2012-03-02

Using Agent-Based Models to Understand Multi-Operator Supervisory Control

Yisong Guo
Brigham Young University - Provo

Using Agent-Based Models to Understand Multi-Operator Supervisory Control

Yisong Guo

A thesis submitted to the faculty of
Brigham Young University
in partial fulfillment of the requirements for the degree of

Master of Science

Michael A. Goodrich, Chair
Christophe Giraud-Carrier
Teppo Felin

Department of Computer Science

Brigham Young University

April 2012

ABSTRACT

Using Agent-Based Models to Understand Multi-Operator Supervisory Control

Yisong Guo
Department of Computer Science, BYU
Master of Science

As technology advances, many practical applications require human-controlled robots. For such applications, it is useful to determine the optimal number of robots an operator should control to maximize human efficiency given different situations. One way to achieve this is through computer simulations of team performance. In order to factor in various parameters that may affect team performance, an *agent-based model* will be used. Agent-based modeling is a computational method that enables a researcher to create, analyze, and experiment with models composed of agents that interact within an environment [16]. We construct an agent-based model of humans interacting with robots, and explore how team performance relates to different agent parameters and team organizational structures [22]. Prior work describes interaction between a single operator and multiple robots, while this work includes multi-operator performance and coordination. Model parameters include neglect time, interaction time, operator slack time, level of robot autonomy, etc. Understanding the parameters that influence team performance will be a step towards finding ways to maximize performance in real life human-robot systems.

Keywords: Computer, agent-based model, simulation

ACKNOWLEDGMENTS

I am grateful for the wonderful and exceptional guidance and mentoring from my advisor, Dr. Michael A. Goodrich, who provided me with great support, insights and encouragements. I appreciate the environment that exists in the BYU Computer Science Department and the HCMI lab that have enflamed my desire of learning. I would also like to thank my parents, who have been supportive in my quest for further education. Most importantly, I am forever indebted to BYU, which has been a great blessing to me.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

In recent years, there has been an enormous increase in the number of robots used to complete monotonous or dangerous tasks for humans [14]. Although robots used in such tasks exhibit sophisticated autonomy, robots in many applications require frequent attention from human operators to ensure that performance is satisfactory. This interaction between human operators and robots needs to be analyzed and studied in order to increase robot task efficiencies. In this thesis, we use a technique called agent-based simulation to study the interaction between human operators and robots.

1.1 Current Uses of Teleoperated Autonomous Robots

Robots have been designed and used to help, and sometimes replace, humans in many dangerous, tedious and monotonous tasks. These tasks, when performed by humans, often yield low performance or pose a high threat to their safety. With the help of robots, these tasks can be completed more quickly, safely, and efficiently. A few examples are listed here.

When a hiker is reported missing, a search party is organized and sent out to find the missing individual. This party can consist of volunteers, paid professionals and relatives of the missing hiker. The search party will then “comb” the ground where the missing hiker is likely to be, which can be very time consuming. As time goes by, the chances of finding the missing hiker drops exponentially, so time is a critical issue. However, wilderness search and rescue teams can use Unmanned Air Vehicles (UAVs) mounted with a camera that scans the ground and feeds live video to searchers who can use the video to scan an area for the missing hiker. This has the potential to not only save time and resources, but also increase the chance of finding the missing hiker quickly.

Another field where remote robots are greatly utilized is underwater exploration. Many underwater caves and caverns are either too deep or too small for a diver or manned submarine to maneuvre through, so the need for remote diving robots arises. They can go very deep without having problems with pressure, and their compactness makes them able to move around without much difficulty. Since they are unmanned, no human lives are in danger when they are sent to explore an unstable cave or trench miles below the surface.

In the military, remote robots have been mostly used for reconnaissance missions. UAVs, mine sweepers, explosive ordnance disposal robots, etc., have all greatly reduced the threat to human lives in the tasks that they do, by separating the human operators from the physical dangers, such as being shot down by enemy anti-air defense or a possilbe bomb explosion.

Robots have the potential to make our lives much easier, safer and more efficient, but we must understand how to design systems that allow humans and robots to act effectively together. In this thesis, we use the metaphor of a team to explain an ideal interaction between humans and robots.

1.2 Human Robot Teams

Since the beginning of time, humans have been social animals. We have always approached problems as cooperative teams when the problem is too great for a single person to manage. We formulate the problem, devise plans to solve it, then divide and share the work and responsibilities. Each person in the team will engage in the tasks allocated, and work is considered complete when every member in the team has completed his/her given tasks and responsibilities. Note here, that the end results are not the works of any single member in the team, it is the sum of the collective work.

If robot autonomy is sufficiently high and demands on human attention are sufficiently low, prior work indicates that it is possible for a single operator to manage multiple robots [8]. Finding ways for operators to control multiple robots effectively requires a deep understanding of the fundamental principles of multi-robot supervisory control. In this context supervisory control occurs when "one or more human operators are intermittently programming and continually receiving information from a robot that itself closes an autonomous control loop through artificial effectors to the controlled process or task environment" [28].

Given a team of robots, there is a trend in the literature to try to maximize so-called *fan-out*. Fan-out has been described as the maximum number of robots that can be managed by a single operator [26]. Although fan-out is a good metric, the real goal is not to maximize the number of robots, but to maximize team performance; in practice, factors such as stress, cognitive workload and many other confounding factors play an important role in team performance.

While tasks can sometimes be completed with only a single operator controlling one team of robots, often there is a need for multiple teams and operators. For example, in wilderness search and rescue, multiple UAVs and operators could collaborate to search for a missing person to increase the chance of quickly finding the person. Tasks such as this require a lot of synchronization of the operators, which implies that factors like communication and team organizational structure are critical.

The goal of this thesis is to find an optimal way to assign agents to one or more operators that would maximize the performance of the whole team. We approach this by modeling the multi-operator supervisory control problem with different parameters using agent-based simulation. Various key parameters that limit or enhance the performance of multi-operator teams include operator workload (utilization), communication overhead, mishandling, etc. We argue that organizations that properly respect these parameters are more likely to produce high-functioning teams than organizations that myopicly insist on maximizing fan-out.

1.3 Thesis Statement

Agent-based simulation can be used to discover parameters that have strong influences on the performance of multi-operator/multi-robot teams. Organizations that properly respect these parameters produce teams that perform better than teams that myopically focus on maximizing fan-out.

1.4 Thesis Organization

Chapter one is the introduction, it introduces the current uses of autonomous robots and human robot teams.

Chapter two describes the related literature, such as task performance, scheduling, workload, stress, and communication, etc.

Chapter three contain the methods: the assumptions made, how we are modeling performance, slack, communication, etc.

Chapter four has the results. Resuls obtained in the simulations are presented and explained here.

Chapter five is the conclusion and future work.

Chapter 2

Related Literature

In this chapter we review and discuss relevant research in areas that relate to human-robot team performance. This review will provide readers with sufficient knowledge to understand and participate in discussions of how we can exploit and improve the performance of human robot teams.

2.1 Operator Performance

Prior work on operator performance includes fundamental ideas in (a) *task scheduling*, that may have periods of "slack" time and models the operator as a "server" and interactions with robots as tasks [24] [35]; and (b) *cognitive workload and attention*, which analyzes cognitive factors that constrain the operator's judgement and ability to perform. This subsection will elaborate more on these two areas, starting with possible metrics for evaluating performance.

2.1.1 Task Performance

A good metric for evaluating performance would be how satisfactory the tasks are completed. This could include the amount of time taken for completion, amount of fuel used, amount of space covered, etc. It can vary depending on the tasks being evaluated, and some tasks may even require having multiple performance metrics. For example, in wilderness search and rescue, a reasonable performance metric would be how much time it takes to completely search through an area, since the likelihood of locating the missing person drops with time; while for a chemical sniffing task, recall and precision will be good and reasonable metrics.

2.1.2 Scheduling

Work has been done on task scheduling in order to improve operator performance by reducing operator response time due to task waiting [10]. Various techniques to make operators more effective at scheduling tasks include developing automation aids [2], using visual aids [11], etc. Automation aids can significantly decrease operator load by automating the regular routines such as UAV flight path patterns; while visual aids can represent different elements in different color and shapes, thus saving the operator from categorizing all the elements and facilitating greater awareness of the environment and situation. For example, in air combat, friendly aircrafts may be blue dots on the radar while hostile units may be represented by red triangles.

2.1.3 Cognitive Workload, Stress and Performance

Intuitively, there is a tight relationship between operator cognitive workload and factors that impact operator stress level and operator performance. This subsection defines some key components of workload and stress.

Operator workload has traditionally been defined with two components: physical workload and cognitive workload [28]. Physical workload, defined as the energy expended (e.g. in calories) by the operator while performing the task, is not very relevant to this thesis. However, cognitive workload, defined as cognitive actions performed, is relevant. Various factors contribute to mental workload: task complexity, number of robots assigned to the operator, operating environment etc. [12]. Increase in task complexity, increase in number of robots, change in the operating environment, or even an operator's personal issues may all increase the operator's cognitive workload. In this thesis, cognitive workload is directly related to the number of agents that an operator must manage, a term we call *operator utilization* [28].

Both physical and cognitive workload in the real world contribute to operator stress. Stress increases directly proportional to workload [12], but as workload gets higher, stress

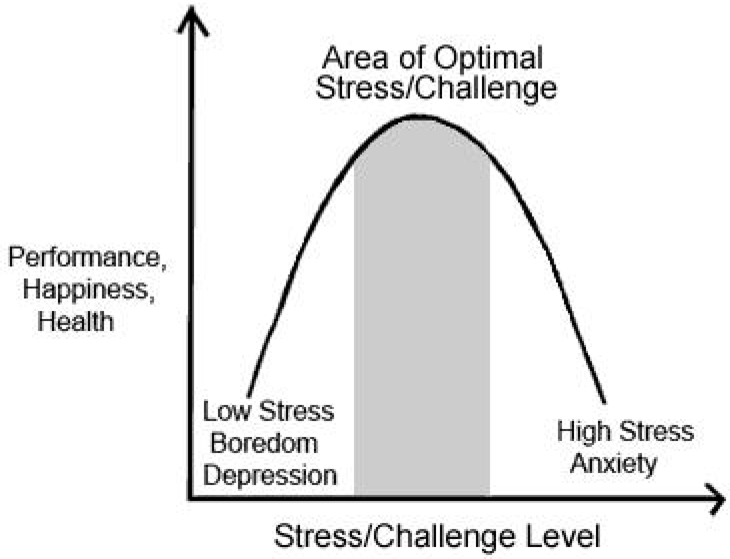


Figure 2.1: Performance versus Stress

increases faster. Studies have shown that humans perform poorly given little or too much workload/stress [19]. The performance versus stress curve looks like the inverted-U shape in Figure 2.1.

2.1.4 Awareness and Performance

In addition to mental workload, situation awareness is another factor that contributes to operator performance [34]. Loosely speaking, situation awareness means how much the operator knows about the current state of the world. More formally, situation awareness has been defined as follows: “The perception of elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future” [12].

Situation awarenss consists of various elements: information about the status of the robot, information about the robot’s environment, and information about the tasks in the environment [34]. A great hindrance to situation awareness is the robot’s field of view. Humans have a great peripheral vision while a robot’s field of vision is rather small and is limited by its camera. Woods et al. made a comparison between operating a remote robot and driving while looking through a 'soda straw' because of the small field of view [32]. Human eyes have about 180 degrees peripheral vision, however peripheral vision is not used for detail finding, but to detect motion and avoid obstacles. Using a wide angle camera may increase the field of view on the robot, but to represent the wider field of view on a flat screen will cause loss of details, images will be distorted on the outside, which are often critical in the missions that use robots, such as search and rescue. However some robots have IR sensors on the side to compensate for the peripheral vision of humans to detect and avoid obstacles, but the cognitive workload of the operator is increased by having to switch back and forth between the camera view and peripheral view.

Woods et al. have also mentioned the issue of remote perception, which they defined as the ability to integrage partial views from a set of robotic resources into a coherent model of the environment for remote human observers [32].

Casper and Murphy [4] also mentioned this difficulty of remote vision in robot-assisted urban search and rescue response at the World Trade Center. Different kinds of robots mounted with cameras were used, with their field of view ranging from 52 degrees to 118 degrees, as opposed to a single human eye's field of view - 95 degrees out, 75 degrees down, 60 degrees in and 60 degrees up. Operators had to use a single color camera to determine the location and status of the robot, search for victims and inspect the environment. This lack of and difficulty in obtaining situation awareness had caused the operators increased cognitive stress [4].

On the contrary, if the operator is more aware of the environmental situation, the cognitive workload on the operator will be reduced, which in turn will decrease stress. Various methods to increase awareness of operators include using different color representations [11], designing ecological interfaces [23], etc. However, an operator's situation awareness is restricted by limited attention and working memory capacity, and is largely affected by the operator's goal and expectations which will influence how attention is directed, how information is perceived and how it is interpreted [12].

2.1.5 Specialization

The idea of "specialization" is analogous to that of assembly lines, where each worker/operator is in charge of one type of task/robot. Jonathan Whetten [30] did experiments on how well operators perform when they are assigned a particular type of robot (specialized operators), against how operators perform when they are assigned heterogeneous robots (unspecialized operators). The experiment consisted of a simulation environment where human operators had the tasks of finding mines and bombs.

Results show that specialization of operators controlling teams of heterogeneous robots reduces workload for the operators, which frees up "spare capacity"; that spare capacity could be utilized to improve task performance within a proper environment.

2.2 Robot Autonomy and Fan-out

Although robots have autonomous abilities, human operators are an irreplacable factor in robot automation for the technologies today. However, more autonomous robots enable a human operator to control multiple robots at once, and the number of robots a human operator can control is modeled using the so-called '"fan-out". Fan-out is a representive or measure of how many simple homogeneous robots a human can manage and is calculated based on two parameters: how much time a robot needs from a human operator (interaction time), and how long it can operate on its own without human input (neglect time) [26]. The less time a robot requires from humans, the more homogeneous robots can be controlled. Fan-out is given by the equation [25]:

$$FO = \frac{NT}{IT} + 1 \tag{2.1}$$

.

2.2.1 Interaction Time

Interaction Time (IT) is defined as the period of interaction by an operator with the robot [26]; it directly depends on how much attention a robot demands from the operator. IT is one of the two parameters that influence fan-out, and factors such as time to gain situation awareness, time taken to switch between tasks, time used for planning, etc. all contribute to interaction time [18].

In order to increase fan-out, one of the ways is to decrease interaction time. IT can be divided into two main categories: time taken to actually interact with the robot, giving commands and receiving information through interfaces (robot-based IT), and the time spent to gain situation awareness, formulate plan and make decisions (operator-based IT).

As mentioned in the previous section, situation awareness is one of the factors that contritube to operator performance. It is directly related to plan formulation and decision making. A higher level of SA will decrease the cognitive workload of the operator and time to react in case of unanticipated events, which in turn increases fan-out [34]. To increase operator situation awareness and decrease the time to gain it, a good interface designed will provide the operator with necessary information required for higher situation awareness [23].

2.2.2 Neglect Time

Another way to increase fan-out is to increase neglect time. Neglect time describes how long a robot can operate on its own without external intervention, or in this case, human interaction [18]. Intuitively, the longer the NT, the less a robot needs interaction from humans, and according to Equation 2.1, a longer NT will increase fan-out.

However, although a higher NT and a lower IT increase fan-out, the performance of a robot improves the longer an operator interacts, that is, performance grows with IT [28][5]. Conversely, performance deteriorates with time (NT) until the next interaction [28].

2.2.3 Robot Attention Demand

When an operator is interacting with a robot (IT), the robot is demanding and consuming time and attention from the operator, which contributes to the operator's stress level. Robot Attention Demand (RAD) describes how much attention a robot demands from the operator. In other words, this is a measure of the fraction of total task time that a user must attend to a given robot [25]. We make the simplifying assumption that RAD=IT in this thesis; this assumes that the robot will never require maintenance, will never get stuck, and will never bug out, which would have extra RAD than just the interaction time.

2.3 Team Performance Parameters

While some tasks can be accomplished by a single operator and a team of robots, some tasks need two or more operators cooperating with one another in order to be accomplished. Thus it is important to study parameters that would impact team performance. Prior work on organizing efficient teams includes (a) *maximizing fan-out*, which describes fundamental limitations on the number of robots that a single human can manage given assumptions about the robots and the tasks being performed; (b) *managing the level of autonomy (LOA)*, which describes how operators delegate work for a robot to do on its own; (c) *identifying team structures*, which includes mechanistic and organic structures defined by [22] and (d) *describing specialization* [30], which describes the effect of operator specializing in particular types of tasks. This thesis identifies several parameters that affect the efficiency of a team, such as the effect of communication, level of autonomy, fan-out, team topologies and specialization. This subsection talks about some of these parameters.

2.3.1 Single Human, Single Robot

A team consists of at least two members, so a single human and a single robot make a team. Robots can be categorized not only by the types of tasks they do, but also by how autonomous they are (LOA). Sheridan and Verplank [27] have categorized autonomy in ten levels (Table 1), level one being completely manual, level ten being completely autonomous, and with different degrees of automation in between:

Table 2.1: Levels of Automation

Automation Level	**Automation Description**
1	The computer offers no assistance: human must take all decision and actions.
2	The computer offers a complete set of decision/action alternatives, or
3	The computer narrows the selection down to a few, or
4	The computer suggests one alternative, and
5	The computer executes that suggestion if the human approves, or
6	The computer allows the human a restricted time to veto before automatic execution, or
7	The computer executes automatically, then necessarily informs humans, and
8	The computer informs the human only if asked, or
9	The computer informs the human only if it, the computer, decides to.
10	The computer decides everything and acts autonomously, ignoring the human.

At lower levels, each robot requires a lot, if not complete attention from the human operator. This results in fewer robots being controllable by an operator. When moving up the automation level, each robot requires less attention from human operators suggesting, by equation 2.1, that more robots can be controlled by an operator.

2.3.2 Multi-Operator Communication

While a lot of tasks could be completed by a single operator controlling one or more robots, some tasks would require multiple operators to coorperate, such as searching through the debris of a collapsed building to search to survivors that are stuck. Operators then will need to communicate and understand what everyone on the team is doing to effectively coordinate and divide the tasks. Communication allows operators that are geographically dispersed to be informed about the environment and actions by other operators. When done in an organized manner, communication can improve performance of the operators by increasing their situation awareness. Operators can communicate through voice, chat, or both. Voice is fast, but could be problematic when the number of operators increases and if multiple voice messages could be transmitted simultaneously. Hence voice communication is usually serial

[22]. Chat allows simultaneous message passing, and the fact that chat is free of background noise, volume effects or operator accents makes it a much clearer communication method [22].

2.3.3 Mechanistic vs. Organic Structures

Another category of team performance criteria is the structure of teams. How well teams are organized could have a huge impact on performance, even the success of tasks. Team structure could play an important role in satisfactory task completion. The types of team structures that have been studied the most are mechanistic and organic teams.

A mechanistic team is one where the operators have rigidly defined roles and responsibilities [22]. For instance, an operator is assigned one type of robot, then that operator has the full reponsibility of controlling robots of that type. If there is a set of robot types and each operator is assigned one robot of each type, then the team structure would be considered organic since any operator can perform the tasks that arise, if he/she has an appropriate robot available for that task [22].

Studies have shown that mechanistic teams perform better than organic teams overall, but perform poorly when the task inter-arrival is erratic instead of constant. This is because erratic inter-arrival times cause events/tasks to arrive in batches, thus increasing the queues [22]. However, organic teams showed no significant difference whether the inter-arrival rate is constant or erratic, and this suggests that organic teams are more robust to environment uncertainties than mechanistic teams, and are able to handle workload spikes well [22]. More work needs to be done to understand different types of organizations.

2.3.4 Human Supervisory Control

Supervisory control is somewhat the middle ground between control and computation. Control is done on a higher level after computation is done by computers. “In highly automated systems, such as Tomahawk and Patriot missiles, operators are rarely in direct control of systems, but are more involved in higher level planning and decision making. This shift from low level skill-based interaction with the agents to higher level knowledge-based interactions is called a Human Supervisory Control" [7], which makes it possible for a single operator to control multiple robots. Supervisory control is relevant to human robot teams because it specifies the framework on how humans should efficiently interact with robots, that is, higher planning rather than low level mechanical operations.

2.4 Agent-based Simulation

Agent-based simulation, also called agent-based modeling (ABM), is a powerful simulation technique that has been used in various applications. In agent based modeling, a system is modeled as a collection of individual entities called *agents* [3], each given a set of simple rules; complex group behaviors can emerge from local interactions.

Since we live in a complex world, the systems that we simulate have inherent complexities that occur because behavior emerges from decentralized decision-making. ABM has been a powerful tool in simulating the actions and interactions of autonomous agents (both individual or collective entities such as organizations or groups) in order to observe their impact on the system or group as a whole.

An agent-based simulation is an abstract description of an environment or the world, the tasks in this world, and how these tasks are completed by agents given a set of decision rules.

An agent in agent-based simulation is an actuator in the environment that can accomplish a set of tasks.

Tasks could be anything that agents or robots do in the real world.

Agents are organized into a team, defined by the topology describing the relationships agents have with each other, including how they communicate and interact.

Agent-based simulation is chosen for modelling human robot teams because it can explicitly represent the environment and the agents in it. The environment in the agent-based simulation could be any work environment in the real world, such as warehouse with boxes, a forest, etc. The tasks in the agent-based simulation are the objectives of agents in the environment, such as lifting and organizing heavy boxes, wilderness search and rescue, etc.

The remainder of this thesis presents modeling and simulation results that explore how multiple operators can efficiently manage a team of robots and comparisons of different results.

Chapter 3

Methods

In the thesis statement, we have claimed that:

- Agent-based simulations can be used to discover parameters that have strong influences on the performance of multi-operator/multi-robot teams.
- Organizations that properly respect these parameters produce teams that perform better than teams that myopically focus on maximizing fan-out.

To verify these two claims, we want to make teams of agents more effective by finding the factors that influence team performance the most. We will use agent-based simulation to do this. In this thesis we want to discover a simple set of assumptions that tell us about how the human robot team works. We will focus on the following three questions:

- How much (many) workload (agents) can an operator handle, and what are the parameters that affect this number?
- In an agent-based simulation, what parameters determine when is it possible for two humans to perform more than twice as well as a single human?
- When can organizations that support specialization free up ”spare capacity” and thus improve team performance?

3.1 What is in the Agent-Based Simulation?

This section lists what needs to go in the simulation to model human-robot teams. The terms included here are not only frequently used in literatures and in the remainder of this thesis, but also formalized and defined enough to put into the model.

3.1.1 Agent

An agent is an individual autonomous entity in an agent-based simulation. When using agent-based simulation of human-robot teams (HRT), an agent is a semi-autonomous robot managed by one or more agent operators. Agents are the most fundamental component in a team and especially in an agent-based simulation because agents complete tasks. This means that performance is measured as how much work the agents get done.

In our simulation, since it is an abtract environment, we have an infinite number of agents of each particular type that can be assigned to operators. Theoretically, operators in the simulations can be assigned as many agents as we want; in practice, the number of agents assigned to each operator is constrained by how much attention an agent demands from the operator. The amount of attention depends on an agent's Level of Autonomy, described next.

3.1.2 Level of Autonomy

Level of Autonomy (LOA) describes how long a robot can work without human interaction. Consequently, LOA determines how much work the robot can do, and how many robots can be assigned to an operator.

An ideal agent would have a high LOA, that is, require very little human interaction, and would do a lot of work on its own and producing high pay-off. In practice, although agents that have high LOA require less human attention, they often have lower performance than agents with low LOA [5] due to lack of human interaction and supervision/real-time evaluation. Agents that have low LOA may require more time from the operator (teleoperation), but they may also be able to achieve a higher pay-off during interaction time due to more human

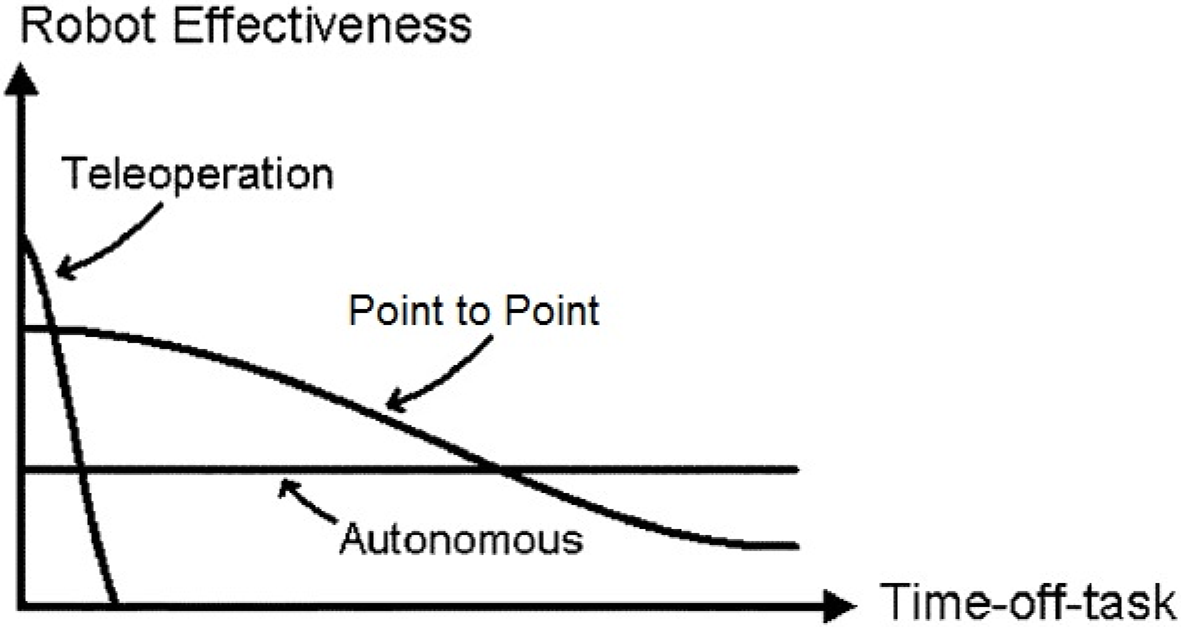


Figure 3.1: Impact of neglect time [5]

interaction. Unfortunately, teleoperated agents and other agents with low LOA often have performance that degrades very quickly during intervals of operator neglect. Crandall et al. [5] studied the effect of neglect and interaction time, and results are presented abstractly in Figure 3.1 and Figure 3.2 for three generic levels of autonomy: teleoperation (low LOA), point-to-point (moderate LOA), and autonomous (high LOA).

3.1.3 Fan-out

Fan-out is the maximum number of homogeneous robots an operator can theoretically control given the agents' NT, IT and LOA. It represents the possible maximum workload of operators and is given by [18, 25, 26]:

$$FO = \frac{NT}{IT} + 1 \tag{3.1}$$

Fan-out is an upper bound of the number of independent homogeneous robots a single human can control [5].

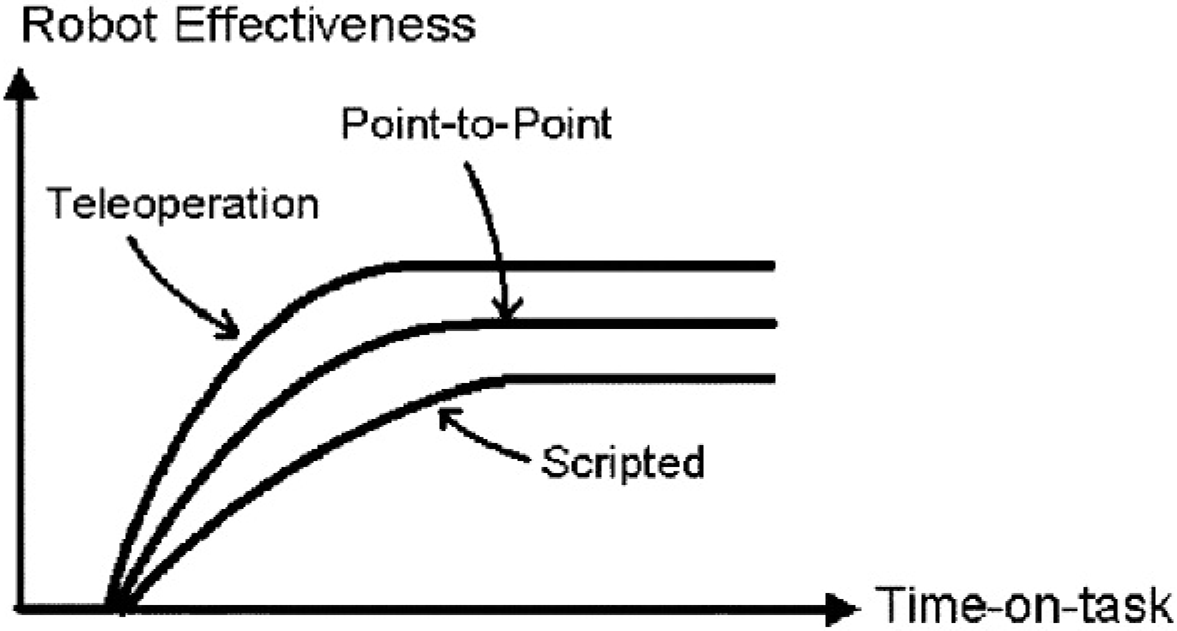


Figure 3.2: Effectiveness of Different LOAs [5]

3.1.4 Cost and Budget

Each robot has a RAD (Robot Attention Demand), modelled as "cost" derived from IT. Budget is an operator's capacity to control a number of agents. When a robot is assigned to an operator, the operator's budget is deducted by that robot's cost. The operator's attention, time and effort are abstracted into a single numeric value, which is consumed by robots according to each robots's demand on the operator. In a real world context, in order to interact with a robot, an operator has to spend time, give attention to the agent in IT, and exert effort; this is modeled by the reduction of operator's budget for each robot assigned. Cost and budget are important because a higher budget means the operator can control more robots.

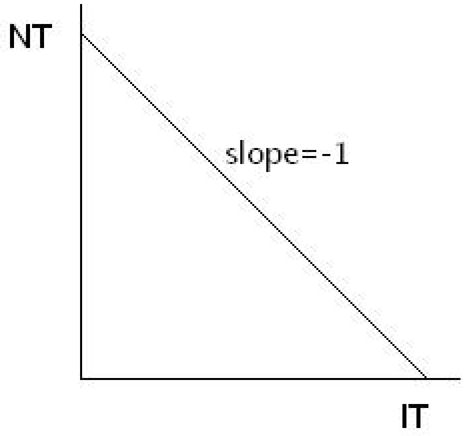


Figure 3.3: Relationship between NT and IT of operator on a robot

3.1.5 Slack

Operator slack is how much free time an operator has. In other words, it is the percentage of operator budget that is not fully utilized. When an operator is spending his or her fully utilized budget, he or she will do the works faithfully and get no break. However, during slack time, many things can be done or happen; it is up to the operator to do whatever is desired, taking a rest to recuperate, or doing various other tasks.

3.2 Modeling Assumptions

From the previous section, LOA, NT, IT, slack, cost and budget are the parameters that determine how a human operator can use his or her time to promote high HRT performance.

To perform effective agent-based simulation, it is useful to reduce the number of parameters.One way to reduce the number of parameters is to note that LOA is associated with a pair of variables: (NT,IT). Thus, we can reduce the number of parameters for LOA by compressing the two-dimensional (NT,IT) space to a single dimensional parameter space.

Figure 3.3 shows how we model the relationship between NT and IT as inversely proportional; that is, when NT increases, IT decreases and vice versa. The inversely proportional relationship between IT and NT can be formalized as:

$$NT = 1 - IT \tag{3.2}$$

This model treats NT and IT as percentages of total budget. For example, when a human interacts with a robot for 10 hours in such a way that interaction begins at the first hour and neglect begins at the second hour and continues to the end, then NT=0.9 and IT=0.1.

Since both NT and IT are percentages, they add up to 1. With this assumption, NT and IT could be combined into a single parameter used to model LOA and budget could be eliminated. Hence LOA is defined to be the percentage of time the robot can operate on its own, and is given by the following equation:

$$LOA = \frac{NT}{NT + IT} = \frac{NT}{1} = NT \tag{3.3}$$

Equation 3.3 equates robot autonomy with how much work it could do on its own without human interaction. If a robot could operate on its own 90% of the time, it has an autonomy level of 90%. In the real world, we often consider a robot to be more autonomous if it requires less time from the human operator and can do more work on its own; the same concept is modeled in the assumption here.

Equation 3.1 represents the relationship between fan-out, NT, and IT. Under the assumption given in equation 3.3, the relationship between fan-out and NT is shown in Figure 3.4. Simply put, fan-out grows with increasing LOA.

3.3 Estimating Performance

An agent's performance (output) and its cost (IT) are generally a function of its LOA. During interaction time, the agent is consuming time, effort and resources from the operator, causing its performance to increase (Figure 3.5). At the end of IT, the operator stops interacting with the agent, either to take a break or to interact with other agents. During the subsequent neglect, the agent's performance level will start to decrease, until it hits a minimum acceptable

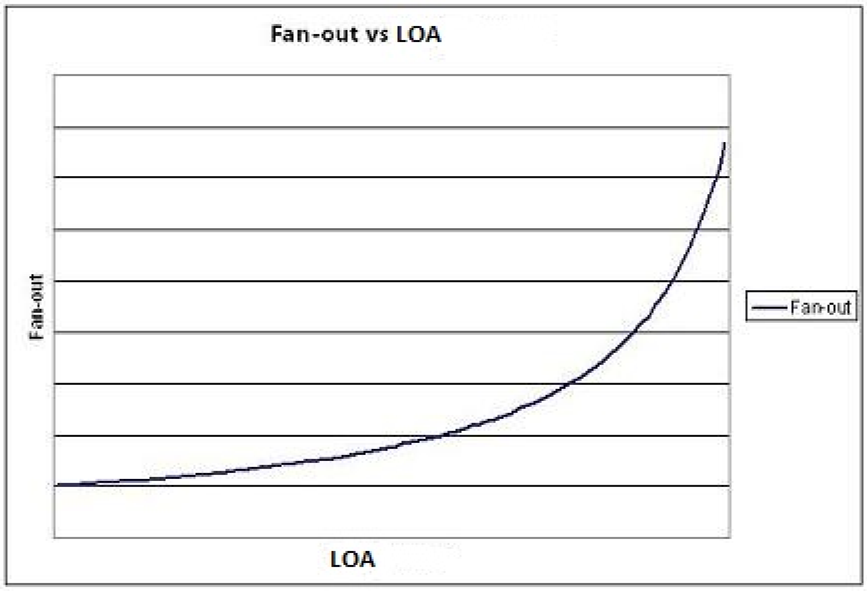


Figure 3.4: Relationship between LOA and Fan-out

performance threshold (end of NT). At that time, the operator must interact with the same agent again to boost its performance. Since how exactly performance decreases during NT has not been explicitly parameterized in the literature and it does not really impact the results, linear decay is assumed; also linear is the most simple approximation of measured decay.

Without loss of generality, the minimum acceptable performance threshold is set to zero. Each individual agent's performance is the area under the graph depicted in Figure 3.5, computed using a Riemann sum [29]. Subsequently, team performance is the sum of each individual agent's performance.

3.3.1 Operator Load & Performance Measurement

The first question we want to answer in this thesis is how much load an operator can handle, that is, how much utilization. To do this, we will estimate the operator's performance at 50% utilization, then slowly increase the operator load (utilization) and measure the respective performance of the team given different utilizations. In our simulation, controlling more

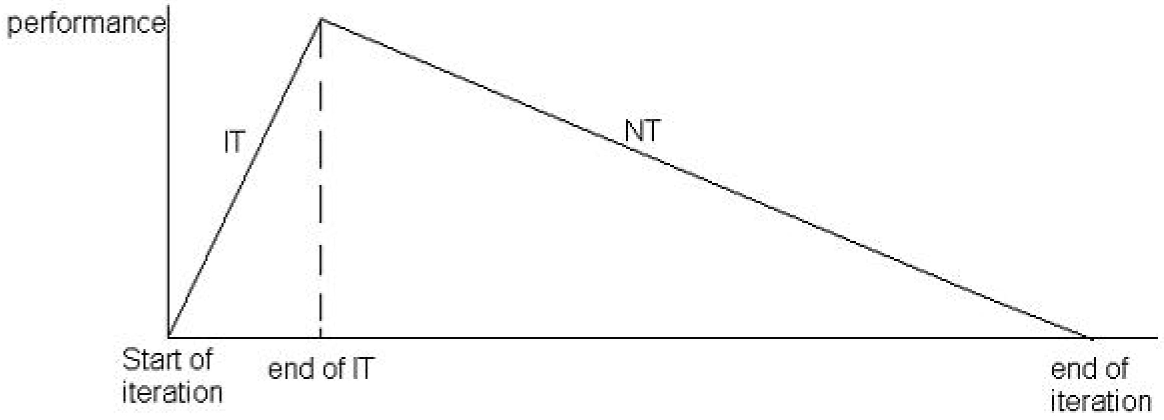


Figure 3.5: Representation of Neglect and Interact Time

agents would seem to yield a higher team performance, but it also increases workload on the operator. From Kavanagh's work [19], predictions are that performance will increase as utilization increases, but will eventually peak out and drop if there is further increase in load. One way that workload negatively impacts operator performance is discussed in the next section.

3.3.2 Operator Load & Operator Stress

Casper [4] demonstrated that humans tend to make more mistakes when stressed. To model this, an implicit parameter called the "mishandling" probability is used and is calculated as:

$$mishandling = \begin{cases} 0 & \text{if utilization} \leq 50\% \\ (utilization - 50\%)^2 & \text{otherwise} \end{cases} \tag{3.4}$$

This equation is used because we try to emulate the human performance and stress curve as much as possible (Figure 2.1). Due to the high cost and the critical nature of missions, the lower end of the U curve is ignored, assuming motivation is not a problem in this scenario. From Equation 3.4, note how mishandling probability increases with increasing utilization as shown in Figure 3.6.

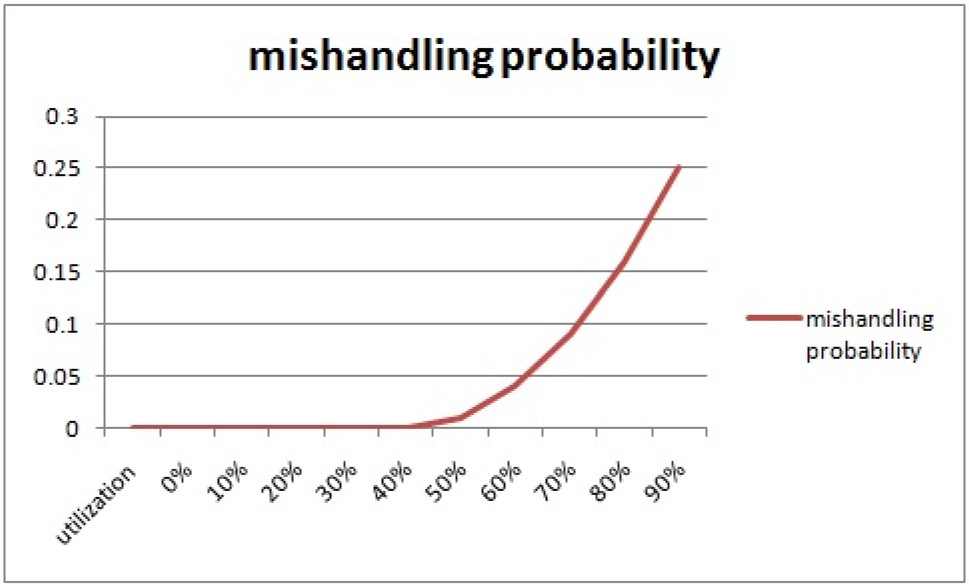


Figure 3.6: Relationship between utilization and mishandling

If an operator mishandles, the performance of the agent in IT will not be added to the team performance, instead, it is “wasted” due to mishandling. In reality, this wasted work may even hinder further progress of the team, which is a negative impact. But since this is totally stochastic and would require more complex modeling, we make the simplifying assumption that a mishandled robot will simply not contribute anything to the team during the mishandled period.

3.4 Multi-Operator Performance

When the operator is overloaded, increasing load will actually decrease the operator’s performance with the team. This leads to the interesting question of what can be done to keep performance up? In the model in this thesis, when the load of a single operator is over the “threshold” where performance starts to drop, we introduce a second operator to share the load with the first operator, and performance is estimated and compared to that of a single operator (question 2).

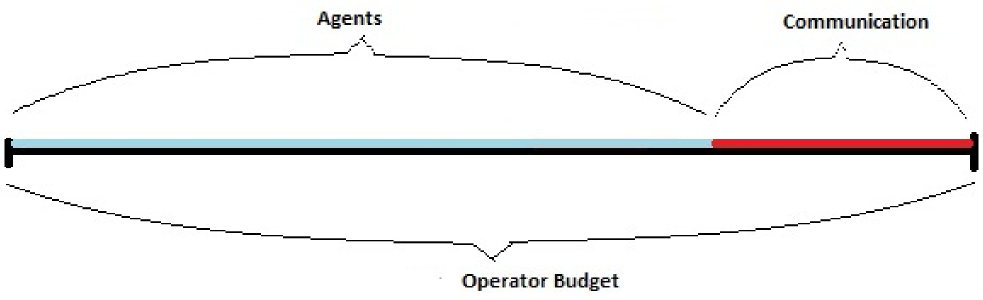


Figure 3.7: How Communication takes up operator budget.

3.4.1 Communication

In real life, an essential component of an effective team is communication [13]. Operators need to communicate with each other, pass information along and sometimes even make collective decisions. Lack of communication may cause redundant work to be done by different operators, or may even cause an operator to undo the work of another operator. For example, in a search and rescue scenario, two independent and non-communicating operators may control different UAVs to scan through an area for the missing person. However, due to lack of communication, some areas that they each covered may have overlapped, which could be a waste of time and resource.

In human-robot teams, the two primary means of communication are voice and text [22]. Each of them has its pros and cons. For example, voice communication is fast and easy, but it is prone to channel noise, background noise, operator accent, etc.; whereas text communication does not have the drawback voice communication has, but it is slow and requires that an operator's visual focus be away from the agents being controlled.

Although detailed comparison of different communcation media is beyond the scope of this thesis, we are nevertheless interested in how communication can help improve team performance. On the downside, communication is modeled in the simulation as a cost on an operator's budget and time (Figure 3.7). On the positive side, we model communication as a multiplier to the team's performance as a percentage in order to simulate team coordination and the ease of tasks due to division of work. Since lack of communication could cause team incoherency, intuitively we would decrease team performance inversely proportional to the amount of communication the operators have; however the optimal amount of communication is unknown. There could also be a point at which adequate communication could even increase team performance. So, instead of reducing performance according to how much communication there is, communication overhead will act as a multiplier to increase both operators' team performance. For example, if the communication between the operators cost 20% of the operators' total budgets, the team's performance will be multiplied by 120%. This is explained more in detail in the next chapter.

There are two types of communication paradigms modeled in the simulation, the peer paradigm and commander paradigm.

Peer

In the peer paradigm, operators are of equal importance and have equal amounts of responsibilities. This means that although they might not control the same type or amount of robots, they spend the same amount of time communicating with one another. In the end, the team performance is increased according to the amount of communication between the operators.

Commander

In the commander paradigm, one operator acts as a "commander" and makes most of the decisions. The commander will have a higher communication cost, which means it controls fewer agents. The other operators act as "subordinates", and consequently they spend less time communicating, because most of their communication is simply reporting to the commander and following orders. In the end, the team performance is multiplied by the communication time spent by the commander.

3.4.2 Specialization

Multiple operators with heterogeneous agents can control the agents in two different ways. One is that each operator can and will control more than one type of agent, and the second way is that each operator controls (specializes) in controlling one particular type of agent. Jonathan Whetten stated that specialization will improve operator performance by "freeing up spare capacities" [30]. We evaluate this claim by introducing multiple types of agents with different LOAs, assign a particular type of robot to each of the two operators (specialization), then assign a mixture of agents to both the operators (non-specialization). When an operator switches from one type of agent to another, there is a "switching cost", because in real life, when a person switches from one task to another, there is usually an "adjusting" period; this switching cost is to simulate the adjusting period. The switching cost works like this: an agent's performance still increases during IT, but when the operator switches over from a different type of agent, the agent in IT will still increase its performance, but at a lower rate (Figure 3.8).

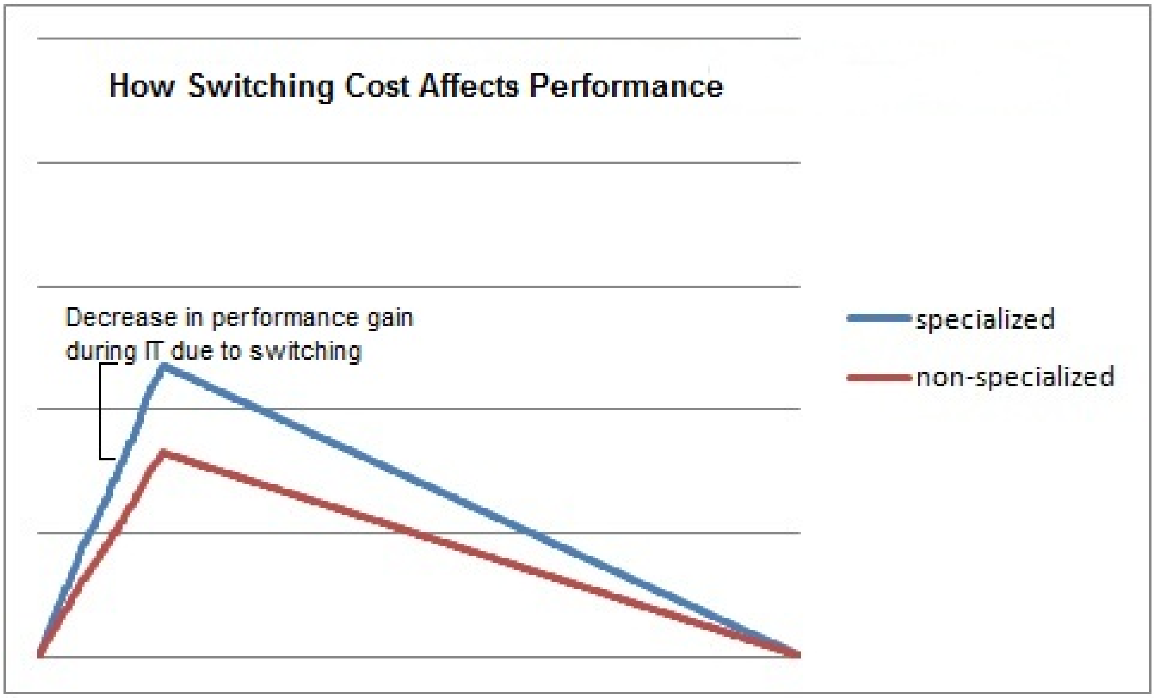


Figure 3.8: How Switching Cost affects performance.

Chapter 4

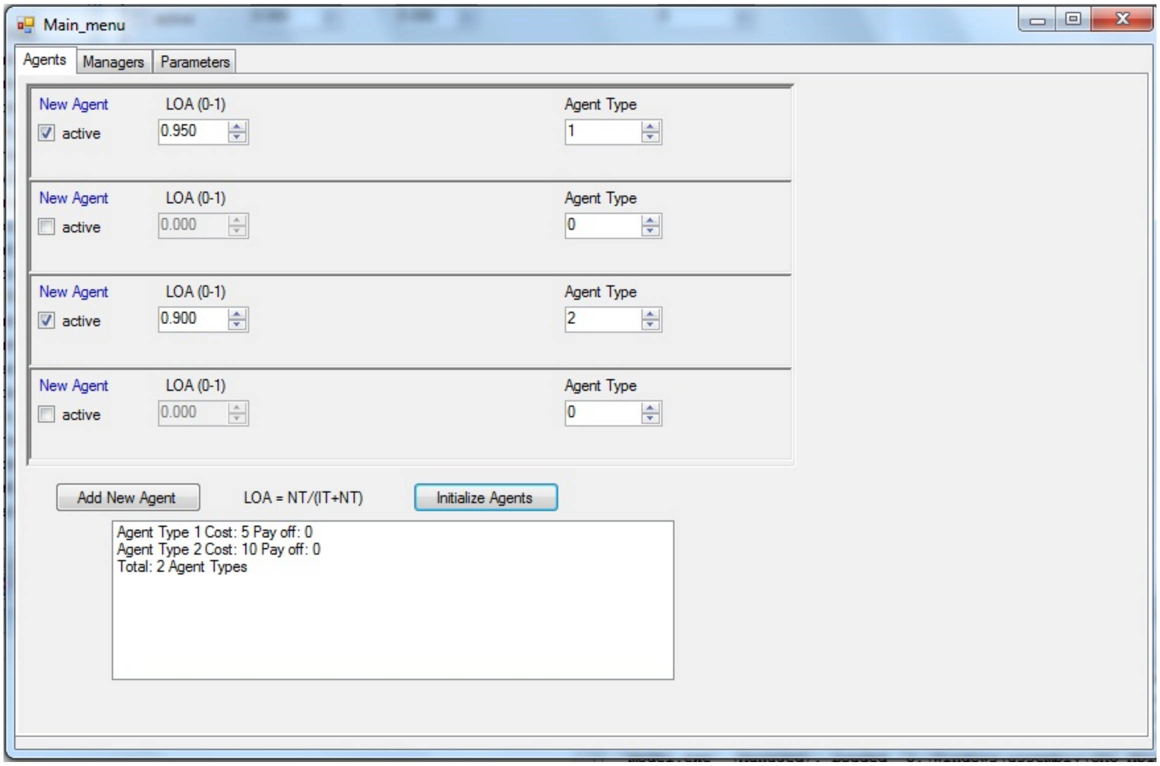
Simulations and Analysis

An agent based model simulation is coded in Visual Studio 2008, with the parameters and specifications in the previous chapter incorporated into it. Various experiments are set up and conducted to analyze, contradict or confirm the following hypotheses:

H1: maximizing fan-out will not produce the best team performance;

H2: adequate communication improves performance;

H3: specialization helps teams to perform better by freeing up spare capacities.



New Agent ☑ active LOA (0-1) 0.950 Agent Type 1

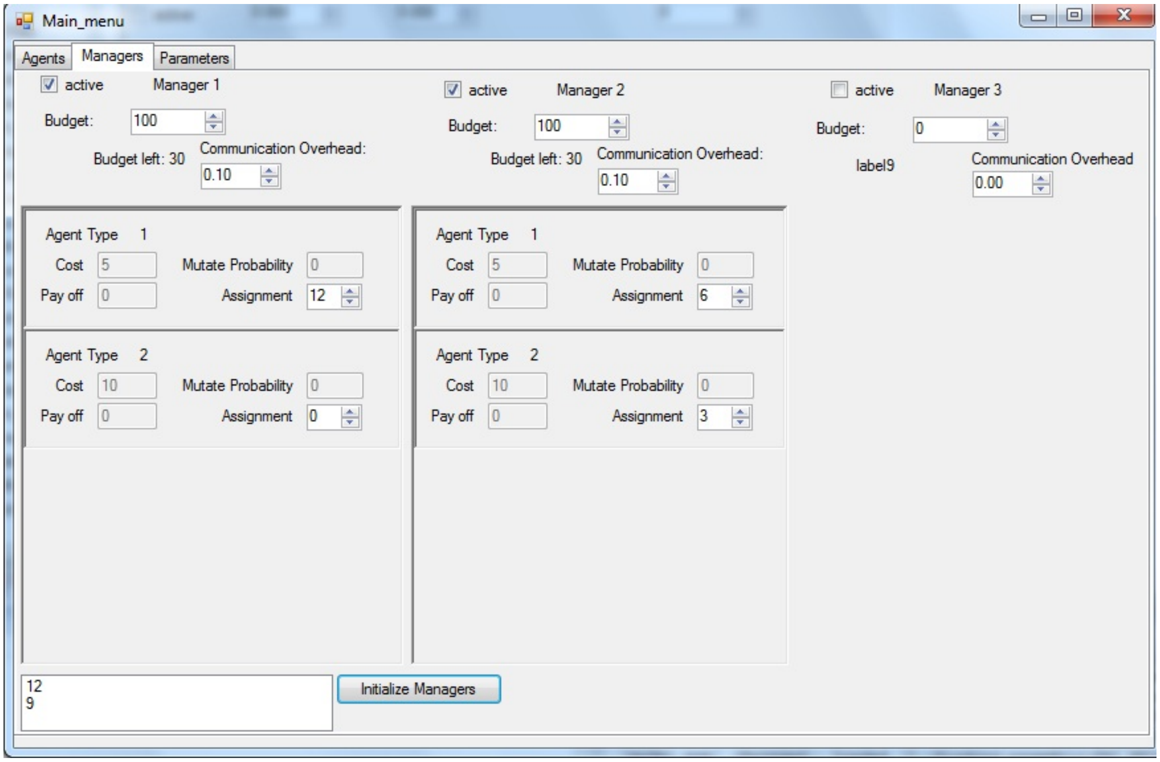
New Agent □ active LOA (0-1) 0.000 Agent Type 0

New Agent ☑ active LOA (0-1) 0.900 Agent Type 2

New Agent □ active LOA (0-1) 0.000 Agent Type 0

```
Agent Type 1 Cost: 5 Pay off: 0
Agent Type 2 Cost: 10 Pay off: 0
Total: 2 Agent Types
```

Figure 4.1: Initiailizing agents.



☑ active Manager 1
Budget: 100
Budget left: 30
Communication Overhead: 0.10

Agent Type 1
Cost 5
Mutate Probability 0
Pay off 0
Assignment 12

Agent Type 2
Cost 10
Mutate Probability 0
Pay off 0
Assignment 0

☑ active Manager 2
Budget: 100
Budget left: 30
Communication Overhead: 0.10

Agent Type 1
Cost 5
Mutate Probability 0
Pay off 0
Assignment 6

Agent Type 2
Cost 10
Mutate Probability 0
Pay off 0
Assignment 3

□ active Manager 3
Budget: 0
label9
Communication Overhead 0.00

12
9

Figure 4.2: Initializing operators.

The simulation code is organized into different classes: GUI, agent, manager (operator), globals and main. The experimenter can intialize any number of agents of any type by specifying the LOA and agent type, as well as indicating that this type of agent is "active" by checking the checkbox for the agent (Figure 4.1). The experimenter then can select an operator/operators to be "active", and assign the agents to the operator(s), together with the amount of budget available and communication overhead of each operator (Figure 4.2). Lastly, the experimenter will set other parameters such as the lazy factor and the number of iterations for the simulation to run. Note that parameters such as mishandling, slack are calculated implicitly according to the load the experimenter assigns to the operator. In the rest of this chapter we will discuss these parameters (Table 4.1) in detail and the results obtained through the simulation.

Table 4.1: Parameters used in measuring optimal operator load

Parameter	**Description**
Agent Type	The type of agent. Different types of agents can have different functionalities, such as Air Vehicle, Ground Vehicle and Underwater Vehicle.
LOA	How autonomous an agent is.
NT	Neglect time, determined by LOA.
IT	Interaction time, determined by LOA.
Budget	The amount of budget units an operator has, this includes time and effort.
Mandatory Budget	The amount of budget units an operator **must** spend on interacting with agents, there is no rest in this period.
Slack	The amount of (free) time an operator has, is a percentage of the operator's total budget that is not mandatory budget.
Lazy Factor	How lazy the operator is to work during slack time. This is a probability.
Mishandling	How likely an operator is to make mistakes. This is also a probability and is calculated based on how much of the operator's total budget is used as mandatory budget.
Communication Overhead	How much time operators take out from controlling agents to communicate with each other.
Iterations	How many iterations the simuation will run. Each iteration ends when the operator has fully expended his/her budget.

4.1 How Much Workload Can an Operator Handle?

The first of the three questions in the beginning of chapter 3 we will answer in this thesis is how much workload an operator can handle. In this experiment, agents of LOA 0.95 are used to simulate highly autonomous robots and this is kept constant. Operator slack and lazy factors are adjusted to produce various results.

4.1.1 Individual Agents

Assume that the operator does not make any mistakes and is controlling the maximum number of homogeneous agents given by fan-out. Each agent's performance looks like what is given in figure 4.3. The maximum performance of an agent is set arbitrarily at 250 units. On the right side of figure 4.3, it shows the shape of the performance curve clearer after zooming in. As shown, the agent's performance increases during IT and decreases during NT as stated

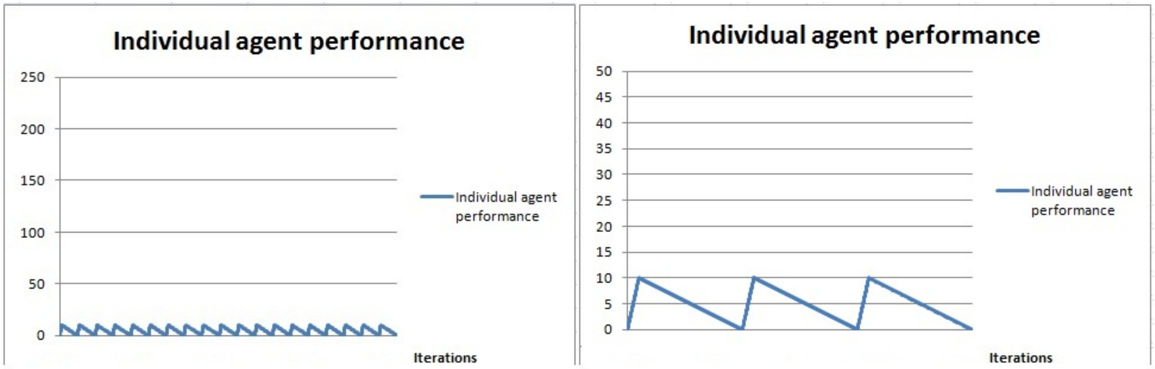


Figure 4.3: The performance curve of an individual agent controlled by an operator without any slack

before, and this cycle continues on without any chance of the agent reaching its maximum potential.

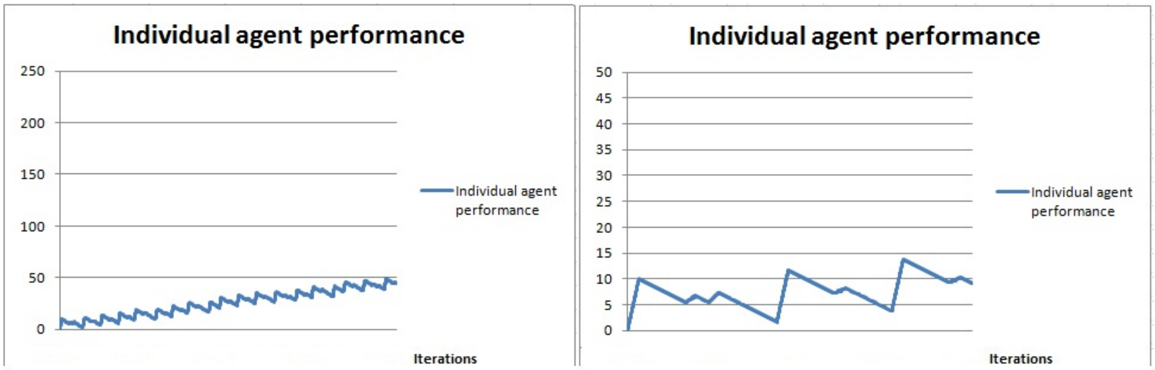


Figure 4.4: The performance curve of an individual agent controlled by an operator with 50% slack and lazy factor = 0.8

However, if the operator is given some slack (for example, 50%), and is willing to take, let's say, only 20% (lazy factor = 0.8) of the effort to continue working on the agent with the lowest performance during slack time, the agents can slowly increase their maximum performance over time. This is because the agent with the lowest performance will receive interactions early, before the end of its neglect time, causing its performance to climb. Thus the agent's performance gets a boost before it reaches the minimum threshold (where it was at the beginning of IT). Figure 4.4 illustrates this phenomenon.

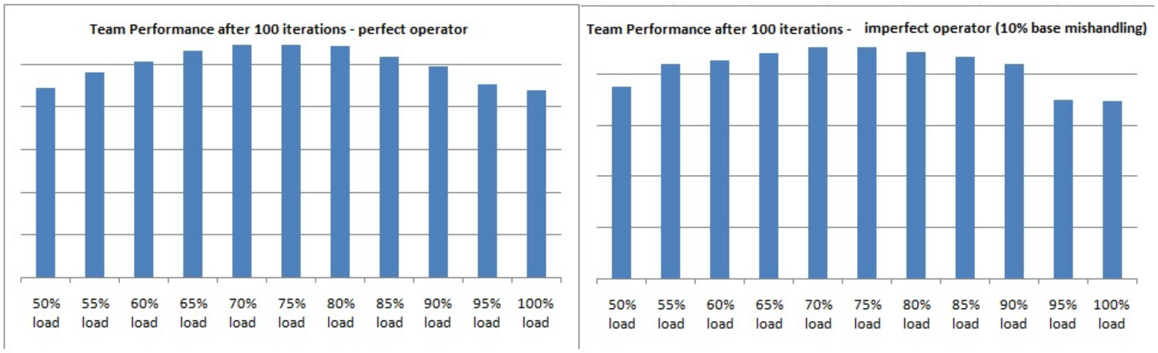


Figure 4.5: Comparison of team performance when the operators are completely lazy during slack (lazy factor =1)

4.1.2 Team of Agents

We consider a team of two operators, and run simulations with different operator loads ranging from 50% to 100%, with increments of 5%. As shown in figure 4.5, simply giving some slack and without being required to do extra work during that time, operators produce a higher performance with the team of agents when there is moderate slack time (about 20% to 30%).

The left of Figure 4.5 shows the comparison of team performance when the operators are completely lazy during slack period (except for 100% load when there is no slack, hence cannot be lazy), and operators make mistakes (mishandles) completely based on the amount of workload they have; that is, operators do not make mistakes when their workload is below 50%, which is a quite optimistic speculation. As shown, given these assumptions, operators have the highest performance at about 75% workload.

The right of Figure 4.5 shows the same set up as the left except that operators have a "leniency" of making mistakes by themselves while not being overloaded; we call this leniency the "base mishandling" probability, therefore Equation 3.4 becomes:

$$mishandling = \begin{cases} basemishandling + 0 & \text{if utilization} \leq 50\% \\ basemishandling + (utilization - 50\%)^2 & \text{otherwise} \end{cases} \tag{4.1}$$

10% is the number used for this base mishandling probability, which is a quite reasonable estimate and is more likely to happen in the real world than "perfect" operators. As shown in Figure 4.5, operators have the performance peak at 70% workload. For the rest of this thesis, we will use 10% as the base mishandling probability, due to human imperfection and how operators can make mistakes regardless of outside influence.

Now, if the operators are willing to continue interacting and working with the agents during slack time with a probability (1 - lazy factor), there will be an even more delicate relationship between operator load and team performance. This is because controlling more agents may add more members and work to the team, but it also means that there will be more mistakes made by the operator. Thus, the operator will have less slack time hence less time to further interact with agents outside IT to boost performance, and of course, this will make the agents' performance curves look more like the ones in Figure 4.3 which yields a lower performance than that of Figure 4.4.

Since each operator is unique as an individual, there is no explicit way of generalizing the laziness of all operators during slack time and how they will behave. So we will not do extensive experiments with different combinations of slack and lazy factors; instead, we will show the effects of different lazy factors on team performance and the trade off between the number of agents and performance gain.

In each set-up in Figure 4.6, the operator has 70% load, but with varying lazy factors. As shown, lazy factor influences the speed at which agents increase their performance during iterations until agents have reached their maximum performance (performance plateaus). With a higher lazy factor, agents will have a steeper increase in performance if the performance gain is greater than the performance loss due to operator mishandling; if not, there will not

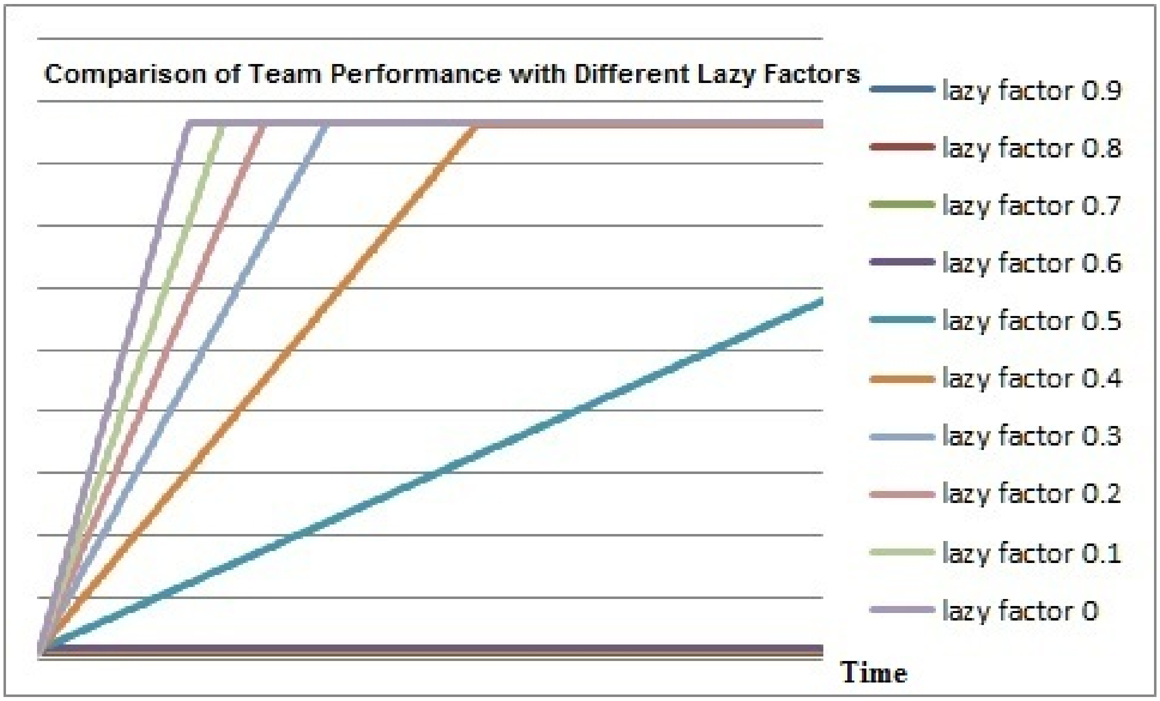


Figure 4.6: Comparison of team performance of the same operator with different lazy factors.

be performance gain during NT. If the operator has a lower lazy factor, that is, the operator is more willing to work with agents during slack time, the performance gain of agents will increase faster. Theoretically, a lazy factor of 0 will be the best, but it is not ideal. A zero lazy factor means that the operator does not get any rest, even during slack time; however willing the operator may be to work, under exhaustion and fatigue poor performance may result [4]. Since each operator is unique, this result aims to provide a better understanding of how activities during slack can increase performance, and not to find a universal lazy factor for all operators to produce the optimal output.

In Figure 4.7, we show the trade off between the number of agents (or load) and performance. A lazy factor of 0.5 is used. As shown, with more agents assigned to an operator, it takes longer for agents to increase their performance and reach their maximum potential. And also, intuitively, more agents mean less slack time, which means more mishandling by the operator, and that in turn requires a lower lazy factor in order for the performance gain to overcome the performance loss due to mishandling. Speculations are that, even if long

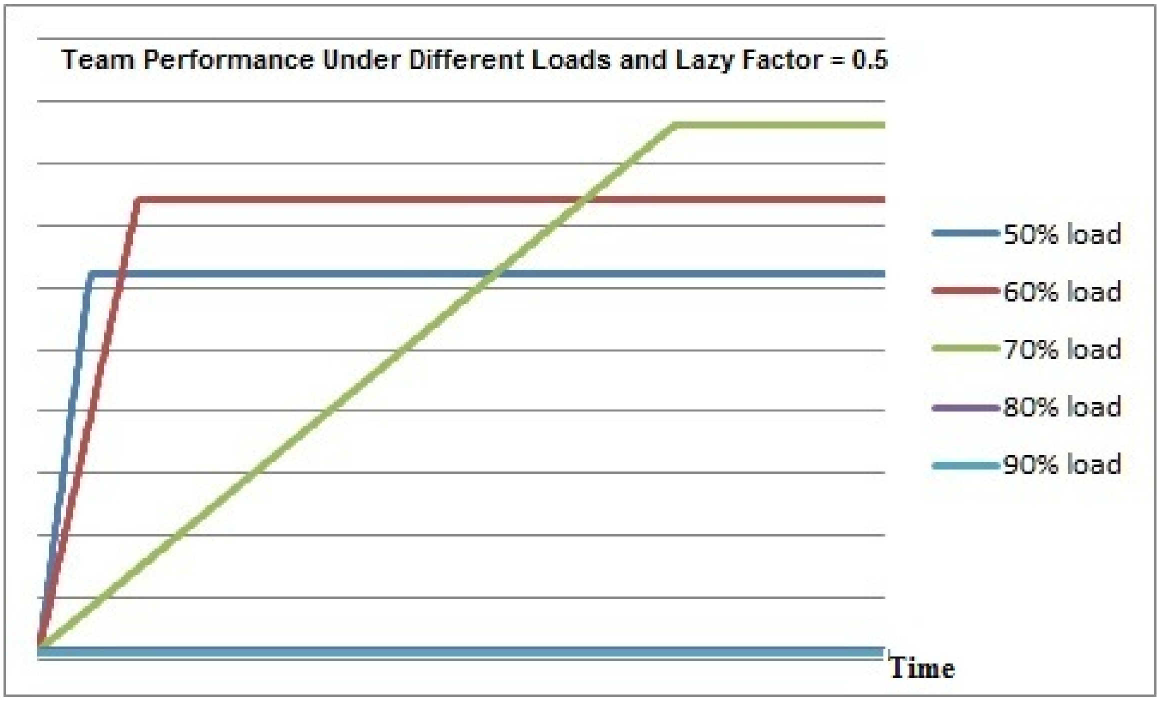


Figure 4.7: Comparison of team performance with operator lazy factor = 0.5 and different loads.

term performance is desired (slow increase but higher output after a long period of time), if the lazy factor required to overcome the performance loss due to mishandling is higher than the operator's maximum acceptable lazy factor, we will have an under-achieving team.

4.2 When Is 2 > 1+1?

In the previous section we have established that when given moderate slack (20% to 30%), operators can perform the best. This introduces our next question, *when can two operators be more than twice as effective as a single operator?* In this section, the same agents of LOA 0.95 are used, and all of the operators are completely lazy. Fixed workload is either carried out by a single operator or divided between two operators.

Our argument is this: since the operator obtains maximum performance at 70%-80% load, when the operator is over that threshold (overloaded), the best way to boost performance and efficiency is to bring in another operator. As shown in figure 4.8, the performance of

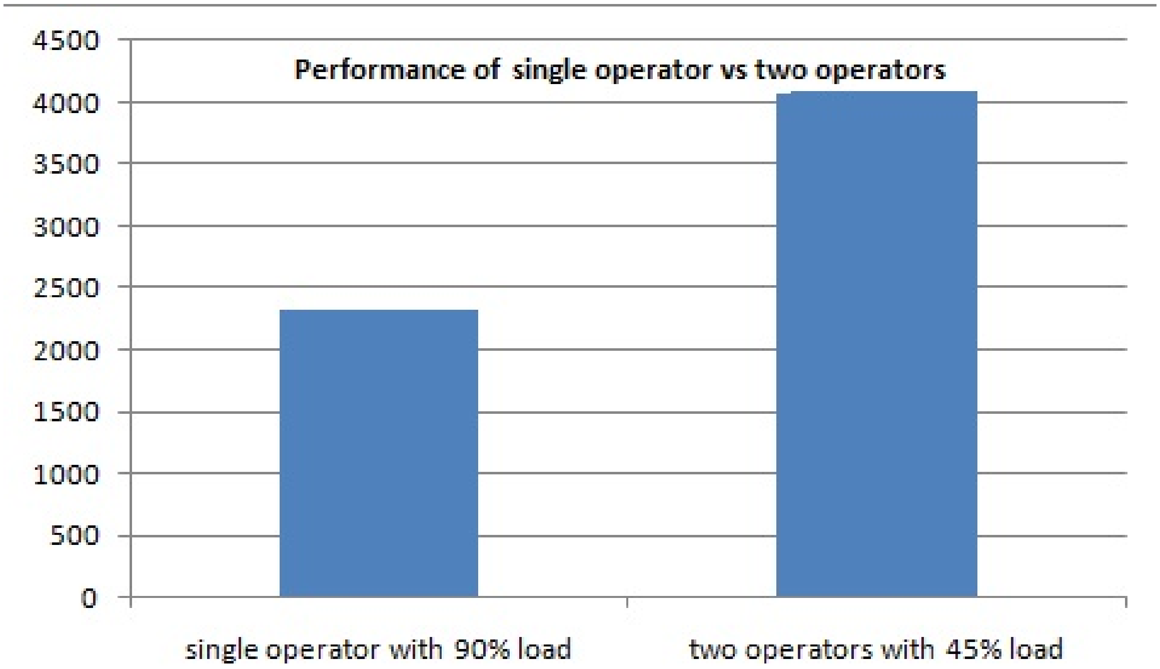


Figure 4.8: Comparison of team performance of a single operator at 90% load and two operators sharing the same load (operators are completely lazy during slack).

the two operators sharing the same load (90%) is roughly twice the performance of a single operator at 90% load. But this is under the assumption that operators are completely lazy during slack, that is, they do not interact with the agents at all during slack. However, as shown in the previous section, even given a little enthusiasm, the operators can boost the performance of the team during slack time. So here is the reason for our argument: two operators sharing the same load as an overloaded operator will have significantly more slack, thus during their respective slack time, the duo will gain a larger performance boost for the team if the operators interact with the agents during slack time.

Figure 4.9 shows the result from a simulation designed to test this hypothesis. The experiment consisted of two scenarios: an operator is assigned 90% workload, and two operators are sharing the same load, i.e., 45% workload each. Figure 4.9 shows the comparison of performance of an agent controlled by the single operator at 90% load and an agent controlled by one of the two operators sharing the same load. As shown, excessive mishandling

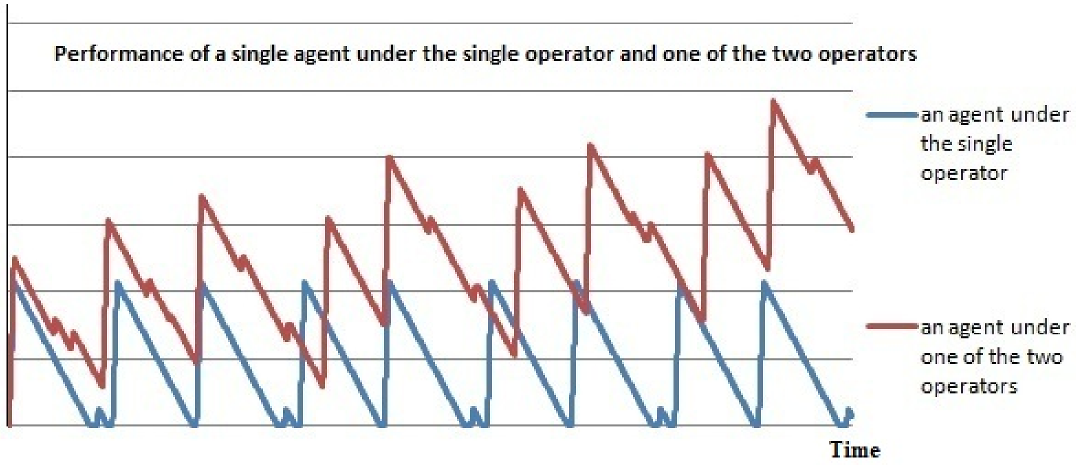


Figure 4.9: Comparison of single agent performance under a single operator at 90% load and two operators sharing the same load (operators are lazy 90% of the time during slack).

caused by high workload and lack of slack hinders agent performance growth even when the operator is not completely lazy during slack; whereas the two operators sharing the load no longer have constant performance with the team of agents, because they have the capacity to increase agent performance over time due to increased slack time and reduced mishandling probability. Again, when operators have a little enthusiasm during slack (90% laziness as opposed to completely lazy), performance of the two operators sharing the same load as the single operator proved to be superior. This is because reduced load contributes to reduced mishandling of the operators, and as mentioned previously, will give the operators more slack time to further interact and boost the performance of the agents, as shown in Figure 4.10.

So far the operators are completely independent of each other. In practice, a team of operators will require communication, and managing this communication will impact team performance. Since the amount of communication required depends on the relative roles of the operators, we explore two operator organizations: peer to peer and commander-subordinate.

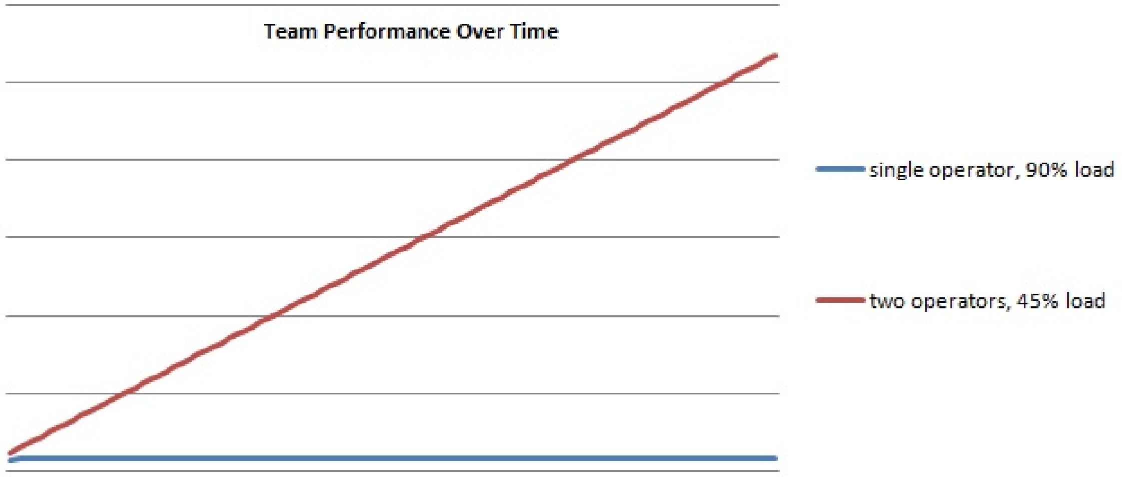


Figure 4.10: Comparison of team performance under a single operator at 90% load and two operators sharing the same load (operators are lazy 90% of the time during slack) over iterations.

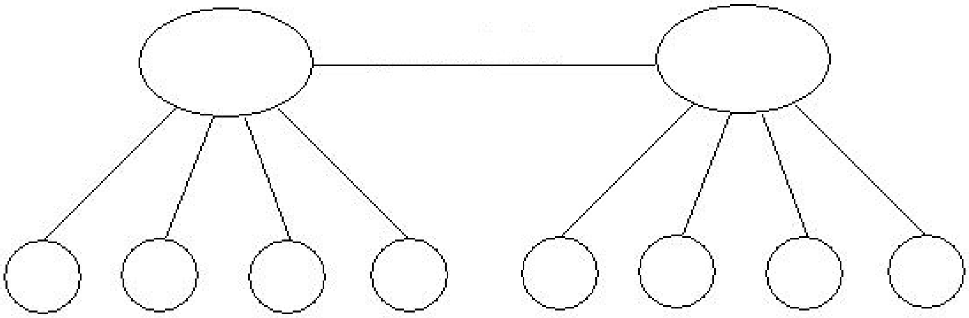


Figure 4.11: Peer to peer topology. Large ovals represent operators and small circles represent agents.

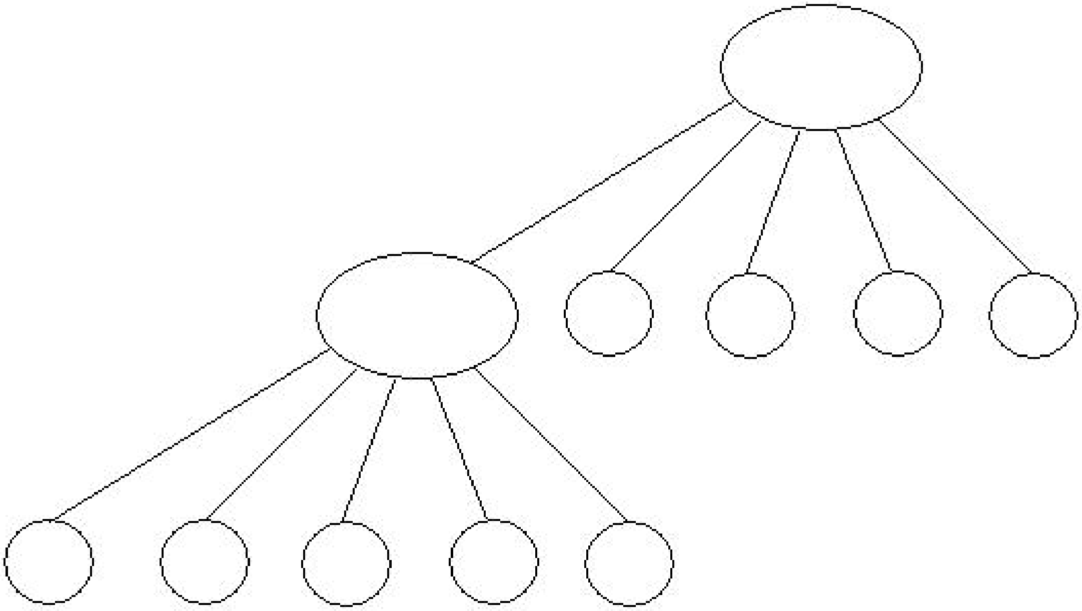


Figure 4.12: Commander topology. Large ovals represent commanders, small circles represent agents.

4.3 Communication and How It Can Help

In this section, agents with LOA 0.95 are used, operators are completely lazy during slack, and the amount of slack time for each operator is kept constant at 30%. Communication overhead is adjusted, and is accomodated by increasing/reducing the amount of workload for the operators since each operator has a fixed slack time. We will discuss the different topologies: peer to peer (Figure 4.11) and commander (Figure 4.12) and discuss the experiment results.

4.3.1 Peer to Peer

We will use the same agents and parameters used in previous simulations (70% load and completely lazy operators during slack), but this time we will have another parameter that is uniform for both operators, which is called “communication overhead”. This communication overhead is similar to an agent, in the sense that it takes up operator budget (Figure 4.15); however, it does not do any tasks and does not contribute to the summation of individual

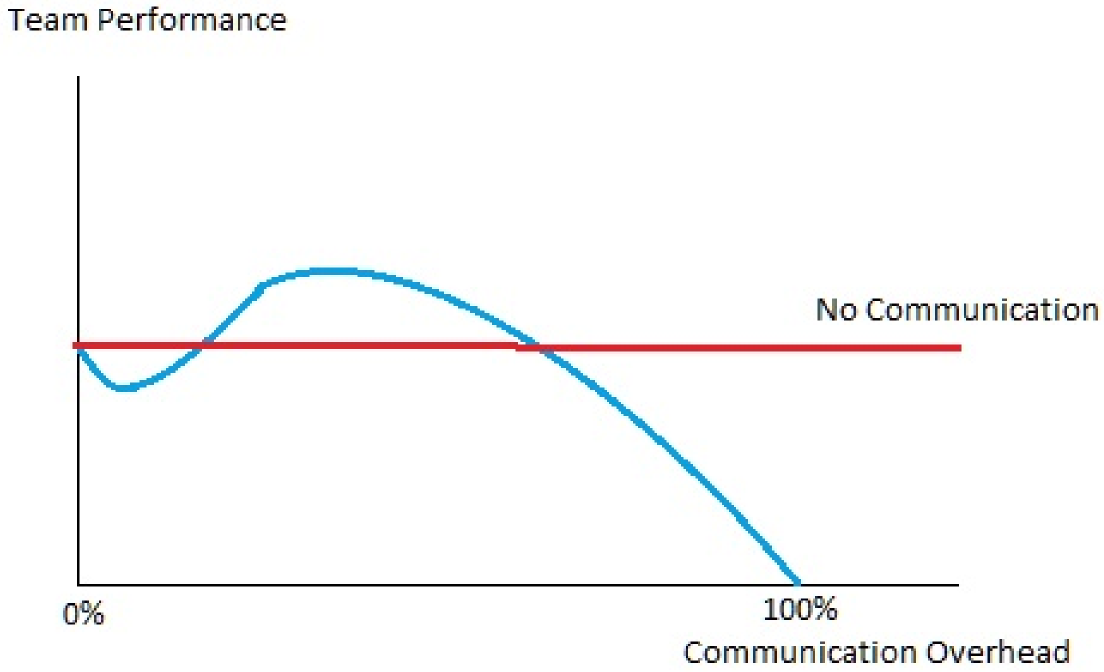


Figure 4.13: How team performance maybe affected by communication in real life.

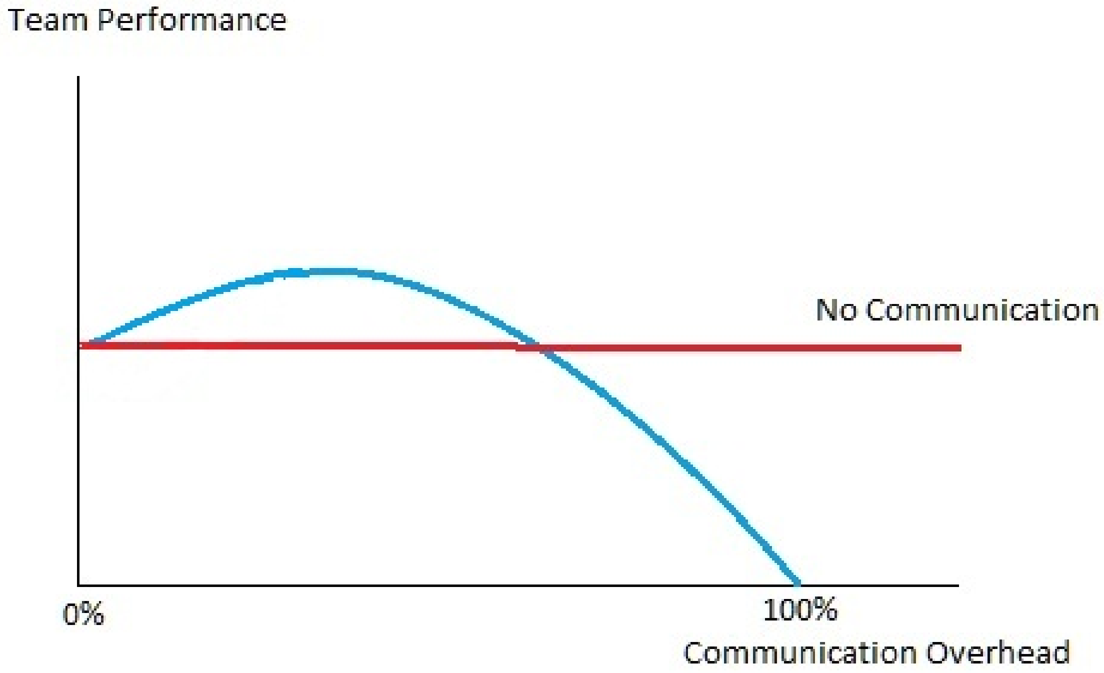


Figure 4.14: How team performance maybe affected by communication in the experiments.

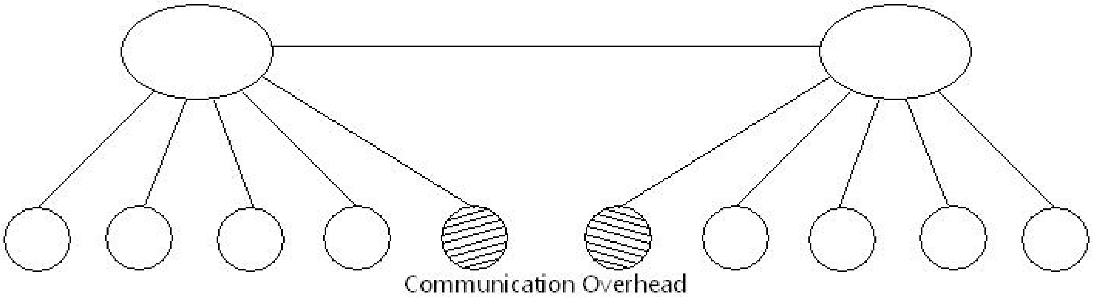


Figure 4.15: How communication overhead is represented in the simulation

agents performance. Since lack of communication could cause team incoherency, intuitively we would expect a decrease in team performance inversely proportional to the communication overhead the operators have (Figure 4.13). But the optimal amount of communication is unknown, and there could be a point at which the benefits of communication outweigh communication overhead producing an increase in team performance, so instead of reducing performance according to how much communication there is, communication overhead will act as a multiplier to increase both operators' teams' performance (Figure 4.14). We are interested in finding that "sweet" spot(s) with the highest team performance.

As shown in figure 4.16, the performance of two non-communcating operators and two operators communcating 10% of the time are approximately the same. However, operators controlled fewer agents when they communicated more. This means that at 10% communication, operators can control fewer agents to produce similar performance output as two non-communicating operators. Not only will this save on operation cost, controlling fewer agents may also reduce the mistakes that operators make.

Now we will move on to three operators. Similar to the two operator set up, three operators also had uniform communcation, and team performance is also increased according to how much communication there is. Figure 4.17 shows the result from our simulations. As shown, for three operators, communication among the operators does not actually help the team, instead, performance is in fact lower than that of non-communicating operators. In real

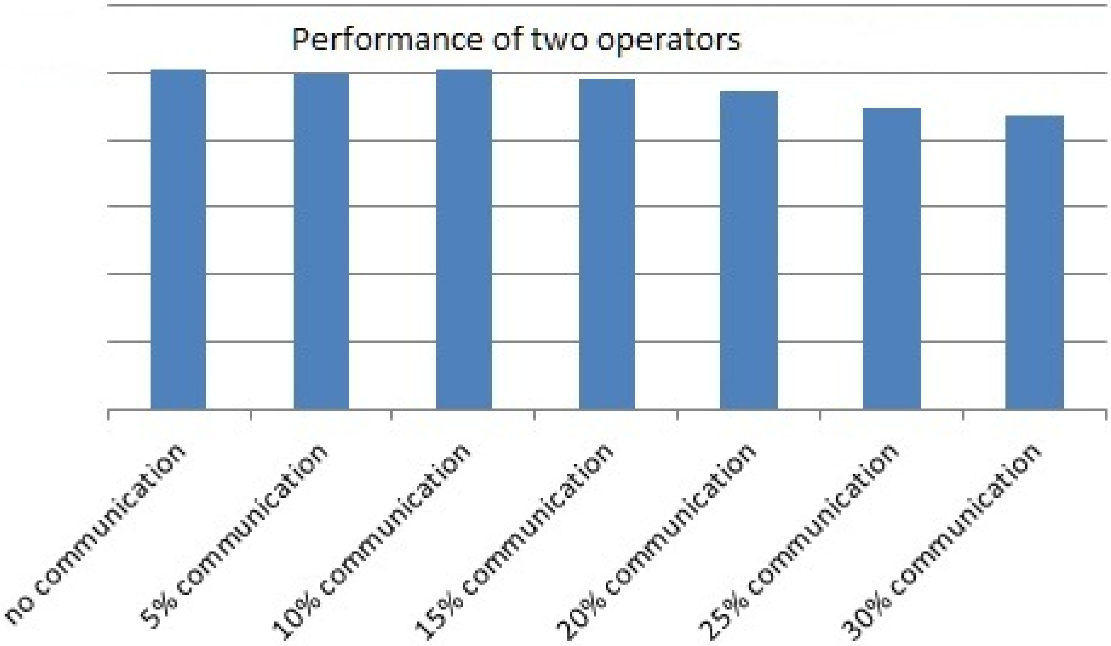


Figure 4.16: Comparison of team performance of two operators with different amount of communication.

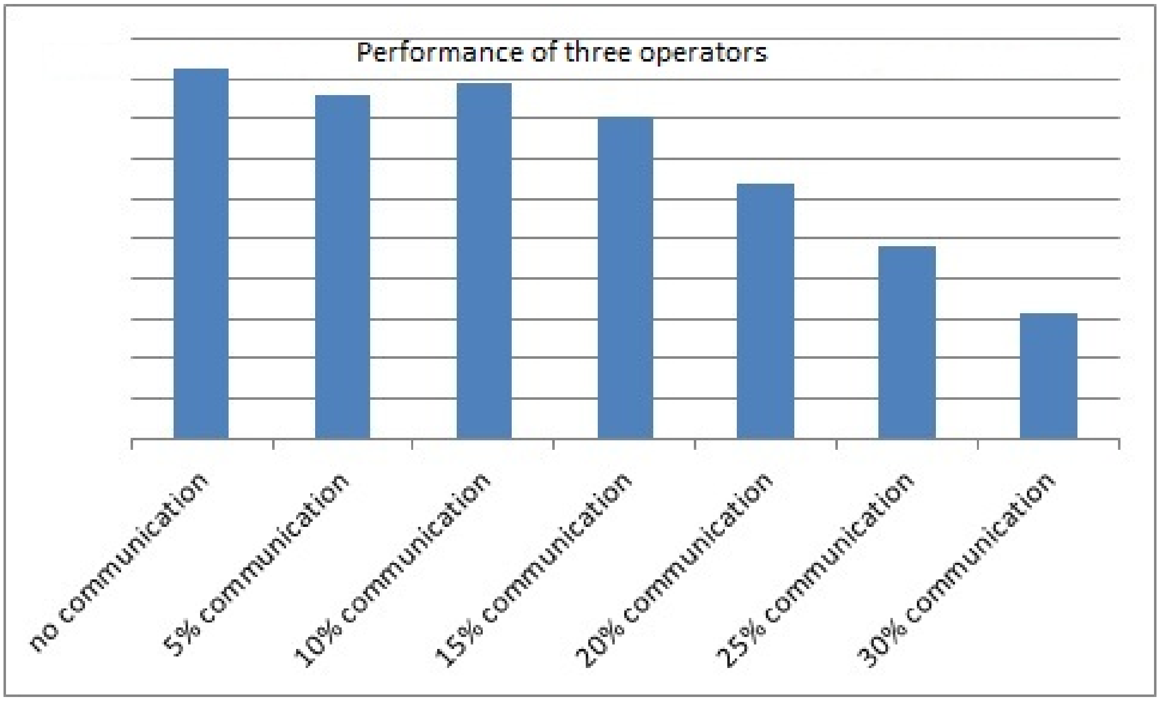


Figure 4.17: Comparison of team performance of three operators with different amount of communication.

life, this could result from various reasons, such as social loafing, difficulty in coordination, etc [20].

In the traditional Chinese culture, there is a parable. A monk lived in a temple on the top of a mountain. The only source of water was a well at the foot of the mountain. Everyday the monk would go down the mountain and haul two buckets of water for himself. Another monk moved in, so he tried to share the load with the new monk using a carry pole to carry one bucket of water in the middle. A third monk moved in, and all three of them refused to go get water since the carry pole can only be used by one or two persons; everyone wanted to do less and enjoy the work done by others.

As portrayed by the parable, peer to peer communication does not seem to work well when there are 3 operators. So is there a way communcation can help? This brings us to the next communication paradigm, commander.

4.3.2 Commander

Think of a manager in a large sized store, like Walmart. The manager does not collect cash at the counter, does not clean the floor, nor does he or she do any of the tasks that the store has other people for. Instead, what the manager does is just to manage the employees. Without the manager, each employee is completely independent and unaware of anything except for the task assigned to him or her, like a cashier would not notice that the store is running out of milk. However, a manager can effectively combine the efforts of the employees and make the entire store function as a whole, and how much time and effort the manager puts in is directly proportional to how well the store runs.

In the commander paradigm, one operator acts as the “store manager” and gives orders to his “subordinates”. The commander will have a high communication overhead since he has to gather information from other operators and formulate strategy, then relay information to each operator. Due to the ease of coordination with only one operator making executive decisions, we let the performance bonus gained by the team equal to the amount

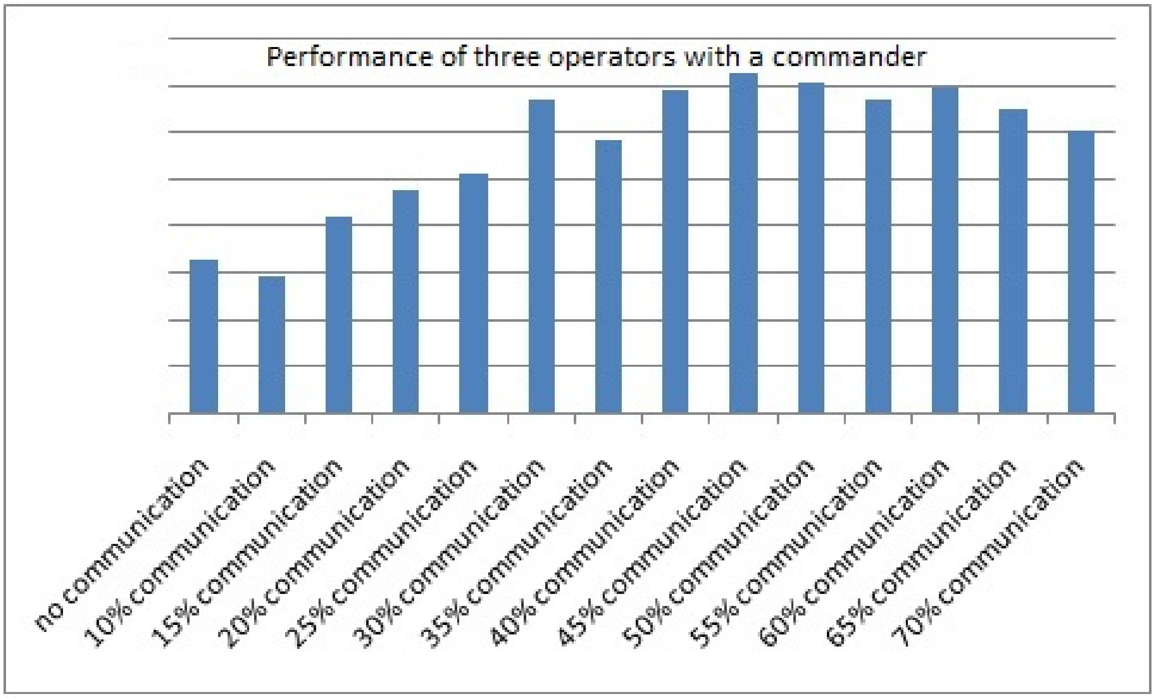


Figure 4.18: Team performance vs different amount of commander's communication. Percentages shown are the commander's communication.

of communication overhead the commander has. For example, if the commander has a communication overhead of 40%, the team performance will increase by 40%. Since the commander makes executive decisions and relays them to the other operators, the amount of communication among the subordinates do not impact team performance in this set up, so we will arbitarily set it to 10% to account for reporting and receiving instructions from the commander; this is also the value that produced the highest team performance in the peer to peer paradigm.

As shown in figure 4.18, at 10% commander communication it is just a peer to peer paradigm since every operator spent the same amount of time communicating. As the commander spends more time communicating, there are three performance peaks. The first peak is at 30%, a second one 45% which is the global maximum, and the third at 60%. The reason for these peaks is this: as the commander spends more time communicating, the team gains more performance boost from the commander; but it also causes the commander to spend less time controlling agents hence fewer agents are controlled by the commander, which means fewer agents are in the team. This delicate relationship between commander communication and number of agents controlled produces different local peaks. But to generalize, the highest performance occurs when the commander spends about 40% to 60% of his time communicating, of course, this includes gathering information, formulating strategies and relaying orders. In real organizations, this performance maxima could mean reduced mission time, less resources expended, or a reduced difficulty due to the commander's coordination.

Hence, when there are three operators, the commander paradigm produces better team performance. As more operators are added to the team, anticipation is that this "sweet spot" of 40% to 60% will shift more to the higher percentages, because a percentage gain in team performance will eventually outweigh the performance contributed to the team by the extra agents controlled by the commander if he spent less time communicating and ultimately, at a certain point it may be better to have a dedicated commander, one who does not control any agents at all but focuses on team coordination.

4.4 Specialization and How It Helps

To model specialization, we have introduced another parameter, called the "switch cost". Basically, switch cost comes into play when the operator switches from interacting with one type of agent to interacting with a different type. In the real world, different types could include an Unmanned Air Vehicle (UAV), Unmanned Ground Vehicle (UGV), Unmanned Underwater Vehicle (UUV), or even robots of similar types but of different models/LOAs. For homogeneous teams of robots, there is no switching cost involved because agent types are the same, hence their functionalities and controls are still the same. For example, an operator is assigned with two UAVs and a UGV; when the operator switches from interacting with one UAV to the other, there is no switching cost because the two vehicles are the same type, they are both flying and have the same LOA, which means they have the same control mechanisms.

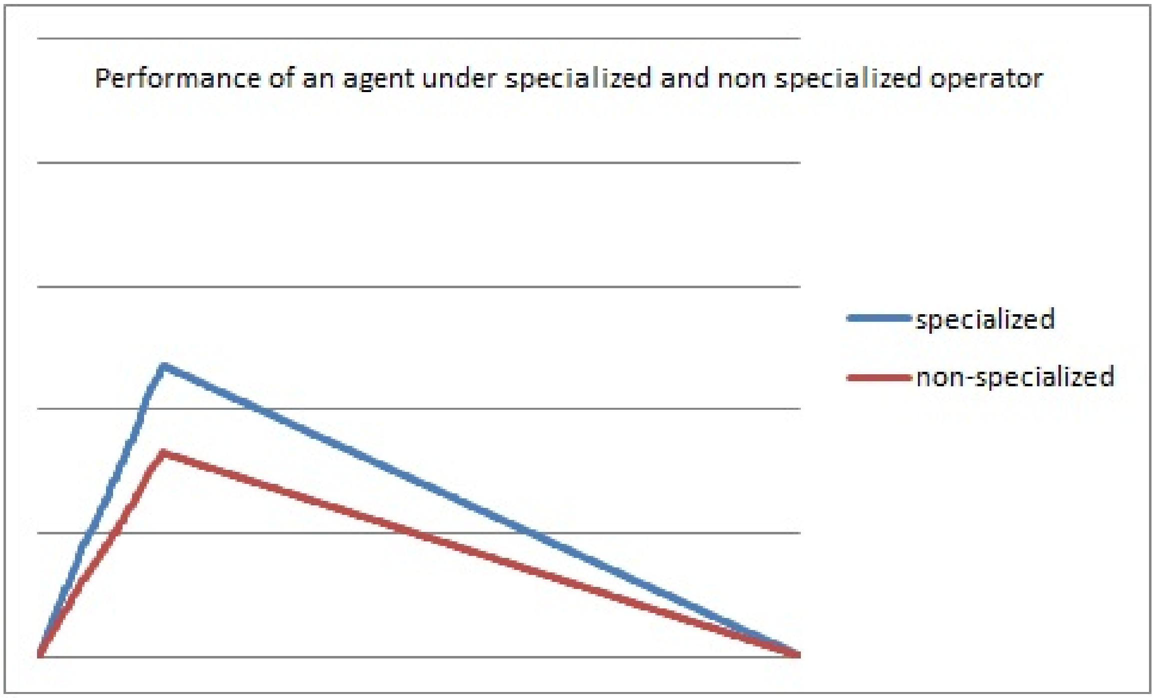


Figure 4.19: Switch cost reduces the performance gain rate of an agent during IT.

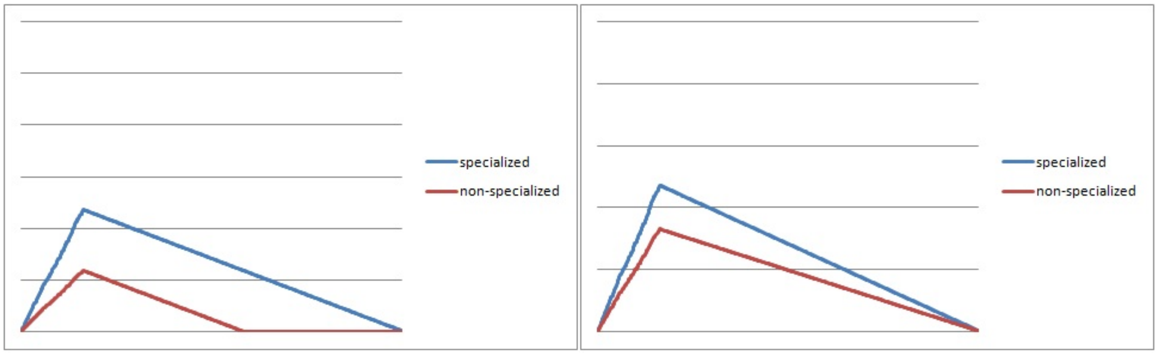


Figure 4.20: How performance could decrease during NT for agents under non-specialized operators.

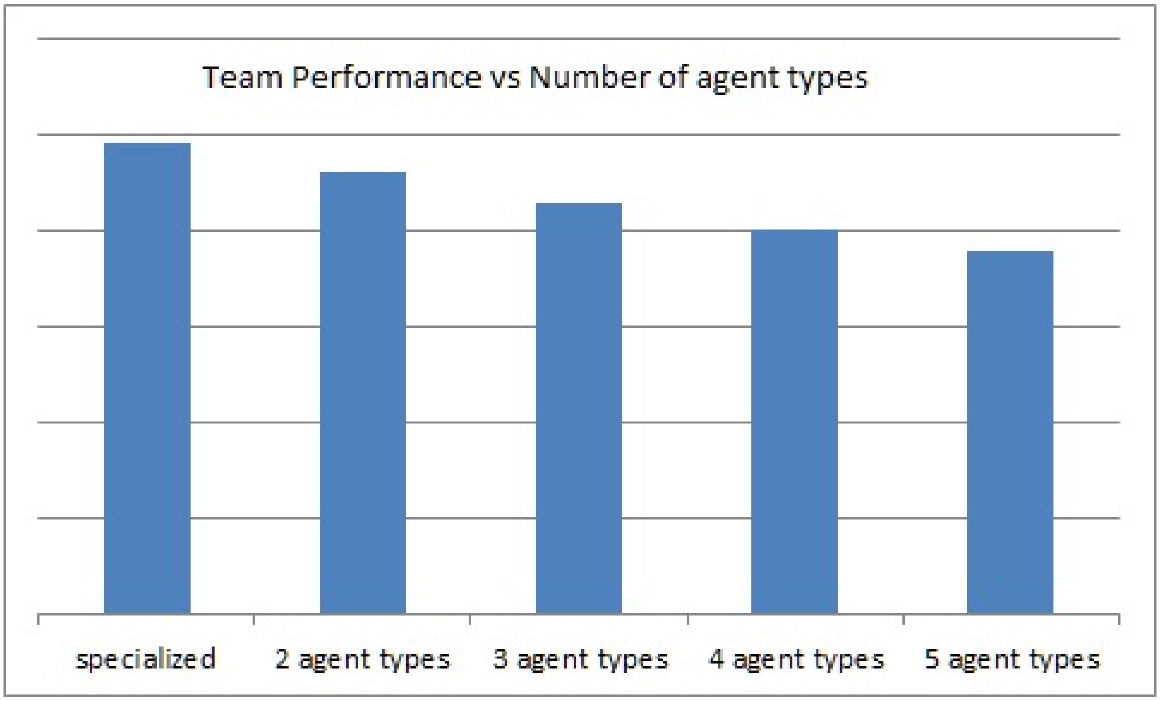


Figure 4.21: Performance of specialized vs non-specialized teams.

However, when the operator has finished interacting with the UAVs and switches to the UGV or vice versa, switching cost will reduce the efficiency with which the operator interacts with the vehicle, in other words, agent performance will increase slower during IT. This is shown in figure 4.19.

With switching cost, during NT the performance of the agents could either drop at the same rate as agents without switching cost, even though the rate of performance increase has decreased due to switch cost, or it could drop according to how much performance gain the agent has had during IT (figure 4.20); we are not terribly concerned with how performance decreases during NT here, because either way the performance of the agent with switch cost will be lower than that of the agent without switch cost (smaller area under the curve). For our purpose of modelling specialization its impact is negligible.

Now we extend this notion to the performance of a team. We will compare the performance of a control group that consists of identical agents with an experimental group that consists of different types of agents. In Figure 4.21, five agents with the same LOA are used. However, except for the specialization scenario, agents have different types. As shown in the figure, team performance decreases continually when there is an increase in the diversity of the agents, due to higher switch cost caused by more agent types. This is supported by John Whetten's study [30]. John Whetten claimed in his study that specialization helps operators to "free up" their spare capacities by lowering their workload from managing heterogeneous robots and hence increase their performance.

Chapter 5

Conclusion

In this thesis we have evaluated human robot supervisory control through the use of an agent-based model. Results show that:

- myopically maximizing fan-out does not necessarily optimize team performance; instead, it can potentially decrease team performance by overloading the operator.

When assigning agents (load) to operators, it is better to leave some room for “slack” time. During the slack time, further improvement of team performance is possible if the operator is willing to continue interacting with the agents. This promotes the maximum efficiency of the agents, the operator, and the team as a whole.

- adequate and yet moderate communication not only helps team members coordinate their activities, it also has the potential to improve team performance by efficient division of work.

When there are two operators, a peer to peer structure works just fine; it is simple to understand, easy to implement, and it produces desirable team performance. However when there are three operators, a peer to peer structure does not give superior performance than individual operators. A sociological phenomenon that produces similar result to this is social loafing. When there are three or more members in the group, one or more of them may choose to slack off and do less work. However, the result of social loafing - reduced team performance, may not just be because the team members are slacking off, it could also be due to the fact that the team structure is simply not effective and is doomed to produce inferior performance. As shown in the results, when there are three or more operators, it is better for the team to have a “commander”.

- specialization helps operators to produce higher performing teams by eliminating the cost associated with managing heterogeneous robots, and thus frees up operators’ spare capacity.

Similar to an assembly line, where each station has one task, when operators are specialized in a particular type of robot, there is no cost when switching from a robot to another of the same type. This ensures that no effort is wasted on adjusting and learning, which is another way to increase team performance.

Ultimately, this thesis has shown that there is a great potential for changes in organization and scheduling from current models, particularly current emphasis on models that seek to maximize fan-out. These changes, if properly studied and implemented, will bring human-robot interaction to a higher level with superior performance.

Limitations

This work assumes that the system is deterministic. In the agent-based model, the human operators, agents, and environment do not have randomness that will affect team performance. A sample of differences between the real world and the model are described in Table 5.1. While randomness pervades real life, speculations are that even with randomness, the results produced here will serve as a mean or average among many sets of results. Future work should validate this and should explore emergent effects of randomness.

Future Work

Future work could explore deeper into:

- more team structures, such as a “net” structure when there are many operators, each of them could also monitor his/her immediate neighbors;

Table 5.1: Assumptions About the Non-randomness of Operators, Agents, and Environments

In the Real World	**In Our Simulation**
Each human operator is unique. They have different skill sets, different amounts of motivation	Each operator has the same skill set and equal amount of motivation due to the critical nature of missions involing robots.
Human operators can get sick or have emotional issues.	Human operators do not get sick nor have emotional issues.
Robots require periodic mechnical maintenance.	Robots do not require periodic mechanical maintenance.
Robots break down.	Robots do not break down.
Bad weather can interfere with mission.	Weather does not matter.

- ways to reduce operator stress, such as visual aid, sound aid;
- more effective team communication methods;
- monotonous tasks versus variety of tasks;
- effects of talking while operation is in progress, such as talking on the cell phone; and
- effects of a noisy operating environment.

While this thesis is not conclusively on how a team should be managed, but a step in the right direction to the ultimate goal of producing the best human robot teams.

References

[1] K. J. Arrow. *The limits of organization.* WW Norton & Company, 1974.

[2] L. F. Bertuccelli, N. W. M. Beckers, and M. L. Cummings. Developing operator models for UAV search scheduling. In *Proceedings of the Guidance, Navigation, and Control Conference*, 2010.

[3] E. Bonabeau. Agent-based modeling: Methods and techniques for simulating human systems. *Proceedings of the National Academy of Sciences of the United States of America*, 99(Suppl 3):7280, 2002.

[4] J. Casper and R. R. Murphy. Human-robot interactions during the robot-assisted urban search and rescue response at the world trade center. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 33(3):367–385, 2003.

[5] J. W. Crandall, M. A. Goodrich, D. R. Olsen Jr, and C. W. Nielsen. Validating human-robot interaction schemes in multitasking environments. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 35(4):438–449, 2005.

[6] M. L. Cummings and S. Guerlain. Developing operator capacity estimates for supervisory control of autonomous vehicles. *Human Factors*, 49(1):1, 2007.

[7] M. L. Cummings, A. Kirschbaum, A. Sulmistras, and J. T. Platts. STANAG 4586 Human supervisory control implications. *Air and Weapon Systems Dept, Dstl Farnborough & the Office of Naval Research*, 2006.

[8] M. L. Cummings and P. J. Mitchell. Operator scheduling strategies in supervisory control of multiple UAVs. *Aerospace Science and Technology*, 11(4):339–348, 2007.

[9] M. L. Cummings and P. J. Mitchell. Predicting controller capacity in remote supervision of multiple unmanned vehicles. *IEEE Systems, Man, and Cybernetics, Part A: Systems and Humans*, 38(2):451–460, 2008.

[10] M. L. Cummings, C. Nehme, J. Crandall, and P. Mitchell. Predicting operator capacity for supervisory control of multiple UAVs. *Innovations in Intelligent Machines-1*, 70:11–37, 2007.

[11] M. L. Cummings and C. Tsonis. Deconstructing complexity in air traffic control. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 49, pages 25–29. SAGE Publications, 2005.

[12] M. R. Endsley. Toward a theory of situation awareness in dynamic systems. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 37(1):32–64, 1995.

[13] S. R. Fussell, R. E. Kraut, F. J. Lerch, W. L. Scherlis, M. M. McNally, and J. J. Cadiz. Coordination, overload and team performance: effects of team communication strategies. In *Proceedings of the ACM Conference on Computer Supported Cooperative Work*, pages 275–284. ACM, 1998.

[14] B. Gates. A robot in every home. *Scientific American*, 296(1):58–65, 2007.

[15] B. Gerkey and M. J. Mataric. A formal framework for the study of task allocation in multi-robot systems. *International Journal of Robotics Research*, 23(9):939–954, 2004.

[16] G. N. Gilbert. *Agent-based models.* Sage Publications, Inc, 2008.

[17] M. A. Goodrich, T. W. McLain, J. D. Anderson, J. Sun, and J. W. Crandall. Managing autonomy in robot teams: Observations from four experiments. In *Proceedings of the ACM/IEEE International Conference on Human-robot Interaction*, page 32. ACM, 2007.

[18] M. A. Goodrich and D. R. Olsen Jr. Seven principles of efficient human robot interaction. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, volume 4, pages 3942–3948. IEEE, 2003.

[19] J. Kavanagh. Stress and performance: A review of the literature and its applicability to the military. Technical report, DTIC Document, 2005.

[20] B. Latane, K. Williams, and S. Harkins. Many hands make light the work: The causes and consequences of social loafing. *Journal of Personality and Social Psychology*, 37(6):822, 1979.

[21] M. Lewis, J. Polvichai, K. Sycara, and P. Scerri. Scaling-up human control for large UAV teams. *Human Factors of Remotely Operated Vehicles*, 7:237–250, 2006.

[22] B. Mekdeci and M. L. Cummings. Modeling multiple human operators in the supervisory control of heterogeneous unmanned vehicles. In *Proceedings of the 9th Workshop on Performance Metrics for Intelligent Systems*, pages 1–8. ACM, 2009.

[23] C. W. Nielsen, M. A. Goodrich, and R. W. Ricks. Ecological interfaces for improving mobile robot teleoperation. *IEEE Transactions on Robotics*, 23(5):927–941, 2007.

[24] N. Nohria and R. Gulati. What is the optimum amount of organizational slack? a study of the relationship between slack and innovation in multinational firms. *European Management*, 15:603–611, 1997.

[25] D. R. Olsen and M. A. Goodrich. Metrics for evaluating human-robot interactions. In *Proceedings of PrivilEge and Role Management Infrastructure Standards*, 2003.

[26] D. R. Olsen Jr and S. B. Wood. Fan-out: measuring human control of multiple robots. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 231–238. ACM, 2004.

[27] T. B. Sheridan. Human and computer control of undersea teleoperators. Technical report, DTIC Document, 1978.

[28] T. B. Sheridan. *Telerobotics, Automation, and Human Supervisory Control.* The MIT Press, 1992.

[29] G. B. Thomas, R. L. Finney, and M. D. Weir. *Calculus and analytic geometry*, volume 30. Addison-Wesley Reading, MA., 1996.

[30] J. M. Whetten and M. A. Goodrich. Specialization, fan-out, and multi-human/multi-robot supervisory control. In *Proceedings of the 5th ACM/IEEE International Conference on Human-robot Interaction*, pages 147–148. ACM, 2010.

[31] Wikipedia. Supervisory control theory.

[32] D. D. Woods, J. Tittle, M. Feil, and A. Roesler. Envisioning human-robot coordination in future operations. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 34(2):210–218, 2004.

[33] H. A. Yanco and J. Drury. Classifying human-robot interaction: an updated taxonomy. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, volume 3, pages 2841–2846. IEEE, 2004.

[34] H. A. Yanco, J. L. Drury, and J. Scholtz. Beyond usability evaluation: Analysis of human-robot interaction at a major robotics competition. *Human-Computer Interaction*, 19(1):117–149, 2004.

[35] J. Zhao, M. Dessouky, and S. Bukkapatnam. Optimal slack time for schedule-based transit operations. *Transportation Science*, 40(4):529–539, 2006.