2011

Agent Modeling In Decision Support System: A Case Study In A Base Hospital System

Ashish Nakhwal
University of Windsor

AGENT MODELING IN DECISION SUPPORT SYSTEM: A CASE STUDY IN A BASE HOSPITAL SYSTEM

by
Ashish Nakhwal

A Thesis
Submitted to the Faculty of Graduate Studies
through Computer Science
in Partial Fulfillment of the Requirements for
the Degree of Master of Science at the
University of Windsor

Windsor, Ontario, Canada
2010

AGENT MODELING IN DECISION SUPPORT SYSTEM: A CASE STUDY IN A BASE HOSPITAL SYSTEM

by

Ashish Nakhwal

APPROVED BY:

Dr. Anne W. Snowdon
Odette School Of Business

Dr. Robert D. Kent
School of Computer Science

Dr. Ziad Kobti, Advisor
School of Computer Science

Dr. Scott D. Goodwin, Chair of Defense
School of Computer Science

Sep 29, 2010

Author's 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

Health practitioners are studying different techniques to provide quality patient care and to prevent injuries in the hospitals, which motivate the ground work to model such a complex system with the objective to understand the chief social factors leading to injury.

The underlying social factors such as socialization, task scheduling, domain knowledge and path finding contribute to the day-to-day activity of the health practitioners and agents in social models which ultimately affect their performance.

The aim of this study is to outline the objective decision support elements in mission critical human social models and critically examine the influence of those factors on the system. The outcome of this research leads to a development of more realistic artificial agents in a social complex modeling for the better understanding of the system's behavior.

Dedication

To the Nakhwal family and my loving parents.

Acknowledgements

First and foremost, I am grateful to my advisor Dr. Ziad Kobti for providing me the opportunity to work in an exciting and challenging field of research. His constant motivations, support, innovative ideas, own research and infectious enthusiasm have guided me toward successful completion of my thesis. My interactions with him have been of immense help in defining my research goals and in identifying ways to achieve them.

My sincere gratitude goes to Dr. Robert D. Kent for his valuable advice and helpful discussions during my thesis research. I would like to thank Dr. Anne W. Snowdon for her valuable comments and suggestions that helped me in my research. I would also like to thank Shamual Rahaman for his advice and long discussions that helped me in learning new things.

Finally I would like to thank my grandparents, parents, uncles and aunts for their unconditional support and love.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Problem Specification

The problem is to outline the key decision support elements in mission critical human social models; specifically in the context of path finding, socializing, task scheduling, and domain knowledge. Furthermore, we seek to evaluate the role of each element in the system and how these elements reveal distinct and substantial effect on the dynamic complex systems.

1.2 Motivation

Modeling human cultures using computer simulations is a challenging task. In these complex systems, there are myriad of factors that account for individual's decisions [2]. These factors may have different effects on the system which makes it difficult to decide the involvement of each parameter on the outcome. We may use probability to decide the contribution of factors on the behavior of a system. However when the decision is critical, such as when one's decision can save somebody's life then it becomes really important to know the ratio of involvement of different factors behind the decision. In order to understand the decision making process we need to account for individual, social as well as environmental factors. To measure the involvement of each factor first we need to develop such a system that is capable of capturing major factors in such a way that we should be able to develop a more realistic artificial social simulation system.

1.3 Thesis Statement

A health care system is highly dynamic and time constrained environment which includes various key factors such as workplace stress, quality of life, human and emotional factors and so on, that is a part of almost every complex social system. In this thesis, we aim to extend artificial social modeling simulation to include key model ingredients and to create a framework for modeling dynamic decision support system for mission critical health care environments.

1.4 Thesis Objective

We start by outlining objective decision support elements in mission critical human social models. We have examined the factors that influence the social models. We built a multi-agent model where each agent represents a human and exhibits human-like qualities and acts as individual. Agents are autonomous entities which can make their own decisions and are capable of responding to the actions perceived from the environment. For example, if a nurse agent receives any request from a patient agent in a hospital then she should perform required actions to serve the request, such as follow a path (depending on the environmental conditions) to the patient using her knowledge about the domain. We examine the social settings that effect an individual's decision, which includes social distractions, task scheduling or any unpredictable event that may happen in an environment. We present such a simulation as a tool for decision support which create a bridge to other disciplines so that they can get a benefit from such a tool design.

The outcome of this research may offer important findings for creating supportive environments for agents in social models that maximizes their performance. For example, the hospital management staff can observe the nurse agent's time utilization or patient service success rate by changing the given different parameters in the simulation. The effect of different parameters, such as ratio of nurse agents to patient agents, socialization rate, time spent on different tasks, probability of random event happening, short-long shift and so on, can be easily examined by changing the particular variable and comparing the results with the base model settings. These findings can allow health care managements to allocate the settings that can maximize the quality of patient care and efficient use of nurse's time in a general hospital.

1.5 Thesis Organization

The remainder of the thesis is organized as follows; chapter 2 provides a literature review and background survey which describes the multi-agent systems, various different components involved in the social complex modeling, decision support systems which helps in organizations to elucidate decisions and problem specific issues with the complex modeling such as, path finding, task scheduling, social factors and domain knowledge. Chapter 3 discusses the approach used in this thesis. Chapter 4 discusses the case study on a base hospital model which describes all the agents and their behavior in a system. Experiments and their results are presented in chapter 5 and their results are discussed in chapter 6. Finally chapter 7 gives the conclusion and direction for future work on the work presented.

Chapter 2

Literature Review and Background Survey

In this chapter, various different components of the multi-agent systems have been discussed which plays an important role in complex modeling. Complex social modeling has been discussed and the role of key elements, such as path finding, task scheduling, social factors and domain knowledge is described. Social complex modeling can further be used as a decision support system which is a useful tool for organizations in decision making activities.

2.1 Components in Complex Modeling

2.1.1 Agency

Agents are autonomous, active, persistent (software) components that perceive, reason, act and communicate [50]. Software agents have evolved from multi-agent systems. Agents are autonomous entities capable of exercising choice over their actions and interactions, act to achieve individual objectives. The concept of an agent provides a convenient and powerful way to describe a complex software entity that is capable of acting with a certain degree of autonomy in order to accomplish tasks on behalf of its user [63]. An agent is defined in terms of it's behavior [54]. The architecture of an agent is the computational structure that, along with the more dynamic knowledge represented within it, generates the agent's behavior in the environment. The architecture must contain structures that enable representing knowledge and achieving goals, interacting with the environment, and coping with unexpected occurrences. Due to the increased technological complexity, the need for complex applications have raised that require systems consisting of multiple agents who can communicate in a peer to peer fashion. Depending on the nature of the environment agent should be capable of performing tasks with co-ordination and collaboration of other agents resulting in human like behavior and affect [58]. For example, the nurse agents in hospital settings receive requests from the patients in a hospital and using the path finding techniques, task scheduling techniques and domain knowledge they respond and do specific actions to serve the patients.

2.1.2 Environment

Agents have their own domain in which they perceive, act, react and communicate which is considered as a working environment of an agent. The agents possess the knowledge about their domain and the level of knowledge they comprise depends upon the type of agent in the environment. The agents are often deployed in the environments in which they interact, and sometimes cooperate with other agents that have possibly conflicting aims [9]; these environments are known as multi-agent systems. The complexity of an environment is decided upon the characteristics of an environment. As the complexity of an environment increases, the system becomes more realistic and the user can obtain more accurate results. In this study, environment is considered as the physical hospital floor plan which include hallways, patient rooms, nurse stations etc. (refer to section 4.4).

2.1.3 Dynamics

Complex environments are dynamic in nature such that, they have high tendency of changing their states frequently. The rapidly changing environment results in different working conditions for an agent at each time step [44]. The agent must adapt to a new situation and overcome possibly unpredictable problems with actions required for autonomous behavior [54][9]. Dynamic environments are helpful in simulating real world environments and observing them because of their similar behavior. The dynamic nature of a simulation helps to imitate the working of a real world to some extent. There are various examples of dynamic environments that change very rapidly which includes a market place, hospitals, shopping malls, and so on. Consider an example of a hospital where multiple number of agents work, the outcome of one agent or happening of any unpredictable event may lead to different working conditions for other agents such as delaying of tasks or spending more time on a task.

2.1.4 Heterogeneity

Complex systems are characterized by a high degree of heterogeneity given the number of involved components [44]. The working environment for an agent consists of various different elements and individuals which have a considerable influence on the performance of agents. Different components may cause unique working conditions for the agents in which they must make different decisions and take actions that can lead the system to mimic real world qualities. In a base model hospital simulation, heterogeneity is expressed by different types of agents and different environmental conditions.

2.1.5 Social Interactions

The application of the social interaction concept to the complex multi-agent systems' functionality is a natural step towards designing and implementing more intelligent and human-like populations of artificial autonomous systems. Knowledge sharing and exchange is particularly important in multi-agent systems. The agents' ability to act in order to achieve their private goals belongs to basic properties defining this class of artificial systems [17]. However, when agents are considered in context of multi-agent systems, it very often happens that their actions become involved in multiple, simple, and complex social relations with other acting and autonomous entities. In such cases, agents belonging to a broader multi-agent population need to take into account other agents while planning and realizing their behavior. According to definitions accepted in social sciences, all actions and practices that involve more than two agents, and affect or take account of other agents' activities, experiences, or knowledge states are called the social interactions. In multi-agent systems, the agents interact with other agents to transfer some sort of information. The whole multi-agent system is created to be capable of reaching goals that are difficult to achieve by an individual agent. In multi-agent systems, an agent usually cooperates with other agents so it should have some social and communicative abilities [43]. The social interactions in these systems between agents could vary with different agents. The interactions depend upon the connection between the individuals. The agents at the same place are more likely to interact with each other. Social interactions are really important in multi-agent systems for acquiring human-like behavior.

2.1.6 Task Scheduling

In dynamic environments, agents' tasks require real time scheduling and execution. In order to finish their tasks and improve system performance they need to coordinate their actions both constantly and extensively [64]. The multi-agent systems need to operate in a real time that involve scheduling and task execution. In the multi-agent systems task scheduling is an important characteristic of the agents in order to finish their task successfully and to keep the service success rate high. Completion of tasks in the hospitals is critically important for the patient safety and quality of care.

2.1.7 Simulation

A simulation is an attempt to model a real life or hypothetical situation to gain a deeper understanding of the behavior of different parameters of the system [14]. The act of simulating something generally entails representing certain key characteristics or behaviors of a selected physical or abstract system. Simulation is used in many contexts, including the modeling of natural systems or human systems in order to gain insight into their functioning. A simulator is any software architecture on which model could be executed to generate it's behavior [39]. Non-linear adaptive interactions are mostly too complex to be captured by analytical expressions due to which computer simulations are most often used. The basic idea of a simulation is to specify the rules of behavior of individual entities, as well as the rule of their interaction, to simulate a multitude of the individual entities using a computer model and to explore the consequences of the specified individual level rules on the level of population as a whole, using results of simulation runs [6]. The simulated entities are usually called agents and the simulations of their behavior and interactions are knows as agent-based simulations. The properties of individual agents describing their behavior and interactions are known as elementary properties and the properties emerging on the higher collective level are known as emergent properties. In a hospital settings, elementary properties are the nurse agent's and the patient agent's behavior (refer to section 4.5), and emergent properties are the services that the nurse agents deliver to the patient agents such as a path nurses follow (social or shortest path) using their domain knowledge, task scheduling, socializing, and so on.

Like deduction, simulation starts with a set of explicit assumptions. But unlike deduction, it does not prove any theorems. Instead, a simulation generates data that can be analyzed inductively. Unlike typical induction, however, the simulated data comes from a rigorously specified set of rules rather than direct measurements of the real world. While induction can be used to find patterns in a data, and deduction can be used to find consequences of the assumptions. Simulation means driving a model of a system with suitable inputs and observing the corresponding outputs. These purposes include: prediction, training, entertainment, and education.

- **Prediction:** Simulation is able to take complicated inputs, process them by taking hypothesized mechanisms into account, and then generate their consequences as predictions. For example, if the goal is to predict the nurse's performance by allocating the different number of nurses in a hospital setting then simulation would be the best and economic technique.

- **Training:** Many of the earliest and most successful simulation systems were designed to train people by providing a reasonably accurate and dynamic interactive representation of a given environment. Flight simulators for pilots are an important example of the use of simulation for training.

- **Entertainment:** From training, it is only a small step to entertainment. Flight simulations on personal computers are fun. So are simulations of completely imaginary worlds.

- **Education:** From training and entertainment, it is only another small step to the use of simulation for education such as hospital simulations.

2.1.8 Verification and Validation

Demonstration of a simulation model's validity such as its ability to accurately represent the system under investigation is a key to the acceptance of a simulation as a technique. Verification is concerned with building the model right [52]. It is utilized in the comparison of the conceptual model to the computer representation that implements that conception. It asks the questions: Is the model implemented correctly in the computer? Are the input parameters and logical structure of the model correctly represented? On the other hand validation is concerned with building the right model. It is utilized to determine that a model is an accurate representation of the real system. Validation is usually achieved through the calibration of the model, an iterative process of comparing the model to actual system behavior and using the discrepancies between the two, and the insights gained, to improve the model. A common method of validation involves comparing the output of the model to that of the real situation using the data obtained during the analysis phase. This process is repeated until model accuracy is judged to be acceptable. The validation and verification phase is especially difficult with computer simulations of complex social systems due to the specific characteristics of the complex systems (non-determinism and so on). The validation and verification phase must be completed before any experimentation with the simulator; otherwise the results will be unreliable.

2.1.9 Feedback and Reporting of Results

Developing a simple model in the beginning phase is very helpful for the future validations, and to report the results to their clients in order to obtain their feedbacks and work accordingly [9]. After reviewing the initial output, client's objectives for the project may change and one can save a lot of time in the initial model development. Feedback is a mechanism, process or signal, that is looped back to control a system within itself; such a loop is called a feedback loop. In systems containing an input and output, feeding back part of the output so as to increase the input is positive feedback; feeding back part of the output in such a way as to partially oppose the input is negative feedback. Feedback loops provide generic mechanisms for controlling the running, maintenance, and evolution of software and computing systems. Feedback loops are important models in the engineering of adaptive software, as they define the behavior of the interactions among the control elements over the adaptation process to guarantee system properties at run-time. Feedback loops and foundations of control theory have been successfully applied to computing systems [9].

2.2 Multi-Agent Systems (MAS)

Multi-agent systems are one of the fastest growing areas in computer science and it has generated a lot of excitement in the recent years because of its new paradigm for conceptualizing, designing, and implementing software systems. As more Artificial Intelligence (AI) applications are being formulated in terms of spatially, functionally, or temporally distributed processing, MAS are emerging as an important discipline of AI. In general, MAS is a computational system where multiple autonomous entities having different information and/or diverging interests interact with one another in order to satisfy certain goals [7][41][42]. Multi-agent system can also be defined as a loosely coupled network of problem solvers that interact to solve problems that are beyond the individual's capability or knowledge of each problem solver [59]. These problem entities or solvers are often called agents. The agents in such systems may be homogenous or heterogeneous such that, they may have common goals or individual goals [37][38]. The agents are capable of operating in dynamic and open environments and often interact with other agents including both people and software. The agents are a way to manage interactions between different kinds of computational entities, and to get the right kind of behavior out of large-scale distributed systems [17].

Multi-agent systems are usually characterized in terms of their internal behaviors and external interactions between the agents [46]. Some of the characteristics of the agent's internal behavior are the type of cognition used and the performance measure they utilize when choosing how to behave in model based, reactive, goal based and utility based environments. Some of the characteristics of agent's external behavior are how the agents interact with each other to share information and to perform tasks. Multi-agent system's main internal and external behaviors characterize them as a unique type of distributed computation system which has to be supported by a generic distributed computer infrastructure or a set of middleware services. Software agents communicate with each other to find out the tasks other agents are able to perform.

In MAS, each agent has incomplete information or capabilities for solving the problem and thus has a limited viewpoint. There is no global system control; one can only monitor and observe the behavior of a system. In MAS, the agents often cooperate to solve a common problem rather than achieving their individual goals. Multi-agent systems are suitable for the domains that involve interactions between different people or organizations, with different goals and proprietary information [58]. Data in the MAS are decentralized and computations by the agents are asynchronous. The absence of centralized data coordination makes it difficult to determine the current state of the system and/or to predict the effects of actions. During this process, agents access the environment's resources and services and occasionally produce results for the entities that initiated these software agents.

The agents interact in a concurrent, asynchronous and decentralized manner, this kind of system can be categorized as a complex system. The most important reason to use MAS when designing a system is that some domains require it [43]. In particular, if there are different people or organizations, with different (possibly conflicting) goals and proprietary information, then a MAS is needed to handle their interactions. Even if each organization wants to model its internal affairs with a single system, the organizations will not give authority to any single person to build a system that represents them all; the different organizations will need their own systems that reflect their capabilities and priorities.

There are several possible reasons for using MAS. Multiple agents could speed up a system's operation by providing a method for parallel computation [17]. For instance, a domain that is easily broken into components, several independent tasks that can be handled by separate agents could benefit from MAS. Furthermore, the parallelism of MAS can help dealing with the limitations imposed by time-bounded reasoning requirements. Another benefit of MAS is their scalability. Since they are inherently modular, it should be easier to add new agents to a MAS than it is to add new capabilities to a monolithic system. The systems whose capabilities and parameters are likely to need to change over time or across agents, can also benefit from this advantage of MAS.

Multi-agent technology is assumed to be one possible solution to meet the requirements coming from complex and dynamic environments. Due to their adaptability and flexibility, agent based information systems have the potential to significantly improve the competitiveness of enterprises. Their application should allow more effective and efficient (logistic) processes and may also generate new customer welfare by making product improvements available.

2.2.1 Application of MAS

Agent-based computing has already transformed the processes such as automated financial markets trading, logistics, industrial robotics, and so on. Some complex systems require MAS for the representation of domain. An example of a domain that requires MAS is hospital settings. This domain from an actual case study requires different agents to represent the interests of different people within the hospital. The hospital employees have different interests, from nurses who want to minimize the patient's time in the hospital, to x-ray operators who want to maximize the throughput on their machines. Since different people evaluate candidate schedules with different criteria, they must be represented by separate agents if their interests are to be justly considered. Multi-agent system applications cover a variety of domains, military demining, wireless collaboration and communications, military logistics planning, supply-chain management, financial portfolio management, software agents participating in online auctions or bargaining [31][48], electronic institutions [30], developing schedules for air traffic [40] and decentralized resource distribution in large storage facilities [62][29].

2.3 Social Networks

A social network is a set of nodes of individuals, groups, organizations, and related systems that tie in one or more types of interdependencies: these include shared values, visions, and ideas, social contacts, kinship, conflict, financial exchanges, trade, joint membership in organizations, and group participation in events, among numerous other aspects of human relationships[15] [65]. People are represented as nodes in the network and relationships among the people are represented as edges. The structure of the network can contain many sub structures such as, groups and cliques. By analyzing these substructures we can find out the likely behavior of the network as a whole. In it's simplest form, a social network is a map of all of the relevant ties between all the nodes being studied. The shape of a social network helps determine a network's usefulness to its individuals [51]. Social network maps and measures formal and informal relationships to understand what facilitates the knowledge flows that bind interacting units, who knows whom, and who shares what information and knowledge with whom. For example, in a hospital environment, there are various different networks such as nurse-to-nurse, patient-to-nurse, nurse-to-doctor, and so on. Social networking is spreading over the internet at high rate in these days. People are connecting with their friends, family and business partners with the help of social networking sites such as, Facebook, Orkut, Twitter and so on.

2.4 Complex Social Systems

Complex social systems are systems that involve numerous interacting agents whose aggregate behaviors are to be understood. Such aggregate activity is nonlinear and typically exhibits hierarchical self-organization under selective pressures; hence it cannot simply be derived from summation of individual components behavior [15]. The observations of such a system requires visualization tools or a running simulation with which one can observe the behavior of a system and elucidate conclusions. In many systems, we can distinguish a set of fundamental building blocks, which interact nonlinearly to form compound structures or functions with an identity that requires more explanatory devices than those used to explain the building blocks. This process of emergence of the need for new, complementary, nodes of description is known as hierarchical self-organization, and systems that observe this characteristic are defined as complex. A truly complex system is assumed to be completely irreducible, means that it would be impossible to derive a model from the system without losing all its relevant properties [27]. However, in reality different levels of complexity exist. In the real world there are various influential factors and properties. In order to make the complex system we need to ask a question that to what extent can we make an abstraction of microscopic interactions in order to understand macroscopic behaviors? In what measure are microscopic interactions linked in a non-reducible way with the laws that govern more structured behaviors? The reduction of complexity is an essential stage in the traditional scientific and experimental methodology (also known as analytic) [65]. After reducing the number of variables (deemed most relevant), this approach allows systems to be studied in a controlled way. The following are the properties of the complex systems [45]:

- **Non-Determinism and Non-Tractability:** A complex system is fundamentally non-deterministic. It is impossible to anticipate precisely the behavior of such systems even if we completely know the function of its constituents.

- **Limited Functional Decomposability:** A complex system has a dynamic structure. It is therefore difficult, to study its properties by decomposing it into functionally stable parts. Its permanent interaction with its environment and it's properties of self-organization allows it to functionally restructure itself.

- **Distributed Nature of Information and Representation:** A complex system possesses properties comparable to distributed systems (in the connectionist sense), such that some of its functions cannot be precisely localized. In addition, the relationships that exist within the elements of a complex system are short-range, non-linear and contain feedback loops (both positive and negative).

- **Emergence and Self-Organization:** A complex system comprises emergent properties which are not directly accessible (identifiable or anticipatory) from an understanding of its components.

2.4.1 Problem Specific Issues with Complex Modeling

In complex modeling there are various factors those have influential effects on the system behavior [49], however there are various other factors that may not be important. It is difficult to decide which factor has more influence on the system. In this section, we are describing important factors that are significant in the complex social systems [2].

Path Finding

Real time path finding in a dynamic environment is influenced by social interactions [36]. To achieve realistic agent navigation is a challenge, in the design of every AI application. Path finding techniques are usually deployed at the core of any AI movement system. This ability becomes critical when the agents operate in a dynamic environment, where unpredictable and sudden changes occur [8]. It is usually required that the path should be safe, optimal and natural between the starting and target location [57]. The optimality criterion for the path is determined on the surrounding environment and running conditions [33][23][22]. There are several methods for computing both the safest and the shortest paths, which can be classified as either local or global [21] [56][34]. Global path planning takes into account all the information in the environment whereas local path planning (reactive) algorithms are designed to avoid obstacles within a close vicinity of the agent. The path could be the safest, shortest or hybrid path. Extensive literature exists on path planning for multiple agents in virtual environments and optimal path planning for multiple agents [60][11][35][5].

Task Scheduling

A key component of any practical agent system is agent scheduling. Task scheduling is a big concern in the multi-agent simulations where agents always try to keep their performance higher and reprioritize the tasks at different time intervals [19]. The agents have to take into account their own preferences and availability for the execution of the tasks. Scheduling is a distributed task which is time consuming, iterative, and tedious [64]. It can take place among two persons or several persons, for example scheduling of visits in a day by the nurse to the patient in a hospital. In a hospital settings where various different agents work they have different tasks, for example the nurse agents in a hospital have various different tasks, such as serving a patient agent, medicating, writing reports, taking breaks, consulting, and so on. Multi-agent planning and scheduling problems arise in many contexts such as supply chain management, coordinating space missions, or configuring and executing military scenarios. In these situations, the agents usually need to perform certain tasks in order to achieve a common goal. The multi-agent systems need to operate in a real time that involve both scheduling and task execution. The characteristics of an environment requires substantial coordination among agents, but exclude time consuming and elaborate coordination activities.

Social Factors

In complex models, agents have an important role to play as social actors [65]. In human cultures, in absence of social interactions it is very difficult for the agents to work in coordination and have other agent's corporation. Social interactions are important to relieve from stress and maintain one's own vigor. In complex agent models, where environment changes rapidly resulting in different situations for agents, and where agents work under high stress; it is really important for the agents to interact for mutual help. Social interactions are also important for the agents to work in corporation and in coordination in order to keep their performances high and to remain socially active [51][15].

Domain Knowledge

The knowledge which is an object of discourse is known as domain knowledge [61]. Domain knowledge is knowledge about the environment in which the agents operates. Domain knowledge is used to achieve intelligent behavior and it is necessary to represent that knowledge in order to draw conclusions from it [25]. Domain knowledge is an essential and central part of AI for any entity to behave intelligently [50]. Moreover, to solve any problem every intelligent entity or agent requires extensive knowledge about the world. In AI, knowledge in the computer systems is thought of as something that is explicitly represented and operated on by inference processes. Domain knowledge could be a declarative or a procedural. Declarative knowledge is represented as a static collection of facts with a set of procedures for manipulating the facts. Procedural knowledge is described by executable code that performs some action. Procedural knowledge refers to how-to do something. In this research the nurse agents have a procedural knowledge in context of finding path, communicating and doing different tasks of serving, medicating and report writing (refer to section 4.5.2). Declarative knowledge is used with the patient agents in context of waiting for nurse agents for specific interval of time. Usually, there is a need for both kinds of knowledge representations to obtain an intelligent behavior in a particular domain.

2.5 Decision Support Systems (DSS)

Making decisions concerning complex systems, for example, the management of organizational operations, industrial processes, or investment portfolios, the command and control of military units, the control of nuclear power plants and so on often strains our cognitive capabilities [47]. Even though individual's interaction among the systems' variables may be well understood, predicting how the system will react to an external manipulation such as a policy decision is often difficult. There is a substantial amount of empirical evidence that human intuitive judgment and decision making can be far from optimal, and it deteriorates even further with complexity and stress. Because in many situations the quality of decisions is important, aiding the deficiencies of human judgment and decision making has been a major focus of science throughout history [13]. Disciplines such as statistics, economics, and operations research developed various methods for making rational choices. These methods often enhanced by a variety of techniques originating from information science, cognitive psychology, and artificial intelligence have been implemented in the form of computer programs, either as stand-alone tools or as integrated computing environments for complex decision making. Such environments are often given the common name of

DSS.

Decision support systems have become popular because of their capability to fill the need of decision making by given information. It is a class of information systems that support business and organizational decision making activities. It couples the intellectual resources of individual's with the capabilities of the computer to improve the quality of decisions [13]. It is a computer-based support for management decision makers who deal with semi-structured problems. A properly designed DSS is an interactive software-based system intended to help decision makers to compile useful information from a combination of raw data, documents, personal knowledge, or business models to identify and solve problems and make decisions. A DSS is a general term for any computer application that enhances a person or group's ability to make decisions. It can also be used as a tool in which user inputs the data and the software component process the data and decisions are made on the basis of the information given. In order to make the decision making tool all the major components of the system should be considered in the system to get an optimal result. Figure 2.1 shows the decision support system flow diagram in any decision support system.

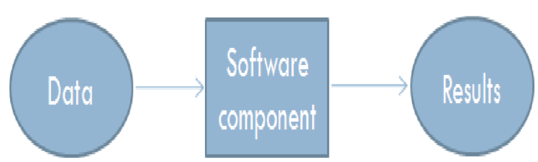

Figure 2.1: Decision support system flow diagram

For example: a national online book seller wants to begin selling its products internationally although they need to determine if that will be a wise business decision. The vendor can use a DSS to gather information from its own resources (using a tool such as OLAP) to determine if the company has the ability or potential ability to expand its business and also from external resources, such as industry data, to determine if there is indeed a demand to meet. Another example is, a hospital management wants to examine the various different factors that affect the nurse's performance in the hospital in order to make any decisions regarding the allocation of nurses or changing other environmental factors after observing the results. The DSS will collect and analyze the data and then present it in a way that can be interpreted by humans. Some decision support systems come very close to acting as artificial intelligence agents. DSS applications are not single information resources, such as a database or a program that graphically represents sales figures, but the combination of integrated resources working together.

Decision support systems are valuable in situations, where the amount of available information is prohibitive for the intuition of an unaided human decision maker and in which precision and optimality are of importance [55]. Decision support system can aid human cognitive deficiencies by integrating various sources of information, providing intelligent access to relevant knowledge, and aiding the process of structuring decisions. Decision support system can also support choice among well defined alternatives and build on formal approaches, such as the methods of engineering economics, operations research, statistics, and decision theory. Decision support system can also employ AI methods to address heuristic problems that are intractable by formal techniques. Proper application of decision making tools increases productivity, efficiency, and effectiveness, and gives many businesses a comparative advantage over their competitors, allowing them to make optimal choices for technological processes and their parameters, planning business operations, logistics, and investments. Decision making is a fundamental managerial activity. It may be conceptualized as consisting of four stages: intelligence, design, choice and implementation.

Important Aspects of the DSS :

(1) The most important consideration is, DSS's ease of use, such that its ability to allow non-technical people to deal with it directly. The single greatest and most enduring problem with computers has been their inflexibility, their inability to let the person who actually needs the data to deal directly with the computer.

(2) The ability to access information should not be restricted to only the part of an organization or to only certain managerial or professional groups. Instead the resource should be distributed to all of the people and part of an organization needing it without widespread access; the power of advanced distributed processing system will go untapped as they typically have in the past.

(3) The ideal DSS in sharp contrast to previous method of designing applications should not be a 'system' at all in the strict sense of the term. Rather, it should be a highly adaptive decision support generator that can easily be used by professionals to quickly design data support prototypes suited to each specific decision-making task.

(4) To adequately support the human element, this highly adaptive support capability must be able to provide access to operational data and as well as to summary data that already has been processed by application programs designed for other specific operational tasks.

(5) The organizations need to access original data sometimes because efficiency is related to how well the original data is organized in the system; the decision support

generator should be able to interface with a true data base management system.

(6) The management or professional information workstation would incorporate a keyboard, display screen and an interface to a printer which could print everything from straight text to graphics like pie charts, bar charts and line charts.

(7) The support tool must interface with several different systems and capabilities, it must be compatible with all of them, the tool must provide users with a single easily used language to access manipulate and present data in a way that will best support the end-user.

Abilities of the DSS:

(1) Support decision making in ill-structured situations in which precisely owing to the lack of structure, problem do not lend themselves to full computerization, and yet require computer assistance for access to and processing of voluminous amount of data.

(2) Help to rapidly obtain quantities results needed to reach the decision.

(3) Support various stages of the decision making process.

(4) Foster high quality decision making by encouraging decisions based on the integration of available information and human judgment.

(5) Offer flexibility as opposed to a preordained pattern of use, making it easy to accommodate the particular decision making style of an individuals.

(6) Facilitate the implementation of the decisions which frequently cut across department boundaries.

(7) Support group decision making particularly through group DSS (GDSS).

(8) Give organization the opportunity to gain a better understanding for their business by developing and working with models.

Decision support systems are used in various domains from military, education and any other industry. Decision support system tools are been used to improve the effectiveness of the system rather than examining the efficiency of the systems. These tools are potential measuring tools which are used for management purposes. In DSS the decisions are largely dependent on the quality of information available. While concluding any decisions always best case scenarios are considered with no faults in a system. Problems arise when the quantity of available information is huge, and non uniform and is difficult to make any conclusion from the information given. The quality of decisions in DSS could not be stated in advance. To build a DSS for observing a complex system is a challenging task. Complex systems are highly dynamic in nature which consists of various parameters that affect the system. Once we have all the important parameters of the complex system in the DSS it is easy to conclude the decisions but it is hard to decide that which parameter has more influence on the system.

In a complex system where various agents work in cooperation with other agents under different environmental conditions, there are myriad of factors behind an individual's decision [14][16]. Under these circumstances it becomes very difficult to decide which factor has more influence on the particular decision. For example, in a hospital setting where nurses work under high pressure and stress and when they have to take critical decisions like which patient's life they have to save, probability can tell us very little about what is going behind the scene. In order to understand the decision making process in a given human system and conditions, we need to account for the individual, social and environmental factors for the decision making process.

Chapter 3

Approach

In [15] “A complex system is a system for which it is difficult, if not impossible to restrict its description to a limited number of parameters or characterizing variables without losing its essential global functional properties”. Modeling social human environments in MAS is always a complex and difficult task. The social systems are modeled using simple agents. Social complex systems consist of many individuals (autonomous agents) who interact non-linearly within the environment and their behavior is unpredictable. The interactions and agent’s decisions often affect their behavior resulting in different working conditions for them within the system. The result is that it becomes impossible to analyze a society as a whole by studying individuals one at a time. In complex systems, components have properties of self organization which makes them non-predictable and the behavior of the society is said to emerge from the actions of it’s units or agents. Simulation is a third way of doing this science with which one can observe the behavior of a system and elucidate conclusions [4]. Complex environments in which the agents should reflect human like qualities and behaviors, includes various parameters to be considered in a simulation. Different properties can be included in the system to have it behave more like a human environment such as, communication, agent's movement, work stress, and so on [49]. However, system's complexity increases and it becomes more prone to errors which could lead to incorrect results.

In theory, all models are simplifications of reality, on the contrary there is always a trade-off as to what level of detail should be included in the model [20]. If too little detail is included in the model, one runs the risk of missing relevant interactions and the resultant model does not promote understanding. If too much detail is included in the model, the model may become overly complicated and actually preclude the development of understanding. In order to observe and understand the behavior of a system as whole, it is important to include the key factors that have a substantial effect on the system. However, there are various other factors that can be ignored in such a way that the system should not lose its relevant behavior. In social modeling, there are various factors (such as path finding, task scheduling, socialization, domain knowledge, and so on) that are common for different social environments. In a hospital domain there are various different factors that are important and have a significant affect on patient care and safety. These factors are communication, situation awareness, decision making, stress, fatigue, and working environment [18].

Multi-agent systems have an important role in the building of complex systems, and agents are the main actors in these systems. The behavior of the agents in these systems is not predictable but it can be observed by visualizing these agents in their environment with the help of simulation. Multi-agent systems lack in various properties that makes them less likely to be predictable in their behavior, however it is a base of building an artificial social complex systems. Including different factors to the MAS is an attempt to develop a more realistic artificial social simulation models. In the beginning, it is important to keep the complexity of the social complex systems low to validate the working of the system, and we can add more properties to the system later.

3.1 Important Key Factors in Social Complex Modeling

In this research, we have chosen a hospital domain for the study of artificial social complex modeling. In a hospital domain, patient safety and patient care are the central concerns for the hospital staff management for the quality patient service delivery. There are various factors underling that have considerable influence on the performance of hospital staff that ultimately affects the patient care in hospitals. In order to observe the effects of the hospital staff behavior and different hospital environmental conditions on the patient service, we have included the important key factors such as path finding techniques, task scheduling techniques, socialization, and domain knowledge to the agent modeling. These factors are included in the agent modeling on the basis of the different studies done on the factors affecting the patient safety in hospitals [18].

3.1.1 Path Finding

The path taken by the agents depends upon the external environmental factors and agent's own beliefs [57]. In certain domains the shortest path is the best way to serve the clients, especially when the clients are eager to get the service as quickly as possible. In these domains agents follow the shortest path and serve other agents. In contrast, in realistic environments it is not always feasible for the agents or the humans to navigate on the shortest path, because of the social distractions. However, the path agents follow could be the shortest path. There are various factors that influence the path of an agent such as social distractions, personal preferences, domain knowledge, and so on. Due to the influence of these factors the shortest path remains no longer shortest and becomes a social path, or in the other words, socially influenced path.

Agents usually follow the shortest path in a domain where service is required quickly. Dijkastra's algorithm is used to calculate the shortest path from one vertex to another. In this model, the nurse agents use same algorithm to calculate their path from their initial location to the patient agent's room.

The Dijkastra algorithm is a graph search algorithm that solves the single-source shortest path problem for a graph with non-negative edge path costs, producing a shortest path tree [28]. This algorithm is often used in routing. For a given source vertex (node) in the graph, the algorithm finds the path with lowest cost (such as the shortest path) between that vertex and every other vertex. It can also be used for finding costs of shortest paths from a single vertex to a single destination vertex by stopping the algorithm once the shortest path to the destination vertex has been determined. For example, if the vertices of the graph represent cities and edge path costs represent driving distances between pairs of cities connected by a direct road, Dijkastra's algorithm can be used to find the shortest route between one city and all other cities.

3.1.2 Task Scheduling

In dynamic environments unpredictable events happen which affect the agent's behavior and their priority for a particular task. In social environments, agents execute their tasks in the order the tasks come in, or on the basis of their priority. In some domains, such as hospitals where nurses have different priority for the different patients depending on the patient's physical condition, nurses have different priorities to serve them.

In social environments, the agents try to keep the success rate of the execution of primary tasks higher to keep the overall service success rate high and balance their own vigor. In reality, when someone knows that they will not be able to finish the current task on time, and if they can estimate that if they will start the current task they will definitely miss the second following task. Then one can intentionally miss the current task and start doing the second following task to keep the successful execution of tasks higher and can serve the missed task after finishing the other task (refer to section 4.5.2).

In order to keep the success rate of the execution of primary tasks higher, agents pre-empt their current tasks so they can successfully execute the tasks following. Pre-empting of tasks serves the current waiting task later, and does the second waiting task first to keep the service success rate of execution of tasks high. In some cases, the agents have long waiting times and they wait until they get service.

3.1.3 Social Factors

Social factors are very substantial for the agents which influences them to decide their path [27]. The tendency of an agent to interact with the other agents can detour it's path which effects it's own performance and the overall success rate of tasks. In social environments, agents work in co-ordination with peer agents and agents get distracted by the peer agents because of various different reasons (such as general discussions, enquiring and so on) which may lead the agents to detour from it's actual path to some other path where they spend more time along their way. These social distractions ultimately affect the quality of service to the patient and the overall performance of the system.

3.1.4 Domain Knowledge

Domain knowledge in complex systems help the agents to decide their path, such that which path they should follow and with which agent they should talk to. These decisions are very crucial in the complex systems which help agents in mutual coordination and co-operation to resolve the adverse events in a system or to service other agents [14]. Domain knowledge help agents to decide on the priority of the tasks, such that which tasks are more important for them to do first in order to keep the success rate of the execution of tasks high. Domain knowledge, that agents posses is very helpful to achieve human like behavior and to obtain accurate results from the complex systems by means of simulation [16]. In this hospital simulation model, the nurse agents posses the procedural knowledge about the working environment such as the floor plan of the hospital which helps them to navigate on the path to the patient agents. The nurse agents are acquainted with the patient agents and their physical conditions, which helps them in providing service to the patient agents.

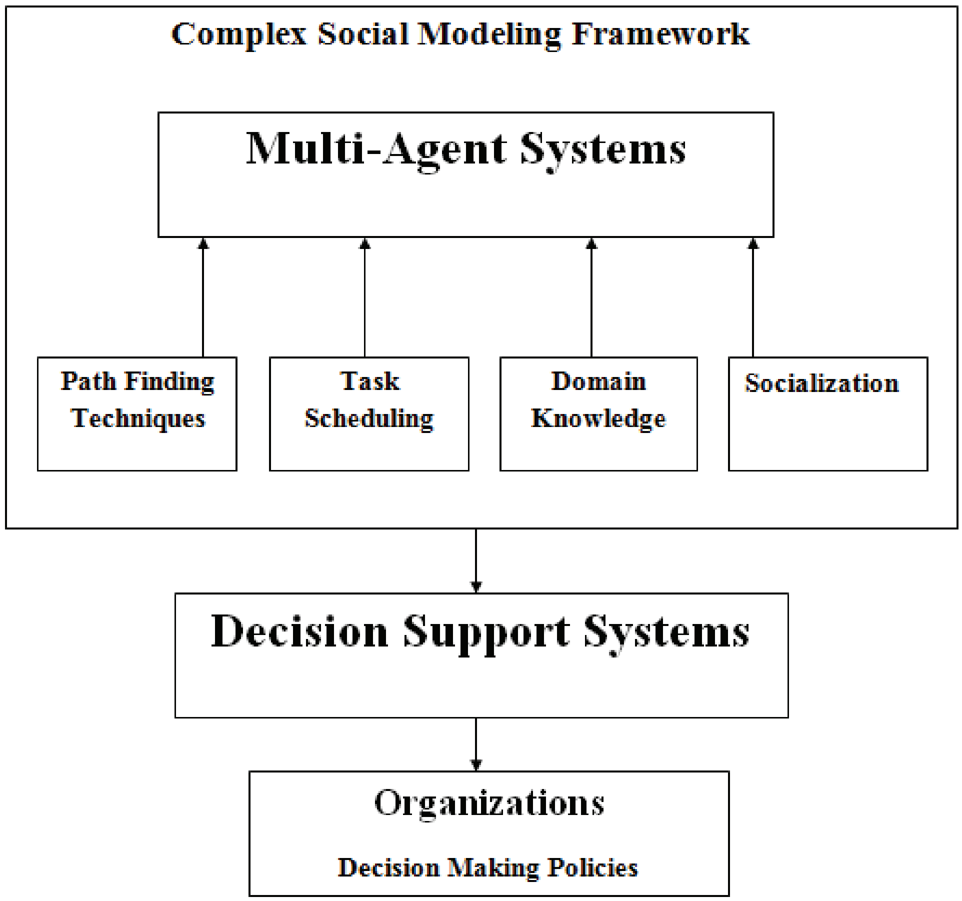

Figure 3.1: Complex social modeling framework

These are the important key factors that can be included in the MAS with an attempt to imitate the working of complex social systems. The artificial complex social systems can be further used as a DSS by large organizations for decision making policies and for system's behavior prediction. The properties of the systems can be changed and their behavior can be observed with different settings. Figure 3.1 shows the complex modeling framework which includes the MAS that is comprised of path finding techniques, task scheduling techniques, domain knowledge and socialization in the agents. This framework could be helpful in developing a decision making tool for different domains. In this thesis, we have used this framework for the hospital domain.

Chapter 4

Case Study: Base Hospital System

4.1 Health Care and Simulations

Health care is the largest industry in Canada and the U.S., and is characterized by very complex and dynamic environments where health services are delivered. Health care is characterized by dynamics, variety, and fragmentation of distributed medical prevention, diagnosis, treatment, and rehabilitation processes. The health care institutions struggle to meet the increasing demands for services and the rising costs of these services. The sustainability of Canadian and U. S. health care systems may be dependent on achieving innovation in productivity and cost effective strategies for health care delivery [32].

While attempting to achieve more cost effective health care delivery, patient safety and risk must be a central concern. Patient safety is a serious global public health issue in a health care systems that represent a fundamental principle of health care [53]. Adverse Event (AE) is defined as an injury caused by medical management rather than underlying disease that prolongs hospitalization, produces a disability at the time of discharge, or both. Adverse events are also referred to as untoward incidents, therapeutic misadventures, iatrogenic injuries, or other adverse occurrences directly associated with care or services provided within the jurisdiction of a medical center, outpatient clinic, or other facility [53]. Adverse events increase the length of hospital stay and have a substantial impact on patients which includes discomfort, injury, increased morbidity or even death. Adverse events result in psychological implications for the patient with a decrease in self-confidence and build fear. This contributes to a decrease in mobility and culminates in a significant reduction in a quality of life. Although only a small number of failures in service produce significant injury, the impact on the health care system can be massive, resulting in an increased length of hospital stay, possible further surgery and a more unhealthy patient population.

Simulation has been used for modeling health care systems for over 40 years [10] to improve health care quality and efficiency of operations, reduce medical errors, identify new approaches, and to reduce delivery costs [26]. Agent-based modeling and simulation is a contemporary approach to modeling artificial systems comprised of autonomous, interacting agents [39]. The field of agent navigation is continuously evolving [60]. In the design of realistic artificial intelligence, it is a challenge to capture the reality of how agents act and interact [21]. In this model, the nurse agent's movement is illustrated using path finding techniques which are usually deployed at the core of every AI movement system. This ability becomes critical when agents operate in dynamic environments, where unpredictable changes may occur. The optimal criterion for the nurse agent's path is determined by the surrounding hospital environment and the running conditions of the model.

The use of simulation into the current healthcare environment is beneficial, because there is a critical need for powerful tools which can help clinicians and administrators to make good decisions on reducing costs while maintaining high quality care in the hospitals.

4.2 Patient Care in Hospitals

Patient safety represents a fundamental principle of health care, and is simply defined as ”the prevention of harm to patients”. It is a serious global public health issue in health care system. [53] [1].

Improvement of health care quality and patient safety are of critical importance to nurses since they have the most consistent presence at the patient’s bedside. In real time dynamic hospital environment, nurses endeavor to spend time with their patients in order to meet their varied and complex needs. However, there are various parameters which have an impact on the nurse’s ability to deliver patient services. Socialization, fatigue, number of patients assigned to each nurse, shift work (such as, day shift, night shift, long shifts (12 hours)), nurse scheduling, patient care planning and time for which the patients are able to wait for nurses to come (which is different for different patient) are all important factors which affect patient service. With all of these influences on nurses’ ability to care for patients, it is difficult to identify how these multiple factors contribute to the patient service. The more closely a patient is monitored by a nurse, the more likely the nurse will be able to observe and assess for any changes in their condition and can respond quickly before the patient’s condition worsen. In this study, it is assumed that the more time a nurse is able to spend with a patient, and the more readily the nurse is able to respond for patient calls for assistance; the better will be patient care. Agent based modeling provides us an important tool for examining these assumptions in health care delivery. With the help of agent modeling, one can simulate the social environment, such as hospital environment and the key factors that are important in the patients care. Once we have the running simulation, it’s behavior can be observed by running the simulation multiple times and one can observe the behavior of a particular parameter by changing its values and comparing it with the base model.

Social factors are very substantial for nurses' to decide their path. Social networks are a set of people, organization, or social entities with social relationships and informal connections between people in a society [27]. The relationships are often ones of communication, awareness, trust, and decision-making [51]. In our simulation, we have two different social networks, nurse-to-nurse and nurse-to-patient network which defines the nurse-to-nurse and nurse-to-patient communication and connections. The nurse-to-nurse social network is an immense network with various connecting nodes relative to the nurse-to-patient network, because only a few patients are assigned to a particular nurse and patients have communication limited to that nurse. Each nurse has two networks, one with working colleagues (nurses) and another with the assigned patients. However, a patient is only connected to the nurse assigned. The task scheduling is another big concern in the multi-agent simulations where agents always try to keep their performance higher and reprioritize the tasks at different time intervals. In our simulation the nurses do the same, in that way they always try to keep the patient service higher by missing fewer number of patients waiting for service. The nurses reprioritize their tasks and try to serve more patients on time.

4.3 Hospital Case Study

A hospital case study is a simulation that aims to model the patient service delivery and nurse's time utilization on different tasks in a typical hospital. In this model, a complex array of selected variables responsible for the social environments such that the factors contributed by the nurses and the patients behavior are implemented and their effects are examined on the outcome of nurse response time to the patient requests. Using detailed simulation modeling, a nurse's time utilization in such a flexible dynamic workload environment was investigated, which includes the time she spent on different tasks in a daily shift. With this tool design decision makers can identify and prevent such conditions that effect service quality levels. The goal of this study is, to use artificial agent simulation to critically examine the factors and influences that contribute to patient service in a hospital setting. In this study, we describe the use of agent modeling simulations to examine the influence of the hospital environment (length of shift, time of day etc.), the behavior of the nurse (path taken, social interactions, task execution etc.), and the patient agent (waiting time, severity etc.) on the outcome of a nurse response to a patient request.

A hospital case study is a study of the real environment that can be used to maximize the quality of service delivery of the nurses to the patients within community hospital settings using the appropriate identification and care for people at risk. The hospital case study simulation has been developed with the guidance of a team of nurses working in a small rural Ontario hospital and different studies conducted on patient care in hospitals [53][32][18][3][24]. The majority of their patients were elderly (such that, over 70 years of age) and were considered at high risk for injury. The research team spent time with the team of nurses to more closely understand the working environment of nurse in a hospital.

In hospital case study, the dynamically changing environment leads to the different working conditions for the agents in a hospital. In this agent simulation, the nurse agents have the familiarity with the floor plan of the hospital which is considered as their domain knowledge. We assume that nurses take the shortest path to serve the patient as quickly as they are able to, without any distractions from other agents. In reality, the path nurses follow is not always the shortest path because of the social influences during any particular shift. A nurse's path is influenced by social distractions in a hospital environment, which lead nurses to detour from an original shortest path and they start following another path.

Nurses schedule their tasks during their shifts to keep the patient service success rate high and take breaks for balancing their own energy.

4.4 Hospital Floor Plan

In this model, the actual floor plan of the hospital was embedded in the model to make the simulation more realistic and to achieve greater validity of the results. The floor plan of the hospital included areas such as hallways, staircases, elevators, waiting areas, cafeterias, doctor and staff offices, patient rooms, nurse stations as well as patient and public washrooms. The physical establishment is represented as a weighted directed graph, where the vertex v represents a particular area and each edge e represents the ability to travel between two adjacent areas. Thus, edge e connects out-vertex vo and in-vertex vi. The weight of each edge represents the travel time from vo to vi. Other properties such as area temperature, humidity, floor friction and lighting are set to constant in this iteration of the model. Model time is the global clock with an event queue where a time unit represents the amount of time it takes a normal person to walk a distance of n meters. The walking speed of a nurse agent has also been set to the speed of a normal person. The nurse agents in this model move at the same speed as a normal person does in any work environment.

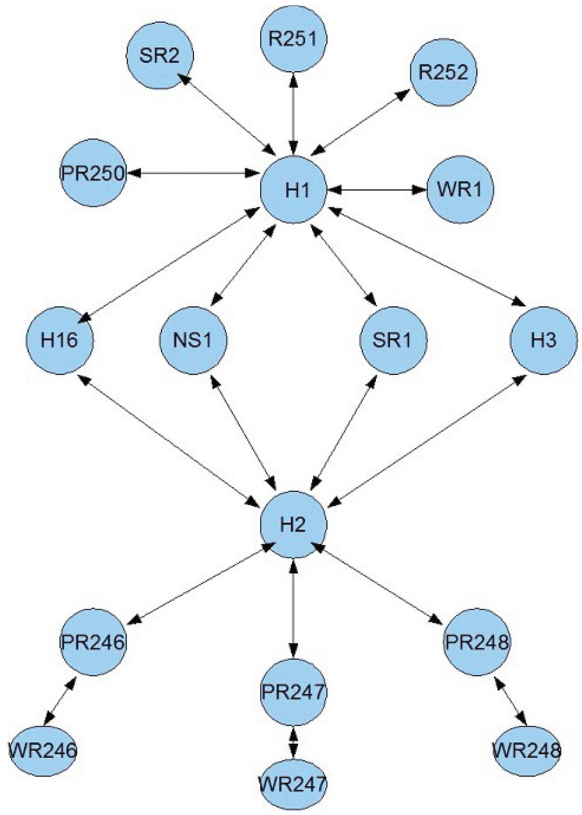

Figure 4.1: Connectivity of the rooms and hallways in a hospital floor plan

Figure 4.1 is the example of a floor plan of a hospital and how rooms and hallways are connected with each other. In the Figure 4.1 hallway H1 is connected to patient rooms PR250, PR251, PR252, nurse station NS1, store rooms SR1, SR2, and another hallways H16 and H3. Hallway H1 is connected to another hallway H2 via H16, NS1, SR1 and H3.

4.5 Agents in Hospital

Agents are autonomous, active, persistent (software) components that perceive, reason, act and communicate. In hospitals, there are various kinds of agents such as, doctor agents, nurse agents, patient agents, visitor agents, cleaning staff agents, other staff agents and so on. In this model, it is assumed that the majority of a nurse's time on the hospital unit is spent providing care to patients and their families. Considering this fact, we have limited the implementation of agents to a nurse agent and a patient agent, to provide clarity in the model. Base model hospital simulation runs for different time steps in which one time step is equivalent to 12.5 seconds in a real world.

4.5.1 Patient Agent

Characteristics and Modeling Assumptions

(a) Patient agents are at fixed locations and do not move.

(b) In hospitals, the patients have different physical conditions, some patients are very critical and characterized by a very severe condition, some patients have moderate severity and other have low severity. In this model, patient agent's severity level is measured on the scale ranging from 0 to 1. Patient agent severity is always known by the nurse agent assigned.

Scheduling of Visit Time

Nurse agents are suppose to visit the patient agents after every hour in a hospital. One hour is 288 time steps in our simulation. As the severity goes higher the visit time is scheduled more frequently, for example if the severity is 1 the visit time would be the current time in addition to the time corresponding to severity 1 which is 0. According to the equation visit time would be,

$$\text{Visit Time} = -288 * 1 + 288 + \text{Current time} = \text{Current Time}$$

This shows that if severity of the patient is 1 then he should be visited right away. Every patient agent has different waiting times (TotalTimeToWait) to wait for a nurse agent to respond to patient's call for assistance. Some patient agents wait for a short time, some wait for longer period of time. Patient agents wait for a nurse agent until their TotalTimeToWait. If the nurse agent visits a patient agent within its waiting time, then the patient agent is considered a "patient served" and if it nurse arrives late to the patient then it is considered as "serviced late".

Severity of Patient Agent

In this model, it is assumed that patient agents stay in their rooms and do not move from their locations. The patient agents have different severity levels which defines their physical condition or degree of illness. There are different values of severity, which are random numbers generated between 0 and 1 and are known by the nurses assigned. The patient agents with higher severity such as severity greater than 0.7, are considered critical condition patients. Visit time is scheduled based on the severity level. If a patient agent is more severe than other patient agents, then the nurse agent is expected to visit the patient agent more often. After every visit by the nurse agents, the severity level of the patient agent changes and the visit time is scheduled again on the basis of that severity level. Visit time is scheduled for every patient agent and the nurse agent visits the patient agent within the expected time schedule. Following is the equation to calculate the visit time for the patient agent:

$$\text{Visit Time} = -288 * \text{Severity of Patient Agent} + 288 + \text{Current Time}$$

This equation illustrates that as the severity of the patient agent increases, nurses are suppose to visit them more frequently.

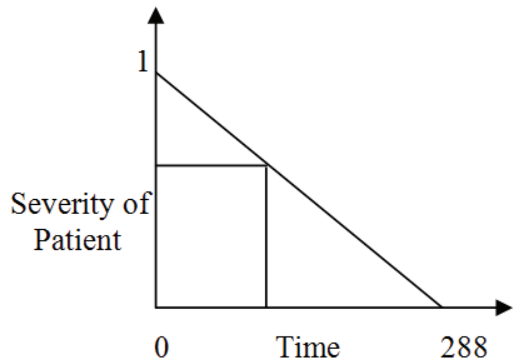

Figure 4.2: Slope for calculating visit time

4.5.2 Nurse Agent

Nurse agents are assumed to be at nursing station in the beginning of the simulation. Nurse agent possesses the knowledge about the floor plan of the hospital which is their domain knowledge. Nurse agents walk at the speed of 0.8 m/sec which is equivalent to the speed of normal person's walk [12]. The equal numbers of patient agents are assigned to the nurse agents. Nurse agents start their shift by serving the patient agents assigned starting from the patient with the higher urgency of need. When nurse agents reach the patient agent they must finish the multiple tasks of serving, medicating and writing report for the patient agent. Nurse agents can consult or get distracted by the other nurse agents in the hospital during their shift.

Characteristics and Modeling Assumptions

(a) Nurse agents are initially positioned at nursing stations.

(b) Nurse agents are familiar with the floor plan of the hospital such that, they know which path is a shortest path to the patient room.

(c) Nurse agent's velocity (walking speed) is .8 m/sec [12].

(d) A particular number of patient agents are assigned to each nurse agent and only the assigned nurse agent serves those patient agents (for example, 5 patients assigned to each nurse).

(e) Nurse agents visit the patient agents after every hour and if a patient agent is more severe then the frequency of their visits increases.

(f) Nurse agents serve the patient agent for MinTimeServePatient, which is a minimum amount of time for which nurse agent spent serving the patient agent.

(g) Nurse agents do medication of the patient agent after serving for MinTimeMed, which is an amount of time nurse agents spent to bring the medication and give it to the patient agent .

(h) Nurse agents write the report for the patient agent after medication task for MinTime-Doc, which is an amount of time nurse agents spent on writing the report.

(i) Once the nurse agent started serving the patient agent she should finish all the consecutive tasks that are medication and documentation.

(j) Nurse-to-nurse interactions happen frequently in any particular shift, however in this model nurses interact a maximum frequency of every 40 minutes.

(k) All nurse agents takes 10 minutes break after every 2 hour. Only few number of nurses go for a break at any given time.

(l) Nurse agents normally work for eight hour shifts which can be extended to 12 hours.

(m) There are always fewer number of nurse agents at night shifts than who work on day shifts.

Task Done by Nurse Agent

(a) **Travelling and consulting task:** The nurse agents begin their shifts by visiting their assigned patient agents, starting with the patient agent having greatest urgency of need (due to the illness). The model assumes that the nurse agents always try to go through the shortest path to the patient, which is decided on the basis of Dijkastra algorithm. Along the way the nurses could distracted from their original path due to the social interactions, because interactions with health professionals are important for patient care delivery in the hospitals. On the contrary, the nurse agents also try to meet the patient agent's needs as quickly as possible with the shortest possible time to reach the patient agent to respond to their request for assistance.

Consider the figure 4.1 again if a nurse agent has to go to the patient agent room PR247 from PR250 she can follow four different paths,

- Path 1: PR250-H1-H16-H2-PR247
- Path 2: PR250-H1-NS1-H2-PR247
- Path 3: PR250-H1-SR1-H2-PR247
- Path 4: PR250-H1-H3-H2-PR247

In order to serve the patient agent quickly, nurse agent would follow the shortest path calculated by Dijkastra algorithm, which is Path 2. But along the way she could have different circumstances and external distractions which would lead her to follow different path which could be longer. The shortest path would become the socially influenced path which could be shortest or longest.

(b) **Serving patient task:** Once a nurse agent responds to a patient agent she must start serving the patient agent which includes regular check-up, patient assessment, and other observations. A nurse agent serve the patient agent for *MinTimeServePatient* which is a minimum time for which every nurse agent has to be with the patient agent and *MinTimeServePatient* could vary according to the severity of a patient agent.

$$MinTimeServePatient = MinTimeServePatient + (MinTimeServePatient * PatientSeverity) + (MinTimeServePatient * ProbRandEventRate)$$

Where *PerRandEventRate* is a percentage of any unpredictable AE that could happen during the patient agent service. *ProbRandEventRate* is a number between 0 and1, which represents a probability of any AE that could happen during the patient agent service and prolong the nurse agent's stay with a patient agent.

(c) **Medication task:** After a nurse agent done with serving the patient agent she start doing the medication for *MinTimeMed* which is an amount of time a nurse needs to get the medication and give it to the patient agent. Medication task also depends on the unpredictable events that could happen in the hospital. After medication the severity of the patient agent changes and the time for next visit is scheduled for the patient care.

$$MinTimeMed = MinTimeMed + (MinTimeMed * PatientSeverity) + (MinTimeMed * ProbRandEventRate)$$

(d) **Documentation task:** After finishing the medication task with the patient agent, nurse agent has to write a report regarding the observations of the patient agent and what type of medication she has given. The nurse agents does this task for *MinTime-Doc*, which is a variable and an amount of time she needs to write a report.

Decision Rules and Algorithm for an Individual Decision Making

(a) **Preempting of tasks:** When a nurse agent starts her shift she has a particular number of patient agents assigned and a task list to serve the patient agents. In her shift, a nurse agent always attempt to serve more patient agents on time to keep the service success rate of patient agents higher. In order to do that she sometimes misses first waiting patient (PA1) intentionally and serves the next waiting patient (PA2) and serves the first waiting patient (PA1) later. Pre-empting of tasks is to serve the current waiting task later and do the second waiting task first to keep the service success rate of execution of a task high. In order to miss one patient agent she does an estimate before starting, for the first waiting patient agent and the second waiting patient agent in a task list. The nurse agent serves the second waiting patient agent (PA2) first only when she does an estimate that she will not be able to serve the first waiting patient agent (PA1) on time. If the nurse agent goes to the first patient agent (PA1) she will definitely miss the first patient agent (PA1) and the second patient agent (PA2) too and also when she checks that first patient agent (PA1) is not critical, such that severity is smaller than 0.7. Following are the conditions in algorithm for the nurse agent to pre-empt her task and all the conditions must be true before she can pre-empt any task,

- *Condition 1:*If time to reach first waiting patient agent (PA1) is greater than the current time subtracted from the visit time of first waiting patient agent (PA1).

- *Condition 2:*If the visit time of second waiting patient(PA2) is smaller than the current time added to the time to reach first waiting patient agent (PA1) added to the time to reach from first waiting patient agent (PA1) to second waiting patient agent (PA2).
- *Condition 3:*If the first waiting patient agent (PA1) has severity smaller than 0.7.

(b) **Nurse agent break:** In an 8 hour shift, nurse agents are assumed to take breaks approximately after every 2 hours. The break lasts for 10 minutes. When nurse agents work 12 hour shifts, fatigue factor effects their performance [18] and they start taking breaks more often (after one and a half hour) and for longer interval that last for 15 minutes. Nurse agents do not go on breaks together, only 30% of nurse agents go on break at time, for example if a nurse agent has to take a break, she will check a break log if 30% of nurse agents are on break, then she will wait until any nurse returns. Nurse agents always take breaks when they are idle, such that they have no task to do at that time. The following are the conditions for a nurse agent to decide whether she should go on break or not and all must be true before she goes on break,

- *Condition 1:*If nurse agent is Idle.
- *Condition 2:*If number of nurse agents on break is less than 30%.
- *Condition 3:*If nurse agent has not been on break for last three hours.
- *Condition 4:*If a nurse agent does not have any patient agents to visit within her break time.

These estimates on a nurse agents work hours and break time were determined in collaboration with the nurse agents working in the rural Ontario hospital that partnered with the research team. Visit times are scheduled on the basis of the level of severity of the patient agent. The greater the severity of the patient agent's condition, the more frequent the visit will be for that patient agent by the nurse agent as described earlier.

(c) **Modeling of visit time:** Nurse agents serve patient agents for *MinTimeServePatient* which is the minimum time for the nurse agent to be with patient agent and *MinTimeServePatient* could vary according to the severity of patient agent.

*MinTimeServePatient = MinTimeServePatient + (MinTimeServePatient * PatientSeverity) + (MinTimeServePatient * ProbRandEventRate)*

(d) **Unpredictable events:** Unpredictable events are the events that happen in a hospital without any notification. These events are also called AE that prolong nurse agent's service time because nurse agents have to overcome these AEs. For example, if a nurse agent is doing a medication task where *MinTimeServePatient* is 15 time steps (15 * 12.5 seconds) and patient severity is 0.5 and probability of happening of AE in hospital is .2 then according to the following formula,

*MinTimeServePatient = MinTimeServePatient + (MinTimeServePatient * PatientSeverity) + (MinTimeServePatient * ProbRandEventRate)*

MinTimeServePatient = 15 + (15 * 0.5) + (15 * 0.2) = 25.5 This is 25.5 * 12.5 = 318.75 seconds. Where *PerRandEventRate* is a percentage of any unpredictable AE that could happen during the patient service. *ProbRandEventRate* is a number between 0 and 1.

4.6 Difference Between Pre-empting, Unpredictable Events and Socialization

- **Pre-empting:** Pre-empting of tasks is a rescheduling of tasks when a nurse agent knows that the first waiting patient agent is not very critical and she will not be able to serve the next waiting patient agents successfully at a given time. If nurse agent will start doing the first waiting task she will not be able to finish the second task successfully. On the basis of the pre-empting rules nurse agents reschedule the visit time of the first waiting patient agent.

- **Unpredictable Events:** Unpredictable events are any AEs that could happen during the serving and medicating the patient that can prolong the nurse agents stay with the patient agent during the serving time and the medicating time, that could lead nurse agents to pre-empt the following tasks. When AE happens during the serving of a patient agent the nurse agents have to stay with the current patient agent for longer interval of time, depending upon the intensity of the adverse event due to which she misses the next waiting patient agents.

- **Socialization:**Socialization is a type of interaction that happens along the way to the patient agent between any two or more nurse agents. These interactions engage the nurse agents which leads them to spend more time along their way to the patient agent resulting them to reach to their destination late.

4.7 Parameters Used in Model

Repast is an open source agent modeling toolkit which was used for this simulation. Repast is a very useful tool that captures the dynamic nature of the social interactions in the model. With the help of Repast, we are able to control some of the parameters of the nurse agents and the patient agents. The following are the parameters which have been used in the model to observe how the system behavior changes by changing these parameters. There are various parameters that could have been included in the system but following are the most important parameters that should be assigned to the agents and model behavior.

Parameters	Description
MaxPatientWaitTime	Time for which a patient agent waits for a nurse agent
MinPatientWaitTime	The minimum time for which every patient agent waits for nurse agent
NumOfNurseAgents	The total number of nurse agents caring for patient agents
MinNurseTalkTime	The minimum time for which one nurse agent consults with other nurse agents
MaxNurseTalkTime	The maximum time for which one nurse agent consults with other nurse agents
NumOfPatientAgent	The total number of patient agents in the hospital
ProbOfNurseTalking	The probability that a nurse agent engaged in an interaction by other nurse agents
RunWithoutNurseTalking	(Boolean) Nurse agents are interacting with other team members or not
StoppingTime	Short shift (8 hours) long shift (12 hours)
SeverityLevel	Percentage of number of patients have severity level greater than 0.7
MinTimeMed	Minimum amount of time for which nurse agents do medication of the patient agent
MinTimeDoc	Minimum amount of time for which nurse agents write the patient agent's report
MinTimeServePatient	Minimum amount of time for which nurse agents stay with patient agent for servicing
Pre-empt	(Boolean) Run with pre-empting or without
PerRandEventRate	Percentage of random AE that could happen in a hospital model

Table 4.1: Description of parameters used in the model

Each of these parameters can be altered or changed in order to observe influence of the change on the nurse agent's ability to respond to the patient agent. For example, one can observe the change in the patient service success rate as an outcome of the nurse agent's ability or limitation in responding to patient agent requests for assistance, or the outcome of increasing the probability of a nurse agent talking to another nurse agent, or increasing the number of nurse agents on duty at the hospital. The model's behavior reflects the subsequent changes while changing these parameters.

4.8 Working of the Model at Each Step

When simulation starts, Repast loads all the floor model files and agents are created and placed onto their initial positions. Nurse agents are at the nurse stations and patient agents are in their rooms and different severities are assigned to the patients. Patient agents are assigned to the nurse agents equally and their levels of severity are known to the nurse agents who are serving the patient agents. When the shift starts, nurse agents have their patient agents list to visit them. Nurse agents start their day by visiting the patient agent with higher urgency of need (patient with high severity). If two patient agents have the same severity level then the patient agent which comes first will get served first by the nurse agent. After reaching the patient agent she does different tasks which include observing the patient agent, give medicines to the patient agent, writing report about the patient agent and so on. While doing her tasks, any random event could happen in a hospital which can prolong the task time. Nurse agents visit their patient agents after every hour in this hospital model. When nurse agents are done with their tasks they schedule the next visiting time of the patient agent depending upon the physical condition of the patient agent. If the patient agent is more sever he is supposed to be visited earlier. The nurse agents can be distracted by the other nurse agents along their way which could detour their path. Nurse agents take breaks after every two hours for 10 minutes and only 30% of nurse agents go for a break at a particular time. For example, if there are 10 nurse agents in a hospital, then after every two hours only 3 nurse agents will go for a break and rest will wait untill others return. This process is repeated in a day for 8 hours and if the shift is long, then 12 hours. Every patient agent in a hospital waits for a specific interval of time for nurse agent service which is their tolerance time or waiting time, if the nurse agent comes within their waiting time, then the patient service is considered a successful service or service on time and if nurse agent arrives late then the service is considered late.

An eight hour shift is 2304 time steps and a 12 hour shift is 3456 time steps. A small percentage of task is done at each time step and simulation is run for 2304 time steps to finish the 8 hour shift and 3456 time steps to finish the 12 hour shift. For example if a nurse agent has to serve the patient agent for 15 time steps, so for 15 time steps of the simulation the nurse agent has to be with the patient agent.

4.9 Repast (Recursive Porous Agent Simulation Toolkit)

Repast is a simulation tool that consists of a set of API to build agent based simulations easily and rapidly. This is an object oriented tool and has many platforms such as, Java, Microsoft .Net and Python. It is open source software that was developed by Sallach, Collier, Howe, North and others at the University of Chicago and later maintained by Argonne National Laboratory (ANL). It provides an integrated library of classes for creating, running, displaying, and collecting data from an agent-based simulation. Now it is maintained by nonprofit Organization Repast Organization for Architecture and Development (ROAD). Even though it is mainly intended for developing agent based simulations it is also suitable for network analysis and visual descriptions of algorithms, concepts and programs. The

Repast version that is built on the Java platform is Repast J 3.0. It is an integrated simulation development framework that provides some functionality to develop and run simulations. These functionalities are,

1. Graphical display features to demonstrate the simulation.

2. To create charts and graphs to capture the outcome of the simulation visually.

3. The user settings menu which allows the user to initialize, run, stop or pause the simulation while running.

Repast behaves as a discrete event simulator whose quantum unit of time is known as a tick. The tick exists only as a hook on which the execution of events can be hung, ordering the execution of the events relative to each other. For example, if event x is scheduled for tick 3, event y for tick 4, and event z for tick 5, then event y will execute after event x and before event z. If no events are scheduled for a certain tick, then it is as if that tick never occurred. Ticks are merely a way to order the execution of events relative to each other.

Repast simulation is primarily a collection of agents of any type and a model that sets up and controls the execution of these agent's behaviors according to a schedule. This schedule not only controls the execution of agent behaviors, but also actions within the model itself, such as updating the display, recording data, and so forth. Scheduling can be automated via the model or manually implemented by the modeler. In addition, this model is typically responsible for setting up and controlling simulation visualization, data recording and analysis. The model is said to be composed of these additional components (the schedule, the display, and so forth). Since Repast J is open source software, the users can modify its behavior any way they want. Some of the features of Repast are explained below.

4.9.1 Repast Setup

The main function routine is given below. The first routine that starts the environment is loadModel in the instantiated init object of class SimInit() that loads the object model of class BaseHospitalModel. The engine class is responsible for setting up, manipulating, and driving a simulation. The SimModel interface is the super-class for all models written with Repast. The SimpleModel class can be used to automate event scheduling as described above. The controller classes (BaseController, Controller, and BatchController) are responsible for handling user interaction with a simulation either through a GUI or by automating such interaction through the use of a batch parameter file. In addition, the engine package contains the classes that make up the scheduling mechanism.

The gui classes are responsible for the graphical animated visualization of the simulation as well as providing the capability to take snapshots of the display and make QuickTime movies of the visualization as it evolves over time. DisplaySurface, the LocalPainter class handles the actual display of these spaces on the screen, and the DisplaySurface itself allows for the probing of the displayed objects.

4.9.2 Repast Tool Bar

The repast tool bar consists of Play, step, initialize, stop, pause and exit/close buttons. Pressing initialize as self described will initialize the repast simulation plus setup the cones and provide initial locations for the agents. Initialization is not required to be pressed prior to start of the simulation. Either pressing play or step will initialize the simulation prior to continuous or stepping through the simulation. Once play has been pressed the simulation can be stopped or paused.

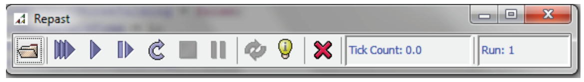

Figure 4.3: Repast tool bar 1

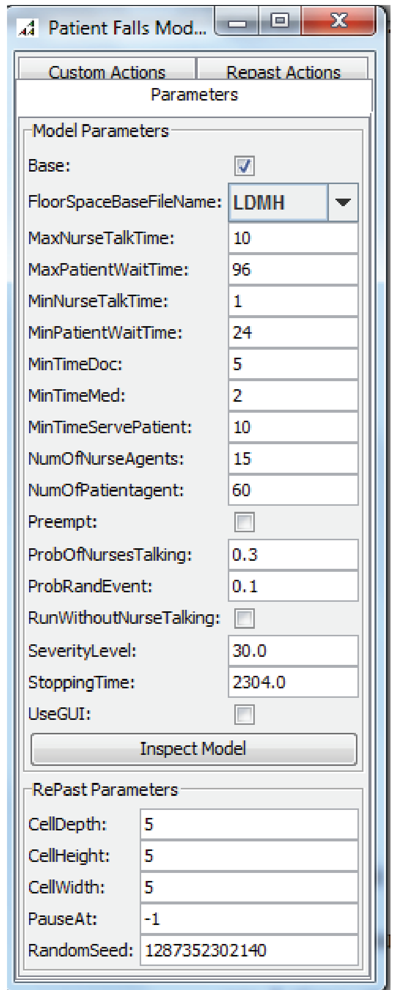

Figure 4.4: Repast tool bar 2

Chapter 5

Experiments and Results

To analyze the presented model, several experiments have been conducted to show how different factors affect the patient service success rate in different conditions. The simulation is capable of running for a specified number of hours, such as an 8 hour shift and a 12 hour shift. Different experiments have been done with different parameters and each experiment has been completed with 70 repetitions so that the average can be taken for a more valid result. In human modeling the probability of a modeling error increases due to the variability within the simulation. The variability that occurs within the simulation can be considerable because of the natural erratic randomness in simulation that is similar to the behavior of a human system. The average has been taken for the consistency and the smoothness of the variability. At any given time step the simulation outputs different data due to which standard deviation has been calculated for each time step which shows how disperse data is from the mean. In the base model, the following parameter settings were established: In the base model, the following settings have been used: maximum patient agent time is 96 time steps which is 20 minutes, minimum patient agent wait time is 24 time steps which is 5 minutes; patient agents wait for the nurse agents after the scheduled visit time at least for

Parameters	Description
MaxPatientWaitTime	96 Time Steps
MinPatientWaitTime	24 Time Steps
NumOfNurseAgents	15
MinNurseTalkTime	1 Time Steps
MaxNurseTalkTime	10 Time Steps
NumOfPatientAgent	60
ProbOfNurseTalking	0.3
RunWithoutNurseTalking	False
StoppingTime	2304
SeverityLevel	30%
MinTimeMed	2 Time Steps
MinTimeDoc	5 Time Steps
MinTimeServePatient	10 Time Steps
Pre-empt	False
PerRandEventRate	0.1

Table 5.1: Values of parameters used in the model

this time interval. The number of nurse agents in the hospital is set to 15, their minimum talking time with other nurse agents is 1 time step which is 12.5 seconds and maximum talking time is 5 time steps which is one minute. The nurse agents consult with each other for this time interval when they travel. The probability of nurse agents talking to other nurse agents is 0.3 which means 30% of times nurse agents start their talk with other nurse agent. Numbers of patient agents in a hospital are set to 60. Checkbox run without nurse talking is set to false which means nurse agents are talking and consulting with each other. Stopping time of the simulations is 2304 tick counts which are 8 hours. The severity level of patient agents in the hospital is set to 30% which means that 30% of the patient agents in a hospital are critical such that their severity level is greater than 0.7. Once the nurses reach the patient agent, their minimum time for serving the patient agent is 10 time steps which is 2 minutes. After the nurse agent is done serving the patient agent minimum time for medication is set to 2 time steps which is 30 seconds and after she is done medicating, she spend at least 5 time steps for documentation which is one minute. The percentage of unpredictable events happening in the hospital is set to 10%, which means 10% of times adverse event happens in the hospital which prolong the stay of the nurse agents for particular task.

5.1 Base Model Results

5.1.1 Patient Service and Nurse Task Activity

The base hospital model was run with the base model settings to analyze the behavior of agents in the hospital settings. Fig 5.1 illustrates the bar graph of patient service status of all the patient agents in the hospital, such as number of times they were served on time and served late.

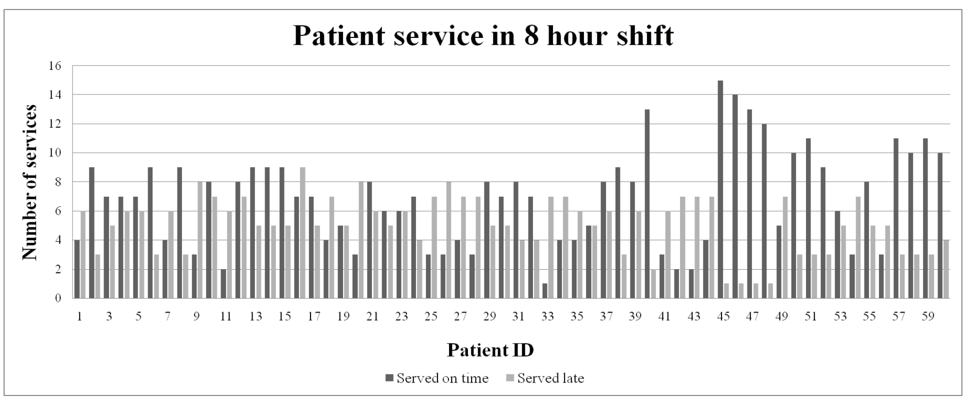

Figure 5.1: Patient service in a 8 hour shift

In Fig 5.1, x-axis shows the different patient agents with different Ids and y-axis shows the number of times a patient agent received service in an 8 hour shift. For example, the patient agent with Id 1 was served 8 times in which he was served twice on time and 6 times late. Fig 5.2 illustrates the bar graph of number of tasks done by the nurse agents in an 8 hour shift, in which x-axis shows the number of nurse agents and their Ids and y-axis shows the number of tasks done. For example, nurse agent with Id 1 has done 49 tasks in a complete shift, which includes serving the patient, medicating the patient and writing report for the patient.

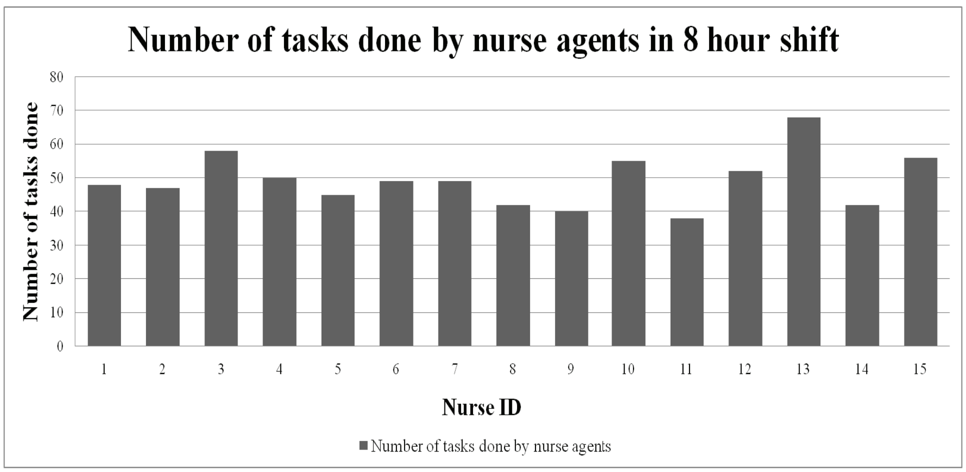

Figure 5.2: Number of tasks done by nurses in a 8 hour shift

Figure 5.3 illustrates the pie graph of nurse agents average task activity in an 8 hour shift. Nurse agents spend their day by doing different tasks in a hospital. In this simulation, only important tasks were included to reduce the complexity of the simulation, which includes travelling, consulting, serving patient agents, medicating patient agents, documenting and rest of the time nurse agents remain idle. With the base model settings, when the simulation is run for 8 hour shift, the following results were obtained in which nurses spent 29% of their time on travelling and consulting with other agents and take breaks, 31% of time they spent on serving the patient agents in the hospital, 14% of time nurse agents spent doing the medication of patient agents after serving them, 25% of time nurse agents spent writing the reports after the medication of patient agents and 1% of total time they spent doing nothing. All the results are average percentage of total time spent for each task by all the nurse agents in a hospital, such that the average of the total time spent by all the nurse agents on a particular task in a shift.

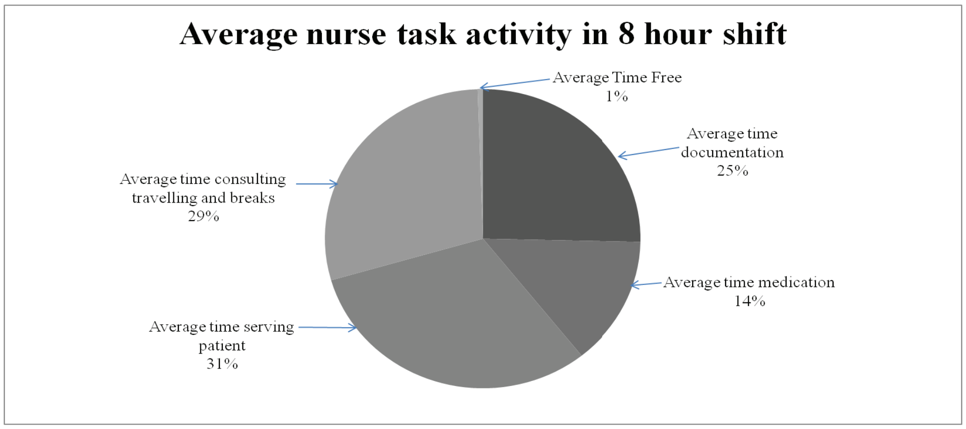

Figure 5.3: Average task activity and average task done in 8 hour shift

5.2 Effect of Different Parameters on Simulation

While maintaining consistency with the base model parameters, only one parameter is varied at a time to test the effectiveness and to observe the behavior of the parameter, on the average task success rate. The percentage of average success rate is the percentage of total number of successfully served patient agents in a hospital in every 10 minutes. Each experiment was run 70 times and their average was taken for the smoothness of the variability of results and their standard deviation is plotted with the mean to show how far the data is distributed from the mean, which represents the standard error at each time interval. All the results are compared with the base model settings to observe the difference.

5.2.1 Social Path versus Shortest Path

In hospitals, nurses' ultimate goal is to keep the patient service success rate higher and to give quality service. In order to serve the patients, nurses schedule the visit according to the patient's need and to keep the patient service success rate high, nurses try to serve the patient as quickly as they can. In this model, the nurse agents visit the patient agents at their scheduled time and they always follow the shortest path to reach the patient agents. The nurse agents follow the shortest path because they possess the knowledge of the hospital environment. On the contrary, the path nurse agents follow is not always a shortest path because of the distractions and intrusions from the other agents due to which the nurse agent's path gets deviated and shortest path becomes social path. As the nurse agent's path gets deviated from the actual path, it also affects the patient service.

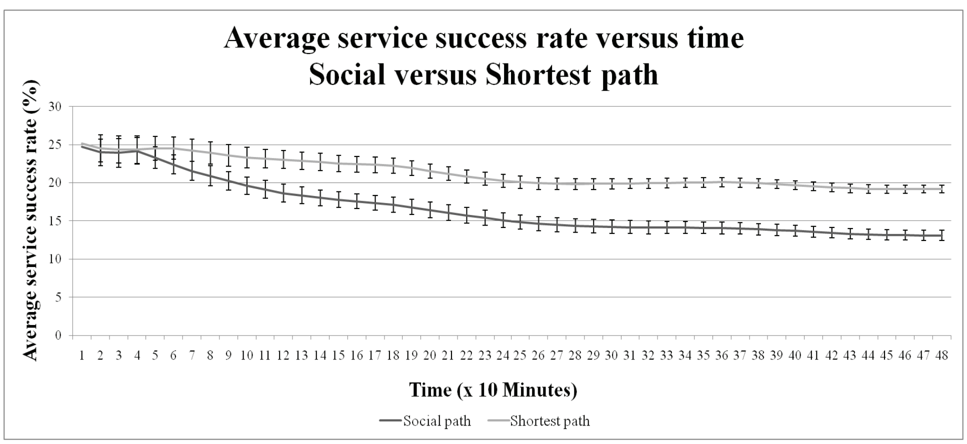

Figure 5.4: Percentage of average success rate of patient service versus time, social path versus shortest path

Figure 5.4 indicates that average success rate of patient service is higher when the nurses follow the shortest path as compared to when nurses follow the socially influenced path by the social distraction from the other agents. The data around the average is not very distributed which indicates that both the averages are statistically diverse.

5.2.2 Day versus Night Shift

In the real world, there are a number of environmental differences in a hospital which effects the nurse's ability to respond to the patient (i.e. service success rate). The main factor which influences the patient's service is the number of patients assigned to each nurse.

In hospitals, there are relatively few nurses at night than compared to a day shift. Different studies on the nurse to patient ratio suggest that there should be different ratio of nurse-to-patient at different sections of hospital, such that 1:1 in the operating room and 1:2 in the intensive care, critical care, and neonatal intensive care units, as well as in post-anesthesia recovery and labor and delivery. The ratio is 1:4 in ante-partum (before delivery), post-partum (after delivery), pediatric care, and in the emergency room and other specialty care units. In general medical-surgical units (regular hospital units), the ratio is 1:5 [3]. According to the data given by the hospital staff and management in an interview, the ratio of nurse agents to patients at night shift is 1:6, and 1:4 on a day shift. More patients are assigned to each nurse during the night shift based on the assumption that patients will sleep and will require less care or need for assistance. In this experiment, the *NumOfNurseAgent* is set to 10 for the night shift and 15 for the day shift, the remaining parameters are held the same as in the base model.

In figure 5.5, the day shift has more nurses than the night shift which significantly affects the percentage of average success rate of patient service. The percentage of average success rate of patient service is higher when there is more number of nurse agents are assigned to the patient agents as compared to when the patient-to-nurse ratio is lower such that, when fewer number of patient agents are assigned to each nurse agent. The average success rate peek is higher when nurse agents start their shift but as more number of requests comes to nurse agent, the peeks slowly goes down. Horizontal arrows around the average success rate reflect that how far the data is distributed from the average at any time step.

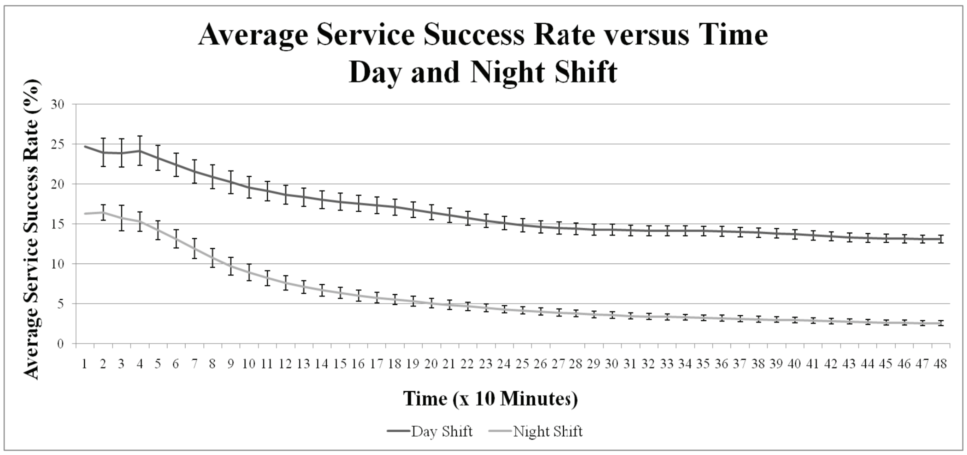

Figure 5.5: Percentage of average success rate of patient service versus time, day shift and night shift

5.2.3 Severity Level

Severity level is the percentage of the patient agents whose physical condition is considered severe, thereby requiring greater urgency for the nurse agent in ensuring the patient's care and safety. If the patient agent severity is greater than 0.7, which implies that the patient agent needs service and the nurse agent should visit patient agent more frequently. The effect of percentage of different levels of severe patient agents in the hospital is observed on the average success rate of patient service. The average patient service success rate has been observed with different severity levels set at 20%, 50%, and 80% with the rest of the parameters set to the same value as in the base model. In figures 5.6, the percentage of the average success rate of patient service delivery by nurse agents with different severity levels has been shown.

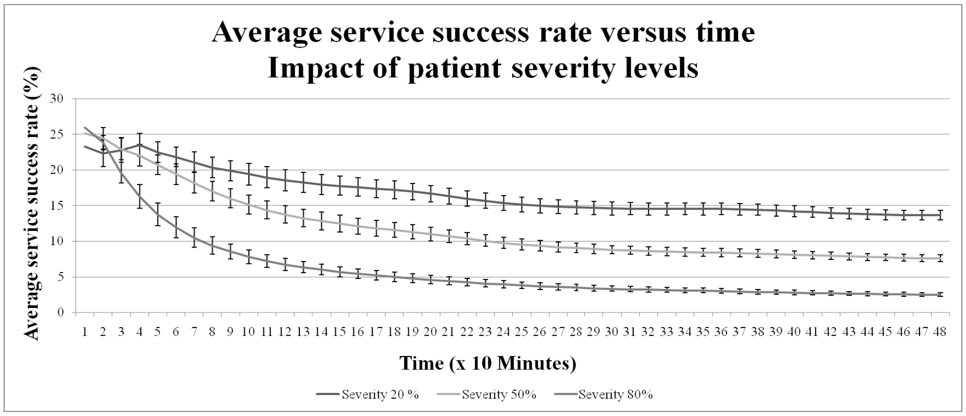

Figure 5.6: Percentage of average success rate of patient service versus time with severity level 20%, 50%, 80% respectively

The figure 5.6, indicates that as the percentage of severe patient agents increases in the hospital, the average success rate of patient service decreases. If the number of patient agents with severity level greater than 0.7 will be present in a hospital then the patient agents would need service more frequently and the nurse agents have to visit them often. This will lead nurse agents to miss the next waiting patient agents because of the fewer time stamps between the scheduled visit times of the patient agents.

5.2.4 Socialization Rate

Socialization rate is the percentage level of the interaction or interruption that happen during the shift from the other nurses which leads the nurse to deviate from their path plan to respond to a patient. Three different experiments were completed, each with a different level of social interactions, 30%, 50%, 80% with the remaining parameters held at the same as the base model. The impact of the nurse agent's social interactions on patient service success rate is observed. Figure 5.7 indicates that as the social interaction levels goes up, the service success rate falls. As the socialization rate increases, the nurse agents are deviated from their path more often which causes them to arrive late for the patient service and results in patient service success rate fall.

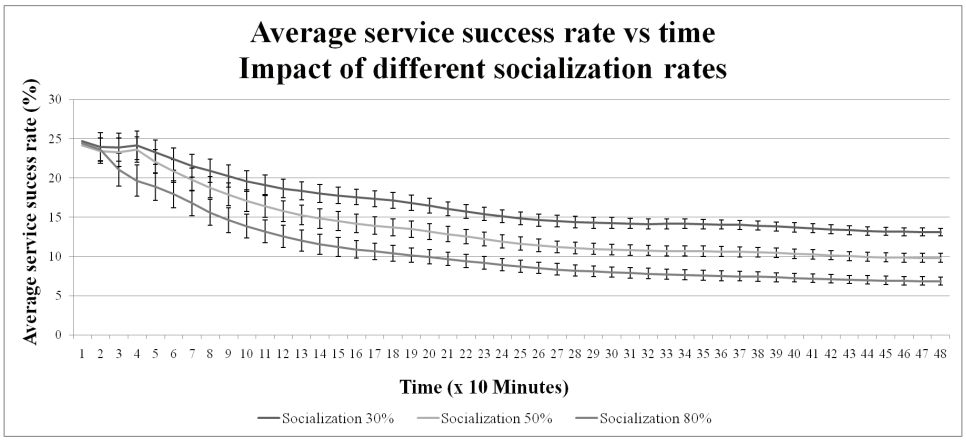

Figure 5.7: Percentage of average success rate of patient service versus time with different socialization rate 30%, 50%, 80% respectively

5.2.5 Short (8 hrs.) versus Long (12 hrs.) Shift

In hospitals, nurses usually work in an 8 hours shift in which they take breaks after every two hours for 10 minutes. In the long day shift (12 hours) which is usually the case, the nurse tires during the last four hours of the shift and then the fatigue factor [18] influences their performance. In this model, it is assumed that after eight hours of work the nurse agents start taking breaks more frequently and breaks last for longer periods of time (for example, 15 minutes break after every one and a half hour). We have done experiments with a short shift and a long shift setup with the rest of the parameters held the same as in the base model which shows the percentage of average success rate. In figure 5.8, the average patient service success rate is shown for long and short shifts which reflects that after 8 hours, the patient service success rate continues to fall at higher rate. This is because nurses starts going to the breaks for the longer period of time. As nurse agents remain on breaks for longer time they tend to serve fewer patient agents on time which decreases the overall service success rate of patient agents in a hospital.

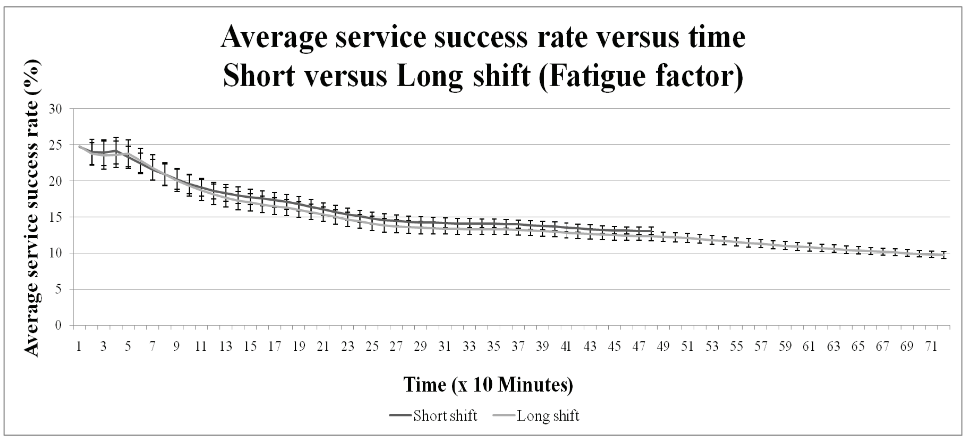

Figure 5.8: Percentage of average success rate of patient service versus time; short shift and long shift

5.2.6 With and Without Pre-empting (Task scheduling)

In a dynamic time constrained environment, it is really important for an agent to keep their performance high and also to keep the success rate of successful execution of tasks high. The pre-empting of tasks helps agents to do so. Figure 5.9 illustrates the percentage average success rate of patient service when nurse agents do not pre-empt their tasks and when they do pre-empt their tasks with the base model settings. Referring to the figure 5.9 one can easily observe the significant difference between the performance of the nurse agents when they pre-empt the tasks rather than doing the tasks in the queue fashion. The x-axis refers to the time which is multiple of 10x minutes and y-axis refers to percentage of average success rate. The percentage of average success rate is percentage of total number of successfully served patient agents in a hospital in every 10 minutes. When nurse agents pre-empt the tasks, they might miss a few tasks intentionally and do those tasks later, but nurse agents finish more tasks successfully on time and they can serve the missed tasks later. In some cases missed patient agents have long waiting time and they keep waiting for the nurse agent to come, due to which average success rate of task doing of nurse agents can go higher.

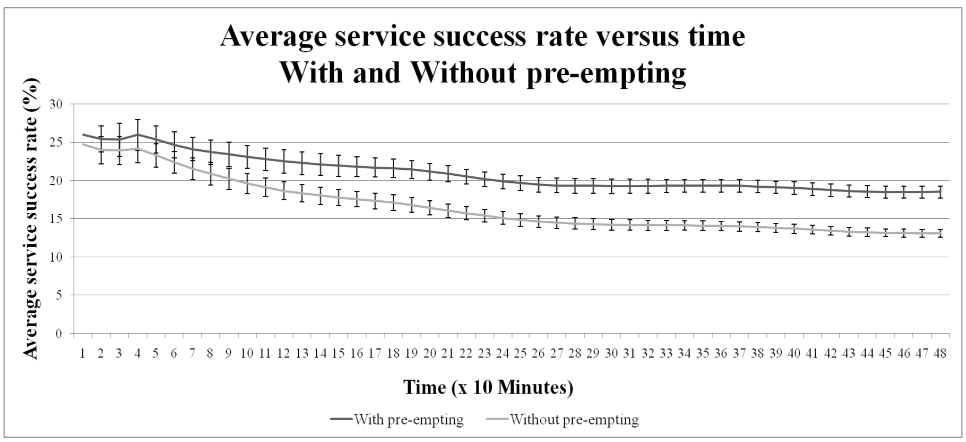

Figure 5.9: Average success rate versus time with pre-empting and without pre-empting

5.2.7 Unpredictable Event Rate

There are various different unpredictable events happen in a hospital which prolong the service time, medication time and documentation time which lead the nurse agents to spend more time on the task. As the nurse agents spend more time with the patient agents they gets less time to reach the next serving patient agent and she arrives late. This ultimately affects the service success rate in a hospital. In figure 5.10, the average patient service success rate is shown for different random event rates such that 20% and 50%. As the random event rates increases, the average service success rate drops.

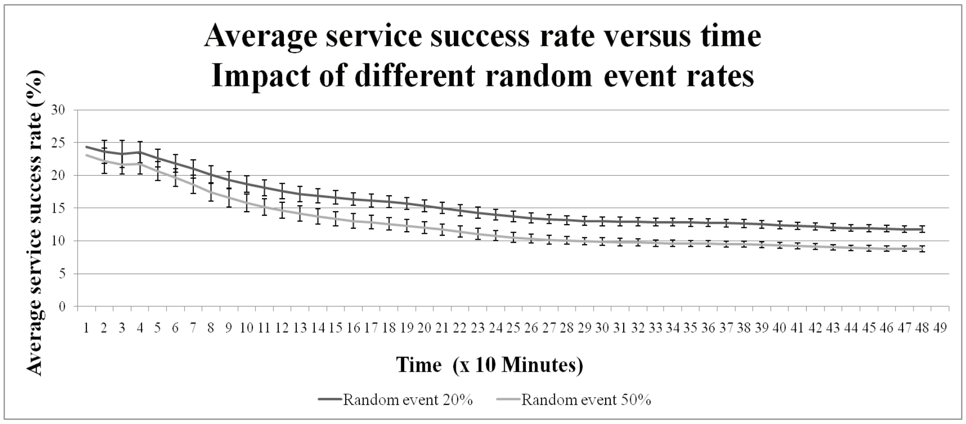

Figure 5.10: Percentage of average service success rate versus time with different random event rates 20% and 50% respectively

Chapter 6

Discussion

By analyzing the test results that were presented in previous chapter, we can clearly observe how different approaches of path finding, socialization, task scheduling, domain knowledge and other factors such as severity levels, fatigue factors and shift hours affect the average patient success rate in a hospital. All the experiments have been done with the base model settings and only one parameter is changed in each experiment to observe its behavior. The average of the results has been taken for consistency. As seen from the graphs, it is observed that data is more dispersed in the beginning of the simulation which concludes the random behavior of the simulation which is similar to the working of human social systems. This reflects that as the nurse agents start their shifts, the variability in their behavior is slightly higher and similar with the different variables. This reveals that different variables affect the system as time progresses; than at the beginning of the simulation.

In figure 5.5, results indicate that night shift has the maximum effect on the nurse agent's performance and average service success rate. There are fewer number of nurse agents work in a hospital during the night shift than during the day shift which increases the nurse-to-patient ratio and nurse agents have to take care of more patient agents. Secondly, in figure

5.6, patient severity rate has the maximum effect on the nurses performance because as the number of patient agents increases with the higher severity in a hospital, nurse agents have to visit them more often and they have to stay with them for longer interval of time which leads them to miss the following patient agents and average service success rate drops. At last, in figure 5.7, socialization rate has the maximum effect on the nurse agent's performance because as the nurse agents get engaged in interactions along their way they tend to miss more patient agents which results in fall of average service success rate.

In figure 5.8, average service success rate of the patient agents is compared between short and long shift which reflects that service success rate is similar until 8 hours but after 8 hours the fatigue factor comes into play due to which nurse agents start taking more frequent breaks and for longer interval of time which results in fall their performance. The nurse agent's performance affects the average service success rate, which continues to drop at higher rate after 8 hours. These findings show that fatigue factor has a significant effect on the nurse agent's performance and also on the service rate.

In figure 5.9, average success rate of the patient service is higher when the nurse agents served the patient agents with task scheduling such as pre-empt the tasks according to the situation and the condition of the patient agents in the hospital. Nurse agents attempt to serve the patient agents first who have critical condition such that their severity level is high. Nurse agents also try to reschedule the visit for the patient agents she pre-empts due to the social factors.

In figure 5.4, average success rate of patient service is compared between the shortest path and the social path which shows that nurse agents deliver better service and average success rate of patient agent is also higher when the nurse agents take shortest path. In figure 5.7, effects of social interactions were shown which reflects that as the social interaction increases the average success rate of patient service decreases. As the nurse agents get involved in social interactions, they spent more time along their way due to which nurse agents arrive late to the patient agents and service success rate drops.

6.1 Verification and Validation of Results

Different studies have been done on the time spent by the nurses in a hospital for the efficient use of the nurse's time and energy and the safe delivery of the patient care. A hospital time and motion study has been done in US where 36 hospitals and 767 nurses were participated. In this study different techniques and instruments, such as PDA, RFID and Armbands were used to calculate the time spent on nurse practices and nonclinical practices. Nurse practices include medication, documentation, patient care coordination, patient assessment and patient care. and non clinical activities include personal time, administration and patient/family care. According to the study the nurses spend large amount of time on medication, documentation and patient care [24]. For the ease of readers and to verify with our model we have combined patient care coordination, patient assessment and patient care whole as the patient care which is 33%, nurses spent 24% of time on documentation, 12% of time on medication and 31% of time on non-clinical activities.

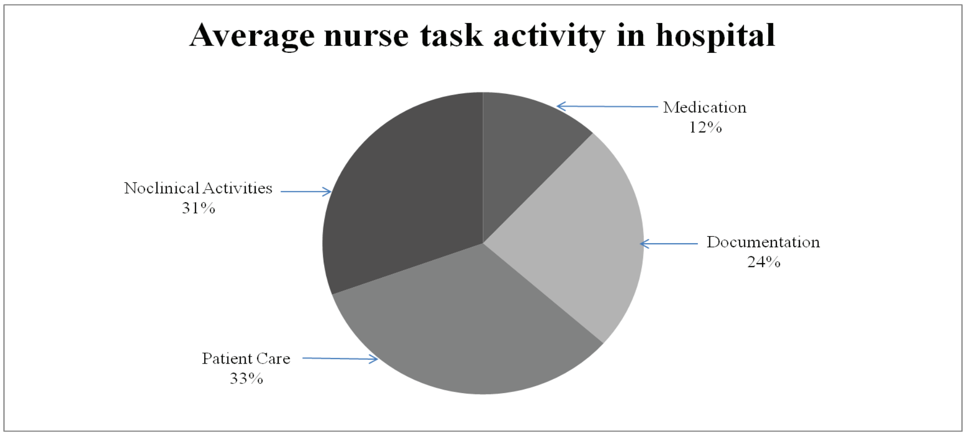

Figure 6.1: Nurse average task activity

The following are the results were obtained from this simulation which indicates the average of the nurse agent's task activity and task done in 8 hour shift. The percentage of average task is measured as the average of total time spent by all nurse agents on each task from 70 runs. The results show that nurse agents spend 31% of time on patient agent servicing, 14% of time on medicating, 25% of time on documenting, 29% of time on consulting, travelling and breaks and 1% of time nurse agents are idle.

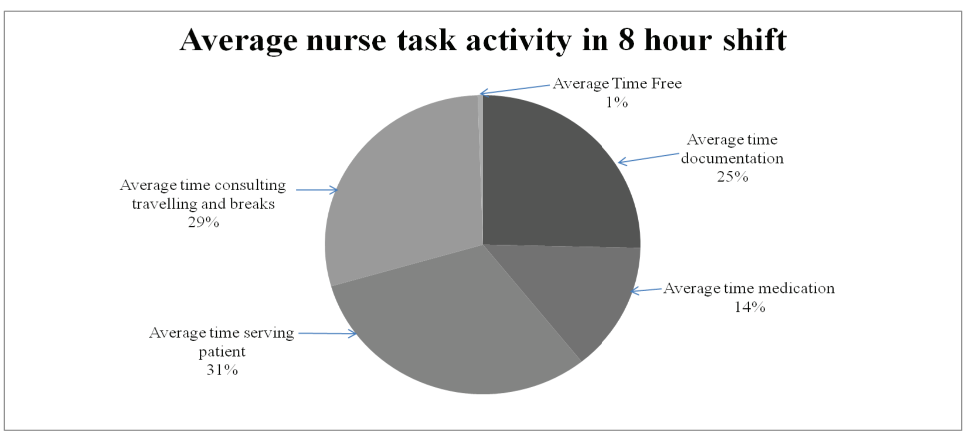

Figure 6.2: Average task activity and average task done in 8 hour shift

The comparison of results reflects that the simulation produces nearly the same results as in the studies, which conclude that, this simulation can be further used as a tool for decision making policies and for the future validations.

Chapter 7

Conclusion and Future Works

7.1 Conclusion

In this research a partial multi-agent hospital simulation is designed for examining the influence of the environmental and individual's behavior on the performance of nurse agents on patient service time. In some aspects, in implementing the autonomous agents we have separated from the traditional optimal behavior towards a more realistic artificial response that better depicts human-like qualities. For instance, the path a nurse would traverse from one location to the next in the hospital would be influenced by the nurse's social interactions with other nurses along the way rather than the undisturbed shortest path. Task pre-empting and prioritizing patient service tasks to maintain a certain level of care in the dynamic environment are implemented. In a setup where patients may initiate unscheduled service requests, for instance by means of having their severity change over time, a nurse is presented with the task re-scheduling problem in order to cater to the changing needs of the patients while maintaining an acceptable level of care. In this model, the level of care or quality of service is measured in terms of the rate at which the patient is serviced on time.

The simulation model is based on real world rural hospital settings that provide care to an elderly population of patients. Different studies and input from actual nurses and domain experts contributed in this design. Despite the simplified hospital model, some important conclusions can be drawn first about our understanding of the complexities of a an artificial hospital model and second about the contributing role of multi-agent systems to better model human-like qualities. In the latter aspect we address the role of a path traversal algorithm in a setting where a shortest path is desirable but not optimal. Here an optimal path is defined as a balance between minimizing traversal time cost and maximizing social interaction gains. Increased social interaction can lead to an overall increase in the total time towards the goal, in this case the patient to be serviced. Yet reducing this social interaction result in inadequate portrayal of the nurse's regular duties that are being abstracted such as consultation with other nurses, families, doctors, pharmacists, etc. The results of this setup lead to conclude that a realistic path must include both social interaction and time minimization for traversal. Furthermore, by measuring the number of patients the nurses miss servicing on time represent a measure of their performance. It follows that if the goal is to achieve a zero number of missed service times, in other words in order for all the nurses to service all the patients on time while maintaining adequate social interactions levels a decision support tool is presented to the hospital manager. If all else constant, one can ask and find out from the simulation what minimum number of nurses is required to achieve this goal.

The hospital simulation has been verified with the different attributes of nurse agents that include social interaction time, task scheduling in terms of prioritizing and pre-empting, and path taken towards a determined destination. The results confirm that nurse performance (in terms of number of patients serviced on time) decrease with a higher rate of social interactions. With a reduced number of active nurses working, factoring in breaks taken, the simulation confirms expectations with an increase in the number of missed service times for patients. Fatigue is also modeled by introducing longer shifts. Fatigue triggers longer breaks and again affect the overall service level by observing an increase in the number of missed patients.

The tested behavior of the simulation is in line with the expected behavior of the hospital service model we intend to implement. Much works still needs to be done in order to improve the realism of the simulated environment. This venture requires deeper collaboration between the modelers and the domain experts. These social simulations can indeed be adopted as decision support tools in a partial extent for domain experts, limited by the underlying model assumptions and implemented details.

7.2 Future Works

In future work, this model will be extended to include more agents such as physicians, pharmacists, social workers, community care case managers, visitor agents, maintenance and administrative staff, and others. Additional features of the environment will also include seasonal changes in patient admissions, physical features of the environment like temperature, humidity, day and night, dark and brightness, friction of floor, wet floor and hindrances in hallways or in rooms. In this model, the patient agent cannot move but in the next model they will be moving to simulate a patient's normative activity in a hospital unit. The reputation of agents will also be included which will help agents to decide to interact with other agents.

Bibliography

[1] Reducing patient fall project. URL http://www.premierinc.com/safety/topics/falls/.

[2] Center for the study of complex systems. URL http://www.cscs.umich.edu/links/artsoc.html.

[3] Nurse-to-patient ratios: The science and the controversy. URL http://healthcarehacks.com/nurse-to-patient-ratios-the-science-and-the-controversy.

[4] R. Axelrod. Advancing the art of simulation in the social sciences. volume 3, page 2. ACM, 1997.

[5] M. A. Batalin, G. S. Sukhatme, and M. Hattig. Mobile robot navigation using a sensor network. In *International Conference on Robotics and Automation*, pages 636–642. IEEE, 2004.

[6] M. Batty and B. Jiang. Multi-agent simulation: Computational dynamics within gis. In *Innovation in GIS VII: Geocomputation*, pages 55–71, 2000.

[7] F. Bellifemine and G. Rimassa. Developing multi-agent systems with a fipa-compliant

agent framework. In *Software Practice Expertise*, volume 31(2), pages 103–128. ACM, 2001.

[8] J. V. D. Berg. Path planning in dynamic environments, 2007.

[9] J. M. Bradshaw. Software agents. MIT Press, 1997.

[10] S. C. Brailsford. Advances and challenges in healthcare simulation modeling. In *Winter Simulation Conference*, 2007.

[11] M. Cesati and L. Trevisan. Parameterized complexity analysis in robot motion planning. In *25th IEEE International Conference on Systems, man and Cybernetics*, volume 1, pages 880–885. IEEE, 1995.

[12] M. C. Chiu and M. J. Wang. Professional footwear evaluation for clinical nurses. In *Appl Ergon*, volume 38(2), pages 133–141. Elsevier, 2007.

[13] K. A. Delic, L. Douillet, and U. Dayal. Towards an architecture for real-time decision support systems: challenges and solutions. In *International Symposium on Database Engineering & Applications*, pages 303–311. ACM, 2001.

[14] A. Drogoul and J. Ferber. Multi-agent simulation as a tool for modeling societies: Application to social differentiation in ant colonies. In *Artificial Social Systems*, volume 830, pages 3–23. Springer, 1992.

[15] K. Ehrlich and I. Carboni. Inside social network analysis. In *IBM Waston Research Center*, 2005.

[16] J. M. Epstein and R. Axtell. Growing artificial societies. MIT Press, 1996.

[17] J. Ferber. Multi-agents systems, 1999.

[18] R. Flin, J. Winter, C. Sarac, and M. Raduma. Human factors in patient safety: Review of topics and tools, 2009. URL `https://www.who.int/patientsafety/research/methods_measures/human_factors/human_factors_review.pdf`.

[19] L. Garrido and K. Sycara. Multi-agent meeting scheduling: Preliminary experimental results. In *2nd International Conference on Multi-Agent Systems*, pages 95–102. AAAI, 1996.

[20] N. Gilbert. Agent-based social simulation: dealing with complexity. *The Complex Systems Network of Excellence*, 2004.

[21] R. Graham, H. McCabe, and S. Sheridan. Realistic agent movement in dynamic game environments. In *DiGRA 2005 Conference: Changing Views Worlds in Play*, 2005.

[22] M. S. Hassoua, A. E. Abdel-Hakim, and A. A. Farag. Robust robotic path planning using level sets. In *International Conference on Image Processing*, volume 3, pages 473–476. IEEE, 2005.

[23] M. S. Hassouna, A. Hakim, and A. A. Farag. Pde-based robust robotic navigation. In *In: IEEE International Conference on Image Processing ICIP*, volume 3, pages 473–476. IEEE, 2005.

[24] A. Hendrich, M. Chow, B. Skierczynski, and Z. Lu. A 36-hospital time and motion study: How do medical-surgical nurses spend their time? *Permanente Journal*, 12(3): 25–34, 2008.

[25] B. Hjsrland and H. Albrechtsen. Toward a new horizon in information science: Domain-analysis. In *The American Society For Information Science*, volume 46(6), pages 400–425. Wiley, 1995.

[26] S. H. Jacobson, S. N. Hall, and J. R. Swisher. Discrete event simulation of health care systems. In *Patient Flow: Reducing Delay in Healthcare Delivery*, volume 91, pages 212–241. Springer, 2006.

[27] M. Jamali and H. Abolhassani. Different aspects of social network analysis. In *International Conference on Web Intelligence*, pages 66–72. IEEE, 2006.

[28] D. B. Johnson. A note on dijkstra's shortest path algorithm. pages 385–388. ACM, 1973.

[29] S. Jong, K. Tuyls, , and I. S. Kuyper. Robust and scalable coordination of potential-field driven agents. In *IAWTIC/CIMCA*. IEEE, 2006.

[30] A. Juan. On the design and construction of agent-mediated electronic institutions, phd thesis. Technical report, Monografies de lInstitut dInvestigaci en Intelligncia Artificial, 2003.

[31] J. Kalagnanam and D. C. Parkes. Auctions, bidding and exchange design. In *Operations Research and Management Science*, volume 5, 2004.

[32] S. Kirn, C. Anhalt, H. Krcmar, and A. Schweiger. Agent hospital: Health care applications of intelligent agents. In *INFORMATIK*, volume 1, pages 64–82. Springer, 2003.

[33] F. A. Kolushev and A. A. Bogdanov. Multi-agent optimal path planning for mobile robots in environment with obstacles. In *Third International Andrei Ershov Memorial Conference on Perspectives of System Informatics*, pages 503–510. Springer, 1999.

[34] F. A. Kolushev and A. A. Bogdanov. Multi-agent optimal path planning for mobile

robots in environment with obstacles. In *PSI*, volume 1755, pages 503–510. Springer, 2000.

[35] S. M. LaValle. Planning algorithms. cambridge university press, 2006. URL `http://msl.cs.uiuc.edu/planning/`.

[36] D. V. Lebedev, J. J. Steil, and H. J. Ritter. The dynamic wave expansion neural network model for robot motion planning in time varying environments. In *Neural Networks*, volume 18(3), pages 267–285. Elsevier, 2005.

[37] R. Lelouche. Team of agents cooperating for intelligent tutoring. In *Distributed Artificial intelligence*, volume 1544. Springer-Verlag, 1998.

[38] J. Liu, H. Jing, and Y. Tang. Multi-agent oriented constraint satisfaction. In *Artificial Intelligence*, volume 136, pages 101–144. Elsevier, 2001.

[39] C. M. Macal and M. J. North. Agent based modeling and simulation: Desktop abms. In *Winter Simulation Conference*, 2005.

[40] X. Mao, A. Mors, N. Roos, and C. Witteveen. Agent-based scheduling for aircraft deicing. In *18th Belgium - Netherlands Conference on Artificial Intelligence*, pages 229–236. Citeseer, 2006.

[41] P. Mi and W. Scacchi. A knowledge-based environment for modeling and simulating software engineering processes. In *Knowledge and Data Engineering*, volume 2(3), pages 283–289. ACM, 1990.

[42] A. Moreno and D. Isern. A first step towards providing health-care agent-based services to mobile users. In *Autonomous Agents and Multiagent Systems*, pages 589–590. ACM, 2002.

[43] N. T. Nguyen and R. Katarzyniak. Actions and social interactions in multi-agent systems. In *Knowl Inf Syst*, volume 18(2), pages 133–136. Springer, 2009.

[44] H. V. D. Parunak, S. Brueckner, J. Sauter, and R. Matthews. Distinguishing environmental and agent dynamics: A case study in abstraction and alternate modeling technologies. In *Engineering Societies in the Agents World (ESAW00)*, volume 1972, pages 19–33. ACM, 2000.

[45] B. Pavard and J. Dugdale. An introduction to complexity in social science. URL `http://www.irit.fr/COSI/training/complexity-tutorial/complexity-tutorial.htm`.

[46] S. Poslad. Specifying protocols for multi-agent systems interaction. In *Transport Autonomous Adaptive System*, volume 2(4), pages 15–24. ACM, 2007.

[47] D. J. Power. Decision support systems : concepts and resources for managers. Springer, 2002.

[48] C. Preist and Maarten. Adaptive agents in a persistent shout double auction. In *Information and computation economies*, pages 11–18. ACM, 1998.

[49] R. W. Proctor and T. V. Zandt. Human factors: In simple and complex systems. 1994.

[50] S. J. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Pearson Education, 2003.

[51] J. Sabeter and C. Sierra. Reputation and social network analysis in multi-agent systems. In *1st International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 475–482. ACM, 2002.

[52] R. G. Sargent. Verification and validation of simulation models. In *37th Conference on Winter Simulation*, pages 130–143. ACM, 2005.

[53] R. Schwendimann. Patient falls: A key issue in patient safety in hospitals base. Technical report, Universitt Basel, 2006.

[54] K. Smith, R. Paranjape, and L. Benedicenti. Agent behavior and agent models in unregulated markets. In *Applied Computing Review*, volume 9(3). ACM, 2001.

[55] R. H. Sprague and Watson. Decision support systems, 1990.

[56] A. Sud, E. Andersen, S. Curtis, M. Lin, and M. Manocha. Real-time path planning for virtual agents in dynamic environments. In *In: International Conference on Computer Graphics and Interactive Techniques*, 2008.

[57] A. Sud, E. Andersen, S. Curtis, M. C. Lin, and D. Manocha. Real-time path planning in dynamic virtual environments using multiagent navigation graphs. In *Transactions on Visualization and Computer Graphic*, volume 14(3), pages 526–538. IEEE, 2008.

[58] K. Sycara and D. Zeng. Coordination of multiple intelligent software agents. In *Intelligent and Cooperative Information Systems*, volume 5(3), pages 181–211. Citeseer, 1996.

[59] K. P. Sycara. Multiagent systems. In *American Association for Artificial Intelligence*, pages 79–92. ACM, 1998.

[60] R. Urniezius and S. Bartkevicius. Robot navigation planning problems in dynamic environments. 2008.

[61] M. Uschold. Knowledge level modelling: Concepts and terminology. In *ACM*, volume 13(1), pages 5–29. Knowledge Engineering Review, 1998.

[62] D. Weyns, N. Boucke, T. Holvoet, , and W. Schols. Gradient field-based task assignment in an agv transportation system. In *EUMAS*, pages 447–458. ACM, 2005.

[63] M. J. Wooldridge and N. R. Jennings. Software engineering with agents: pitfalls and pratfalls. In *Internet Computing*, volume 3(3), pages 20–27. IEEE, 1999.

[64] R. Xie, D. Rus, and C. Stein. Scheduling multi-task multi-agent systems. In *Fifth international Conference on Autonomous Agents*, pages 159–160, 2001.

[65] B. Yu and M. P. Singh. Searching social networks. In *Second international Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*, pages 65–72. ACM, 2003.

Vita Auctoris

Ashish Nakhwal was born in 1986, in Hoshiarpur, Punjab, India. He received his bachelor's degree from RIMT Punjab in computer science in 2007. His research includes MAS and social complex models.