mechanism:

The agents in this model interact with others with each time period. The agents will most of the time interact with those agents whom they have a higher probability of choosing. The two agents will exchange knowledge pieces. For each exchanged knowledge piece a number will be drawn either 0 or 1. If the number is under the learning rate for that agent, the receiving agent will have a new link to the communicated knowledge piece in the network.

4.2.1 Probability of interaction:

The probability of interaction depends on two factors Relative similarity and relative expertise. It is the sum of those two factors. The probability value never exceeds one and if such case happens it automatically reset to one as the probability values cannot be more than one.

P*ij*=RS*ij*+RE*ij*

Table 5 Probability of interaction variables

RELATIVE SIMILARITY (RS)	Ratio reflects similarity in knowledge Between choosing and chosen agent. $$RSij = \frac{\sum_{k=0}^{K} (Sik * Sjk)}{\sum_{j=0}^{I} \sum_{k=0}^{K} (Sik * Sjk)}.$$
RELATIVE EXPERTISE (RE)	Ratio reflecting the amount of knowledge chosen agents has and the chooser agent does not have. $$REij = \frac{\sum_{k=0}^{K} (Xjk)}{\sum_{j=0}^{I} \sum_{k=0}^{K} (Xjk)}.$$

4.3 Isolation Strategies': The threat scenario that was used here is isolating set of agents. We select the agents that are to be isolated by using a Social Network Analysis tool called ORA. This SNA has developed some measures to detect key players in a network. The six measures are

- Cognitive demand
- Total degree centrality
- Clique count
- Eigenvector centrality
- Betweenness centrality
- Task/knowledge exclusivity

Table 6 Measures for selecting top ranked agent (Carley et al 2006)

Cognitive Demand	Measures the total amount of effort expended by each agent to do its tasks.
Total degree Centrality	The total degree centrality of a node is the normalized sum of its row and column degrees.
Clique count	The number of distinct cliques to which each node belongs.
Eigenvector centrality	Calculates the principal Eigenvector of the network. A node is central to the extent that its neighbors are central.
Betweenness Centrality	The betweenness centrality of a node v in a network is defined as: across all node pairs that have a shortest path containing v.
Task/knowledge Exclusivity	Detects agents who exclusively perform tasks or have singular knowledge.

After isolating the agents the agent interaction takes place. The agents check for the partner and if the partner is removed the agent will take over the link from removed agent and updates its own knowledge strings.

4.4 Knowledge Diffusion:

After all this process, we need to calculate the impact caused by isolations. The measure that is used here is knowledge diffusion. It stands for how much amount of knowledge is shared between the agents during the interaction. The formula that is used here is

$$KD = \frac{\sum_{i=0}^{k} \sum_{j=0}^{n} AK_{ij}}{kn}$$

$n = (\text{num. of agents in a network at the time})$

$k = (\text{num. of knowledge bits})$

$\mathrm{AK} = (\text{Adjacency matrix of Agent - Knowledge})$

4.5 Agent Behavior:

The agents in our solution framework follow a certain process. These agents tend to interact with each other and update their own knowledge strings. Our solution framework does not have any transitive memory that share the information about other agents. Once the interaction phase is done the knowledge is shared between he agents and the output is calculated with a degree called knowledge diffusion. The figure 5 clearly shows the life cycle of agent step by step.

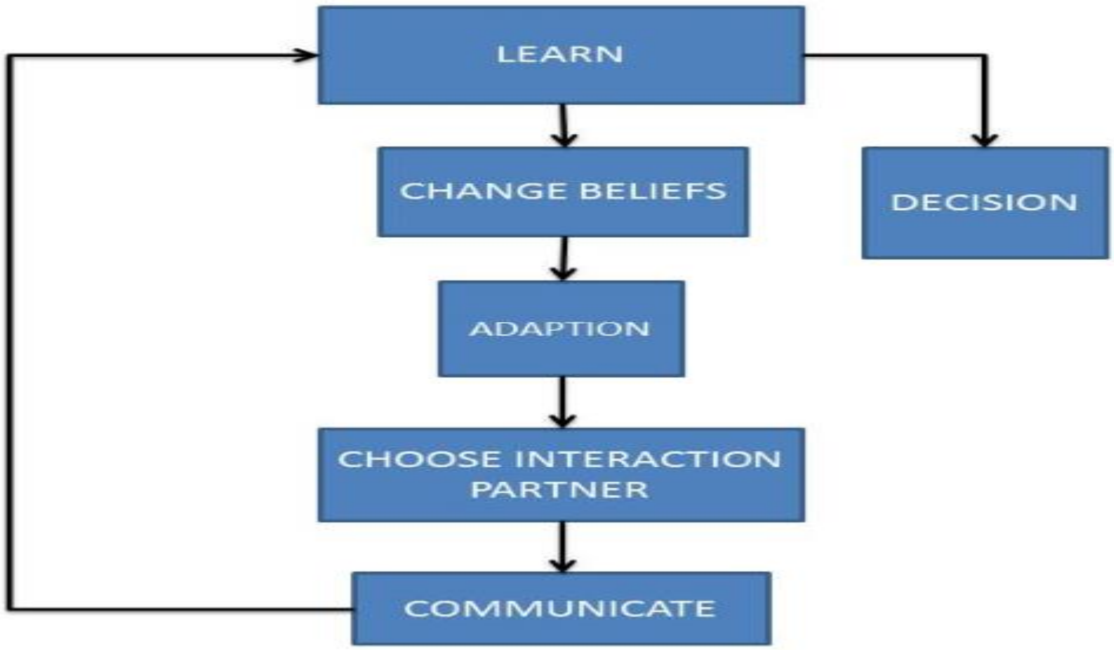

Figure 5 Agent Behavior

An algorithm for agent behavior in my method as follows:

- Step 1: This model simulates each individual agent and agent interactions with others.
- Step 2: Select the interaction partner based on probability of interaction.
- Step 3: Share the knowledge pieces and updates the knowledge strings.
- Step 4: Unlike the “Near term analysis” my method doesn't have transitive memory that share the information about other agents.
- Step 5: Knowledge diffusion is calculated at the end as it is the output measure.
- Step 6: An agent interact and learn, their behavior will eventually change the organizational structure and improve organization performance.

CHAPETR-5 EXPERIMENTAL SETUP

We designed a framework based on the idea of near term analysis. We used java and easy java simulation tool(EJS) for that. The input that is given to our framework is Meta-network. It represents a network organized structure which consists of agents and knowledge pieces. The output that is used here is knowledge diffusion. It is a performnce metric showing how accurate the information is diffused across the network. There are certain parameters that are used here are simulation run time, and number of replications. The internal variables that are used here are relative similarity and relative expertise. The complexity of this framework

5.1 Dataset description:

For the test of this proposed solution framework the project development team (PD) dataset in a software company was used. We used this dataset because it has contextual knowledge about the software company organization. The PD team contains 104 employees in an organization. The PD team is simulated because it has to handle future projects in a company.

The employees in a PD department will have subsequent knowledge on various levels of a project. The agent-knowledge matrix was designed based on which employee has significant knowledge on the particular knowledge bit. As we discussed earlier 0 indicates that agents does not have significant knowledge on particular bit and 1 indicates the otherwise. During live simulation the network structures were extracted from the survey. This network contains 104 agents and 80 knowledge bits. This software company dataset (PD) courtesy of ORA goggle group members.

There are some assumptions associated with this dataset. The data should be in the form of a meta - matrix. Each employee in a company should have at least one knowledge bit known to them. The knowledge must be represented in the form of 0 and 1 only. When the employees represent the knowledge bit as 1 then they should have significant knowledge on that particular bit. The employees need not to have information of other employees. The table 7 indicates the whole experimental setup.

Table 7 Experimental Setup

Input	Meta-network	Consists of Agent-Agent network, Agent-knowledge network.
Output	Knowledge diffusion	The amount of knowledge that an agent is diffused across the network.
Parameters	a) Simulation run setup (default: Time Stamp 50) b) Number of replications (3)	Total simulation, run-time. Number of times the model runs.
Variables	a) Relative Similarity. b) Relative Expertise.	Interactions caused by homophily. Interactions caused by

	c) Probability of interaction	expertise. Weighted sum of two factors Relative similarity and Relative expertise.

We tested our framework by varying the number of agents. First, we tested with 20 agents and then we increased that number to 104. On both the occasions we compared with "Near-term Analysis" framework.

5.2 Isolating of Top ranked agent

To isolate an agent, we need to know the top ranked agent in a network. For that we use ORA tool which runs some social network measures to know the top ranked agent. After knowing the top ranked agent, we isolate an agent from the network and calculate knowledge diffusion. The input which we have given is a meta-matrix and the parameters are simulation run time and number of replications. Variables like relative similarity, relative expertise and probability of interaction, etc. Were used. The table 8 summarizes the experimental configuration 1.

Table 8 Experiment Configuration 1

Threat Scenario	Isolating of top ranked agent
Input	Meta-matrix (Agent-Knowledge matrix)
Parameters	a) Simulation run setup (default: Time Stamp 50)

	b) Number of replications (3)
Variables	a) Relative Similarity. b) Relative Expertise. c) Probability of interaction
Output	Knowledge diffusion

5.3 Isolating randomly selected agents:

After isolating the top ranked agent, we performed some experiments by randomly selecting some agents. We calculated the knowledge diffusion by removing each agent of a network. The inputs that we have given are same and variables and parameters are also same. . The table 9 summarizes the experimental configuration 2.

Table 9 Experiment Configuration 2

Threat Scenario	Isolating randomly selected agents
Input	Meta-matrix (Agent-Knowledge matrix)
Parameters	a) Simulation run setup (default: Time Stamp 50) b) Number of replications (3)
Variables	a) Relative Similarity. b) Relative Expertise. c) Probability of interaction
Output	Knowledge diffusion

5.4 Best and Worst case scenarios:

After removing the top ranked agent and isolating the randomly selected agents, we performed some experiments to know the best and worst case situations in a network. Here the best case indicates that by removing agents the knowledge diffusion value increases. The parameters that were used here are same as above experiment. . The table 10 summarizes the experimental configuration 3.

Table 10 Experiment Configuration 3

Experiment	Best and Worst case scenarios.
Input	Meta-matrix (Agent-Knowledge matrix)
Parameters	a) Simulation run setup (default: Time Stamp 50) b) Number of replications (3)
Variables	a) Relative Similarity. b) Relative Expertise. c) Probability of interaction
Output	Knowledge diffusion

5.5 NTA and our solution framework:

After finding out the values by isolating the agents at various levels we performed some experiments with a Software company dataset of size 20 and software company dataset (PD) of size 104. The input that is given here is same and all the parameters that are used are same as above experiment. The table 11 summarizes the comparison experiments between the two frameworks.

Table 11 Experiment Configuration 4

Framework	Near-term Analysis
Dataset 1	Software company (group size 20)
Dataset 2	Software company (group size 104)
Input	Meta-matrix (Agent-Knowledge matrix)
Parameters	a) Simulation run setup (default: Time Stamp 50) b) Number of replications (3)
Variables	a) Relative Similarity. b) Relative Expertise. c) Probability of interaction
Output	Knowledge diffusion

We performed the same experiment with our solution framework. The table 12 below illustrates the details of an experiment.

Table 12 Experiment Configuration 5

Framework	Our solution Framework
Dataset 1	Software company (group size 20)
Dataset 2	Software company (group size 104)

Input	Meta-matrix (Agent-Knowledge matrix)
Parameters	a) Simulation run setup (default: Time Stamp 50) b) Number of replications (3)
Variables	a) Relative Similarity. b) Relative Expertise. c) Probability of interaction
Output	Knowledge diffusion

CHAPTER-6

6. RESULTS & DISCUSSIONS

The solution framework which was discussed earlier provides two research purposes. Firstly it simulates the threat scenarios and secondly it calculates the outcomes of those scenarios. Therefore the key agent in a network has to be identified by using some social network analysis measures. After generating the threat scenarios, knowledge diffusion was calculated as output measure. Furthermore, we are able to find out best and worst scenarios by using this solution framework.

6.1 Social Network Measure Values:

We applied our designed solution framework to Software company (PD) team dataset. The threat scenario that is deployed here is the isolation of agents. In order to decide which agent has to be isolated social network analysis introduced some metrics. We generated some reports by using an organizational risk analyzer (ORA) tool. The six measures that are used are

- Cognitive demand
- Total degree centrality
- Clique count
- Eigenvector centrality
- Betweenness centrality
- Task/knowledge exclusivity.

Cognitive Demand:

We applied cognitive demand metric to our data set. "*It calculates the total amount of effort expended by each agent to do its tasks*".

Table 13 Cognitive Demand Values (ORA Tool)

RANK	AGENT	COGNITIVE DEMAND VALUE
1	A32	0.120
2	A82	0.120
3	A13	0.118
4	A18	0.118
5	A23	0.117
6	A28	0.116
7	A58	0.116
8	A63	0.115
9	A68	0.115
10	A73	0.114

Minimum Value: 0.000

Maximum value: 0.120

Standard Deviation: 0.031

Total degree centrality: The total degree centrality of a node is the normalized sum of its row and column degrees. The table 14 indicates the value of Total degree centrality for each corresponding agent.

Table 14 Total Degree Centrality Values (ORA Tool)

RANK	AGENT	TOTAL DEGREE CENTRALITY VALUE
1	A17	0.674
2	A63	0.661
3	A101	0.652
4	A43	0.652
5	A51	0.652
6	A47	0.633
7	A01	0.629
8	A97	0.620
9	A70	0.606
10	A33	0.602

Minimum Value: 0.086

Maximum value: 0.674

Standard Deviation: 0.047

Clique count: The number of distinct cliques to which each node belongs. The table 15 indicates the value of clique count measure for each corresponding agent.

Table 15 Clique Count Values (ORA Tool)

RANK	AGENT	CLIQUE COUNT VALUE
1	A17	0.144
2	A47	0.143
3	A63	0.143
4	A01	0.141
5	A51	0.139
6	A70	0.139
7	A05	0.137
8	A24	0.135
9	A20	0.131
10	A97	0.130

Minimum Value: 0.00

Maximum value: 0.144

Standard Deviation: 0.038

Eigen Vector: Calculates the principal Eigenvector of the network. A node is central to the extent that its neighbors are central. The table 16 indicates the value of Eigen vector value for each corresponding agent.

Table 16 Eigen Vector Values (ORA Tool)

RANK	AGENT	EIGEN VECTOR VALUE
1	A17	0.174
2	A63	0.173
3	A47	0.173
4	A51	0.172
5	A01	0.167
6	A33	0.167
7	A70	0.166
8	A24	0.165
9	A02	0.164
10	A97	0.164

Minimum Value: 0.035

Maximum value: 0.174

Standard Deviation: 0.303

Betweenness Centrality: The betweenness centrality of a node v in a network is defined as across all node pairs that have a shortest path containing v. The table 17 indicates the value of betweenness centrality for each corresponding agent.

Table 17 Betweenness Centrality Values (ORA Tool)

RANK	AGENT	BETWEENNESS CENTRALITY VALUE
1	A101	0.018
2	A43	0.016
3	A57	0.013
4	A51	0.012
5	A97	0.012
6	A17	0.011
7	A70	0.011
8	A37	0.010
9	A62	0.010
10	A02	0.010

Minimum Value: 0.000

Maximum value: 0.018

Standard Deviation: 0.004

Task Exclusivity:

It detects the agents who exclusively perform tasks or have singular knowledge. The table 18 indicates the value of task exclusivity measure for each corresponding agent.

Table 18 Task Exclusivity Values (ORA Tool)

RANK	AGENT	TASK EXCLUSIVITY VALUE
1	A61	0.017
2	A101	0.017
3	A43	0.017
4	A70	0.016
5	A89	0.016
6	A15	0.016
7	A51	0.016
8	A24	0.015
9	A05	0.015
10	A85	0.015

Minimum Value: 0.002

Maximum value: 0.017

Standard Deviation: 0.004

By employing these measures on a Software dataset (PD) team, we get the values of agents ranked from 1 to 10. This can be done by using ORA tool.

In order to isolate agents, we need to know the top ranked agent. By taking the average values of the top 10 agents of each social network analysis measure we get to know the top ranked agent. The figure 5 shows the agent that is repeatedly top ranked in the measures that we discussed earlier.

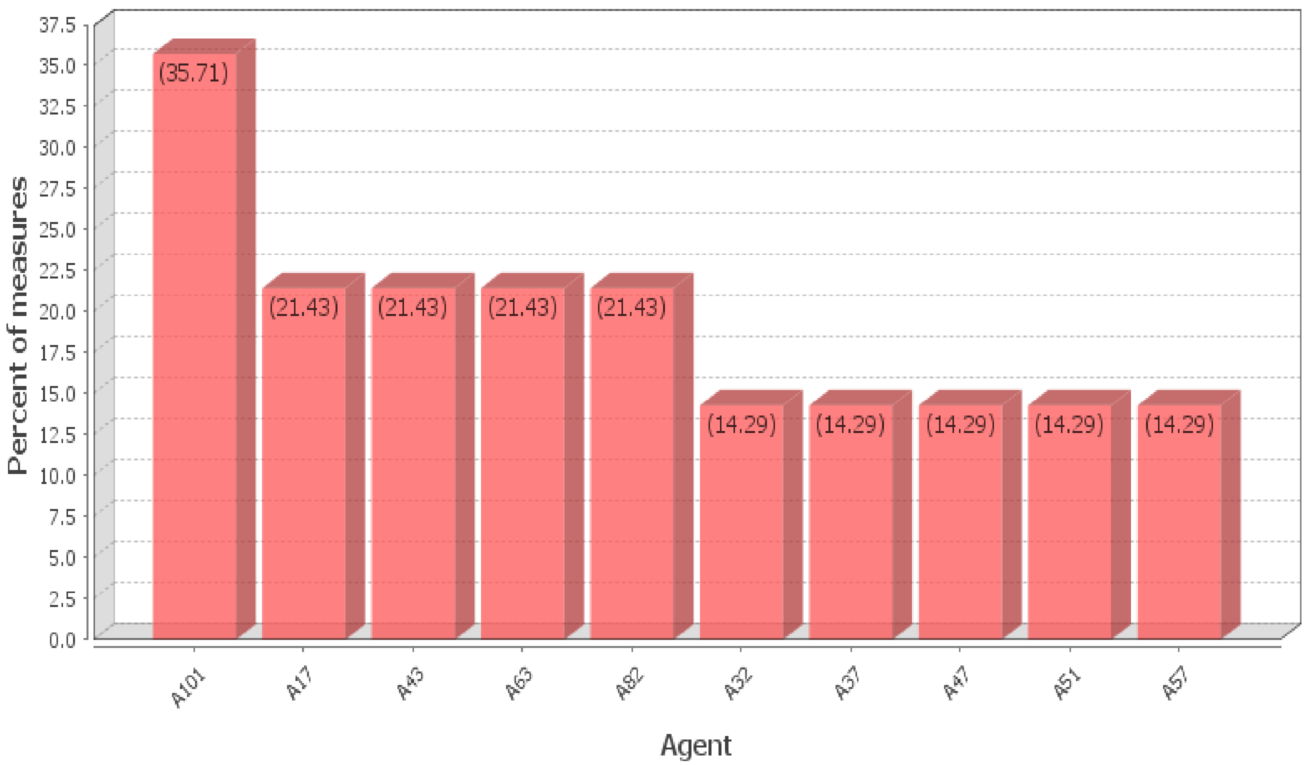

Figure 6 Graph representing Top ranked agent

6.2 Isolating Top Ranked agent:

Now we have top ranked agent in a network. The next step is to isolate an agent and calculate the output measure called knowledge diffusion. The knowledge diffusion has to be calculated before the isolation of the agent and after the isolation of the agent. On both the occasions the agent interaction mechanism takes place. The probability of interaction matrix is calculated (refer to table (19)) to decide which agent interacts with whom. Now the top ranked agent A101 was removed from the network and knowledge diffusion was calculated.

Table 19 KDF value before and after Isolating Top ranked agent

CASE	Condition	Knowledge diffusion (KDF)
1	Before removing agent (A101)	0.478
2	After removing agent (A101) at Time stamp (15)	0.477

The table 19 indicates there is a slight difference between the knowledge diffusion values. After removing the agent there is a slight reduction in the value of knowledge diffusion. This indicates the performance of organization reduces if we remove the Agent (101) from a network. The probability of interaction matrix also calculated before the agent removal and after agent removal.

6.3 Isolating randomly selected agents:

After removing the top ranked agent, we randomly select some agents to remove from a network during the simulation phase. We check the values of knowledge diffusion after removing each particular agent. We compared that value with that of the value of knowledge diffusion before the agent removal.

Table 20 KDF value for randomly selected agents (ORA Tool)

AGENT	Significant Impact
A8	Increase
A9	Increase
A12	Decrease
A14	Increase
A17	Increase
A28	Decrease
A35	Decrease
A44	Increase
A59	Decrease
A67	Increase
A79	Increase
A88	Increase
A94	Decrease
A102	Decrease

The table 20 indicates the performance of organization corresponds to particular agent removal. There is a slight increase and decrease in the values of knowledge diffusion. The above results indicate that the agent that has significant knowledge and maximum number of links performs well in a network. For example the agents A08, A09, A67, and A88 have less number of links and they hold a less number of knowledge bits. So by removing those agents the knowledge diffusion rate increases than that of the threshold value. More amount of knowledge has been diffused during the interaction phase. Removing of those subsequently decreases the performance of the organization. Further, there is only slight difference between knowledge diffusion values. The standard deviation value is 0.038.

6.4 Best and Worst Case Scenarios:

Furthermore, we investigated on best scenarios and worst scenarios. We isolated group of agents and we note down the performance of the organization. The table 21 indicates the best case and worst case scenarios. The agents A8, A9 and A14 stand out best to improve the performance of the organization. Isolating those agents will improve the diffusion rate. The second best case will be A17, A44 and A67. There is a slight difference between the best case 1 and best case 2. But on both occasions the knowledge diffusion rate increases predominantly that in turns increase the performance of the organization.

Table 21 Best and Worst case scenarios.

Best Case 1	A8, A9, A14
Best Case 2	A17, A44, A67
Worst Case 1	A12, A28, A35
Worst Case 2	A59, A94, A102

The worst case situations are A12, A28, and A35. The knowledge diffusion rate suddenly decreases than that of standard value. Isolating of these agents will decrease the performance of the organization. This indicates that these agents have maximum number of links and these agents also hold more number of knowledge bits. Although there is only slight difference between the values of diffusion rate these agents in a network stands out best and worst case scenarios.

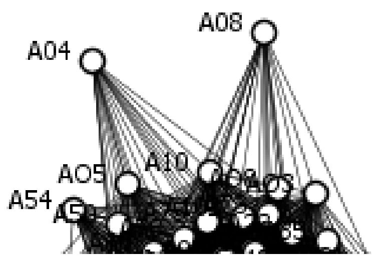

Figure 7 Visualization of Agent 8 in a network

In figure 7 Agent 8 (A08) has less number of links when compared to other agents The performance of an organization will not be reduced if we remove that agent. This is the reason why the knowledge diffusion value increases if we remove A08. Considering all this A08 can be termed as one of the best scenarios to increase the performance of an organization.

6.5 Comparison of NTA and Our Solution Framework:

The solution framework which we designed was able to predict the performance of an organization for group size over 100. Now we compare our results with the "Near term Analysis (NTA) framework. First, we compared the NTA framework and our designed framework with the software dataset of group size 20. We calculated the knowledge diffusion rate on both the occasions. First, we have given software company dataset as input to NTA and isolated the agent at time stamp 10. The knowledge diffusion is noted here. Now the same software dataset was given to our solution framework and knowledge diffusion was noted here. On both the occasions the parameters that are used are same. The results indicate that knowledge diffusion values are almost the same. The table 22 summarizes the comparison of two frameworks with a dataset of size 20.

Table 22 Comparison with dataset of group size 20

Dataset (Group Size 20)	Framework	KDF before isolation.	Isolation	KDF after Isolation.
Software company	NTA	0.497	Agent 5 at Time stamp 10.	0.565
Software company	Our solution framework	0.497	Agent 5 at Time stamp 10.	0.565

The knowledge diffusion values after the isolation and before the isolation indicates that there is no much difference between the NTA framework and our solution framework. Now we need to test our result with the large dataset and NTA.

Table 23 Comparison with dataset of size 104.

Dataset	Framework	KDF before isolation.	Isolation	KDF after Isolation.
Software company dataset (PD) Group size (104)	NTA	-------------	Agent 5 at Time stamp 10.	---------------
Software company			Agent 5 at Time	

dataset (PD) Group size (104)	Our solution framework	0.478	stamp 10.	0.479

The tables 22 and 23 indicate that the Near Term analysis unable to produce agent interactions for large group size. The performance of the organization cannot be predicted for large group size by using NTA. This is because the agents in NTA framework hold the information about all other agents forming a transitive memory. This makes the system complicated and it cannot be able to produce interactions. The agents in our solution framework do not have transitive memory that stores the information of others. Thus, it is very easy to produce agent interactions for large group size.

The value of knowledge diffusion has a significant role in the real world. In the above experiment the knowledge diffusion value decreases after isolating the particular agent from a network. This indicates the performance of an organization decreases after isolating the agent. The amount of knowledge that is diffused from one agent to another agent decreases as a result the performance also decreases. Although the values of knowledge diffusion varied marginally they have some significant importance in terms of health care department.

CHAPTER-7

CONCLUSION AND FUTURE WORK:

In this thesis, we used both multi-agent simulation and also social network analysis methods to predict the performance of an organization. We did this by isolating some of the agents in a network by using social network analysis measure. Our solution framework takes meta-network as input and isolates the agents during simulation. We calculate the output with a degree called knowledge diffusion as an output from a simulation. Our solution framework is immediately capable of handling the large group size (over 100).

The Software company (PD) team dataset is given as input to our framework and we observed the capabilities of our framework. The framework detects A8, A9 and A14 as a best case scenario to increase the knowledge diffusion value. These results indicate that this agent does not possess exclusive knowledge when compared to others. They interact with other agents, but the amount of knowledge diffused from one agent to another agent is more during the interaction phase. The knowledge diffusion rate increases predominantly as other agents interact with each other agents more accurately because the inefficient agents are isolated from the node in a network.

On the other hand the agents who receive more knowledge when compared to others, in that instance the knowledge diffusion value decreases. If the agent who has more knowledge is removed from the network than the amount of knowledge diffused during the interaction phase also decreases.

Future work is to validate the agent interactions with all other kinds of networks. The agent-knowledge matrix was applied in this experiment. Other networks like task network, Cognitive networks are also can be practiced. Further the use of performance measurement holds key in predicting the organization functioning. Knowledge diffusion is to be validated by using a number of datasets. By developing different performance measures we can improve the performance of an organization. The time complexity of this framework and NTA is almost same when they compared with dataset of size 20. The time complexity is not calculated for large dataset as we don’t have any other algorithm to compare.

In the real world this work can be used in corporate offices and in also military organizations. This study can be used to disable terrorist organizations. If we find a key person in a terrorist network and if we can remove the person from a network then we can collapse an entire network. This study can be applied in health care departments and also in many other systems.

REFERENCES

BOHTE, J. AND MEIER, K. J. 2001. Structure and the Performance of Public Organizations: Task Difficulty and Span of Control. Public Organization Review. Volume 1.341-354.

CARLEY, K. 2005. Dynamic network analysis for counter-terrorism. http://www.casos.cs.cmu.edu/projects/ora/publications.php as of may 25, 2015.

CARLEY, K.M. AND CARLEY, P.K.M. 2006. Destabilizing Terrorist Networks. http://www.casos.cs.cmu.edu/projects/ora/publications.php as of may 25, 2015.

CARLEY, K.M. AND KIM, E.J. 2008. Random Graph Standard Network Metrics Distributions in ORA . http://www.casos.cs.cmu.edu/projects/ora/publications.php as of may 25, 2015.

CARLEY, K.M. AND PFEFFER, J. 2004. Dynamic Network Analysis. http://www.casos.cs.cmu.edu/projects/ora/publications.php as of may 25, 2015.

GRAHAM, J. 2005. Dynamic Network Analysis of the Network-Centric Organization: Toward an Understanding of Cognition & Performance, Doctoral degree dissertation,CASOS lab, Carnegie Mellon University, PA.

JIN, V. Y. AND LEVIS, A. H. 1990. Effects of Organizational Structure on Performance: Experimental Result. Research Report. LIDS-P ; 1978. Laboratory for Information and Decision Systems. Massachusetts Institute of Technology. Boston MA.

KUNZ, J. C., LEVITT, R. E., AND JIN, Y. 1998. The Virtual Team Design: A Computational Simulation Model of Project Organizations, Communications of the Association for Computing Machinery, 41(11), pp 84-92

LIN, Z. AND CARLEY, K. M. 1997. Organizational Decision Making and Error in a Dynamic Task Environment. Journal of Mathematical Sociology. 22(2). pp 125-150.

HIRSHMAN, B., MORGAN, G., CHARLES, J.S., AND CARLEY, K. 2010. Construct Demo Input Deck. http://www.casos.cs.cmu.edu/projects/ora/publications.php as of may 25, 2015.

MACAL, C.M. AND NORTH, M.J. 2006. Modeling and Simulation. http://www.casos.cs.cmu.edu/projects/ora/publications.php as of may 25, 2015.

MCCULLOH, I., LOSPINOSO, J., AND CARLEY, K.M. 2010. The Link Probability Model : A Network Simulation Alternative to the Exponential Random Graph Model. http://www.casos.cs.cmu.edu/projects/ora/publications.php.

MOON, I., C, K.M., JAMES, D.H., JOHN, M.G., STATES, U., AND A, M. 2008. Pittsburgh , PA T hesis Committee Submitted in partial fulfillment of the requirements. http://www.casos.cs.cmu.edu/projects/ora/publications.php.

MOON, I. AND CARLEY, K.M. 2006. "Estimating the near-term changes of an organization with simulations." *AAAI Fall Symposium*.

MOON, I. AND CARLEY, K.M. 2006. Modeling and Networks in Social and Geospatial Dimensions. http://www.casos.cs.cmu.edu/projects/ora/publications.php.

MOON, I.-C., CARLEY, K.M., AND KIM, T.G. 2013. Modeling and Simulating Command and Control. Springer London.http://www.casos.cs.cmu.edu/projects/ora/software.php as of may 25, 2015.

PESTOV, I. AND VERGA, S. 2009. Dynamical networks as a tool for system analysis and exploration. *2*009 IEEE Symposium on Computational Intelligence for Security and Defense Applications Cisda, 1–8.

REMINGA, J. AND CARLEY, K.M. 2004. DyNetML : Interchange Format for Rich Social Network Data. http://www.casos.cs.cmu.edu/projects/ora/publications.php as of may 25, 2015.

SCHREIBER, C. 2006. Human and Organizational Risk Modeling : Critical. http://www.casos.cs.cmu.edu/projects/ora/publications.php as of may 25, 2015.

SCHREIBER, CRAIG, AND CARLEY,K,M. "Validating agent interactions in construct against empirical communication networks using the calibrated grounding technique." *Systems, Man, and Cybernetics: Systems, IEEE Transactions on* 43.1 (2013): 208-214.

WEI, W., PFEFFER, J., REMINGA, J., AND CARLEY, K.M. 2011. and Disconnected Networks in ORA. http://www.casos.cs.cmu.edu/projects/ora/publications.php as of may 25, 2015.

WEBER, R., RICK, S., CAMERER, C. 2004. The Effects of Organizational Structure and Codes on the Performance of Laboratory 'Firms'. Working Paper. Department of Social and Decision Sciences. Carnegie Mellon University. Pittsburgh PA.

VITA AUCTORIS

NAME:	Anivesh Reddy Minipuri
PLACE OF BIRTH:	Hyderabad, India
YEAR OF BIRH:	1990
EDUCATION:	MSc. (Computer Science),
	University of Windsor,
	2015.