HUMAN CONTROL OF COOPERATING ROBOTS

by

Jijun Wang

B.S. in Elec-Mechanic Engineering, Tsinghua University, P.R.C., 1993

M.S. in Elec-Mechanic Engineering, Tsinghua University, P.R.C., 1998

Submitted to the Graduate Faculty of

The School of Information Sciences in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2007

UNIVERSITY OF PITTSBURGH

SCHOOL OF INFORMATION SCIENCES

This dissertation was presented

by

Jijun Wang

It was defended on

Dec 12th, 2007

and approved by

Paul Munro, Associate Professor, DIST, University of Pittsburgh

Michael Spring, Associate Professor, DIST, University of Pittsburgh

Paul Scerri, Systems Scientist, Robotics Institute, Carnegie Mellon University

Katia Sycara, Professor, Robotics Institute, Carnegie Mellon University

Dissertation Advisor: Michael Lewis, Professor, DIST, University of Pittsburgh

HUMAN CONTROL OF COOPERATING ROBOTS

Jijun Wang, Ph.D.

University of Pittsburgh, 2007

Advances in robotic technologies and artificial intelligence are allowing robots to emerge from research laboratories into our lives. Experiences with field applications show that we have underestimated the importance of human-robot interaction (HRI) and that new problems arise in HRI as robotic technologies expand. This thesis classifies HRI along four dimensions – human, robot, task, and world and illustrates that previous HRI classifications can be successfully interpreted as either about one of these elements or about the relationship between two or more of these elements. Current HRI studies of single-operator single-robot (SOSR) control and single-operator multiple-robots (SOMR) control are reviewed using this approach.

Human control of multiple robots has been suggested as a way to improve effectiveness in robot control. Unlike previous studies that investigated human interaction either in low-fidelity simulations or based on simple tasks, this thesis investigates human interaction with cooperating robot teams within a realistically complex environment. USARSim, a high-fidelity game-engine-based robot simulator, and MrCS, a distributed multirobot control system, were developed for this purpose. In the pilot experiment, we studied the impact of autonomy level. Mixed initiative control yielded performance superior to fully autonomous and manual control.

To avoid limitation to particular application fields, the present thesis focuses on common HRI evaluations that enable us to analyze HRI effectiveness and guide HRI design independently of the robotic system or application domain. We introduce the interaction episode (IEP), which was inspired by our pilot human-multirobot control experiment, to extend the Neglect Tolerance

model to support general multiple robots control for complex tasks. Cooperation Effort (CE), Cooperation Demand (CD), and Team Attention Demand (TAD) are defined to measure the cooperation in SOMR control. Two validation experiments were conducted to validate the CD measurement under tight and weak cooperation conditions in a high-fidelity virtual environment. The results show that CD, as a generic HRI metric, is able to account for the various factors that affect HRI and can be used in HRI evaluation and analysis.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

AA – Adaptive Autonomy

CD – Cooperation Demand, the percentage of time spent in controlling relevant robots while a operator neglects a robot

CE – Cooperation Effort, the extra cooperation effort required for a robot

CSCW – Computer-Supported Cooperative Work

FO – Fan-Out, the maximum robots (or robot subsets) that one operator is able to control

FT – Free Time, the off-task time in robot control

HCI – Human-Computer Interaction

HRI – Human-Robot Interaction

HMRI – Human-Multi-robot Interaction

IE – Interaction Effectiveness

IEP – Interaction Episode, the period of time during which the operator interacts with the robotic system, which may be a team or one or more individual robots, to pursue a sub-task

IEP model – an extended NT model that measures and analyzes human's coordination behaviors in robot control

(IEP,NT) – a pair of interaction episode and neglect time

IT – Interaction Time, time spent in actively controlling a robot

(IT,FT) – a pair of interaction time and free time

(IT,NT) – a pair of interaction time and neglect time

LOA – Level Of Autonomy

MrCS – Multi-robot Control System

NASA-TLX – NASA Task Load Index, a workload evaluation approach

NT – Neglect Time, the time during which an operator ignores a robot

NT model – Neglect Tolerance model

ODE – Open Dynamic Engine, an open source physic simulation engine

OT – Occupied Time, time occupied by coordinating relevant robot(s)

PTP – Point To Point, a robot control style

RAD – Robot Attention Demand, the attention demand from a robot

RFT – Relative Free Time, the fraction of the task time that a operator can relax without paying attention to a robot

ROI – Region of Interest, a robot control style

SA – Situation Awareness

SART – Situation Awareness Rating Technique, a SA evaluation approach

SAGAT -- Situation Awareness Global Assessment Technique, a SA evaluation approach

SOSR – Single Operator Single Robot control

SOMR – Single Operator Multiple Robots control

SWAT – Subjective Workload Assessment Technique, a workload evaluation approach

TAD – Team Attention Demand, the total attention demand from controlling a group of robot

TOL – Teleoperate and Landmark, a robot control style

UI – User Interface

USAR – Urban Search And Rescue

USARSim – Urban Search And Rescue Simulation

WA – Workspace Awareness

WP – Workload Profile, a workload evaluation approach

WTQ – Queuing Wait Time

WTSA – wait time caused by a loss of SA

1.0 INTRODUCTION

The advances in robotic technologies and artificial intelligence allow robots to emerge from research laboratories into our lives. Roomba is available in Home Depot to clean our floor automatically. Kids play with Robosapien, the humanoid toy robot, to entertain themselves at home. Talon is being used in Afghanistan to explore bombs. Field robots were deployed at the World Trade Center site (2001) and in New Orleans (2005) to search for victims. These and other uses of robots in recent years show that we have underestimated the importance of human-robot interaction (HRI) and that new problems arise in HRI as robotic technologies expand. For example, in the Roomba and Robosapien applications, in which the robot and the human share the same space, building social interaction became a new problem for the field. When robots were used at the World Trade Center site (2001) to explore the rough terrain for victims, situation awareness was found to be more critical than previously thought [14]. Other field studies have shown that robot autonomy is not always helpful if the human operator does not trust the robot [56, 86]. Although a robot team may improve task performance, it will require the operator to maintain a more complex situation awareness to shift between an individual robot and the robot team [37, 47, 78, 85].

The interaction between a human and a robot is usually rich, complex, and concrete because the robot is a situated agent that lives in and interacts with a dynamic environment in imperfect and unreliable ways. The noisy sensors and effectors, narrow communication bandwidth, limited data processing capabilities, and other characteristics are obstacles to the goal of building an efficient robotic system [65, 88]. The frontier of HRI study is to extend the single-operator-single-robot (SOSR) interaction to single-operator-multi-robot (SOMR) control. Cooperating robot teams have emerged in recent years because of the advances in multi-agent technologies. In this paper, we are interested in studying the new HRI problems that arise in human interactions with cooperating robot teams.

In the remainder of the introduction, we describe the scope of our study and then compare HRI with human-computer interaction (HCI) to further clarify the study that we present in this proposal. Finally, we give the overview of the paper.

1.1 THE FOUR ELEMENTS OF HRI STUDY

The content of HRI is very broad because of the complex interactions that exist between the robot, the human, and the working environment. Classifying HRI will help us to identify the scope and to better understand the content of HRI. [91] attempts to identify and classify the content of HRI in terms of its five application domains, and the numeric, spatial, and authority relationships between the human and the robot. [119, 120] propose 11 taxonomy categories in order to include all possible classifications of HRI. We believe that there are four essential elements in HRI: human, robot(s), world, and task. Along these four dimensions[1], we are able to systematically identify the HRI categories. All of the above classifications can be interpreted as

[1] Time is the fifth dimension. Because of its obviousness, we ignore it here.

either about one of these elements or about the relationship between two or more of these elements.

1.1.1 Human

The human in a human-robot team can be an operator who controls the robotic system or a person who implicitly affects the robotic system as a decision-maker, investigator, or communicator. In terms of personal skill, background knowledge, and experience, the human in an HRI system can be a novice, trained person, or expert. Defining these characteristics of the human element in HRI is critical because these characteristics can significantly impact the human-robot system. For example, [1] shows that many HRI systems have failed because the human element of these systems' designs was based on the roboticist instead of the potentially novice end user. Furthermore, when more than one person is involved in a system, the humans become a human group. If the human group cooperates among themselves, they compose a human team that shares the same team goal. For example, in the army, a team of soldiers typically control one unmanned aerial vehicle (UAV).

1.1.2 Robot

There are many definitions of a robot. It can be software that responses to a user, a vehicle that is controlled remotely by the operator, a mechanical device that performs manufacturing tasks, or any other program or object that has some degree of autonomy. In this thesis, without special declaration, we restrict the definition of a robot to a mechanical device that directly interacts with the workspace.

In terms of locomotive features, we further define a robot as in a fixed position or possessing the ability to move around. From the perspective of robot morphology, a robot can have an anthropomorphic (human-like) or zoomorphic (animal-like) appearance or simply a functional appearance [120]. The desired workspace of the robot can be ground, aerial, space, or nautical. In describing the autonomy level that specifies the desired level of human intervention with the robot(s), we follow Sheridan and Verplank's (1978) autonomy level spectra. [72] lists ten autonomy levels that range from teleoperation (fully manual control performed by a human) to mixed or shared control to full autonomy of the robot(s). These ten autonomy levels appear below.

1. The human has full control.
2. The computer suggests all possible alternatives.
3. The computer selects from all possible alternatives to suggest only a few.
4. The computer suggests one recommended alternative.
5. The computer executes the alternative if the human approves.
6. The computer executes the alternative, which the human can veto.
7. The computer executes the alternative and informs the human of the execution.
8. The computer executes a selected alternative and informs the human only if asked.
9. The computer executes a selected alternative and informs the human only if it decides to.
10. The computer acts entirely autonomously.

One or more robots can be involved in a human-robot team. Similarly to humans, robots can be formed in groups or teams. If all the individuals are the same type, then they construct a homogeneous team. Otherwise, they construct a heterogeneous team that usually has a higher cooperation requirement. The size of the team has a significant impact on team cooperation and control as well. Approximate ranges of team size include small (four or fewer robots), medium (four to 20 or 30 robots), and large (20 to 30, or more, robots). When a huge robot team is constructed of homogeneous robots with simple functions, it is a robot swarm that possesses superior capabilities as well as a special requirement in cooperation and control. Robot teaming is currently a hot topic that is related to multi-agent technologies. From the view of multi-agent systems, other classifications of robot teams exist, for which the criteria can be types of communication, cooperation, and organization, are listed in [26, 27, 34].

1.1.3 Task and world

Robots and human organize and function together to achieve a task. [91] summarizes five task domains, which include search and rescue, personal assistance, museum docent, robot fleets, and physical therapy. [105] enumerates five task domains as navigation, perception, management, manipulation, and social. According to the information processing and functional framework of a robotic system [3, 87], an HRI task is comprised of four essential sub-task components: mobility, perception, planning, and social. These components vary from a low abstraction level to a high abstraction level. The mobility sub-task involves mechanical motions, such as the locomotion of the robot or the manipulation of a robot's arm or a target object. The perception sub-task involves information acquisition with the goal of understanding the current situation; for example, a robot or robot team can help to map an unknown environment or to find targets in a building. The planning sub-task involves making decisions for the future, such as planning a path for the robot or coordinating two robots to solve a conflict. The social sub-task is the highest-level sub-task and involves maintaining a social relationship. Entertaining the human player or helping the human operator to trust the robotic system is example of social sub-task. This higher-level sub-task is usually based on a lower task. For example, trust is based on the robot's predictable behaviors, planning is based on perception, and perception is based on the robot's locomotion during which sensory data is collected. Although each sub-task can be distributed between the human and the robot, from a low to a high level of abstraction, the human tends to take more responsibility. Figure 1 illustrates the sub-task types listed in [91] and [105]. For example, an urban search and rescue (USAR) task should require more human involvement than a navigation task because the former involves higher requests in the perception and planning sub-tasks. Moreover, in practice, identifying a victim relies heavily on a human's input because of current limitations in pattern recognition. Therefore, the overall effect of the sub-tasks' allocations between a human and a robot makes USAR a typical HRI task because of the human's necessarily deep involvement.

In addition, based on extrinsic task characteristics, we can classify tasks according to their significance, urgency, frequency, risk, and reward. For instance, the classification of a task according to its frequency yields the categories of unique, periodical, and routine. The classification of a task according to its urgency yields high, medium, and low critical task levels [120].

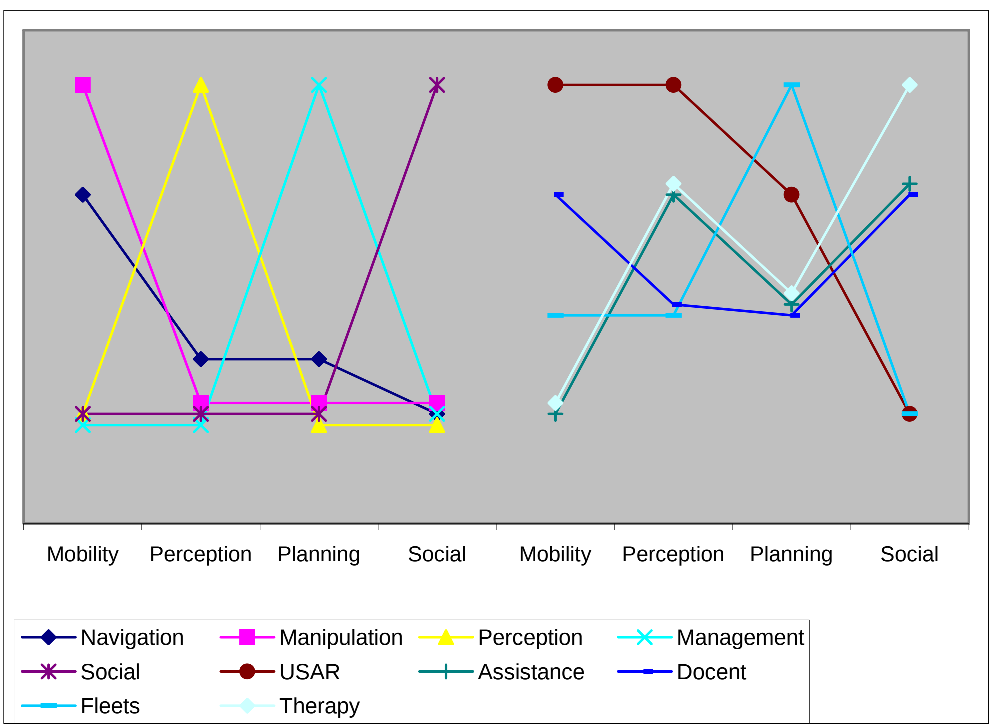


Figure 1. The tasks among the four basic sub-tasks.

Although the workspace usually reflects the task, we cannot simply combine them. For instance, a navigation task in an office-like environment, over rough terrain, or in a forest-like environment will present different challenges to the human-robot team and therefore require different robotic systems and human-robot interactions. [19] estimates world complexity from the branch factor and the clutter of the workspace. [73] measures rough terrain in terms of traversability with respect to the robot's coverability and crossability. [101] characterizes the debris field in disaster environments. In addition to these terrain features, the world can be open or closed and static or dynamic. Features of the ambient environment can characterize the world as light or dark, cold or hot, and clear or dusty, among other variables. For example, during the rescue activities at the World Trade Center, [14] reports that the type of weather (i.e., high temperatures and rain) and noise had a significant impact on the robot-assisted search and rescue.

1.1.4 The relationships

The relationships among the human, the robot, the task, and the world constitute the interactions in HRI. In this Section, we describe three main relationships of human-robot, human-robot-task, and human-robot-world.

Human-Robot

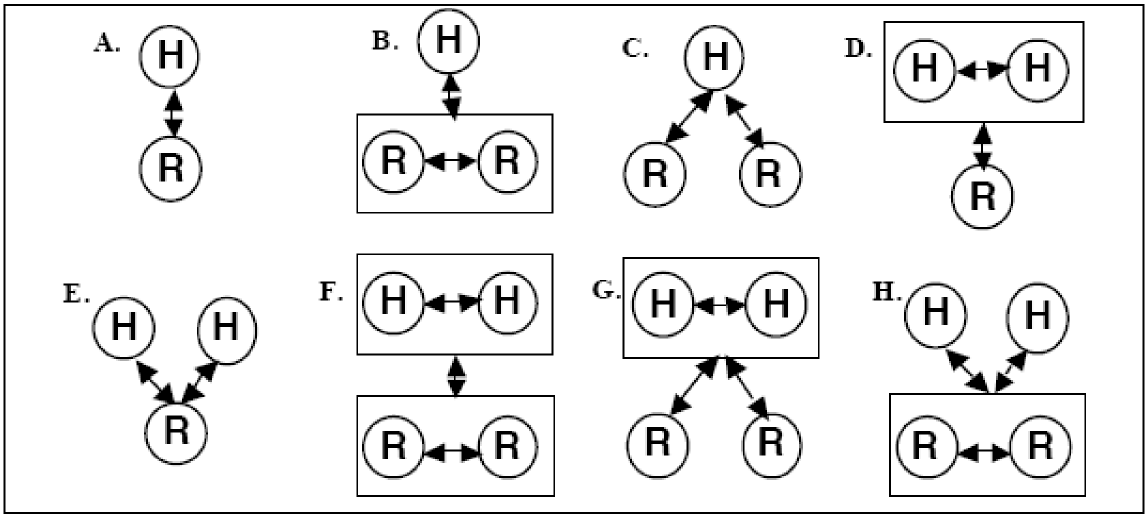


Figure 2. The level of shared interaction among teams (reprinted from [120]).

Ignoring the task and the world, the relationship between the human and the robot includes the numeric relationship, called *human-robot ratios*, that describes the number of people involved in controlling a certain number of robots. The ratio is the number of people to the number of robots. Possible ratios include one-one, one-many, many-one, and teams-teams, according to [91]. It can also be a range when the number of people or the number of robots varies in the control process. [120] further classifies the human-robot relationship as a *level of shared interaction among teams* to describe the interaction between humans and robots at both the team and the individual levels. Figure 2 is an example taken from [120] that lists all possible relationships when the human and robot team sizes are less than two. Cases A through H represent the one-one, one-team, one-many, team-one, many-one, team-team, team-many, and many-team relationships, respectively.

In addition to classifying human-robot relationship in terms of the existing interactions between human and robot, they can be classified according to the type of interaction. *Interaction scheme* refers to the particular combination of autonomy and interface that characterizes how the human operator affects the robotic system, i.e., the interaction style [19]. The types of interaction include teleoperation [44, 59], point-point (waypoint) [14, 35], scripted [19], region of interest [79], and delegation [85]. [120] identifies the interactions in terms of the information flow via *decision support for operators*, which classifies the available sensors, provided sensors, sensor fusion, and pre-processing of the sensors.

Human-Robot-Task

Humans can play different roles in the possible relationships among the human, the robot, and the task. [96] defines a set of five roles:

Supervisory: The human or human team monitors the robot or robot team and changes the plan when necessary.

Operator: The human or human team directly interacts with the robot or robot team to change its behaviors.

Mechanic/programmer: The human or human team physically intervenes with the robot or robot team to change its capabilities through modifying hardware or software.

Peer: The human or human team works with the robot or robot team to perform a task. Usually a peer interaction between human and robot occurs at a high level of behavior.

Bystander: The human or human team is not directly involved in controlling the robot or robot team, but affect how the robot or robot team accomplishes the task. For example, humans in the same building can affect how a robot navigates the environment by blocking the robot or opening or closing a door.

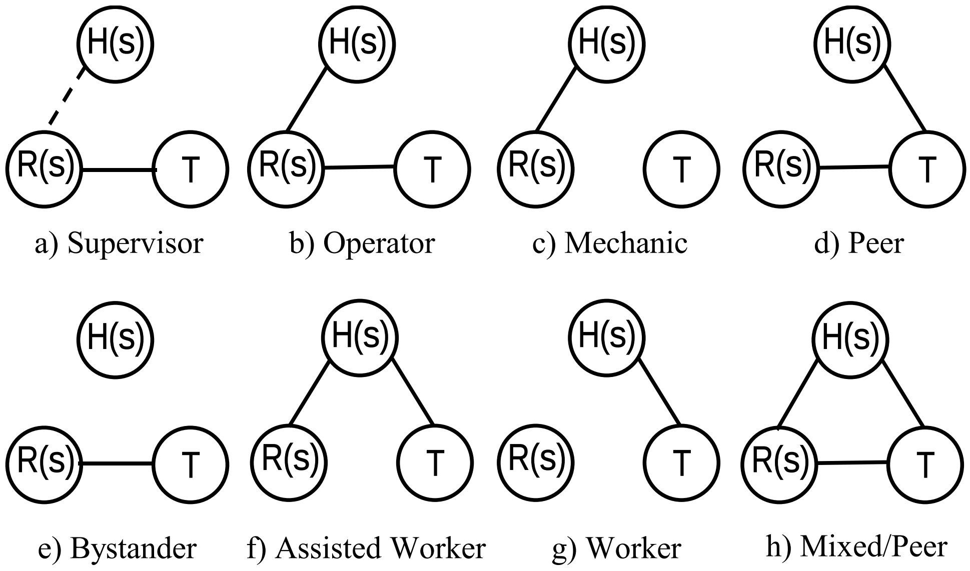


Figure 3. The human-robot-task relationships.

Figure 3 lists the possible combinations of relationships. The relationships represented by (a) through (e) correspond to the above five roles. In this figure, cases (a) and (b) can be merged together to represent a human or human team who works as a controller. In case (d), a human or human team and a robot or robot team work together to perform a task; however, no direct human-robot interaction occurs. In case (h), a human or human team, a robot or robot team, and the task are in direct interaction with each other. In case (f), a human or human team perform a task and a robot or robot team provides assistance; for example, an expert robotic system can help a doctor diagnose a patient. Case (g) involves a robot or robot team that exists as a bystander and that may implicitly affect a human or human team.

Human-Robot-World

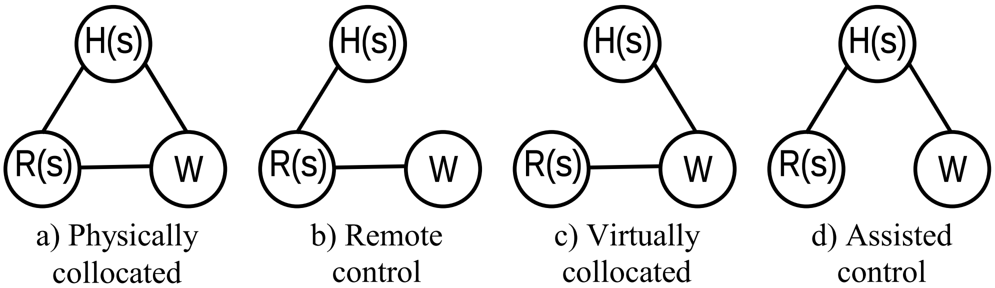


Figure 4. The human-robot-world relationships.

There are three fundamental, direct relationships between the human, the robot, and the world: human-robot, robot-world, and human-world. Figure 4 lists four meaningful combinations of these relationships. Case (a), or *physically collocated*, represents the situation in which the human and the robot exist in the same world. Both the human and the robot directly affect the world and can interfere with the other's actions to the extent that a social relationship between the robot and the human exists. For example, the robotic museum docent will interfere with the visitors such that both the robot and the visitors change the environment. Case (b), or *remote control*, describes the situation in which the human and the robot exist in different worlds. The human perceives the world through the robot's sensors and interacts with the workspace via the robot. For example, a robot can be deployed in a disaster environment where a human is not allowed to enter but can control the robot's search in the environment. Case (c) represents the situation in which both the human and the robot can directly change the world but cannot directly interact with the other. For example, a human can play chess with a robot where the human and the robot affect each other only implicitly through the world of the chessboard. Case (d) represents the situation in which the robot does not exist in the workspace. The robot affects the world via a human's interaction with the workspace. The decision support robot is an example of this type of situation.

1.1.5 The scope of the present study

In this thesis, we will focus on one person who remotely controls multiple mobile ground robots. The person can be a supervisor or an operator. The robots are cooperating ground robots that construct either a homogeneous or a heterogeneous robot team. Navigation and perception are the primary tasks that the human and robot team will perform; we will ignore management, manipulation, and social tasks at present. We are particularly interested in urban search tasks because the disaster environment presents many rich and complex challenges to HRI in which the human is heavily involved in robot control. The world will be an indoor disaster environment with even or rough terrain. In our study, human and robots exist in different worlds. The operator must rely on the robots' feedback to perceive the workspace and to control the robots.

1.2 HRI VS. HCI

HRI can be treated as a subset of human-computer interaction (HCI) and computer-supported cooperative work (CSCW) [25, 119]. Many research results in HCI can be applied to HRI research. However, according to [96], "Human-robot interaction is fundamentally different from typical human-computer interaction." [37] notes that the complex and dynamic features in both the control system and the real world, as well as the autonomy and cognition model embedded in a robotic system, distinguish HRI from HCI. [96] lists six ways that HRI is different from HCI. First, the object that the human is interacting with is different. As mentioned above, a robot is a situated agent that makes an interaction more complex. A robot dynamically interacts with a world according to its own "world model"[2]. This "world model" must be represented clearly to any human who will interact with the robot. The differences between the real world, the robot's world model, and the human's world model, as well as the difference between the robot's world model and the human's understanding of the robot's world model, pose significant failures in and challenges to HRI. These differences cause, for example, situation awareness problems and social interaction problems, including issues of emotion and trust, that are new to the typical HCI. The robot platform interacts with the environment in imperfect ways, introducing into interactions an uncertainty that is rare in HCI. Examples of this uncertainty in HRI include an incorrect moving distance, a broken robot arm, and a degraded sensor. Humans must live with such errors to interact with robots. On the other hand, in HRI, the workspace is the real-time world of physics, which may change from time to time. Unlike the typical HCI, the results of HRI are not constant. The real world does not pause, yielding time-related problems in HRI like delay and synchronization. Moreover, damages may occur that force the operator to respond to the incident and that introduce stress into the interaction. The HRI operator may face a range of working conditions, such as a dark, dusty, or underwater environment.

In addition to the dynamic and uncertain features of the robot platform and the world, the degree of robot autonomy in HRI and HCI differ. The degree of robot autonomy changes the role that a human or human team will play in the interaction as well as how deeply a human or human team will be involved in robot control. The ten autonomy levels and five possible roles described in previous Sections require different types of information (abstract or concrete data), control styles, intervention frequencies, and workloads in HRI. Furthermore, the roles played by humans

---

[2] For example, the robot uses its own map to move around or uses range data to navigate in a building.

and a robot's degree of autonomy may change. These present both advantages and challenges to HRI.

Finally, in HRI, it is possible for a human to control multiple robots simultaneously. During such interactions, the human must handle issues such as cooperation, conflict, interference, and competition among the robots. When the robots work as a team, the operator must maintain team awareness and possibly individual awareness as well. In typical HCI, however, one user usually interacts with only one computer. Furthermore, in HRI, humans may work as a team to control one or more robots. For example, in the World Trade Center (2001) rescue activity, two people controlled one robot [10]. The U.S. Army plans to have multiple soldiers control multiple UAVs and UGVs in the battlefield of the future.

1.3 OVERVIEW

The remainder of the dissertation begins with a background chapter that introduces the current HRI field's interest in a single operator controlling a single robot. We then explore the current effort in extending HRI to a single operator controlling multiple robots. A summary and the potential questions in human multi-robot control will be discussed. We next introduce our efforts in the study of multiple robots. At the outset, we built an original high-fidelity HRI-oriented robot simulator to provide a cheap and realistic HRI testing bed. Then, based on a multi-agent architecture, a flexible and scalable multi-robot control system was developed to support the control of cooperating robots. With the simulator and robotic system, we conducted a pilot experiment to study the impact of autonomy in human-controlled cooperating robots. Finally, we propose the interaction episode methodology, which allows us to study the interaction between humans and cooperating robots as well as the related validation and study experiments.

2.0 BACKGROUND

Single-operator-single-robot interaction has been studied extensively in recent years. This research falls into three main areas: how user interface improves effectiveness in HRI, how the robotic system's capability benefits HRI, and the metrics that can be used to evaluate and guide HRI design. The field studies [10, 11, 14, 97], the controlled experiments [18, 44, 64], and the robot competition activities [24, 99, 121] demonstrate that most of the problems in HRI connect with situation awareness (SA), which emphasizes the robot as a situated agent. This study in SA plays an important role in HRI, so we begin with SA and then proceed to discuss three other areas in HRI. Finally, we discuss the challenges that we will face when we shift from single-robot control to multi-robot control.

2.1 SITUATION AWARENESS

2.1.1 Definition

Situation awareness is generally defined as "the perception of the 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" [28].

This definition implies three levels of SA. At the level of perception (Level 1 SA), people are aware of the elements in the environment. For example, in robot driving, we may see the range data, a hallway from the video feedback, and a speed meter on the interface. At the level of comprehension (Level 2 SA), we synthesize these data to form an overall understanding of the environment in terms of what is happening at the current time. For example, we can understand that the robot is moving down the hallway at a high speed and not quite pointing at the center of the hallway. At the level of projection (Level 3 SA), we apply our knowledge and comprehension of the current situation to predict the future status of the environment. Continuing the previous example, we can predict that, even though the robot points only slightly off-center, its high speed will cause it to bump into the wall in the near future. Through situation awareness, we are able to decide either to slow down the robot so that we can finish another task before intervening with it or to adjust the robot's direction immediately.

2.1.2 The SA errors

[29] discusses the potential errors that can result from incomplete or incorrect SA. For Level 1 SA, incomplete SA may occur when we lack data about an element or when we provide data about an element but fail to represent it in a noticeable or distinguishable way. For example, in the RoboCup 2003 USAR competition, [99] reports that one of the final teams with two DOF cameras was able to view the robot's front wheel and thereby explored wider areas because of the better SA of location and surroundings. In contrast, the teams with fixed front cameras had more difficulty in exploration because of the comparative lack of SA. Another team used a 360-degree omni-camera to aid navigation, but the distorted image made the obstacle very difficult to distinguish; the omni-camera ultimately provided no help in local navigation, although it showed

better performance in one run. In addition to reasons of the lack or improper presentation of information, incomplete SA can arise from human limitation in sampling data, paying attention, and sharing attention among tasks. This significantly impacts SA when several people control multiple robots. Incorrect SA occurs when people misinterpret data. For example, robots rolled over in the attitude experiment [59] and field study in Sandia [69] when crossing desert-like terrain because operators underestimated the robots' attitudes from video and meters.

For Level 2 SA, failures in understanding the current situation mainly arise from lacking a mental model, having an incorrect mental model, or using the mental model improperly. In the field study of the Fire Chief controlling robot, [121] reports that when the Chief controlled a robot, which was working under safe mode, he continuously drove the robot forward when the robot stopped because of the self-protection function. In this example, lacking the mental model of the robot control mode caused a misunderstanding of the situation. Another example is reported in [44] in which users were allowed to pan and tilt the camera while driving the robot. The user interface used a clock-like meter to represent the camera's pan angle and the robot's heading direction. Although the users were instructed in how to interpret the pose meter, they still became confused in robot driving from time to time for one or both of two reasons: (1) the user failed to build a mental model of the pan/tilt camera and (2) the user had built a mental model for both the fixed camera and the pan/tilt camera, but had selected the wrong model in some situations[3]. [121] reports another type of Level 2 SA error caused by incorrectly using mental model in the field study of the Fire Chief controlling robot. In one of the trials, the Chief

[3] This was reported as the user becoming confused and then resolving the confusion by centering the camera so that he could use the fixed camera mental model or by correctly controlling the robot by shifting to the pan/tilt camera mental model.

thought that the robot was caught on a cable that did not exist. When this mismatch occurred, the Chief had failed to use the mental model correctly.

Level 3 SA is based on a highly developed model to predict future situations. Similar to the Level 2 SA errors, the lack of a mental model, the use of an incorrect mental model, and the improper use of a correct mental model will cause failures. For instance, the rollover incidents reported in [59, 69] occurred when the operator continued to drive the robot although the robot was in a dangerous attitude. Usually Level 3 SA requires more cognitive resources and longer response times. When a person is under high stress or must distribute mental resources among tasks, the person's SA will likely suffer. In the simulated experiment [110], the users had no difficulty in driving a single robot. However, when they were asked to control two or four robots simultaneously, they responded to the robotic system instead of proactively supervising. These users' SA decreased due to the limitations on their cognitive resources.

2.1.3 Measurement

Measuring SA is very difficult. Three types of measurements are commonly used [118].

The subjective measure asks the subject evaluate his own SA by answering a questionnaire. This approach is straightforward but, as the name reveals, the result is subjective. SART (Situation Awareness Rating Technique) is one of the most accepted examples of the subjective measure. SART was originally designed to assess pilots' SA in terms of the attention demand, the supply of attention from the system, and the pilots' understanding of the situation. One experiment shows that SART strongly correlates with the subject's confidence level as well as performance [31].

The implicit performance approach measures SA in terms of task performance. It assumes that a strong correlation between SA and performance exists and that improved SA always leads to better performance. This approach provides objective measurement. However, the assumption of correlation may be violated during the measurement test. SA is not the only factor that impacts task performance. Decision-making, the complexity of the world, the robotic system's failures, the human operator's physical fatigue, and other factors can affect task performance as well [97]. For example, in the attitude experiment, abnormal performance occurred in a specific region although the authors believed the users' SA should have been higher [59]. In practice, subjective and implicit performance approaches are usually used together to compensate for the other's drawbacks.

The explicit performance approach directly probes SA by temporarily suspending the task and asking questions designed to measure SA. SAGAT (Situation Awareness Global Assessment Technique) is the most widely used approach that directly measures SA in a simulation scenario. During a task into which SAGAT has been integrated, the simulation periodically freezes the task and blanks all the visual displays. A series of questions designed to facilitate deep cognitive task analysis are provided to the operator to evaluate his three levels of SA, such as another robot's location or whether a robot will block a path [31]. The explicit performance approaches are usually conducted in a simulated environment because of the requirement of task freezing. Moreover, although the explicit performance approach can objectively measure SA, the process of task freezing and answering questions during task implementation can change the task.

2.1.4 SA in HRI

In general, for HRI involving multiple people and multiple robots, SA encompasses human-robot, human-human, robot-human, robot-robot, and humans' overall mission awareness [24].

Human-robot awareness involves the human's understanding of the situation of the robot and its environment, which includes the robot's states, intentions, actions, local environment, global environment, and environmental events. The examples given above to demonstrate SA errors are violations of human-robot awareness.

Human-human awareness is needed when multiple people work together to perform a task and involves the understanding of another person's states, activities, and the surrounding environment and events. Sharing SA among operators can significantly improve HRI performance. For example, [10] reports that, when two operators controlled one robot, they communicated with each other to share SA and therefore to improve SA; these operators were able to find nine times as many victims than a single operator in an USAR (Urban Search And Rescue) task. In a similar study, the data analysis shows that more communication led to improved SA and a greater number of discovered victims [11].

Robot-human awareness involves the robot's knowledge about the human or human team and allows the robot to adapt itself to the operator or to cooperate more effectively with a team of people. For instance, robot-human awareness allows a robot to execute a default command instead of wasting time while waiting for a temporarily unavailable operator to intervene. [109] proposes a perspective-taking approach that allows the robot to take the human's perspective in order to effectively collaborate on a task. Although the authors did not connect perspective-taking and SA, this approach implies a robot-human awareness based on robot behavior control. The cognitive model described in [109], Polyscheme, in fact generates the three levels of SA.

Robot-robot awareness is the robot's knowledge about another robot and its environment, which includes states such as location, speed, actions, and plans as well as the surrounding environment. This awareness is useful when the robot interacts or cooperates with other robots. The main difference between robot-robot awareness and types of human-involved awareness is that a robot can directly receive another robot's information via data exchange without data loss or distortion if we assume a perfect data connection. Any SA exchange involving humans is more subtle and usually difficult. To share SA between robots and humans is one of the main challenges in HRI.

The humans' overall mission awareness involves the understanding of the overall goal of the task and the progress in reaching the goal—that is, the joint awareness of human, robot, task, world, and time. With overall mission awareness, the operator(s) can change the plan or strategy to keep the task in progress.

In terms of the four elements in HRI, human-world and robot-world awareness necessarily involve people's and robots' understanding of the environment. However, because the world interacts with both the human and the robot, we include world awareness within any given robot's or human's awareness. Therefore, we omit human-world and robot-world awareness in the above list.

In HRI, when SA evaluation is used as a tool in usability or system design studies, researchers usually use implicit performance measures to assess SA in terms of overall performance parameters, such as task completion time and the number of collisions [44, 51, 53, 59, 80]. Subjective measures are also used to support the implicit performance measure [44, 59]. However, in the SA analysis, post-data analysis is the main technique that allows us to examine the details. By coding the video tap, [25, 98, 99] use short situation reports to measure the match between an operator's SA and the actual situation. Since incidents usually occur when an SA error occurs, critical incident analysis is employed in these studies. Based on the involved sub-tasks in the incident, researchers have analyzed five categories of incidents in USAR: local navigation, global navigation, obstacle encounter, victim identification, and vehicle state. [118] uses similar approach to measuring SA but, instead of analyzing incidents, the authors measured the time spent on sub-tasks and used the "think-aloud" approach to attain the real-time subjective SA assessment. In [97], a SAGAT-based approach was used to measure the SA in ground vehicle driving.

USAR is a typical HRI task in which the human is deeply involved in robot control. The use of USAR in the disaster environments of the World Trade Center, Hurricane Charley, La Conchita mudslide as well as in RoboCup 2003 and the AAAI competition in USAR provide us with rich data to analyze SA in HRI. Based on the their USAR practice and filed exercises, [75] finds that the operators spent significantly more time on building or maintaining situation awareness than on navigating the robot. SA, not autonomous navigation, is the major bottleneck in USAR. The analysis of the operators' communications shows that more than 60% of communications were related to building and maintaining SA and that only 28% were spent in using SA. In another study based on the analysis of control behaviors, like robot and camera control, [118] shows that on average operators dedicated 30% of their time to SA activities. They "had less SA of the space behind the robot than in front or on the sides" as well as more difficulty in maintaining SA when the robot worked autonomously. The analysis of the RoboCup 2003 rescue competition [99] shows that incomplete Level 1 SA is the main cause of critical incidents. No team provided the operator with the sufficient information needed to build correct SA. The teams with more information on the user interface usually had fewer incidents.

Furthermore, the unfamiliar low and narrow view from the robot caused Level 2 SA errors in understanding the size of robot and its surrounding objects and the distance between them.

2.2 USER INTERFACE IN HRI

Based on the study of SA, we can improve effectiveness in two areas of the human-robot user interface (UI): (1) providing the right information required in building SA and (2) properly representing the information to enable the attainment of SA with the least amount of effort. However, the interface should still allow effective intervention with the robotic system.

2.2.1 The main modality

On every user interface, based on the task and interaction style, there is a main human robot communication modality that is extensively used by the operator. For instance, in vehicle teleoperation, visual feedback is usually the modality on which the operator spends most of his time [9, 36, 80]. In contrast, in high-level robot control tasks such as supervised navigation, the map is often the main modality for the operator.

2.2.1.1 The video modality

Camera configuration

Visually-based robot control is similar to people's daily behavior control in the rich and detailed image, and different in the low and narrow view, and the unnatural control style [117].

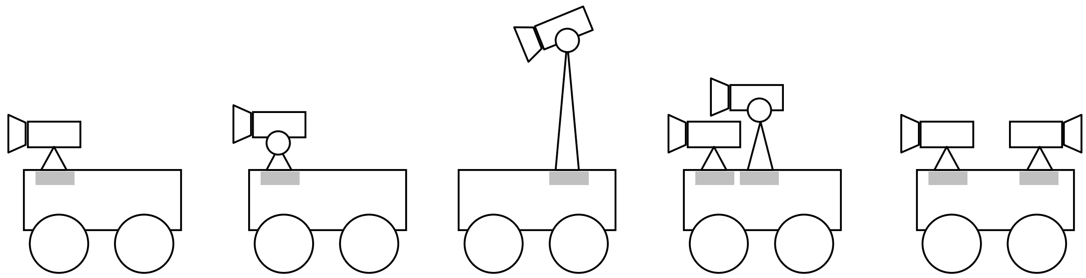
There are many studies of the configuration and representation of video feedback. In summary, the camera configuration can be classified according to four aspects:

The *reference* describes the movement domain of the camera. When we mount a camera on the robot's body, it will translate and rotate with the robot. However, for a gravity-referenced camera, it will not rotate because the robot may pitch and roll. When we drive a robot in a room from a camera mounted on the ceiling, the camera is a room-referenced camera that will not move with the robot. As the robot works in a spaceship, reference plays a critical role because the robot lacks proper frame reference [109].

The *placement* is the camera's position relative to its reference. A chassis-referenced camera, in which a camera is mounted on the robot's body, yields a front camera, rear camera, and overhead camera.

The *control DOF* is the degree of freedom to which an operator can control the camera. For example, a camera fixed on the robot's body has zero DOF and a pan/tilt camera has two DOFs. If we include a zoom function as the third dimension of camera control, a pan/tilt/zoom camera has three DOFs. It is possible for a camera to have more than three DOFs if it is mounted on a robot arm with multiple joints.

The *number of cameras* specifies how many cameras are used in single robot control. The camera can be real and physical or virtual. For instance, if we periodically point the camera in two directions to provide two different views on the interface, then we virtually have two cameras but physically have one camera. Multiple cameras are usually used in the following ways: stereo cameras provide distance information, panorama-style cameras provide a wide field of vision, and multi-placed cameras provide different visual perspectives.



a) Forward fixed camera b) Forward pan-tilt camera c) Overhead camera d) Fixed-controllable dual cameras e) Front-rear fixed cameras

Figure 5. Common camera configurations in the literature.

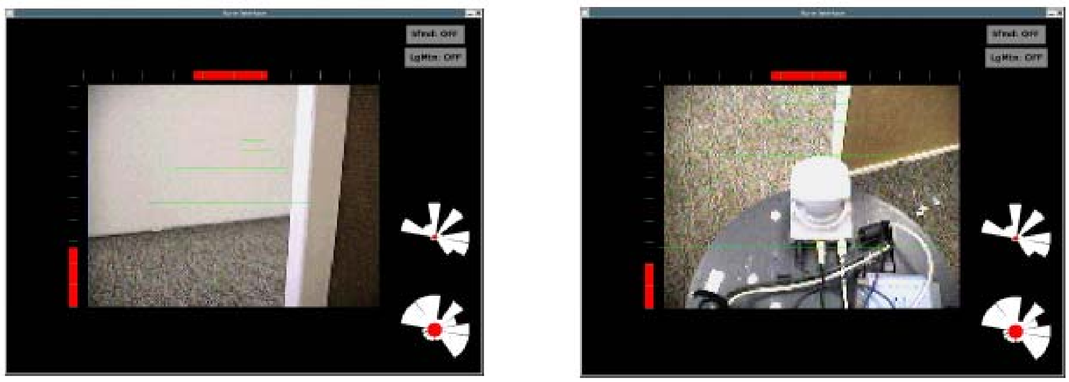


Figure 6. Forward (left) and overhead (right) camera views [53].

Figure 5 lists five camera configurations most used in the HRI studies. All of the cameras are chassis-referenced cameras. The experiment shows that with type (b) pan/tilt cameras, the operators were able to have significantly ($p<.05$) superior overall performance than those with type (a) fixed cameras [44]. However, the pan/tilt camera may lead to confusion of the camera's and the robot's directions [44, 121]. This type of camera is usually mounted on the front of the chassis to provide a first person's view. The compact mounting allows the robot to freely navigate small voids. In contrast, type (c)'s overhead camera configuration mounts the camera high over the chassis to provide a partial third person's view, i.e., showing part of the robot's body and its surroundings. Figure 6 shows the camera views of the same scene using types (b) and (c) configurations. The comparison of these two configurations shows that the overhead view provided better SA and explains why the operators preferred the overhead camera configuration three times as much in the experiment reported in [53]. The disadvantage of the overhead camera configuration is that looking down shortens the view distance. On the other hand, occupying more space will limit the travelable environment and cause safety problems. The configuration of an overhead camera is ideally mounted at human eye level and behind the robot's body to simulate the view that a person is following the robot. However, in practice this is usually infeasible. Instead, researchers are trying to virtually generate the followed third person's view by using previous video images [106].

Figure 7. Front (left) and Front-rear (right) camera view [53]

A type (d) configuration utilizes the forward fixed camera and the pan/tilt camera, which allows us perform two tasks, such as navigation and inspection, independently. The simulated target searching experiment [44] demonstrates that, when compared with the type (a) configuration, the pan/tilt camera in the type (d) configuration significantly ($p<.05$) benefited the overall performance and that different control strategies were found in types (d) and (b) configurations. However, the two independent camera controls are unfamiliar to us, and no significant difference in performance was found between types (d) and (b) configurations. Instead, the type (e) configuration utilizes fixed front and rear cameras to simulate the multi-view situation of the real world. The image at right in Figure 7 is an example of this type of interface configuration in which the front and rear views simulate the driver's view and rear mirror view in vehicle driving. In the real world experiment [53], the results reveal that the front and rear camera views caused fewer collisions in robot driving than the type (b) camera configuration. When only one camera view was shown and the operator was forced to switch between the front and rear cameras to acquire SA, higher-quality performance was found than when the type (b) configuration was used. Compared with the front and rear views condition, [53] reports that the single view display seems to produce less confusion in multi-camera conditions, at least in rear SA.

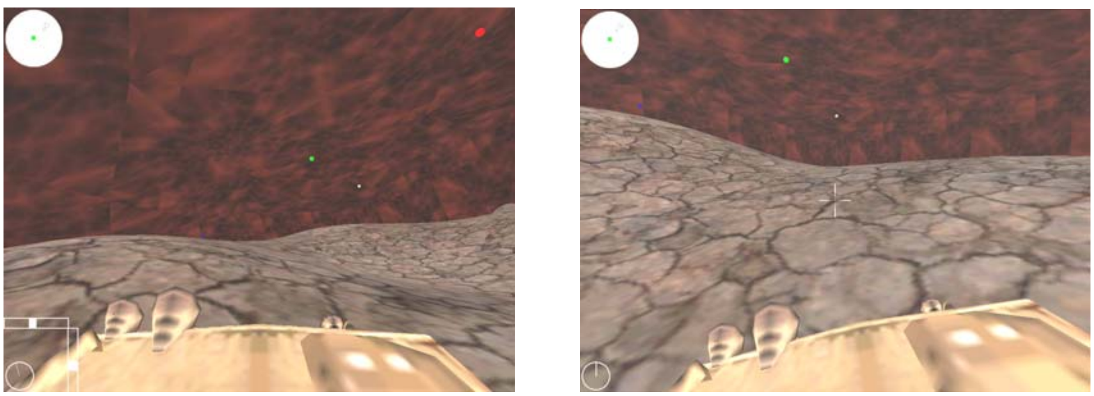


Figure 8. Fixed (left) and gravity-referenced (right) camera views [59].

All of the above configurations are based on chassis-referenced cameras. The flaw of the chassis-referenced camera is that, without the reference cues from the environment, we cannot use only video feedback to identify the robot's attitude. For instance, when the operators teleoperated vehicles at the desert-like Sandia site, the vehicles rolled over because the operators lacked awareness of the robot's pose [69]. [59] compares the chassis-referenced fixed camera with the gravity-referenced camera that holds a constant zero roll angle with respect to absolute vertical. Figure 8 illustrates that, with the gravity-referenced camera, it is possible to maintain awareness of the terrain surface and the robot's pose. The experiment conducted under lacking reference cues and confused reference cues environments shows that, with the gravity-referenced camera, the operators had improved control behaviors (e.g., small accumulated roll and pitch angles and more control correction behaviors) and shortened the task completion time, although no significant difference was found in the number of rollovers. However, it is technically difficult to build this kind of camera. Virtually generating the gravity-referenced camera view (e.g., turning the camera view back) is a potential solution.

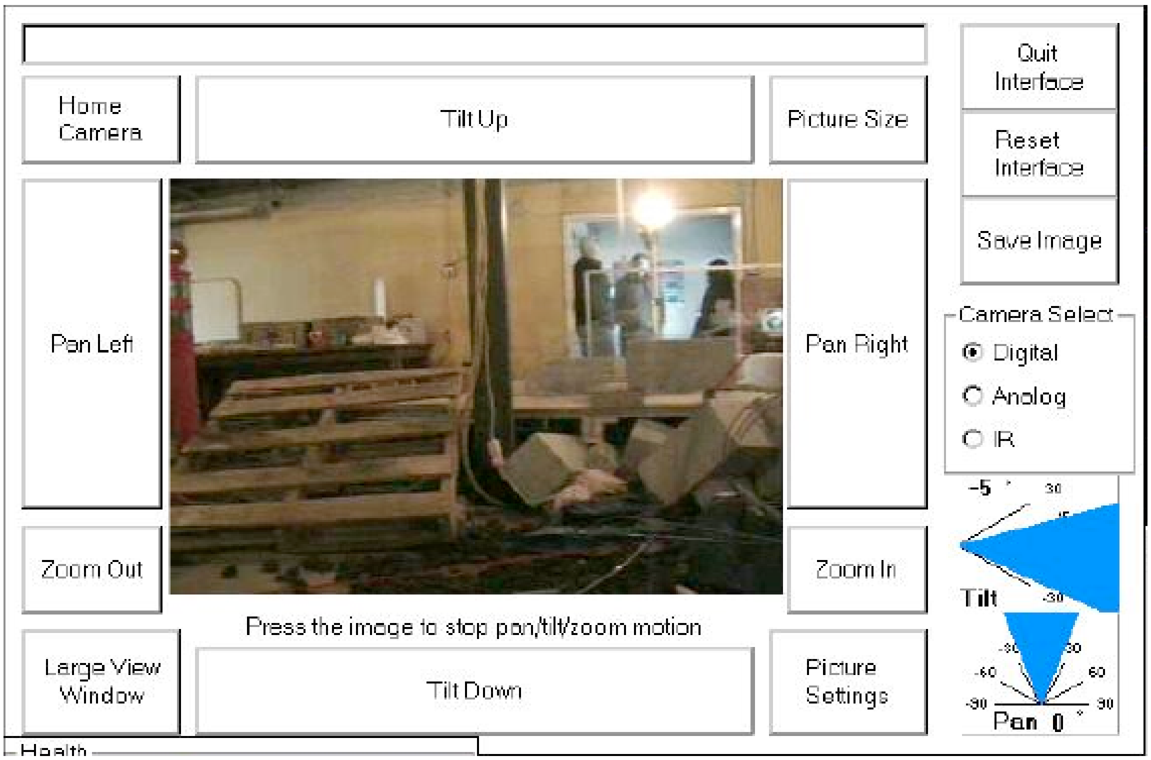


Figure 9. The INEEL interface reprinted from [4].

In summary, the above experiments show that proper camera configuration can enhance SA. However, new requirements, such as configuration awareness, arise as well. As shown above, lacking SA about the configuration features like the camera's orientation and position will mitigate the benefit or even cause incidents. The configuration of the camera is the cue that allows us to link the camera with the robot and the local environment to build Level 2 SA. Conventionally, we represent the camera view as an image window associated with configuration indicators (Figure 9). It is the operator's responsibility to mentally synthesize the information to maintain SA. In the following Section, we discuss current efforts to resolve camera SA issues.

Camera SA

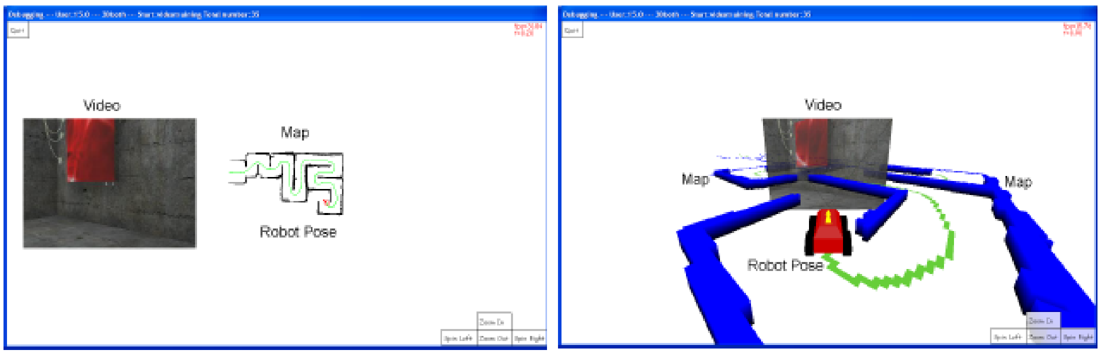


Figure 10. 2D (left) and 3D (right) map and video interfaces [80].

The first major problem in camera SA is that a narrowed field of vision and a moving camera make it difficult to maintain SA of the environment. Although placing the map next to the video (see the left window in Figure 10) significantly improves the environment SA, a novice user is more likely to be attracted to the video and to ignore the important information on the map [80]. [80] proposes a 3D interface (see the right window of Figure 10) that places the camera view in the virtual 3D environment to intuitively combine spatial and visual information. With this interface, the operator can take a snapshot and leave it on the map to help remember a scene. Both the simulated and real-world experiments show that the 3D interface is superior to the 2D side-by-side interface in terms of exploration with the fewest collisions [80]. The disadvantages of the 3D embedded camera view are that (1) the camera view is too small and the operator may loss detailed information on the video and (2) the distance between the projection plane and the robot is unknown such that the guessed value may be misleading. A variation of the 3D interface is fixing the camera view above the 3D world.

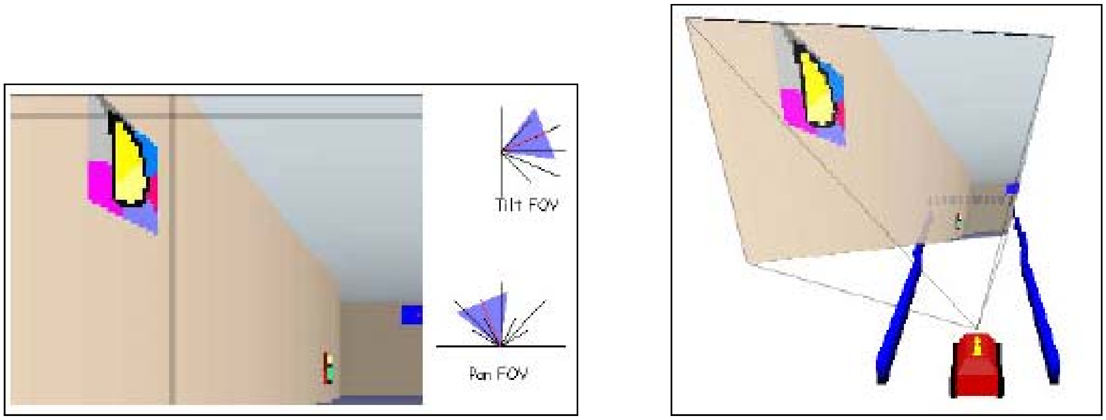


Figure 11. Crosshair (left) and 3D (right) representations of camera poses [81].

The second major problem in camera SA is the relationship between the robot's pose and the camera's orientation. The gravity-referenced camera shows how break the connection between robot pose and camera pose can benefit us. However, besides the technical difficulty in implementation, when the camera is controllable we still have problems in understanding the robot's pose. Conventionally, we use extra pose indicators to provide the information (see the right-hand lower corner in Figure 9). An improved approach is showing a crosshair on the video to represent the pan/tilt angles (see the left panel in Figure 11). This approach avoids shifting attention between the indicators and the camera view. However, the operator still needs a mental model to combine the pan/tilt information with the camera view. Based on the 3D interface, [81] proposes a 3D camera view representation (see the right of Figure 11) that skews the image and draws a perspective cone to present the view's projection. To avoid severe distortion (for example, the camera pointing to the right or left side), the interface automatically changes the operator's view angle to compensate for the skewed camera view and robot's heading direction display. The user study of searching for targets in a maze-like environment, which was conducted within a simulation world, reports that with the 3D interface the operator significantly completed the task more quickly (p<.04), with less collisions (p<.001), and in a safer way (farther from the wall) (p<.001) than the crosshair interface [81].

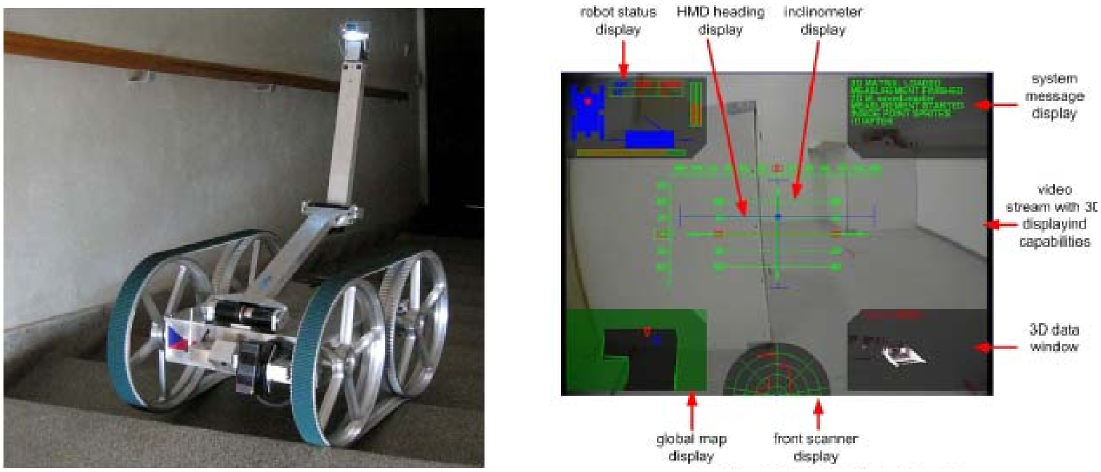


Figure 12. Orpheus (left) and its user interface (right) [122].

Other problems in camera SA include the camera's position, such as the front and rear cameras discussed previously. Awareness of the camera's height is also critical because it impacts size and distance perceptions and, in terms of Level 3 SA, the passable safe space. For example, the Orpheus robot shown in Figure 12 gives the user the flexibility to remotely inspect the environment through a head-mounted display and a head movement sensor as though the operator is present in the remote space [122]. Although both the robot's and arm's poses are displayed on the upper left-hand corner of the interface, the changed camera position relative to the robot might cause SA errors. When the camera has multiple control DOFs or when multiple cameras are used, the main problem is how to help the operator build and maintain SA of the spatial relationship between camera and robot and the relationship among cameras. Building a virtual 3D world might be the intuitive solution.

2.2.1.2 Other modalities

Another commonly used modality in an HRI interface is the map. Unlike video feedback that provides rich and concrete information, the map presents high-level abstract information to facilitate maintaining an overall understanding of the environment. The robot's capability, task, and world will influence whether video feedback or a map is the major modality because different levels of abstract information and local or global SA may be required. For instance, when using a high-speed robot with auto-navigation capability, the operator might prefer the map-based interface. For a search task, however, video feedback will be the main modality because of lacking advanced technology in pattern recognition. When we drive robots on rough terrain, the high demand on local SA may require a video-based interface. Given a robotic system, task, and world complexity, determining the proper primary modality is an interesting problem.

[80] compares video-based and map-based robot teleoperation in navigation tasks in two environments. In the office-like maze environment in which the camera view provided close range scenes because the wall blocked it, the operators who used the map interface completed the task significantly more quickly and with fewer collisions than the operators who used the video interface. However, in the environment in which the maze was constructed with low boxes, the camera was able to provide a view beyond the boxes and therefore increased the number of navigation cues. With the video-based interface, the operators completed the task in a significantly ($p<.001$) shorter period of time. In the RoboCup rescue competitions, both video-based and map-based interfaces were used. [98] selected four of the top five teams in RobotCup 2004 to compare auto-mapping and overhead camera-based techniques. The analysis of the critical incidents that occurred in one run shows that the two map-based teams had about the

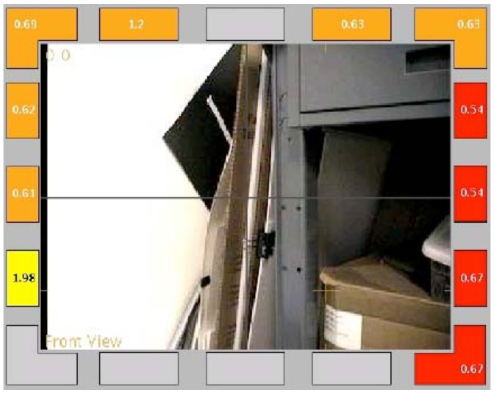
same local SA as the two overhead camera teams. Roughly the same obstacle encounter incidents were identified in the post-data analysis. Although this study is based on four entirely different robotic systems in one run with many potential confounds, the result reveals the necessity of comparing these two modalities. [110] compares map-based and camera-based interfaces for the semi-autonomous multi-robot system in a 3D simulated world during a navigation task. Again, no significant results were found in terms of the task completion time. However, we should note that the simulator in this experiment had low fidelity in that the situated events and behaviors were ignored. The simulated robot is indeed a fail-safe vehicle.

2.2.2 Auxiliary modalities

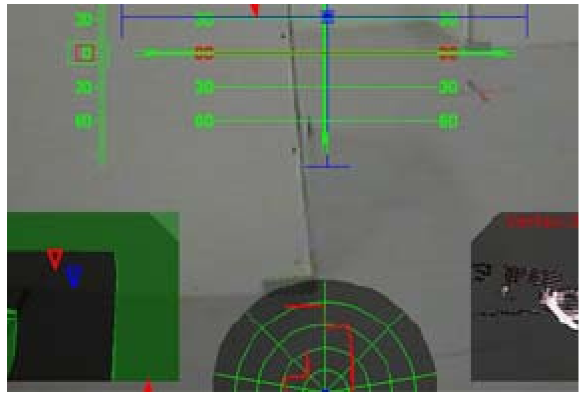
There are many possible modalities in HRI because of the diversity of the sensors. Information processing plays a very important role if we treat the robot as an active information source [75]. When multiple sensors are available, improper data representation will cause heavy mental demands or even confounds that will in turn lead to degraded or incorrect SA. In general, there are four ways to represent multiple sensors. We explain the four approaches using the frequently cited range data sensor as the example.

The *association* approach combines the relevant data to produce a chunk of information that reduces the mental resource requirement. Figure 13 (a) shows an example of this approach. The colored sonar data around the camera view represents the distance between the obstacles and the robot. Red indicates a dangerously short distance [4]. This layout facilitates the operator's processing of the image and range data as crucial elements of building local SA.

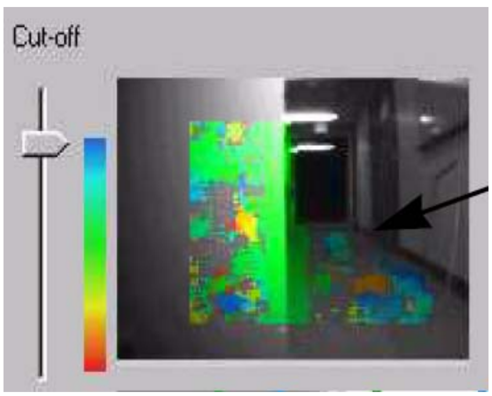
The *overlay* approach overlays multiple sensor data so that the user is forced to pay attention to all of the relevant data. For instance, the Orpheus user interface overlays range data on the camera view (see Figure 13 (b)) to allow the operator to see the camera image and the laser data at the same time.



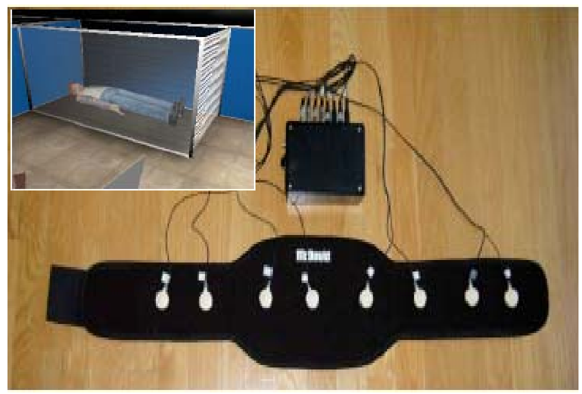
a) Association



b) Overlay



c) Fusion



d) Diversion

Figure 13. The four representations of range and visual data [4, 70, 102, 122].

The *fusion* approach, unlike the previous approaches that simply combine data, synthesizes multiple sensor data to generate new data. The association and overlay approaches implicitly facilitate information perception by attracting the user's attention, but the fusion approach directly helps the user in processing the data. Figure 13 (c) provides an example of stereo images fused with sonar data [70].

The *diversion* approach addresses the limitations of the 2D computer screen in representing all information as text or image, causing the operator's visual channel to become tremendously occupied such that information is perceived inefficiently. The diversion approach tries to represent sensor data in different channels to allow humans to simultaneously attain and process multiple sensor data with a lower cognitive workload. For instance, [102] transfers the range data to vibration via the TactaBelt (see Figure 13 (d)) to provide the directional vibrotactile cues. The experiment conducted in [60] shows that, compared with the video-based building-clearing testing, participants with the extra vibrotactile cues were able to explore significantly more areas with fewer exposures to potential enemies (the uncleared areas). The drawback of the diversion approach is that extra fatigue may be introduced. Representing the sensor data in ambient format, such as ambient light, audio, may be a useful alternative.

2.2.3 Intervention

Another approach to improve HRI effectiveness is to facilitate human interaction with the robotic system. Many research results in HCI, such as the GUI layout, attention attraction, and unambiguous options, can be directly used in a human-robot interface. Here we will focus on intervention, the special issue that significantly affects the effectiveness of any HRI.

Traditionally, humans fully control robots in a master-slave style in which all HRI initiations come from the operator. Due to the robot's ability to automatically perform part of the task, the human must cooperate with the robot at different levels, and interruptions will occur when the operator must stop his current task in order to respond to the robot. Interruption is important in HRI for two main reasons. First, the interactions between human, robot, and world inherently cause many interruptions, and improper responses to the interruptions significantly impact the effectiveness of HRI. Second, the consequence of the response may cause critical incidents, including damages.

In general, human interruption is defined as "the process of coordinating abrupt changes in people's activities" [67]. In terms of human responses to interruption, four kinds of interruption exist [68].

An immediate interruption stops the operator in his current task and forces him to respond to the robot immediately. This forced task switch usually causes problems in resuming the original task because of the difficulty in maintaining SA for the original task. Researchers have proposed several approaches to aid the resumption of the original task, such as a warning message before the interruption occurs so that the user can rehearse the point of interruption to benefit later memory retrieval when resuming the task; providing information about the original task via transparency panel, background sounds, or some other means to help the user maintain SA; reminding the user of the original task's information to help the user recover from interruption; and temporally freezing the original task to allow the user to rebuild SA after the interruption.

Negotiated interruption allows the user to select how and when to respond to the robotic system. Usually there are four possible responses: (1) immediately respond; (2) acknowledge the request for a response but respond later; (3) explicitly refuse to respond; and (4) ignore the request for a response, which is an implicit refusal to respond. The representation of this kind of interruption should at first attract users' attention when it arises and then can be ignored after the user has noticed it. Therefore, "silent" representation, like a change in color, size, or marking, is usually used in interface design.

Mediated interruption introduces a mediator as a buffer between users and tasks who helps the user respond to interruptions with a lower mental workload. The mediator can be based on: (1) the prediction of interruptibility that monitors a user's activities to find the boundaries between two tasks or sub-tasks and delivers the interruptions at these boundaries to reduce the disruption to performance; (2) the cognitive workload and dynamic task and function allocation that counterbalance the user's decision-making workload between a human and an autonomous system; and (3) a cognitive model so that the mediator is sufficiently intelligent to infer the user's intention and interrupt the user at the best time and in a preferred style.

A scheduled interruption occurs when the user has prior knowledge of the incoming interruptions and therefore can plan his activities to prepare for and to minimize the impact of future interruptions. A well-arranged schedule could render some would-be interruptions as ordinary planned activities. Explicit agreement and convention are the two scheduling techniques that in turn arrange one-time events and periodic events.

Many factors can affect the selection of the responding approach. For instance, an interruption for which response time is critical requires the immediate interruption style while negotiated interruption is appropriate for the time-insensitive interruption. Moreover, a poorly designed mediator may eliminate the benefit of mediated interruption.

Another trend in dealing with the interruptions is considering the intervention from a social cognitive view. The experiments [54, 76] show that, when a human interacts with a non-human, the human consciously or unconsciously treats the non-human object as another human and creates a mental model to estimate its knowledge. Our robotic system is usually selfish and interrupts us rudely without concern about the operator's current state. If we imagine our system as an assistant, following the social rules of daily life, we are able to design the interruption style that allows humans to interact naturally with robots with the least mental workload. For example, when an assistant needs you to sign a document while you are on the phone, she will place the document on your desk with a note and leave your office without any disruption; furthermore, she may remind you if you forget to sign the document and the deadline is approaching. When this approach is applied to a robotic system, a robot needing input from a human operator will first check to see if the operator is busy. If the operator is busy, the robot will attract the operator's attention and then silently leave a message about the suspend task on the interface. If the operator ignores the request for a long time, the robot will remind the operator. In this way, human-computer and human-robot etiquette is similar to the cognitive modeling that underlies mediated interruption. However, here we concentrate on interruption in terms of the social relationship. The experiment in [86] shows that etiquette is able to improve performance and trust in the robotic system, and good etiquette can even compensate for low autonomous reliability.

2.2.4 Summary

Given a robotic system, its task, and its workspace, the user interface has the greatest potential to allow us to improve human-robot performance. The user interface has two main functions: (1) helping the user to build and maintain SA and (2) providing ways to interact with the robotic system. Because of the limitations of human cognitive ability, in user interface design we must distinguish the main modality from auxiliary modalities and find the proper way to present them in order to achieve effective awareness (i.e., maintaining high-quality SA with the lowest cognitive workload). On the other hand, the user interface must support effective human-robot interaction by: (1) physically facilitating interaction via a proper interface design that involves, for example, layout, components' color, and size; (2) mentally helping interaction via providing assistance in, for example, decision-making and reminding the operator of the task context; and (3) socially maintaining or improving cognitive ability via following social convention and pursuing engagement. Based on the analysis of robotic competition and SA study, [25] summarizes that a good interface should enhance awareness via providing a map, lower cognitive load via using fused sensor information, increase efficiency via supporting multiple robots in one window, and provide help in choosing robot modality.

Finally, the user interface should be user-centered such that the end user, rather than the robot or the interface builder, is at the center of the design process. The human-robot interaction is a process that lasts for a period of time, and the interface therefore should include time-related issues, such as maintaining the user's vigilance and helping the user avoid both physical and mental fatigue [1]. The user interface is not necessarily based on the 2D computer screen. Although there is no direct evidence for this in HRI literature, it is possible to extend the interface into a 3D world if we consider SA representation as recreating SA in another world. For example, the UAV's 3D orientation is difficult to represent on computer screen. However, if we use a gyroscope to physically represent it, the operator will be able to intuitively perceive the UAV's pose and adjust the pose intuitively through the gyroscope representation.

2.3 ROBOTIC SYSTEMS IN HRI

In addition to the user interface, the capability of the robotic system significantly impacts human-robot performance. Robotic systems differ from each other in many aspects, such as sensory capabilities, mobile capability, and control algorithms. In HRI, we are more interested in studying how and when human involvement in robot control maximizes effectiveness. How people are involved in robot control is a function of the level of autonomy (LOA) that measures the static function assignments of the human and the robot. Adaptive autonomy occurs when people are involved in robot control such that human involvement dynamically changes the autonomy level, allocating control between the human and the robot [30, 50].

Table 1. LOAs in dynamic, multitask scenarios [50].

Level of automation	Roles			
	Monitoring	Generating	Selecting	Implementing
(1) Manual control	Human	Human	Human	Human
(2) Action support	Human Computer	Human	Human	Human Computer
(3) Batch processing	Human Computer	Human	Human	Computer
(4) Shared control	Human Computer	Human Computer	Human	Human Computer
(5) Decision support	Human Computer	Human Computer	Human	Computer
(6) Blended decision making	Human Computer	Human Computer	Human Computer	Computer
(7) Rigid system	Human Computer	Computer	Human	Computer
(8) Automated decision making	Human Computer	Human Computer	Computer	Computer
(9) Supervisory control	Human Computer	Computer	Computer	Computer
(10) Full automation	Computer	Computer	Computer	Computer

2.3.1 Level of autonomy

Based on Sheridan and Verplank's (1978) hierarchy of LOAs, [50] proposes ten autonomy levels (Table 1) that describe the dynamic and multitask autonomy scenarios that can be directly used in robotic systems. The roles in Table 1 represent the four basic functions in a human-computer (or human-robot) system. These four functions can be understood in the following terms: (1) monitoring is perceiving the system's state, (2) generating is creating the options or strategies for the given task, (3) selecting is deciding to accept or reject an option, and (4) implementing is executing the selected option. The level of autonomy that most benefits human-robot performance is affected by many factors that emerge from relationships between the task, the world, the human, and the robotic system.

For example, in human-machine system studies, [74] compares LOAs 5, 8, and 10 in a process control simulation. The results suggest that, when the task is time critical, the final decision should be allocated to automated processing instead of human processing. [74] states that the best LOA is based on the complexity, difficulty, dynamic, and quality requirements of the task. In another study, [61] compares three levels of computer support in an automated diagnosis system. The computer support showed improved performance under normal conditions. However, under automation failure, a medium LOA led to the worst performance, which was caused by the different information sampling strategies utilized under the three conditions. In the human-robot study, [63, 64] compare the teleoperation (LOA 1), safe mode (LOA 2), shared control (LOA 4), and dynamic control (AA) in robot-assisted search and rescue for both novice and expert users. The results reveal that the preferred LOA is strongly correlated with the user's experience. The novices with no teleoperation experience were more willing to trust automation and utilize the autonomy capabilities, while the experts preferred teleoperation. Furthermore, the experiment shows that both novices and experts could have approximately the same overall efficiency with the proper LOA. In [85], the comparison of medium LOAs and the corresponding lower LOA (in which the human operator has more control over the robotic system) shows that, with four robots, the lower LOA tends to lead to superior performance. However, when the number of robots increases to eight, the benefit is eliminated because of the heavy management workload.

Although many factors may affect the optimal LOA for achieving efficient HRI, in general, a medium LOA is superior to full autonomy or full manual control because an LOA that is too high will cause degradation in manual or mental skill, the loss of SA, decision bias, vigilance decrement, and bad response to unexpected conditions. Under full manual control, the high mental demand, human decision bias, complacency, boredom, and inconsistent control behavior will degrade the performance. In practice, instead of using a uniform LOA, we decompose the robot control into sub-tasks under different conditions and utilize different LOAs[4]. For example, based on the previous conclusions about LOA, an adjustable system may switch to autonomous control when the sub-task is time-critical. [4] proposes to add autonomy suggestions to the human-robot interface in order to help the human operator adjust the LOA of the robotic system.

2.3.2 Adaptive automation

In adaptive automation (AA), there are four approaches to changing the autonomy level [50]. The critical events approach changes the LOA when critical events occur. Performance-based AA adjusts the LOA according to the human monitoring performance measurement; when this measurement is below a certain threshold, the control allocation can be changed to maintain system performance. The psychophysiological assessment approach uses physiological measures to assess the operator's workload in real time and then to adjust the LOA accordingly. Finally, the behavior modeling approach changes the LOA according to the operator's model.

The challenge in AA is that the change between LOAs will require a switch in both the context for and the skill of the operator. When the user is exposed to automation for a long time, his skill will decay and considerable time may be spent in building SA. In contrast, a very high

[4] Within a USAR domain, the task decomposition has been utilized in SA analysis. Unfortunately, the decomposed optimal LOA study, which focuses on identifying the LOAs for local andglobal navigation and search in an office-like environment or on rough terrain, is largely lacking.

frequency of control allocation might cause an incomplete or a lacking of SA because of the limited processing time for information. Who makes the final decision in control allocation affects the effectiveness as well. In general, under time-critical conditions, a robot or a computer can make faster decisions than a person and therefore leads to better overall performance.

As mentioned before, AA and LOA correlate with each other. AA responds when we need to change the LOA and how long we will utilize the new LOA. The LOA responds with which LOA we will use. [50] studies how the interaction of LOAs and AA impact overall performance. The participants switched between manual control and LOA 3, 4, 6, or 9 with different amounts of time dedicated to manual and automated control. The results reveal that the LOA and AA benefit human-machine performance differently. The LOA is the driving factor that affects performance and SA, and the automation allocation cycle time mainly manages the operator's workload. However, the combination of the LOA that leads to best performance and the AA that produces superior functioning (in which the operator has the lowest workload under full automation) did not produce the best overall performance. In addition, the study shows that human's intervention always benefits the human-machine performance.

2.4 EVALUATION OF HRI

In the evaluation of HRI, we are usually interested in assessing the human-robot performance, effectiveness, and efficiency. Unfortunately, we currently lack common metrics to evaluate HRI. Most of the studies are based on task-specific evaluations that indirectly assess HRI. For instance, in the USAR domain, [63, 64] use reported targets and task completion time as their main metrics, [9] utilizes joystick control as an auxiliary metric, and [80] uses the number of collisions as an additional metric. While in the RoboCup Rescue competition, map quality, the number of found victims, the number of collisions, and the number of operators are used to evaluate overall performance [77]. [105] proposes common metrics for HRI based on three aspects: system performance, operator performance, and robot performance.

2.4.1 System performance

System performance is the human-robot performance that measures how the human and the robot effectively and efficiently work together in performing a task. In [105], effectiveness and efficiency are defined as "the percentage of the mission that was accomplished with the designed autonomy" and "the time required to complete a task…. [or] if time constraints are ignored, … all tasks completed," respectively. In this definition, efficiency is measured in terms of time, which is unsuitable for any task that is not time-critical. Instead, similar to Webster's Online Dictionary's definition[5], we can generally define efficiency as the ratio of the completed tasks to the cost of completing those tasks. For example, in the RoboCup Rescue competition, the cost is the number of operators and the time is ignored because every team has the same operating time in the contest. For a tour robot, the cost may be the money spent in building and maintaining it or the consumed power. The cost used in efficiency measurement will be decided by the task, i.e., which input has the most significant effect on it.

[5] Efficiency is defined as "the ratio of the output to the input of any system" in Webster's Online Dictionary (http://www.websters-online-dictionary.org/definition/efficiency).

2.4.2 Operator performance

Operator performance measures the operator's state and efficiency. The state of an operator includes situational awareness of the robotic system and the task, and the workload, all of which affect the operator's robot control capability.

In Section 2.1.3, we introduced the three ways to measure SA: through subjective assessment, a measure of implicit performance, and an explicit measure. Workload can be measured in similar ways. A subjective assessment of workload requires the operator to complete a carefully designed questionnaire. NASA-TLX (NASA Task Load Index), SWAT (Subjective Workload Assessment Technique), and WP (Workload Profile) are the three most common multidimensional workload assessment techniques. NASA-TLX measures the mental workload according to six dimensions: mental demand, physical demand, temporal demand, performance, effort, and frustration level [43]. SWAT measures workload in terms of time load, mental effort load, and psychological stress load, which in turn is divided into three levels of low, medium, and high [89]. WP assesses the used attentional resources in terms of resource dimensions, which include perceptual, central, and response processing, spatial and verbal processing, visual and auditory input processing, and manual and speed output [112]. The subject study based on a single task or dual tasks shows that there is no difference among the three approaches in terms of their intrusiveness and validity, and the WP approach is more sensitive to the task manipulations and demonstrates diagnostic power superior to that of NASA-TLX and SWAT [92]. The implicit measure of workload introduces the second independent task and uses its performance to indirectly assess the workload. High performance in the second task implies a low workload in the primary task. This approach is easy to implement. However, the introduced task may impact the operator and lead to inaccurate results. Using an explicit measure of operator performance, physiological measurements such as cardiac or respiration rates can be used to assess cognitive workload in real time [105].

Finally, when the user operates a robotic system, the usability of the interface significantly impacts the operator's performance. Operation efficiency refers to how efficiently the user operates a device. Usually, matching the interface display and controls to human mental models allows us improve the efficiency [105].

2.4.3 Robot performance

The measurement of robot performance includes the robot's capabilities, such as its autonomy, as well as the robot's awareness of itself and of the operator. Neglect tolerance is the general index that allows us to measure a robot's autonomy. The autonomy enables a robot to work effectively and independently without human intervention. However, because of the limitations of automation, the effectiveness will decline over time. Therefore, we use the amount of time that the robot is able to work independently with satisfied effectiveness as the measurement of the robot's autonomy capability. Noticeably, the satisfied effectiveness during robot control is judged by the operator. The sub-task and the complexity of the local workspace affect satisfied effectiveness as well. In practice, we usually measure neglect tolerance implicitly from the user's perspective, i.e., the human's intervention, to take into consideration the subjective judgment bias. We also measure the complexity of the local environment and sub-task to help us adjust the autonomy assessment [82].

Awareness plays a very important role in human-robot interactions. The robot's self-awareness allows the robot to make proper decisions and therefore requires less human intervention. On the other hand, the robot's awareness of its capability and state will help the operator efficiently interact with it. [105] proposes to measure self-awareness in terms of the robot's knowledge of its physical capabilities, such as mobile and sensory limitations, its current state, its ability to detect, isolate, and recover from failures or abnormal states. The awareness of the human also will help the robot efficiently interact with the operator. For instance, with the perspective-taking technique, the robot is able to solve problems collaboratively with a person [109]. The human awareness measurement can include the capability to perceive, monitor, and model the human's capability and state as well as the ability to utilize these awareness to improve interactions [105].

2.5 THE CHALLENGES IN HUMAN MULTI-ROBOT INTERACTION

In this chapter, most of the studies focus on single-operator single-robot (SOSR) control. Obviously, one operator who can simultaneously control multiple robots will greatly improve the effectiveness. However, the conflict between increased complexity and limited human capabilities brings many new challenges.

Fist of all, to control multiple robots, the operator must maintain each robot's SA and switch between contexts. This will challenge both the operator's cognitive capability and the human-robot interface design. Given that there are multiple robots under the operator's control, the interface should allow the user get the right information about the right robot at right time. Simply listing all robots' information on one interface may overwhelm the user because of a human's limited attention resources and because improper selective information representation will cause a loss in SA. On the other hand, the increased number of robots will lead to difficulties for the human operator in decision-making, planning, and issuing commands.

Clearly, improving the individual robot's autonomy and the cooperation among the robots will mitigate the human operator's workload. Unfortunately, new problems arise from the view of human-robot interaction. The cooperating robot team requires the user to build and maintain team SA that is much more complex than single-robot SA. The operator of a cooperating robot team will use more cognitive resources and take more time in processing information. As the robots cooperate with each other, how the human effectively intervenes with the robot team is another new problem. If the user is able to control the robots at the team and the individual level, then the cognitive workload will increase when the user switches between the contexts of these two levels. On the other hand, a high LOA generally will cause a decline in SA. People may have difficulty in understanding the robots' intentions and lose trust in the robotic system, which will significantly impact the effectiveness of the interaction.

Finally, when a human controls multiple robots, we need to consider the robot team's effect as well as each individual robot's effect. For example, when we assess a robot's performance, we need to evaluate the robot's awareness of other robots as well. The structure, organization, and cooperation of the robot team and the level of human involvement in this cooperation will impact the HRI and should be reflected in the HRI evaluation.

3.0 RELATED WORK

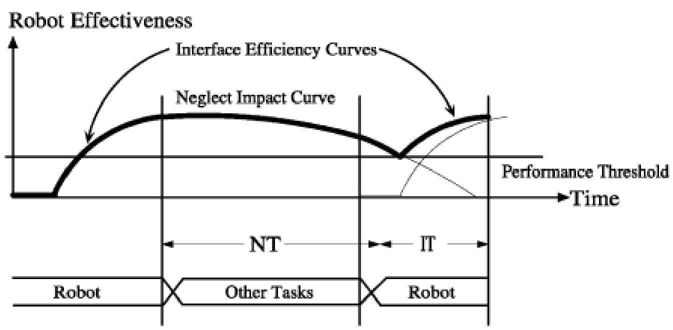
In this chapter, we introduce current studies in human-multi-robot interaction (HMRI). We begin with current theory about human control of multiple robots, although such theory has obvious limitations. Then we analyze SA in multi-robot control. Because of a lack of literature in this area, related work in group awareness will be discussed. Then we describe current efforts that examine how the group size, level of autonomy, and the user interface impact HMRI. Finally, we summarize the current status of HMRI study.

3.1 THE FOUNDATION AND EVALUATION

3.1.1 Neglect Tolerance

In general, humans work in a serial style [16]. If we need to work on multiple tasks, we complete them one by one or temporarily switch to another task. This also happens when we control multiple robots. When we interact with a robot, we try to improve or maintain its performance via our intervention. If we can temporarily ignore it, then it is possible for us to control another robot. The longer that a robot can be ignored, the more robots we are able to control.

When we ignore a robot, the robot works independently but its performance declines over time. The neglect impact curve in Figure 14 shows this neglect effect. The time during which a robot is ignored is called the neglect time (NT)[6]. In contrast, when we control a robot, its performance will increase. However, this increase will not occur immediately because the operator needs to do a minimum of four things before a robot can execute a command: (1) select the sub-task to perform, (2) build SA for the robot, (3) make decisions in order to form a plan, and (4) transfer the plan to control commands and issue them to the robot [82]. In practice, the time that the operator spends on these stages may be long and have a significant impact. For instance, the operator may not be aware that there is a sub-task or may not have enough SA to make a plan [20]. The interaction effect is represented in Figure 14 by the thin line. The time spent in interaction is called interaction time (IT).



Summary of Terms

NT	Neglect Time
IT	Interaction Time
Tolerance	(NT,IT)
Neglect Impact	Decrease in robot performance in the absence of human input
Interface Efficiency	Increase in robot performance in the presence of human input
Interaction Scheme	(Autonomy,Interface)

Figure 14. Neglect Tolerance (NT) and Interaction Time (IT) (reprinted from [19]).

3.1.2 Evaluation metrics

For a homogenous independent robot team, we can theoretically predict and evaluate human multi-robot control in the following metrics using NT and IT [19, 82]:

Fan-Out (FO): The maximum robots that one operator is able to control is

$$FO = (IT+NT)/IT = 1 + NT/IT,$$

[6] Please note, when we ignore a robot we may control more than one robot, and part of the NT time may be off-task time (also called Free Time) during which we don't pay attention to any robot, for example the time we spent in switching from one robot to another.

which reveals that to improve the effectiveness, we should enhance the robotic system's autonomy to increase NT and utilize a good interface to decrease IT. However, usually NT and IT affect each other. As mentioned in the previous chapter, under a higher LOA, the operator will have more difficulty in effectively interacting with the robotic system. Therefore, an increase in NT might increase the IT as well.

Robot Attention Demand (RAD): In measuring the attention demand from a robot, RAD is the ratio of the time that the operator must attend to a robot over the total task time:

$$RAD = IT/(IT+NT).$$

For robot teleoperation, the NT is very small, and the RAD will approach one when the working status is very busy.

Relative Free Time (RFT): RFT is the fraction of the task time that the operator can relax without paying attention to the robot:

$$RFT = NT/(IT+NT).$$

RFT is related to RAD in that the sum of RFT and RAD should be one (i.e., RFT + RAD = 1). RAD is usually difficult to directly measure because it involves measuring the mental actions, such as the time that the user spends on context acquisition and planning. The relationship between RFT and RAD allows us to implicitly assess RAD by measuring RFT. Since RFT is the time that a user can relax, it is possible to ask the operator to spend time on a second independent task and use the performance on the second task to measure RFT. The defect of this indirect measurement is that the second task may impact the operator's control behaviors. [19] demonstrates another approach to directly measuring robot effectiveness by using the generated neglect impact curve and interface efficiency curve to compute the metrics.

In summary, the neglect tolerance opens the door to allow us to theoretically analyze human multi-robot control. Unfortunately, current theory is based on the assumption of a homogenous independent robot team. An extension of neglect tolerance theory is measuring the wait time between NT and IT [20]. When an operator moves his attention from a robot, he needs time to decide which robot to turn to before beginning a new interaction. The time between NT and IT is the wait time, which includes the time spent in being aware of a task in the queue as well as the time caused by a loss of SA. The Fan-Out is updated as the following:

$$FO = NT/(IT+WTQ+WTSA) + 1,$$

where WTQ is queuing wait time and WTSA is wait time caused by a loss of SA.

Since only temporal measurements are used in analysis, [22] further extends the wait time concept to include a cost-performance model. The number of robots that a single operator can control is depicted in Figure 15.

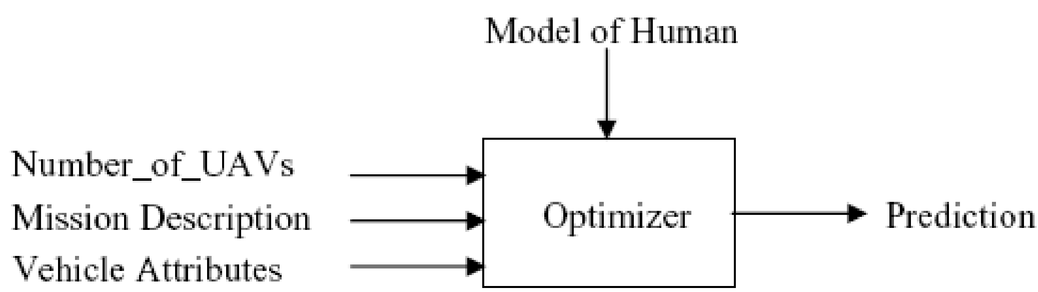


Figure 15. The optimization model (reprinted from [22]).

3.2 WORKSPACE AWARENESS

As mentioned before, when a human controls multiple robots, the operator must maintain his awareness of the robot team as well as of the individual robots. Unfortunately, studies about this kind of awareness are rare in the literature. Instead, we introduce relevant workspace awareness studies in computer-supported cooperative work (CSCW) and apply it to human multi-robot control.

3.2.1 Definition

When people work together via groupware, each person in this group usually has more difficulty in getting to know his coworkers than when the group physically works together in a face-to-face environment. Groupware limits the actions that a team member can take. Information can be acquired from the workspace, but often only part of the available information is represented to each user. Therefore, a group member may lack the awareness of who the other group members are, where the other group members are, and what the other group members are doing. Based on these characteristics, the awareness of a robot group in HMRI is very similar to group awareness in CSCW. The difference is that, in HMRI, the group members are robots and coworkers' "brains" are robotic agents (control software). In CSCW, we refer group awareness as workspace awareness (WA), which is defined as the "up-to-moment understanding of another person's interaction with a shared workspace" [40]. WA is a specialization of SA in which the "situation" is the other group members' interactions with the workspace, which implies awareness of both the domain and the collaboration [41, 42]. [41, 42] propose a framework to indicate the information that makes up WA, how people acquire WA, and how we use WA in collaboration.

3.2.2 The framework

Essentially, WA includes the information that answers the who, what, where, when, and how questions. The “who” question asks if there are other members in the workspace, who they are, and what authorships exist between the group members and the observed actions. The “what” question involves information about a person’s actions and the intention and object of each action. The “where” question addresses the location of a group member and the areas with which a group member can interact, which include where he is gazing and where he can reach. The “when” and “how” questions directly involve time. The answer to the “when” question is the moment at which an event happened, and the answer to the “how” question is the history of an action or an involved object over time. The “who,” “where,” and “what” questions can be associated with time as well, i.e., the awareness of who was there, where he was, and what he was doing. In HMRI, this part of the framework can be directly used when coworkers are replaced with robots. The awareness of the robot group is the information that answers all five of these questions. For example, the “where” question queries the location of a robot, the area the robot is sensing, and the direction in which the robot is moving.

When people work in a shared workspace, they can obtain WA in three ways. The first way is the consequential communication that gathers awareness by watching other group members. For example, we can look at the body’s movement or gestures to know what someone is doing. This communication emphasizes our perceived consequences of other people’s actions because no intentional information is retrieved.

Similarly to watching humans, we can observe artifacts to indirectly obtain awareness about coworkers. This is known as a feedthrough mechanism. Like people acting in an environment, the artifacts will be impacted and the state change can be interpreted as the feedback of a person performing a task. For example, the changed shape of a ball in a human's hand can tell us how hard the person is squeezing it, and the sound of a bounced ball tells us that another person is playing with a ball even if we cannot see the person or the ball. When we see both the artifacts and the actor, the feedthrough is usually coupled with consequential communication.

The final mechanism used in acquiring awareness is intentional communication, which allows us to proactively cooperate with partners by telling us what the partners intend to do. Verbal communication is the main modality through which a person directly talks to each other to learn others' intentions, listens to other conversations to learn others' intention, or talks aloud to himself to allow other people to learn his intentions. Gesture is another modality that can transfer information about intentions. For instance, pointing to the door we are going to enter, gazing at the object we intend to pick up, or heading toward the person we want to interact with can send our intentions to other people. In HMRI, people use similar mechanisms to build awareness of the robot team. However, the major difference may be the intentional communication. In CSCW, humans are intelligent enough to communicate with each other in order to ensure an understanding of intentions. However, robots usually communicate in simple and straightforward ways, such as directly delivering plans to the operator without any explanation. Therefore, the human operator may fail to know why the plan was selected and what intention lies behind the plan. How to represent both the individual's and the team's intentions to the human operator is a challenge in HMRI, especially when the robotic system is complex.

When people have gathered information awareness of each other, there are at least five ways to use this information in collaboration. First, WA helps a person manage when he needs both to work with others and to work independently. Second, WA allows a person to communicate with others more efficiently because it facilitates a common grounding. Third, WA facilitates the use of non-verbal communication, such as deictic references, demonstrations, manifesting actions, and visual evidence, which simplify communication. Fourth, WA helps a person to predict others' actions on both small and large time scales and therefore helps a person to cooperate with coworkers more efficiently and without conflict. Fifth, WA benefits a person through providing a better understanding of others' requests. The use of WA in HMRI is very similar to its use in CSCW because the operator is also a human. All of the approaches mentioned above can be directly used except the simplification of communication, which is restricted by the limited communication capabilities of a robot.

3.2.3 Awareness metaphor

One of the main goals of CSCW is providing support to help people build, maintain, and use WA in collaborative work. Because people cooperate much more efficiently in a face-to-face environment than in a remotely shared workspace, early effort in this area focused on building a "rich" media space in which a face-to-face environment can be virtually presented. Unfortunately, no expected success has been reported because of the discontinuous nature of media space and the fact that facilitating awareness is not simply providing all possible information. Later effort has focused on building a computational environment that is based on an awareness model that allows people to efficiently collect, dispatch, and integrate information involved in collaborative work [95]. Spatial metaphors and reaction-diffusion metaphors are the two models used in CSCW. The spatial model measures awareness as the combination of focus and nimbus, where focus is the subspace where our attention is allocated and nimbus is the subspace that contains the observed objects. The reaction-diffusion model addresses the mutual awareness and interaction between actors. In this model, the reaction means that actors interfere with each other and thereby change their states, and diffusion refers to the space where involved entities exist and interact. The details of these two models can be found in [90] and [103]. In HRI, current studies classify awareness of single robot control in rough grain (see Section 2.1.4) and the study of multi-robot control is still rare. We lack a theory that allows us to formally analyze awareness and use it in evaluation, the robotic system, and interface design.

3.3 THE CURRENT EFFORTS

Although multi-robot control will potentially benefit HRI effectiveness, it challenges the limited cognitive capabilities of human operators. Using multi-robot control in future combat settings, space discovery, and search and rescue attracts many researchers to study ways to improve efficiency in HMRI. How multiple robots impact both human and overall performance is the fundamental question that we need to answer. Proceeding from this question, we are able to find ways to improve or deeply analyze HMRI either from the robotic system or the user interface. In the following Section, we introduce the current efforts in these three fields separately.

3.3.1 Group size

As mentioned in Section 1, the human, the robotic system, the task, and the world affect each other. Therefore, the impact of multiple robots correlates with many other factors. Fortunately, we are able to investigate this impact under different conditions because the comparison studies of different numbers of robots are present in several studies.

[111] compares the navigation performance of two, four, and eight robots controlled by a single operator in a 2D simulated world. Under conditions of waypoint-based independent robot control and simple robot simulation, the experiment shows that the operator is able to control up to four robots, with the larger number of robots resulting in a higher workload and a stronger impact on the operator's monitoring ability than on the control ability. In a later study [110], the authors added autopilot capability to the robots, upgraded the world simulation to a 3D graph rendering system, and improved the simulated robot to a virtual fail-safe vehicle. The comparison of one, two, and four robots controlled during the exploration task shows that, in single robot control, human intervention improved performance but, when participants shifted to multi-robot (two or four robots) control, they "tend to reactive instead of proactive supervisory control" [110]. Overall performances under this latter condition were worse than those under the condition of full autonomy control. Again, increased human workload was found as the group size increased. In the most recent study [46], researchers used a dynamic robot and workspace simulator (USARSim) to compare the robot control behaviors with six and nine independent heterogeneous robots. In this experiment, the high fidelity simulator, which introduces realistic SA problems, such as robot collisions in robot control, makes the results potentially more realistic than those of previous experiments. The participants controlled the robots via teleoperation or prescribed behavior with a scalable interface. The results show that a higher number of robots caused a higher workload; however, the increment was less than the ratio of 1:1, which reveals that a single operator is able to control several robots with the user interface. The interesting result reported in the experiment contradicts the traditional belief that more robots lead to worse SA, yet the experiment found improved overall SA. Either the limitation of

the SA measurement or the specified (e.g., small, simple) testing world may contribute to this contradiction. Further study is needed.

The impact of multi-robot control is studied in multi-agent fields as well. In [85], robots with a high level of autonomy work as a team. Participants control a team of four or eight robots to play a capture-the-flag game with another fully autonomous robot team. The experiment, which was conducted using an abstract 2D simulated world, reveals that players took significantly more time to finish the game with significantly less success rate when controlling eight robots than when controlling four. However, the introduced flexibilities in robot control (controlling an individual robot or a robot team with manual or scripted behavior) allowed the player to control four robots with improved overall performance. The requirement of controlling eight robots produced a greater management workload for the human user, which countered the benefits. In another recent study that examined humans in loop multi-agent control [100], the human operator controlled four, six, or ten fire engines to extinguish fires in a simulated 3D urban environment via task allocation or strategy selection. Although the experiment is based on very small sample size (three subjects), the results reveal that human intervention and additional agents do not always benefit team performance. Instead, the dominant team-level strategy changes as the team size changes.

In summary, multi-robot control impacts the human operator's workload in three ways: (1) building and maintaining awareness, (2) making decisions, and (3) controlling the robotic system. Increasing the autonomy level in robotic system, whether providing decision support or individual robot autonomy, allows us shift the decision-making and robot control workload from the human to the robotic system. On the other hand, the increased intelligence may cause an increase in perception and decision-making workloads. Thus, there is a trade-off between the autonomy level of the robotic system and the level of human intervention. The user interface is another place that allows us to mitigate the workload by facilitating awareness acquisition and issuing commands. A good interface can help us to combine different resources to meet the 1+1<2 equation and therefore to control more robots.

3.3.2 Robotic system

To mitigate the human operator's workload, the robotic system should support or take over the human's work while maintaining the operator's control over the robotic system. This includes constructing the robot group to better support the human operator, improving an individual robot's autonomy to simplify robot control, and introducing cooperation to (partially) replace the human operator's high-level control.

[15] presents an example of the robotic system that enhances SA via a carefully designed robotic system. The 2 UAVs in the robot team are able to provide the overhead view of the UGVs on the ground and thereby significantly improve the operator's robot team awareness. Although the design of MRS is beyond the scope of this paper, other studies that focus on improving the effectiveness of robotic system design is available in the literature.

One of the most recent efforts in HMRI is seeking the optimal interaction scheme[7] that allows us to effectively control multiple robots. [35] demonstrates collaborative robot control when human operators control the robots via waypoint assignment and communicate through verbal dialog. This interaction scheme makes multi-robot control possible by enabling adaptive task allocation and temporally neglecting a robot. [19, 79] compare three interaction schemes—

[7] Recall that an interaction scheme is the combination of autonomy and the human-robot interface.

the teleoprate and landmark (TOL), the point to point and human snapper (PTP), and the region of interest of sealing (ROI)—in a 2D simulated world with three independent robots. The results reveal that the increase in autonomy decreases both workload and performance and that a greater degree of human control increases the workload but also improves performance. More specifically, in this experiment, PTP tends to increase workload and performance, ROI causes decreased workload and performance, and TOL dramatically imposes a high workload and leads to slight performance improvement. In the multiple UAVs study with different levels of decision support under low or high replanning demands [21], the successful destruction of targets with four UAVs demonstrates similar results in that increasing autonomy reduces both workload and awareness. In addition, for this kind of management task, automation bias was found in human decision-making. The operators usually failed in appropriately accounting for uncertainty in their decisions, and the probabilistic prediction support degraded the performance.

Unlike the above experiments that utilize independent robots, [85, 104] compare different interaction schemes with a cooperating robot team for the capture-the-flag task in a 2D simulated world. The results show that the flexibility in switching between autonomy levels (adaptive control) slightly increased workload with significantly improved overall performance. In another recent study [100] in which a human operator controls a robot team to extinguish fires in a virtual 3D city, the comparison among the four conditions—full manual control, a human operator with individual-level autonomy, a human operator with team-level autonomy, and full autonomy—reveals that human involvement did not always benefit the performance of a complex task and that a team-level strategy was not consistently superior in performance to the individual-level strategy in the specific task and multi-agent system. However, the experiment was conducted on three participants with three tests under each condition.

Although the results slightly vary in the above studies because of the differences in the nature of the task, the world, and the robotic system, autonomy decreases workload with degraded awareness for both teams and groups of robots. Adaptive autonomy (the flexibility of switching between different LOAs) tends to increase performance with workload and may be associated with better awareness than restricted autonomy is.

3.3.3 User interface

When we switch from single robot control to multi-robot control, the user interface becomes more complex and new problems arise, such as supporting group awareness and switching between the robot group and individual robots. However, many discussions of UI in SOSR control still can be used here. For instance, similar to the main modality discussion in Section 2.2.1, in HMRI we need to decide the suitable main modality or the awareness metaphor. As an example, [110] compares map-based and camera-based UI for multi-robot control in a 3D simulated world. In this Section, we will focus instead on three new problems in HMRI: the problem of selecting and switching between the group and the individual, the problem of group awareness, and the problem of complex awareness support.

[85] compares individual robot selection and whole group selection UIs. The authors summarize that, in multi-robot control, we need "flexibility to reallocate robotic resources or to compensate for suboptimal robot behavior." The four-robot control experiment shows that, with flexibility in controlling either an individual robot or the entire group, the participant won significantly more games without a significant difference in workload. A later study [104] of task-switching times reports that the switch time varied from two to seven seconds among different interface types, which demonstrates how UI can significantly impact effectiveness.

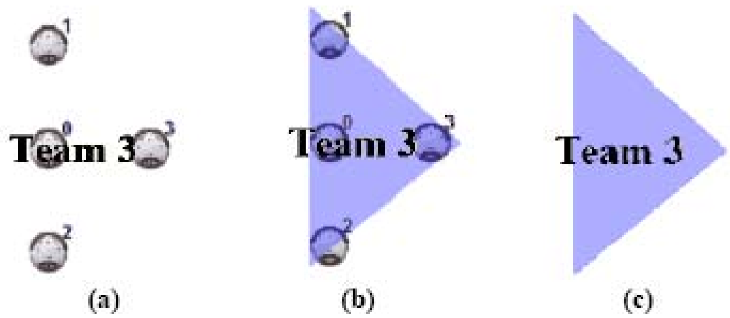


Figure 16. Team representations: (a) individual, (b) semi-transparent, (c) solid (reprinted from [45]).

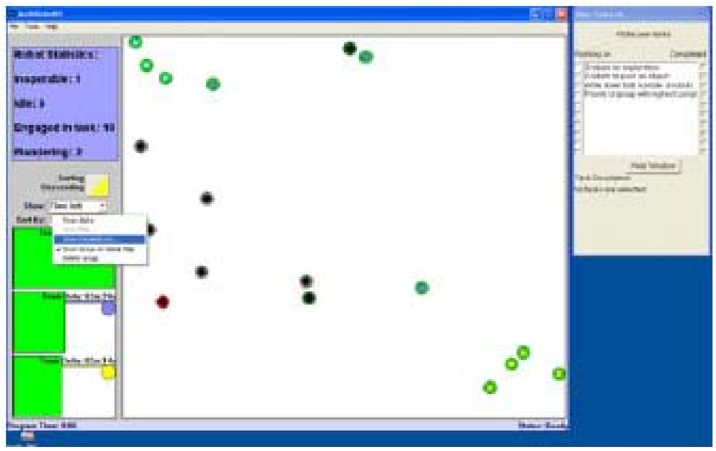


Figure 17. Task-list-based UI in which the user task list window is docked on the right [32].

At the team level, the UI should represent the team appropriately so that the user can maintain group awareness with the least mental workload. [45] proposes a team shape-based visualization that represents the robot team with various geometric shapes (Figure 16). The user study shows that the participants significantly prefer connecting members with semi-transparent or solid geometric shapes to the individual robot visualization. Recalling the awareness metaphors introduced in Section 3.2, possible enhancements to this representation include displaying nimbus and focus or adding interaction visualization to help the user acquire workspace awareness. Knowledge of the team-level tasks is another factor that impacts team-level planning and management. [32] demonstrates a task-list-based user interface (Figure 17) that provides the user with the tasks in queue, task descriptions, and task status. The usability study shows that task list can better guide users under ambiguous situations and improve overall performance. In addition, it makes users' control behaviors more predictable and decreases the overall workload.

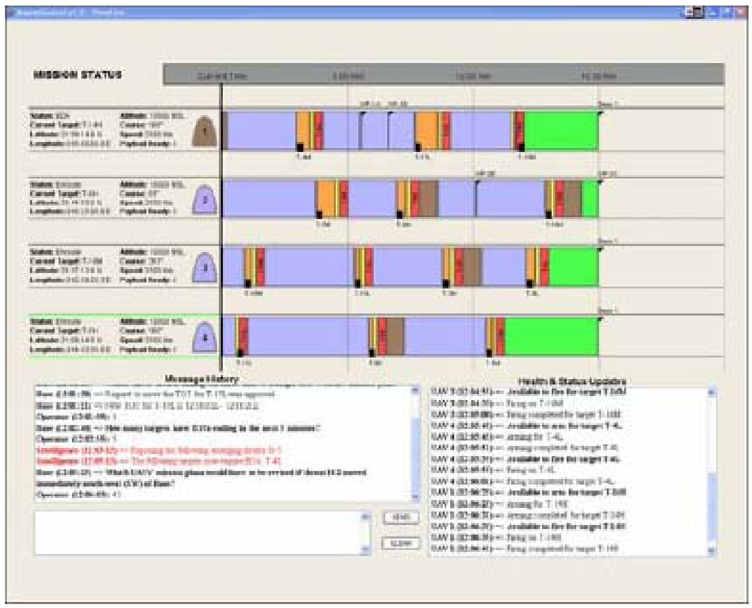


Figure 18. Timeline interface [21]

As mentioned before, under multi-robot control, situation awareness becomes complex and the user requires more time to make a decision. From this complexity the history problem emerges: to make a plan, we must know the group members' histories and predict their futures. In general, the larger the team is, the more complex the decision-making is, such that a larger time scale of history and prediction is needed. [21] presents an example of a user interface that accounts for time issues in the interface design, such as displaying the schedule and the time delay as well as associating the predicted workload with specific tasks (Figure 18). Displaying a group member's plan, such as the planned path, and that group member's current progress is one approach to representing a shorter time scale of history and the future. In a multi-robot user interface, fusing the individual robots' information allows us to mitigate the overall workload and to improve the overall SA. Figure 19 shows the recent effort to design a scalable interface that enables a single person to control up to nine robots. The user study reveals that, with the halo style interface, adding a robot does not linearly increase the demanded workload. Thus, the interface is scalable to the number of robots.

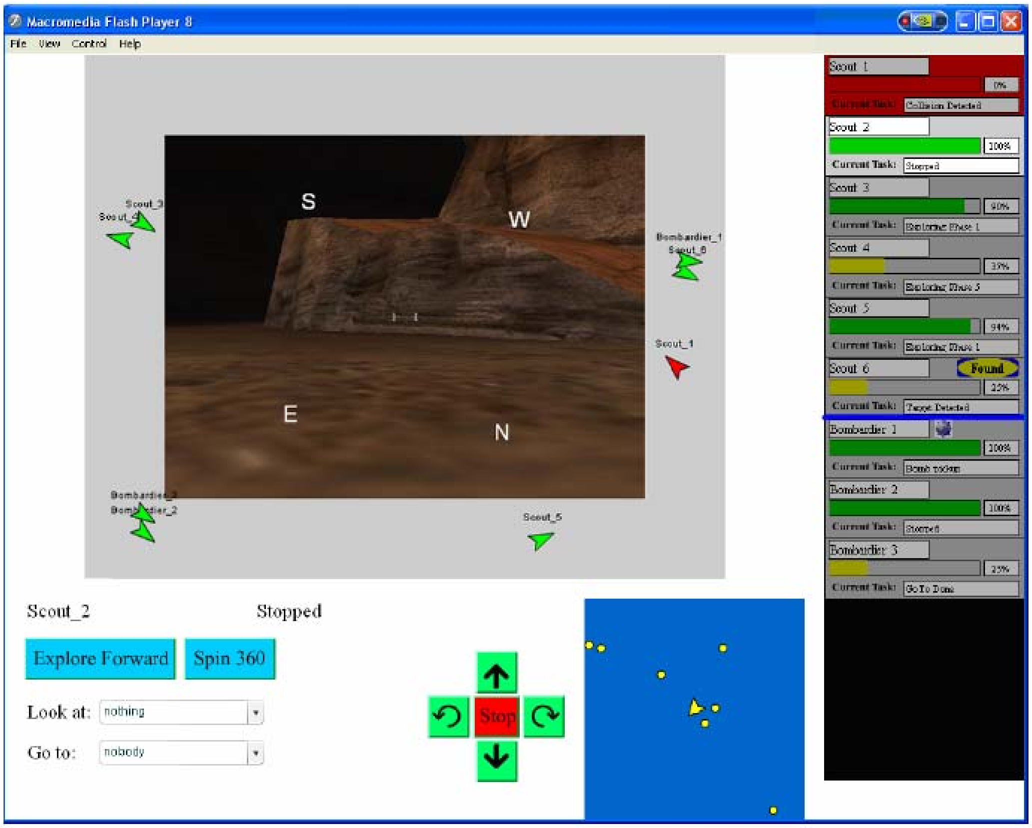


Figure 19. The halo scalable interface [46].

3.4 DISCUSSION

Table 2 lists the current studies in human-multi-robot interaction. In the following Sections, we summarize them in terms of the four elements in HRI, i.e., the human, the robotic system, the task, and the world.

3.4.1 Human

A single person controlling a group of robots is the dominant control style in current studies. Multiple operators controlling a single robot such that performance is significantly improved is reported in a few studies, including [9, 10]. N operators controlling M robots (where N>1, M>1) seems to be the most efficient way to allow multiple people to control a large group of robots. Unfortunately, the study of MOMR (multiple-operators multiple-robots control) is rare in HMRI.

Humans' limited cognitive capabilities are the main bottleneck in HMRI. To effectively control robots, the operator must make decisions, issue commands, and build and maintain awareness with the lowest possible mental workload. Shifting the workload to the robotic system is one solution. For example, if we control a large group of robots via issuing a region of interest, the workload is similar regardless of the number of robots because the operator only needs to draw the interest region. However, as the group size increases, the cooperation among the robots increases. Another approach to decrease the workload is that we help the operator by providing decision support and better presented or organized information. Indicating teams with a particular geometric shape and the presence of a task list and timeline are user interface enhancements that can mitigate human workload. Unfortunately, these studies are based on ad hoc tasks and robotic systems. An awareness study in HMRI is rare. We still lack an awareness model, similar to the one used in CSCW, that allows us to formally analyze awareness and use it to guide user interface design.

Social relationships among operators and between humans and robots present interesting problems that require further study. A study in social robots [8] shows that engagement can enhance a person's cognitive capabilities to allow him to "think and respond quickly." With a carefully designed interface, it is possible to achieve engagement in robot control and therefore potentially improve efficiency.

In most of the user studies, the robotic system was tested by novice users rather than by experts. This is appropriate for HRI because it is closer to real-life situations. However, a novice's learning curve is usually ignored or underestimated in these studies.

3.4.2 Robots

The level of autonomy (LOA) is the feature of the robotic system that most impacts the effectiveness of HMRI. The studies show that autonomy can decrease workload and can degrade SA and performance and that human involvement will increase the workload, SA, and performance. However, when the task or situation is complex, human involvement may instead have a negative effect on the overall performance. Many factors, such as the task, the environment, and the robots, will influence the optimal level of autonomy. Most of the current studies are based on specific tasks and robotic systems, which makes applying the result to another application difficult. The alternative approach to improving effectiveness in HMRI is utilizing adaptive autonomy. For instance, the studies in delegation-type interfaces show that the flexibility (the capability of manually changing the level of autonomy) can improve performance with a slightly increased workload and SA. However, we face the same problem in decomposing the task into sub-tasks and finding the corresponding optimal autonomy for each sub-task. With a decomposition study, we are able to use it to guide the operator, to provide decision support, or to invoke it in automation to decide when we should change the level of autonomy.

A small size and a homogeneous robot group are the main features of the robot organization style used in current HMRI studies. Because of the limitations on human cognitive abilities, controlling a mid- or large-sized robot group poses a challenge. To allow a person to control a large group of robots, we need a scalable interaction scheme such that both the autonomy capability and the user interface are scalable to a large number of robots. In particular, the complexity of the robotic system should not significantly increase and the autonomy capability should not significantly degrade as the number of robots increases. A distributed cooperation system, such as the Machnetta framework [94], is an example of a system that supports various sizes of robot groups. In terms of the human operator, the workload should increase only slightly as the robot group size increases so that the human operator is able to maintain control of the robots. The ideal interface is a task-oriented interface because human input is related to specific tasks rather than to a specific robotic system and because the workload remains constant for any given task. For instance, with the region of interest or sub-goal specification interface, the size of the robot group is irrelevant because the operator controls the robots via drawing a region or issuing a sub-goal according to the task and the current SA. This kind of interface requires a very high level of autonomy of the robotic system. Given a large robot group, the robotic system should be error- and failure-tolerant so that required human intervention in response to failure does not linearly increase with the robot group size. Another possible solution is to divide a large robot group into several sub-teams so that the human is able to deeply intervene with the robots via the robot sub-teams, thereby avoiding a large increase in mental workload. Similar to the military organization that allows a general to efficiently control multiple troops, building a robot organization, such as the 4D/RCS architecture [3], to control robots at different aggregation and abstract levels might be another, more intuitive way to control large robot groups.

Finally, a robot simulator is often used in HMRI studies. Unfortunately, most of the simulations are low-fidelity 2D simulations that largely ignore the dynamic and uncertain features in interaction study. Consequently, the SA problem, which is extensively studied in SOSR control, is not addressed in many of the current HMRI studies. A high-fidelity, 3D, dynamic-level robot simulation is lacking in the HMRI community.

3.4.3 Task and World

Table 2 shows that navigation and exploration comprise the predominant task studied in HMRI. In these tasks, a human user is usually involved via determining how and where to guide a robot to a specific destination. Perception task is another type task that requires a human user's deep involvement because the low-level control is usually based on human input, such as identifying a target. Because of the high human intervention demand, perception tasks are also typical HRI tasks and have been extensively studied in SOSR. Unfortunately, the study of perception tasks like search and rescue are relatively rare in HMRI. Management tasks are a typical task in multi-robot control domains that have been studied by several current researchers. Switching and regaining SA at different levels (i.e., individual, sub-team, team) of management is an interesting problem that deserves more attention.

The effect of the workspace is usually ignored in HMRI studies. However, it is important especially when we consider its effectiveness. For instance, the office-like, forest-like, and open-space environments have different impacts in map-building and path generation for robot navigation. Flat and rough terrain cause different SA problems in robot driving and require different levels of autonomy. When a robot moves around, some regions may be easier than others for the robot to navigate. Therefore, accounting for world complexity in system design and evaluation is necessary. In multi-robot control, although the different local world of the individual robot will require more cooperation, the different features of the local world might benefit overall performance. One of the trends in multi-robot control is combining UAVs and UGVs to improve SA and to benefit the robots' navigation. Few researchers realize the importance of measuring a world's complexity and features, and only a few approaches to such measurement have been proposed (see Section 1.1.3).

As mentioned before, a low-fidelity simulator is still the dominant simulation used in HMRI but we need a high-fidelity simulation of the world as well. Moreover, for interaction studies, fidelity in robot-world interaction simulations is required.

Table 2. Current HMRI studies .

Study	Task	World	Robots[8]	Interaction	Teaming
Fong et al. (2001): robotic system of collaborative control [35]	Surveillance & reconnaissance	Real world with flat terrain.	2 UGVs (PionnerAT & Pioner2-AT)	Dialog + waypoint control	Independent
Trouvain & Wolf (2002): user study of the impact of robot group size [111]	Navigation	2D simulated office world	2, 4, 8 UGVs (homogeneous)	Waypoint	Independent
Trouvain et al. (2003): user study of map based and camera based user interface [110]	Exploration	3D simulated outdoor world (graph rendering system)	1, 2, 4 UGVs (homogeneous)	Supervisory + waypoint control	Independent
Nielsen et al. (2003); Crandall et a (2005): user study of interaction schemes [79]	Exploration	2D simulated office like world	3 UGVs (homogeneous)	Teleoperate and landmark; Point to point and human snapper; Region of interest and sealing.	Independent
Olsen et al. (2004): Fan-out study [83, 84]	Exploration	2D simulated office like world	18 UGVs (homogeneous)	Goal specification + simple/bounce/plan level automation	Independent
	Exploration	Maze like real world	4 real robots (homogeneous)	Direction control + collision monitoring; Goal specification + auto exploration	Independent
Cummings & Mitchell (2005): Time management and scheduling [21]	Attack target	2D simulated world	4 UAVs (homogeneous)	Manual, passive, active, and super active decision support in planning	Independent

[8] A UGV is a ground robot, and a UAV is an aerial robot. An agent is the abstractly simulated robot with supernatural powers.

Parasuraman et al. (2005): delegation-type interface [85]	Capture the flag	2D simulated world (RoboFlag)	4, 8 Agents (homogeneous)	Autonomy, manual (waypoint), and mixed control in robots behavior and selection	Cooperative
Envarli & Adams (2005): User study of task lists [32]	Solve robot failures, team management	2D simulated world	18 Agents (homogeneous)	Team management and task assignment with minor decision support (additional task management constraints)	Cooperative
Schurr et al. (2005): Multi-agent system with human in the loop control [100]	Fire fighting	3D simulated world (DEFACTO)	4, 6, 10 Agents (homogeneous)	High level task allocation and strategy selection	Cooperative
Squire et al. (2006): Task switching time [104]	Capture the flag	2D simulated world (RoboFlag)	4, 6, 8 Agents (homogeneous)	Autonomy, manual (waypoint), and mixed control in robots behavior and selection	Cooperative
Wang et al. (2006): user study of cooperated robot team[9] [116]	Search	3D simulated indoor world (USARSim)	3 UGV (homogeneous)	Manual, mixed-initiative	Cooperative
Humphrey et al. (2006): user study of robot team visualization [45]	Robot selection and position identification	2D simulated world	4 x 4 Agents (4 teams with 4 robots in each team, homogeneous)	None (No robot control)	Independent
Humphrey et al. (2006): user study of scalable interface[10] [46]	Search	3D simulated outdoor world (USARSim)	6, 9 UGVs (heterogeneous)	Teleoperation and scripted behaviors	Independent

[9] This is our previous work. For more details, please see Section 4.3.
[10] This experiment is later than our multi-robot control experiment, which was conducted in 2005.

4.0 THE HMRI TESTBED AND PILOT STUDY

In this chapter, we describe our testbed for human multi-robot control study and our pilot experiment. Based on the literature review, our work focuses on human control of a cooperating robot team to pursue a search task, which is a typical HRI task that requires deep human involvement. We first introduce our robot simulator, which is intended to provide an HRI research platform in a virtual world for HRI researchers. Then we describe our multi-robot system that was based on an off-the-shelf multi-agent system and previous HRI research results. The system was built as a scalable robotic system to allow us to pursue our long-term goal of human control over a large robot group. Finally, we describe our primary experimental study of human control over a robot team searching for victims in a disaster environment.

4.1 USARSIM – THE ROBOT AND ENVIRONMENT SIMULATOR

4.1.1 Introduction

Although many robotic simulators are available, most of them have been built as ancillary tools for developing and testing control programs that are run on research robots. Simulators built before 2000, including [55] and [57], typically have low-fidelity dynamics for approximating the robot’s interaction with its environment. More recent simulators including

ÜberSim [7], a soccer simulator, Gazebo [38], and the commercial Webots [23] that use the open-source Open Dynamics Engine (ODE) to approximate physics and kinematics more precisely. The ODE, however, is not integrated with a graphics library, which forces developers to rely on low-level libraries like OpenGL. This limits the complexity of the environments that can practically be developed and effectively precludes the use of many of the specialized rendering features of modern GPUs. Both high-quality graphics and accurate physics are needed in HRI research because the operator's tasks depend strongly on remote perception [117], which requires accurate simulation of camera video feedback as well as interaction with automation, which in turn requires accurate simulation of sensors, effectors, and control logic.

USARSim is a high-fidelity simulation of urban search and rescue (USAR) robots and environments and is intended as a research tool for the study of HRI and multi-robot coordination. USARSim supports HRI by accurately rendering user interface elements (particularly camera video feedback), accurately representing robot automation and behavior, and accurately representing the remote environment that links the operator's awareness with the robot's behaviors.

4.1.2 Game-Engine based simulation

Real-time "out the window" or "through the camera" simulations have classically been difficult, time-consuming, and expensive to build because they require specialized hardware and personnel. The cost of developing such simulations has grown so much that even in the gaming industry developers can no longer rely on recouping their entire investment from a single game. This has led to the emergence of game engines, or modular simulation code, that can be used for families of similar games. The separation of game logic and rules from simulation dynamics and environmental data allows the core code to be reused for more general simulations. In addition to affordability, today's game engines also offer advanced graphical displays, realistic environments, accurate physics, and dramatically reduced development times [58].

USARSim uses Epic Games' Unreal Engine 2 [33] to provide a high-fidelity simulator at low cost. Unreal is one of the leading engines in the "first-person shooter" genre and is widely used in the gaming industry. It is also gaining a strong following in the academic community as more researchers use it in their work. Recent academic projects included creating VR displays [48], studying AI techniques [62], and creating synthetic characters [107]. In addition to the egocentric perspective, there are several other features of the Unreal Engine that make it particularly appealing for HRI research. These features include graphics, a physics engine, an authoring tool, game programming, and networking, each of which is discussed in more detail below.

In terms of graphics, the Unreal Engine provides fast, high-quality, 3D scene rendering that supports mesh, texture, lighting, and material (e.g., reflective, transparent, and semi-transparent surfaces) simulation, which allow us to simulate realistic camera video. This is one of the most critical features in current approaches to human control of mobile robots.

In terms of a physics engine, the Unreal Engine integrates MathEngine's Karma Engine [66] to support high-fidelity rigid body simulation in instances of collision, friction, joint simulation and force, and torque modeling. This feature allows the simulation to replicate both the physical structure of the robot and its interaction with the environment.

In terms of an authoring tool, the Unreal Engine provides a real-time design tool, UnrealED, for developers to build their own 3D models and environments from scratch or by importing models from other popular modeling tools, such as Maya and 3D Studio Max. UnrealED permits HRI researchers to accurately model both robots and their environments.

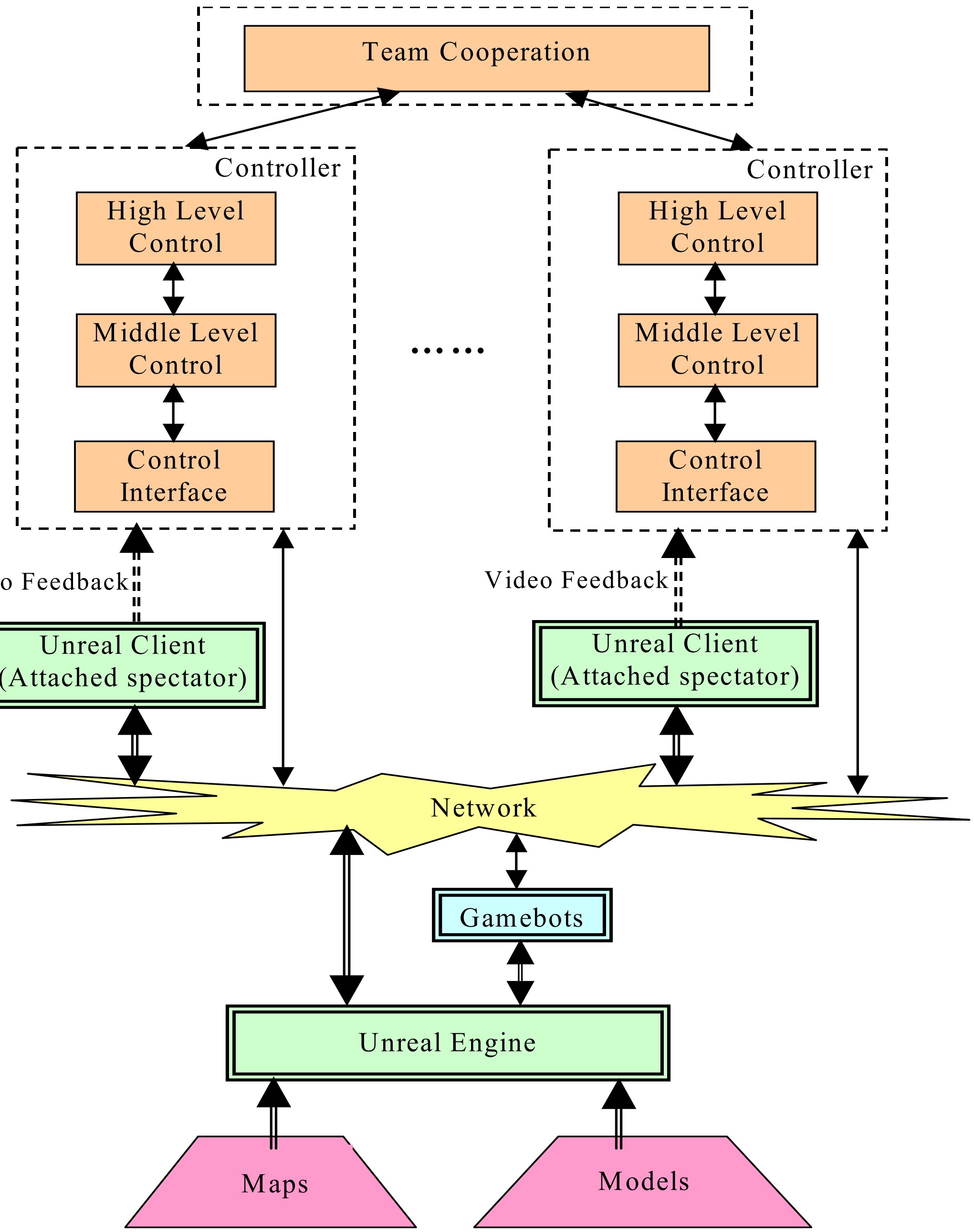


Figure 20. USARSim architecture.

In terms of game programming, the Unreal Engine provides an object-oriented scripting language, UnrealScript, that allows researchers to control the game logic. This affords the ability to customize the interaction with the simulation in order to match the specifics of desired robot behaviors.

Finally, in terms of networking, the Unreal Engine uses efficient client-server architecture to support multiple players. This embedded networking capability allows USARSim to support human control of multiple robots without modification.

Figure 20 shows the Unreal Engine's components and the expandable library of robot-themed models, environments, and control interfaces to acquire sensor data and issue the commands that we have added to create the USARSim simulation.

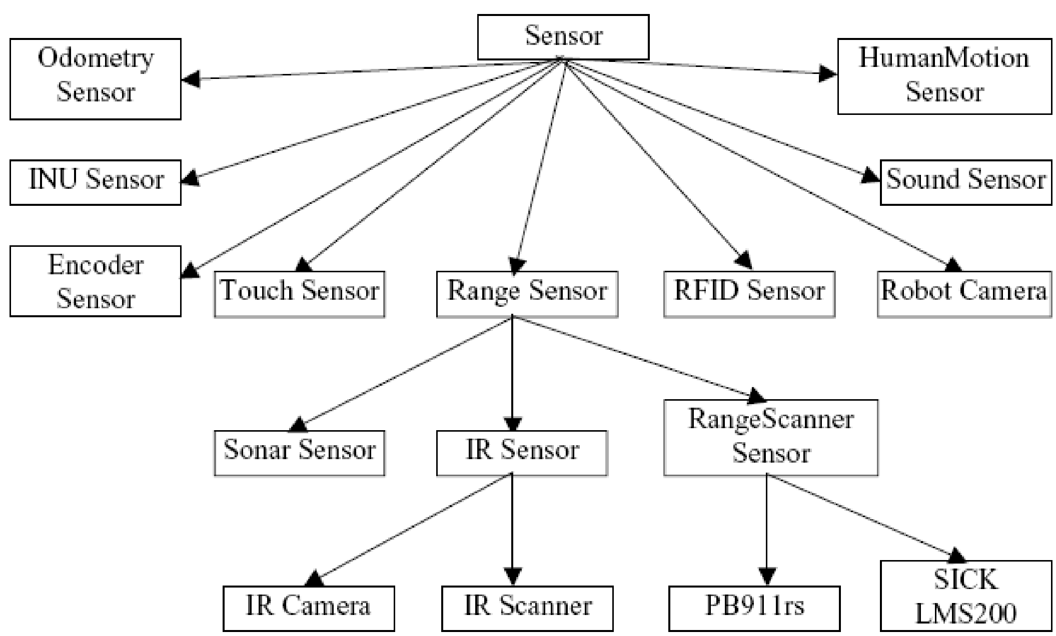


Figure 21. USARSim sensors.

More specifically, USARSim provides a generic robot model that simulates a robot at the joint level and that enables us to create our own robots via assembling robot parts and mounting sensors with minor Unreal programming. The current version, which includes contributions from other researchers, provides ten wheeled robots, two legged robots, one submarine, and one helicopter. The sensor simulation in USARSim provides the perception from a robot's view. USARSim uses a hierarchical architecture to build sensors and to enable adding new sensors without a deep understanding of the Unreal Engine. Figure 21 shows the current available sensors in USARSim. USARSim provides special world simulations, such as mirrors, wire grids, and transparent and semi-transparent boards, in addition to supporting high-fidelity world simulations. Shown in Figure 22 are the real and simulated NIST reference arenas [49]. Finally, USARSim applies a modification of GameBots [52] to communicate with the virtual robot via a network, which makes the simulator independent of both the programming language and the computer platform. Moreover, USARSim provides tool kits to help users control robots via popular software like Player and Pyro. More details of USARSim can be found in the user manual [113]. It is currently maintained at SourceForge (http://sourceforge.net/projects/usarsim) by NIST with more than 19,000 downloads. It is also the official platform of the RoboCup Virtual Robot competition since 2005.

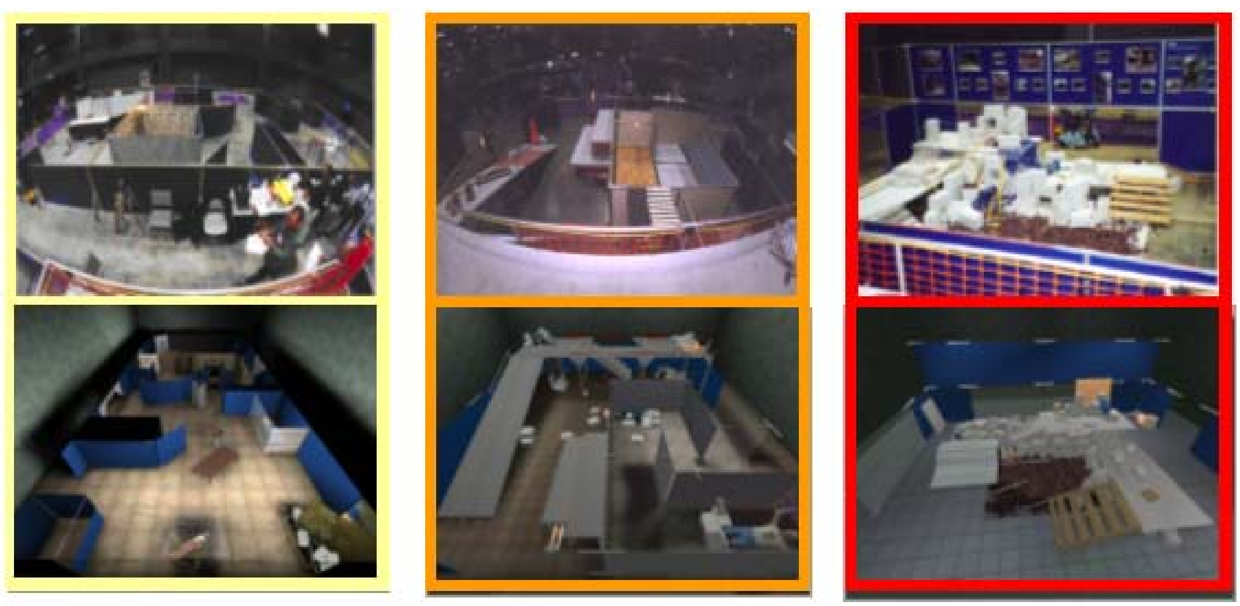


Figure 22. The real (top) and simulated (bottom) NIST reference arenas.

4.1.3 Evaluation

This Section evaluates USARSim according to three aspects: (1) the validation that compares the real world with the simulated world, (2) the comparison of USARSim with other simulators, and (3) USARSim's applications in HRI research. Many of the studies on sensor data validation and simulator comparison that are introduced here are conducted by the third part research groups.

The validation of USARSim can be conducted from inside or outside. The inside validation compares the simulated data with the real-world data. High-quality simulation implies a very closed data set between the real and virtual worlds. [13] selects to validate the most substantial sensor in HRI, which is the range sensor, because most robot control is based on it. The comparison of the world features extracted from the range data shows a high correlation between the simulated and real sensor data. The camera is another essential sensor in HRI study. [12] applies an approach similar to the previous one, i.e., a comparison of sensors via the popular data-processing algorithms in robot control in order to validate the camera images. Both edge detection and OCR testing demonstrate that, although the images in the virtual world show a lower level of noise than the images in the real world, close results were found in the measurement of the extracted features. We validate USARSim from the human user's side by comparing the robot control behaviors [115]. In one experiment, the participants drove PER robot on wood, paper, and lava terrains under teleoperation or waypoint control modes to pass through clear and obstructed environments in real or virtual worlds. Although the sample size was very small (five subjects in each condition), the results reveal a similar learning trend, a similar terrain effect, and very close task completion times in both types of worlds. However, degraded depth perception was found in the simulation.

Clearly, more validations, especially those that involve the robot's dynamic features, are needed.

[17] summarizes the current available commercial and open-source robot simulators in Table 3 according to their physical fidelity, functional fidelity, ease of development, and cost. Physical fidelity measures how the physical simulation behaviors (e.g., looks, sounds, and feels) are like the real world. Functional fidelity is the degree to which the simulated actor acts in the same way as the real equipment would in performing a task. The ease of development refers to how easily the simulator can simulate a new environment or robot and how easily the developer can extend or customize the simulator. Finally, cost is the commercial and time cost for using the simulator. Table 3 shows that USARSim appears at the top of the list. However, as the authors point out, all of the simulators model sensors by adding random noise and ignoring effects from the environment. In HRI research, this is a significant oversight and should guide future development of USARSim.

Table 3. Mobile robot simulators (reprinted from [17]).

Simulator	**Physical Fidelity**	**Functional Fidelity**	**Ease of Development**	**Cost**
USARSim	High	High	Easy	Low
X-Plane	High	High	Easy	Low
FlightGear	Medium	Medium	Medium	Free
MS Flight Simulator	Low	Medium	Medium	Low
Webots	Low	Medium	Easy	Low
Simbad	Low	Low	Medium	Free
Player/Stage/Gazebo	Low	Low	Easy	Free
EyeWyre	Low	Medium	Easy	Low
MS Robotics Studio	High	High	Medium	Medium
MATLAB & Simulink	Low	Low	Easy	High
MissionLab	Low	Low	Easy	Free
SimRobot	High	Low	Medium	Free
SubSim	Medium	Low	Easy	Free

The applications of USARSim implicitly measure the successfulness of our work. Herein lists the main published studies that are based on our simulator. Our lab utilized USARSim to simulate the lack of reference and confused reference environments to analyze attitude SA [114], and in [44] we used USARSim to study camera-based exploration. The Human-Centered Machine Intelligence Lab of Brigham Young University used USARSim to compare camera-based and map-based navigation [80]. The Human-Machine Teaming Laboratory of Vanderbilt University uses USARSim to study human control of multiple robots [46]. USARSim is being used in the robot control domain as well. For example, the Knowledge-Based Systems Research Group of the University of Osnabrück in Germany is using it in 6D SLAM research and education [2]. The University of Rome "La Sapienza", in Italy used USARSim in soccer robot simulations [123]. Utilization in these researches shows that USARSim is successfully accepted in the HRI community and that it meets our design objectives.

4.2 MRCS – THE MULTIROBOT CONTROL SYSTEM

4.2.1 Introduction

The completion of building our testing bed, USARSim, led to the next step of acquiring a robot control system that could be used in our series of HRI studies. To suit our experimental studies, this system had to be scalable to allow us to control different numbers of robots, reconfigurable to enable us to study different human-robot interfaces, and reusable to facilitate testing different control algorithms. With these requirements in mind, we selected the distributed proxy-based multi-agent framework Machinetta as our system's baseline, which we will introduce in the next Section.

In human-robot interaction, how and when the operator intervenes in the robotic system are the two predominant issues [30]. How a human works with the system is a function of the level of autonomy (LOA), which describes the static function assignments between the human and the robot. The LOA can range from full manual control to full autonomy, with intermediate levels of LOA generally being superior to full autonomy or full manual control. This is because an LOA that is too high leads to degradation in manual or mental skill, loss of situation awareness, decision bias, and vigilance decrement and because the low LOA of full manual control leads to high mental demand, human decision bias, complacency, boredom, and inconsistent control behavior, all of which degrade performance. In systems with adaptive autonomy (AA), the allocation of control between the human and the robot can be dynamically changed and is usually triggered by a critical event, performance measurement, operator's workload, or the operator model. In the current human-robot control system, we utilized a middle LOA and a critical-event-based AA to build a multi-robot control system (MrCS) that allows a human to work with the cooperating robot team to construct a mixed-initiative human-robots team.

4.2.2 Team work

The teamwork algorithms used in MrCS are general algorithms that have been shown to be effective in a range of domains [108]. To take advantage of this generality, the emerging standard approach is to encapsulate the algorithms in a reusable software proxy. Each team member has a proxy that he works with closely, and the proxies work together to implement the teamwork. The current version of the proxies utilized in MrCS is Machinetta [94], which is implemented in Java and is freely available on the internet. This type of proxy differs from many other "multi-agent toolkits" in that it provides the coordination algorithms, e.g., algorithms for allocating tasks, as opposed to the infrastructure, e.g., APIs for reliable communication.

The Machinetta software consists of five main modules, three of which are domain-independent and two of which are tailored for specific domains. The three domain-independent modules are designed for coordination reasoning, maintaining local beliefs (state), and adjustable autonomy. The domain-specific modules are designed for communication between proxies and communication between a proxy and a team member. These modules interact with each other only via the local state with a blackboard design and are designed to be "plug and play." This means, for examples, that new adjustable autonomy algorithms can be used with existing coordination algorithms.

The coordination reasoning is responsible for reasoning about interactions with other proxies, thus implementing the coordination algorithms. The adjustable autonomy algorithms reason about the interaction with the team member, providing the possibility for the team member rather than the proxy to make any coordination decision. For example, the adjustable autonomy module can reason that a decision to accept the role of rescuer for a civilian in a burning building should be made by the human who will enter the building rather than the proxy. In practice, the overwhelming majority of coordination decisions are made by the proxies, and only key decisions are referred to the human operators. Teams of proxies implement team-oriented plans (TOPs) which describe joint activities to be performed in terms of the individual roles to be performed and any constraints on those roles. Typically, TOPs are instantiated dynamically from TOP templates at run-time when pre-conditions associated with the templates are filled. Constraints between these roles specify interactions, such as the required execution ordering and whether one role can be performed if another is not currently being performed. It is important to note that TOPs do not specify the coordination or communication required to execute a plan. Instead, the proxy determines the coordination that should be performed.

Current versions of Machinetta include state-of-the-art algorithms for plan instantiation, role allocation, information sharing, task deconfliction, and adjustable autonomy. Many of these algorithms utilize a logical associates network that statically connects all team members. The associates network is a scale free network which allows the team to balance the complexity of needing to know about all the team and maintaining cohesion. The associates network's key algorithms, including role allocation, resource allocation, information sharing, and plan instantiation, are based on the use of tokens that are "pushed" onto the network and routed to where they are required by the proxies. For example, the role allocation algorithm LA-DCOP [93] represents each role to be allocated with a token and pushes the tokens onto the network until a sufficiently capable and available team member is found to execute the role. The implementation of the coordination algorithms uses the abstraction of a simple mobile agent to implement the tokens, leading to robust and efficient software.

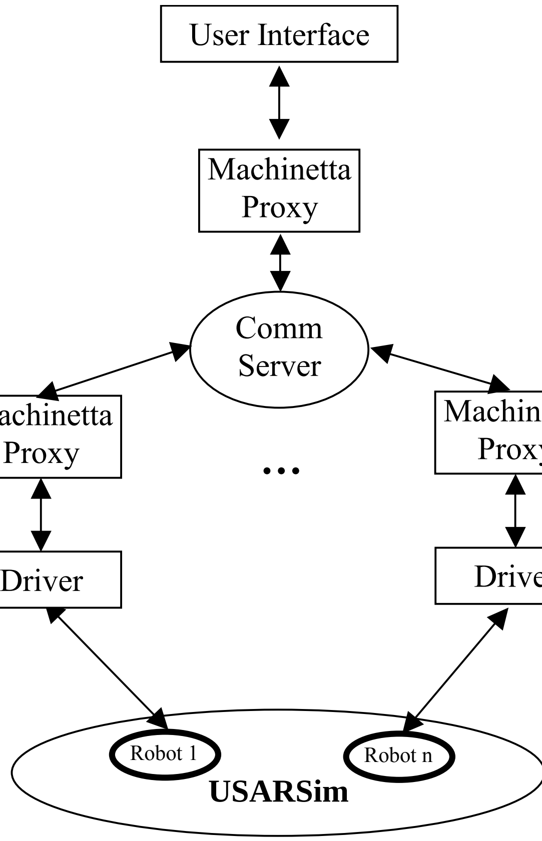


Figure 23. MrCS architecture.

4.2.3 MrCS

The system architecture of MrCS is shown in Figure 23. Each robot connects with Machinetta through a robot driver that controls the robot on both low and middle levels of control. For low-level control, it serves as a broker that translates robot sensory data into local beliefs and that translates the exploration plan into robot control commands (e.g., wheel speed control). For middle-level control, the driver analyzes robot sensory data to perceive its states and local environment. Then, based on this perception, the driver overrides the control commands when it is necessary to ensure safe movement. Possible adjustments include changing the direction of motion to avoid obstacles and recovering from becoming immobilized and from a dangerous pose. When the robot is in an idle state, laser data analysis allows the driver to generate potential exploration plans (e.g., the destination and the path to the destination). In addition, when the robot senses a potential victim[11], the driver immediately stops the robot and generates a plan to inspect the potential victim. However, instead of executing the plans immediately, the driver sends them to the Machinetta proxy to trigger TOPs. With Machinetta's role allocation algorithm, the robots and the human cooperate with each other to find the "best" robot to execute a plan. Here the "best" robot is defined as the robot that can find a route to the destination with the least cost, i.e., the shortest weighted travel length.

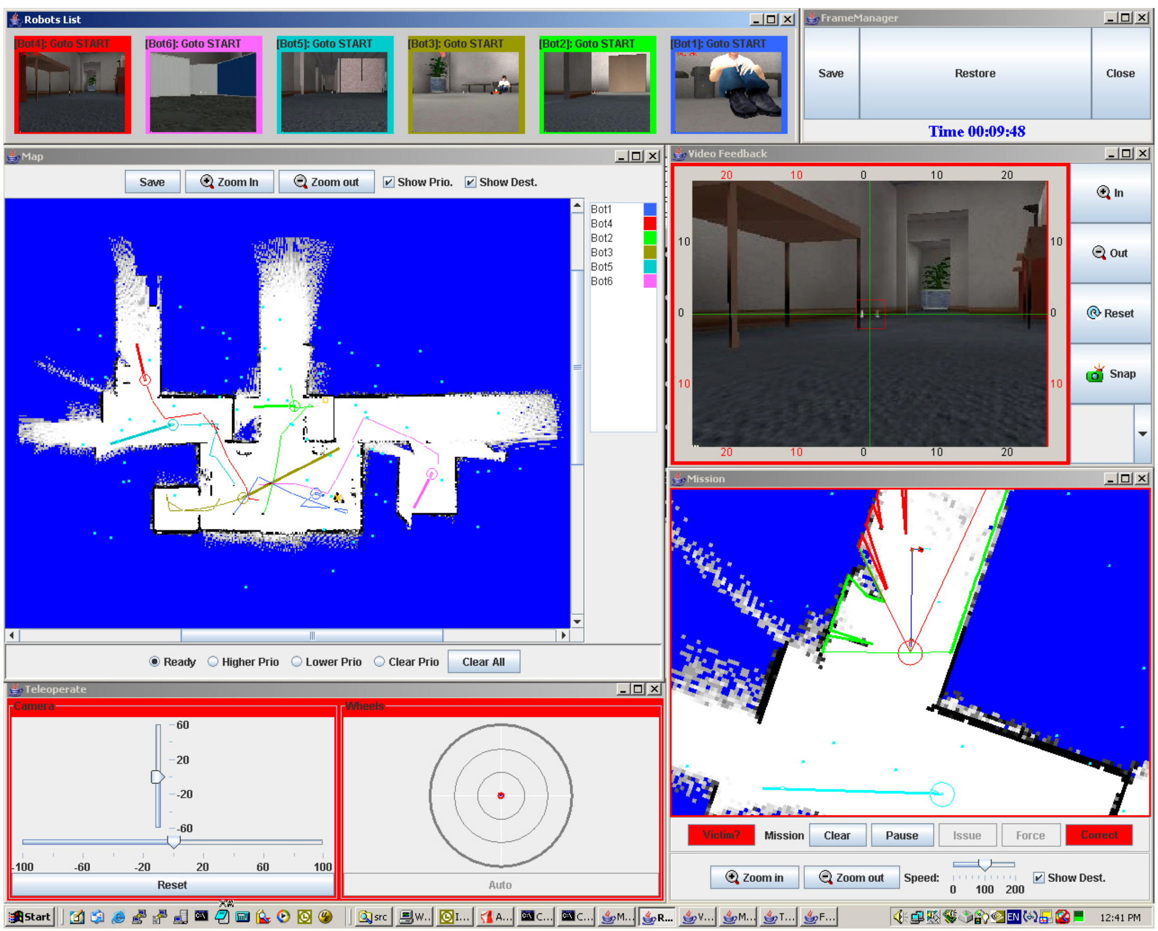


Figure 24. MrCS user interface.

---

[11] This functionality was added for RoboCup. In the competition, a faked "super" victim sensor was introduced to allow a robot automatically sense a potential victim.

The operator connects with Machinetta through the user interface agent. This agent collects the robot team's beliefs and visually represents them on the interface. It also transfers the operator's commands in the form of a Machinetta proxy's beliefs and passes them to the proxies network to allow human intervention in the loop cooperation. The operator can intervene with the robot team on three levels. On the lowest level, the operator takes over an individual robot's autonomy to teleoperate it. On the middle level, the operator interacts with a robot via editing its exploration plan. For example, the operator is allowed to delete a robot's plan to force it to stop and regenerate a plan or issue a new plan (a series of waypoints) to change its exploration behavior. On the highest level, the operator intervenes with the entire robot team via issuing priority areas. The priority will impact the cost calculation in role allocation and therefore affects the regions that the robots will explore.

In this human-robot team, the human maintains the highest authority to adjust the robot team's behavior. For example, the human can change a plan during plan execution, and this plan can be further adjusted by the robot to avoid obstacles or a dangerous pose. When critical events, such as sensing a potential victim or being in a dangerous pose, occur, the robot adjusts its own behavior and informs the operator. In this case, the robot initiates the interaction and the operator can either accept the robot's adjustment or change the robot's plan. One of the challenges in a mixed-initiative system is that the user may fail to maintain situation awareness of the robot team and of the individual robots when control switching and may therefore make faulty decisions. Moreover, as the team size increases, the interventions from the robots may overwhelm the operator's cognitive resources [68] and the operator may be limited to reacting to the robots instead of proactively controlling the robots [110]. We address these issues in the user interface design described below.

The user interface of MrCS is shown in Figure 24. The interface is reconfigurable to allow the user to resize the components or change the layout. Shown in the figure is a configuration that we used in the RoboCup 2006 competition in which a single operator controls six robots. On the upper and center portions of the left-hand side are the robot list and team map panels, which show the operator an overview of the team. The destination of each of robot is displayed on the map to help the user perceive team performance. Using this display, the operator is also able to control regional priorities by drawing rectangles on the map. On the center and lower portions of the right-hand side are the camera view and mission control panels, which allow the operator to maintain situation awareness of an individual robot and to edit its exploration plan. On the mission panel, the map and all nearby robots and their destinations are represented to provide partial team awareness so that the operator can switch between contexts while moving control from one robot to another. The lower portion of the left-hand side is a teleoperation panel that allows the operator to teleoperate a robot. On the interface, interruptions from the robots are mitigated by using principles of etiquette in user interface design [76]. When the robot needs the operator's attention, such as when sensing a victim or being in dangerous pose, the system will not display a pop-up window but will instead temporarily change the mission panel's size and background color or flash the robot's thumbnail picture in the robot list panel to inform the operator that a robot needs to be checked. This silent form of alert allows the operator to work at his own pace and respond to the robots when able.

We utilized MrCS in the RoboCup 2006 virtual robot competition. The four days of practice showed that, with mixed-initiative control and a simple cooperation algorithm (avoiding duplicate exploration effort), a single operator can control six robots. The overall performance during the competition was superior to other human-involved systems, and the final score was comparable to other full autonomy systems even our score was dived by four because of needing an operator [5].

4.3 IMPACT OF AUTONOMY – THE PILOT STUDY

4.3.1 Introduction

This pilot study investigates human interaction with a cooperating team of robots that performs search-and-rescue task. It compares the performance of autonomous teams, manually controlled robots, and operators interacting with a cooperating team in order to identify the contributions of each to system performance. Table 2 on page 89 organizes details of recent MRS studies. All were conducted in simulation and most involve navigation tasks rather than search tasks. This is significant because a search task using an onboard camera requires greater shifts between contexts than a navigation task, which can more easily be performed using a single map display [9, 80]. Our experiment uses USARSim because it provides a physics-based simulation of the robot and the environment that accurately reproduces mobility problems caused by uneven terrain [115], as well as hazards like rollover [114], and provides accurate sensor models for laser range-finders [13] and camera video [12]. This level of detail is essential to posing realistic control tasks likely to require intervention across levels of abstraction. Previous studies have not addressed the issues that arise from human interaction with a cooperating robot team within a realistically complex environment. Results from a 2D simulation [85, 104], for example, are unlikely to incorporate tasks requiring low-level assistance to robots, and experiments with non-cooperating robots [19, 79, 110, 111] miss the effects of this aspect of autonomy on performance and HRI.

4.3.2 Method

4.3.2.1 Participants

Fourteen paid participants, ranging from 19 to 35 years old, were recruited from the University of Pittsburgh community. None had prior experience with robot control, although most were frequent computer users. Only two reported playing computer games for more than one hour per week. The participants' demographic information and experience are summarized in Table 4.

Table 4. Sample demographics and experiences.

	Age		**Gender**		**Education**	
	19	20~35	Male	Female	Currently Undergraduate	Complete Undergraduate
Participants	2	12	5	9	10	4

	Computer Usage (hours/week)				**Game Playing (hours/week)**			
	<1	1-5	5-10	>10	<1	1-5	5-10	>10
Participants	0	2	7	5	6	7	1	0

	Mouse Usage for Game Playing		
	Frequently	Occasionally	Never
Participants	8	6	0

4.3.2.2 Procedure

The experiment began with a collection of the participants' demographic data and computer experience. Each participant then read standard instructions on how to control robots via MrCS. In the subsequent ten-minute training session, each participant practiced each control operation and attempted to find at least one victim in the training arena under the guidance of the experimenter. Each participant then began a twenty-minute session in Arena 1, followed by a short break and a twenty-minute session in Arena 2. At the conclusion of the experiment, each participant completed a questionnaire.

4.3.2.3 Experimental Design

In the experiment, participants were asked to control three P2DX robots (Figure 25) simulated in USARSim in order to search for victims in a damaged building. Each robot was equipped with a pan/tilt camera with a 45-degree FOV and a front laser scanner with 180-degree FOV and resolution of one degree. The participant interacted with the robots through MrCS using the fixed user interface shown in Figure 27. When a victim was identified, the participant marked its location on the map. The testing worlds were simulated versions of the NIST Reference Test Arena, Yellow Arena [49]. Two similar testing arenas (Figure 26) were built using the same elements but with different layouts. In each arena, fourteen victims were evenly distributed in the world. We added mirrors, blinds, curtains, semi-transparent boards, and wire grid to increase the difficulty of situation perception. Bricks, pipes, a ramp, chairs, and other debris were placed in the arena to challenge mobility and SA in robot control. Figure 25 shows a corner of the testing world.

Figure 25. P2DX robot.

a) Arena-1 b) Arena-2

Figure 26. Simulated testing arenas.

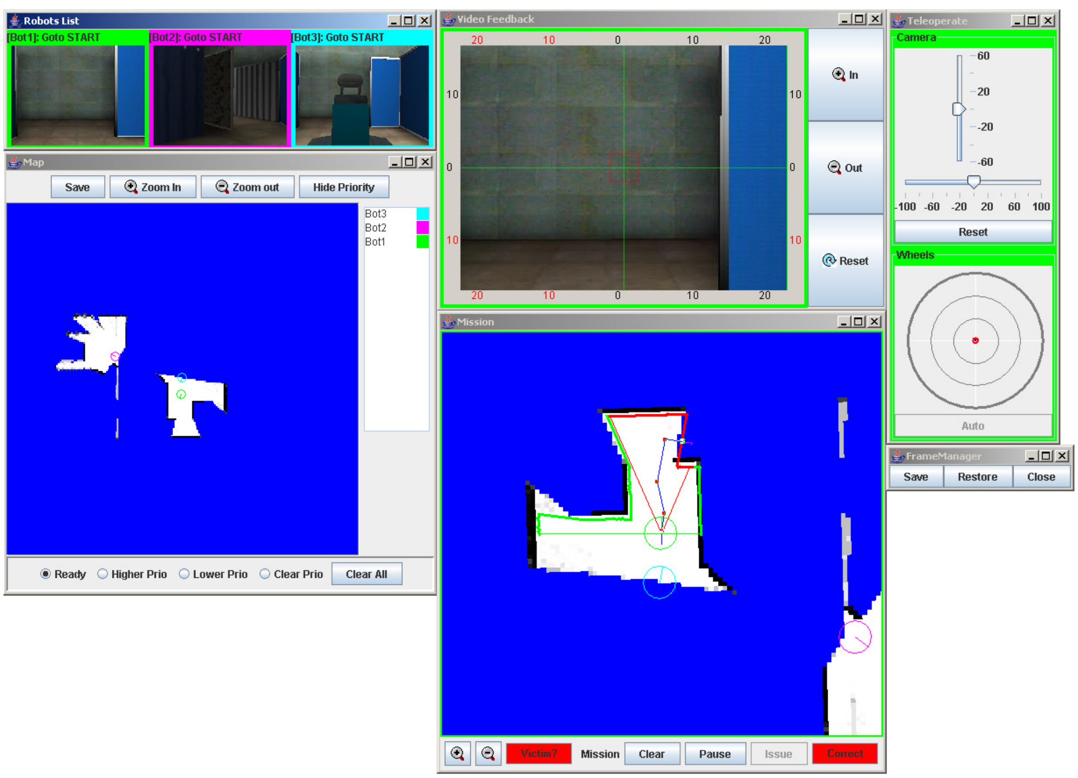


Figure 27. User interface used in the experiment.

We used a within-subjects design with counter-balanced presentation to compare mixed-initiative and manual control conditions. Under the mixed-initiative control condition, the robots analyzed their laser range data to find possible exploration paths. They cooperated with one another to choose execution paths that did not duplicate efforts. While the robots autonomously explored the world, the operator was free to intervene with any individual robot by issuing new waypoints, teleoperating, or panning or tilting its camera. When the operator's command was completed or stopped, the robot would return to auto mode. Under the manual control condition, robots could not autonomously generate paths and there was no cooperation among robots. The operator controlled a robot by giving it a series of waypoints, directly teleoperating it, or panning or tilting its camera. As a control for the effects of autonomy on performance, we conducted "full autonomy" testing as well. Because MrCS does not support victim recognition, based on our observations of participants' victim identification behaviors, we defined detection to have occurred for victims that appeared on camera for at least two seconds and occupied at least 1/9 of the thumbnail view. Because of the high fidelity of the simulation and the randomness of paths picked through the cooperation algorithms, robots explored different regions on every test. Additional variations in performance occurred due to mishaps such as a robot becoming stuck in a corner or bumping into an obstacle, which caused its camera to point to the ceiling so that no victims could be found. Sixteen trials were conducted in each area to collect data comparable to that obtained from human participants.

4.3.3 Results

In this experiment, we studied the interaction between a single operator and a robot team in a realistic interactive environment where human and robots must work tightly together to accomplish a task. We first compared the impact of different levels of autonomy by evaluating the overall performance as revealed by the number of found victims, the explored areas, and the participants' self-assessments. For the small robot team with 3 robots, we expected similar results to those reported in [19, 79, 111] that although autonomy would decrease workload, it would also decrease performance because of poorer situation awareness (SA). How a human distributes attention among the robots is an interesting problem especially when the human is deeply involved in the task by performing low level functions, such as identifying a victim, which requires balancing between monitoring and control. Therefore, in addition to overall performance measures, we examine: 1) the distribution of human interactions among the robots and its relationship with the overall performance, and 2) the distribution of control behaviors, i.e. teleoperation, waypoint issuing and camera control, among the robots and between different autonomy levels, and their impacts in the overall human-robot performance. Trust is a special and important problem arising in human-automation interaction. When the robotic system can't work as the operator expected, it will influence how the operator controls the robots and hereby impact the human-robot performance [56, 86]. In addition, because of the complexity of the control interface, we anticipated that the ability to use the interface would impact the overall performance as well. At the end of this section, we report participants' self-assessments of trust and capability of using the user interface, as well as the relationship among the number of found victims and these two factors.

4.3.3.1 Overall measurement

All 14 participants found at least five of a possible 14 (36%) victims in each of the arenas. The median number of victims found was seven and eight for Arenas 1 and 2, respectively.

Two-tailed t-tests found no difference between the arenas both for the number of victims found and for the percentage of the arena explored. Figure 28 shows the distribution of victims discovered as a function of area explored. These data indicate that participants exploring less than 90% of the area consistently discovered five to eight victims while those covering greater than 90% of the area discovered between half (seven) and all (14) of the victims.

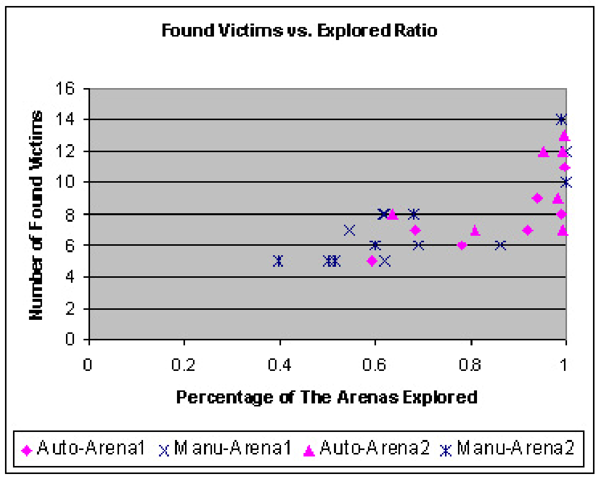


Figure 28. Victims as a function of area explored.

Within-participant comparisons found that wider regions were explored in mixed-initiative mode, $t(13) = 3.50$, $p < .004$, with a marginal advantage for mixed-initiative mode, $t(13) = 1.85$, $p = .088$, in the number of victims found. Comparing the full autonomy and mixed-initiative conditions, two-tailed t-tests found no difference ($p = 0.58$) in the explored regions. However, under the full autonomy condition, the robots explored significantly, $t(44) = 4.27$, $p < .001$, more regions than under the manual control condition (Figure 29 left). Using two-tailed t-tests, we found that participants found more victims under the mixed-initiative and manual control conditions than under the full autonomy condition with $t(44) = 6.66$, $p < .001$, and $t(44) = 4.14$, $p < .001$, respectively (Figure 29 right). The median number of victims found under the full autonomy condition was five.

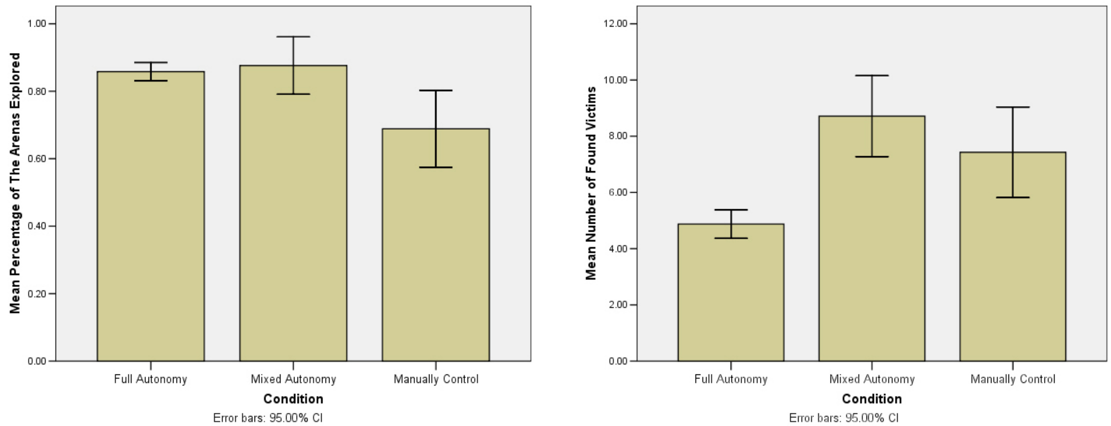


Figure 29. Regions explored (left) and victims found (right) by mode.

In the post-test survey, eight of the 14 (58%) participants reported that they were able to control the robots although they had problems in handling some components. The remaining participants thought that they used the interface very well. Comparing the mixed-initiative and manual control conditions, most participants (79%) rated team autonomy as providing either significant or minor help. Only one of the 14 participants (7%) rated team autonomy as making no difference, and two of the 14 participants (14%) judged team autonomy to worsen performance.

4.3.3.2 Human interactions

Participants intervened to control the robots by focusing on an individual robot and then issuing commands. Measuring the distribution of attention among the robots as the standard deviation of the total time spent on each robot, no difference ($p = .232$) was found between the mixed-initiative and manual control conditions. However, we found that, under the mixed-initiative condition, the same participant switched robots significantly more often than under the manual mode ($p = .027$). The post-test survey showed that most participants switched robots using the "Robots List" component. Only two of the 14 participants (14%) reported switching robot control independently of this component.

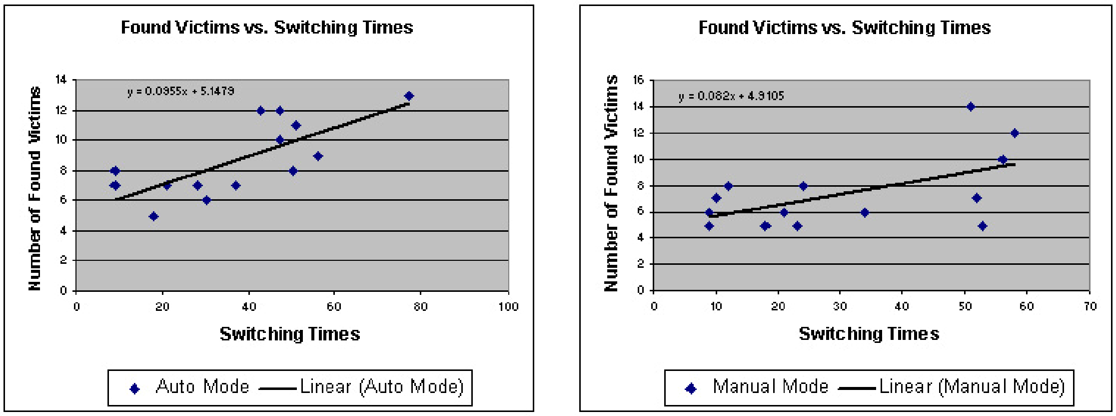


Figure 30. Victims vs. switches under mixed-initiative (left) and manual control (right) modes.

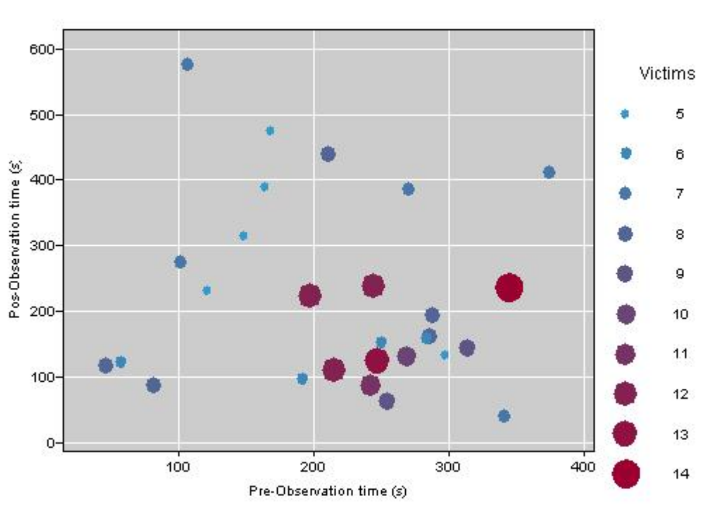


Figure 31. Pre- and post-observation time vs. found victims.

Across participants, the frequency of shifting control among robots explained a significant proportion of the variance in the number of victims found for both mixed-initiative, R2 = .54, F(1, 11) = 12.98, p = .004, and manual, R2 = .37, F(1, 11) = 6.37, p < .03, modes (Figure 30).

An individual robot control episode begins with the pre-observation in which the participant collects the robot's information and then makes a control decision and ends with the post-observation phase in which the operator observes the robot's execution and decides to turn to another robot. Using a two-tailed t-test, no difference was found in either total pre-observation time or total post-observation time between mixed-initiative and manual control conditions. The distribution of found victims among pre- and post-observation times (Figure 31) shows, however, that the proper combination can lead to higher performance.

4.3.3.3 Interaction methods

Three interaction methods, comprised of waypoint control, teleoperation control, and camera control, were available to the operator. Using waypoint control, the participant specifies a series of waypoints while the robot is paused. Therefore, we use the times of waypoint specification to measure the number of interactions. Under teleoperation control, the participant manually and continuously drives the robot while monitoring its state. Time spent in teleoperation was measured as the duration of a series of active positional control actions that were not interrupted by pauses of greater than 30 seconds or any other form of control action. Using camera control, the times of camera operation were used because the operator controls the camera by issuing a desired pose and then monitoring the camera's movement.

Although we did not find differences in overall waypoint control times between mixed-initiative and manual control modes, mixed-initiative operators had shorter, $t(13) = 3.02$, $p < .01$, control times during any single control episode, which is the period during which an operator switches to a robot, controls it, and then switches to another robot.

Figure 32 shows the relationship between the number of victims found and total waypoint control times. In manual mode, this distribution follows an inverted "U" with too much or too little waypoint control leading to poor search performance. In mixed-initiative mode, the distribution is skewed to be less sensitive to control times while holding a better search performance.

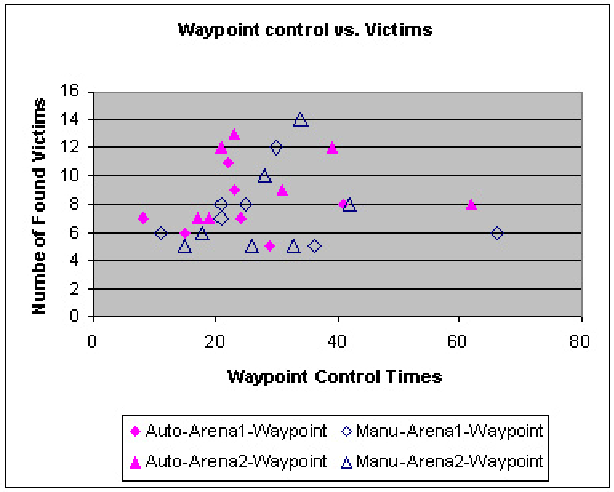


Figure 32. Number of victims found as a function of waypoint controls.

Overall teleoperation control times, $t(13) = 2.179$, $p < .05$, were reduced in the mixed-initiative mode, yet teleoperation times within episodes only approached significance, $t(13) = 1.87$, $p = .08$. No differences in camera control times were found between mixed-initiative and manual control modes. It is notable that operators made very little use of teleoperation (0.6% of mission time) and only infrequently chose to control their cameras.

4.3.3.4 Trust and Capability of Using Interface

In the posttest we collected participants' ratings of their level of trust in the system's automation and their ability to use the interface to control the robots. 43% of the participants trusted the autonomy and only changed the robot's plans when they had spare time. 36% of the participants reported changing about half of the robot's plans while 21% of the participants showed less trust and changed the robot's plans more often. A one tail t-test, indicates that the total victims found by participants trusting the autonomy is larger than the number victims found by other participants ($p=0.05$). 42% of the participants reported being able to use the interface well or very well, while 58% of the participants reported having difficulty using the full range of features while maintaining control of the robots. A one tail t test shows that participants reporting using the interface well or very well found more victims ($p<0.001$). Participants trusting the autonomy reported significantly higher capability in using the user interface ($p=0.001$) and conversely participants reporting using the interface well also had greater trust in the autonomy ($p=0.032$).

4.3.4 Conclusion

In this experiment, the first of a series investigating control of cooperating teams of robots, cooperation was limited to the deconfliction of plans so that robots did not re-explore the same regions or interfere with one another. The experiment found that even this limited degree of autonomous cooperation helped in the control of multiple robots. The results showed that cooperative autonomy among robots helped the operator to explore more areas and find more victims. The fully autonomous control condition demonstrates that this improvement was not due solely to autonomous task performance as found in [100], but rather resulted from mixed-initiative cooperation with the robotic team. The superiority of mixed-initiative control was not a foregone conclusion because earlier studies with comparable numbers of individually autonomous robots [19, 79, 110, 111] found poorer performance at higher levels of autonomy for similar tasks. We believe that differences between navigation and search tasks may help to explain these results. In navigation, moment-to-moment control must reside with either the robot or the human. When control is ceded to the robot, the human's workload is reduced but task performance declines due to the loss of human perceptual and decision-making capabilities. A search task, in contrast, can be partitioned into navigation and perceptual sub-tasks, which allows the human and the robot to share task responsibilities and thereby improve performance. This explanation suggests that increases in task complexity should widen the performance gap between cooperative and individually autonomous systems. We did not collect workload measures to check for the decreases found to accompany increased autonomy in earlier studies [19, 79, 110, 111]. However, 11 of our 14 subjects reported benefiting from robot cooperation.

Our most interesting finding involved the relationship between performance and switching attention among the robots. In both the manual and mixed-initiative conditions, participants divided their attention approximately equally among the robots. However, in the mixed-initiative mode, the participants switched among the robots more rapidly. Psychologists [71] have found task-switching to impose cognitive costs and switching costs have previously been reported [39, 104] for multi-robot control. Higher switching costs might be expected to degrade performance; however, in this study, more rapid switching was associated with improved performance in both manual and mixed-initiative conditions. We believe that the map component at the bottom of the display helped to mitigate losses in awareness when switching between robots and that more rapid sampling of the regions covered by moving robots gave more detailed information about the areas being explored.

The frequency of this sampling among robots was strongly correlated with the number of victims found. This effect, however, cannot be attributed to a change from a control task to a monitoring task because the time devoted to control was approximately equal in the two conditions. We believe instead that searching for victims in a building can be divided into a series of sub-tasks involving, for example, moving a robot from one point to another and turning a robot from one direction to another with or without panning or tilting the camera. To effectively finish the search task, we must interact with these sub-tasks within their neglect time [19], which is proportional to the speed of movement. When we control multiple robots and every robot is moving, there are many sub-tasks for which the neglect time is usually short. Missing a sub-task means that we failed to observe a region that might contain a victim. Switching robot control more often gives us more opportunities to find and finish sub-tasks and therefore helps us to find more victims. This focus on sub-tasks extends to our results for movement control, which suggest that there may be an optimal balance between monitoring and control. If this is the case, it may be possible to improve an operator's performance through training or online monitoring and advice.

We believe the control episode observed in this experiment corresponds to a decomposed subtask of the team and the linear relationship between switches and found victims reveals the independent or weak relationship among the subtasks. For a multi-robot system, decomposing the team goal into independent or weakly related sub goals allowing the human to intervene into the sub goals is a potential way to improve and analyze human multi-robot performance. From the view of interface design, the interface should fit the sub goal decomposition (or sub goal template) and help the operator in attaining SA. Under mixed-initiative control condition, the number of found victims is less sensitive to waypoint specification than under manually control condition. The relation between found victims and waypoint specification can be generalized to the relationship between performance and human intervention. The potential of extending the present experiment to a generic HRI sensitivity evaluation methodology deserves a further study in the future. Moreover, the control episode can be used as a unit of human intervention, rather than the traditional counting of control actions or durations.

5.0 INTERACTION EPISODE

HRI is an active research area that is still in its infancy. The classification of HRI is unsystematic and an update is required [119, 120]. SA studies in HRI are still case studies [24, 25, 44, 59, 96-98], and formal analysis is unavailable. Both the analysis and the evaluation of HRI are usually limited to specific tasks, robotic systems, and interface designs [20, 37, 46, 79, 80, 85, 104, 111, 116]. Clearly, we need to develop the underlying theory of HRI.

In early HRI research, [19] utilizes neglect tolerance to study how humans are able to implement independent tasks. More recently, [20] expands this study in considering the wait time effect and [22] improves it in including a cost-performance model. Unfortunately, these improvements are limited to independent tasks and individual robots. Extending this theory to the control of cooperating robots remains an unsolved problem. Inspired by our observations in the pilot experiment (see Section 4.3), in this chapter, we propose the interaction episode methodology, which utilizes neglect tolerance to study the human control of a robot team in performing complex tasks. Using this methodology, we first investigate neglect tolerance from the human operator's view in which one person proactively controls one or more robots to pursue a task. Second, we view this control as a procedure in which a human operator perceives the SA, makes decisions, and evaluates the overall performance.

5.1 THE INTERACTION EPISODE METHODOLOGY

5.1.1 Neglect Tolerance in a dependent system

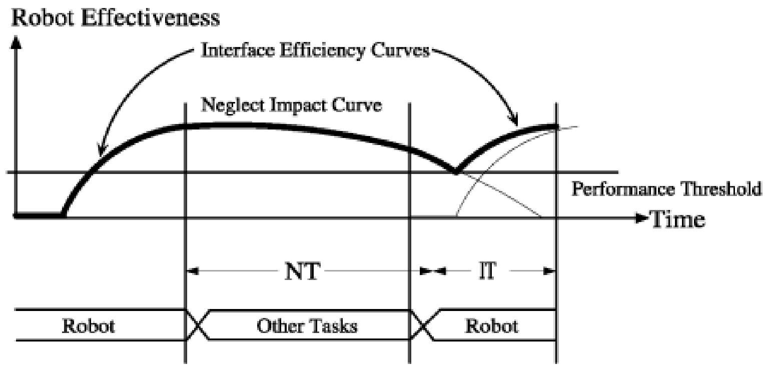


Figure 33. Independent system's effectiveness [19].

Figure 33 depicts the robot effectiveness of single-operator single-robot control for an independent task [19]. If we assume that a human's input always improves the overall performance, then in this situation the effectiveness constantly increases in IT and decreases in NT. For a dependent robotic system, other robots' actions will directly or indirectly affect the currently controlled robot and therefore affect the overall performance. If we assume that the human operator always improves task performance and controls robots serially, then in a dependent robotic system, the effectiveness can be represented in a series of decrease and increase curves in NT (see Figure 34). The increase curve occurs when the operator must control another relevant robot[12] to maintain task performance above the satisfaction level. For instance, when we control two robots to push a box forward, controlling only the left or the right robot will not accomplish the task. We must periodically control both robots to move the

[12] Continuing control of the current robot will not improve performance. The operator must shift to another robot to maintain the performance level because of the constraints imposed by the robots and the task.

box forward. The time spent in controlling the relevant robots is called the occupied time (OT) required by the dependent task. There are two kinds of free time (FT) in the figure below. The first kind is the time spent off-task while establishing team cooperation. The second kind is the time spent off-task after team cooperation has been established. For a complex multi-robot system with N relevant robots, we can define:

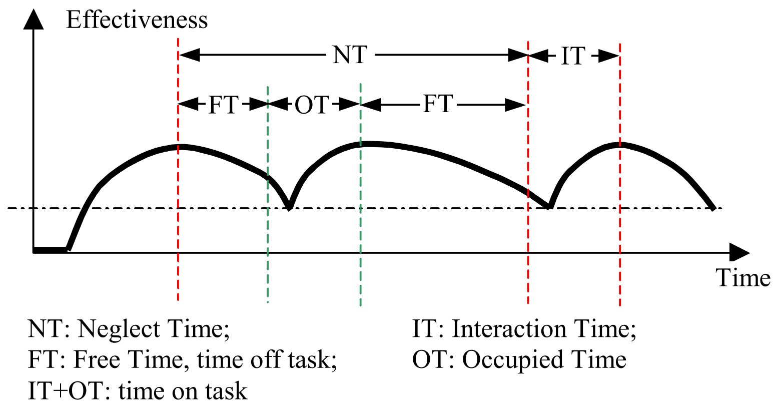


Figure 34. The effectiveness of a dependent system with two robots.

Number of relevant robots: N = number of OTs in one NT.

Neglect time respect to robot j: $NT_j = \sum_{\substack{i=1 \\ i \neq j}}^{N} (FT_{ij} + OT_{ij}) + FT_T$ where FT_{ij} is the free time (time spent off-task) with respect to the j^{th} robot after it interacted with $(i\text{-}1)^{th}$ and before it interacts with the i^{th} robot, OT_{ij} is the occupation time with respect to the j^{th} robot that comes from i^{th} robot, and FT_T is the free time after team cooperation has been established..

Cooperation effort: $CE_j = \dfrac{\sum OT_{ij}}{IT_j}$ is the extra cooperation effort required for a robot. Given a task and a robotic system, a good interface for the robot team should require a low CE. In robotic system design, deploying automatic cooperation among the robots will shorten the time spent in coordinating the robots and yield a low CE.

Cooperation demand: $CD_j = 1 - \frac{\sum FT}{NT_j} = \frac{\sum OT}{NT_j}$ is the percentage of time spent in controlling the relevant robots while the operator neglects the j^{th} robot. When a person is required to teleoperate two robots (e.g., a left robot and a right robot) to push a box, the operator must repeatedly control one robot and then another to move the box. When the operator dedicates his time to control another robot while he neglects one robot (OT=NT), CD will equal 1, which means an extremely high cooperation demand. In contrast, an independent system will hold OT=0, and therefore gives us CD=0, which means no cooperation demand.

Robot Attention Demand with respect to j^{th} robot: $RAD_j = \frac{IT_j}{NT_j + IT_j}$

Team Attention Demand: TAD = $\frac{\sum OT + IT}{NT + IT}$ is the percentage of time consumed in interacting with a robot team. Unlike CD, which measures the fraction of time required in coordinating relevant robots, a TAD includes the IT to measure the fraction of total time spent controlling the entire robot team. It also differs from RAD [82] in including the time occupied by coordinating teammates. From the cooperation view, TAD can be treated as the team cooperation demand.

Team interaction time: For a strongly cooperating system, we can define team interaction time as $IT_T = \sum(FT+OT) + IT$; $NT_T = FT_T$. If we treat the entire team of robots as a single robot, we are able to use all of the metrics used in HRI evaluation of independent tasks [19, 82].

Fan-out: FO is the number of robots that a human can control "simultaneously" [82]. Finding FO for a dependent system is difficult because (1) the minimum number of robots required is determined by the dependent system; (2) a pattern, whether from the task or a system constraint, of added robots may exist (e.g., adding one A-type robot and two B-type robots will give us maximum benefits, but adding only one A-type robot will not[13]); and (3) the system may not support extra robots (e.g., a centralized robotic system) and therefore the FO is a constant. Ignoring the additional constraints from the specific system, the FO can be similarly defined as:

$$FO = \frac{\sum_{\substack{i=1 \\ i \neq j}}^{N} \left(FT_{ij} + OT_{ij}\right) + IT + FT_T}{\sum_{\substack{i=1 \\ i \neq j}}^{N} \left(FT_{ij} + OT_{ij}\right) + IT},$$

which implies that, when we have free time to control more robots, we spend our effort in controlling a subset of robots rather than an individual robot. This subset of robots is the pattern of robots (P_R) that a human can control to reach satisfied effectiveness. The pattern is a function of interaction scheme π, task $\mathcal{T}$, and world complexity C, i.e., $P_R = f(\pi, \mathcal{T}, C)$.

5.1.2 Interaction episode

In the previous section, we assumed that the operator focuses on an individual robot. Indeed, all of the time-based parameters, which include NT, IT, FT, and OT, are defined with respect to an individual robot. For a team of robots, however, it is possible for the operator to acquire team awareness and then issue a team command to control several robots at the same time. For example, the operator can issue an attack command to a robot team when playing a capture-the-flag game with RoboFlag [85]. In these situations, we need to investigate

---

[13] For example, when a robot team of N communicators and M searchers is in a saturated state, adding one search robot will be useless because it will have no communication support. However, adding a communicator and a searcher at the same time might be helpful.

performance from the perspective of the robot team rather than from that of an individual member. Thus, we introduce the concept of the interaction episode (IEP).

An IEP is the period of time during which the operator interacts with the robotic system—which may be a team or one or more individual robots—to pursue a sub-task. For a simple cooperation situation, IEP is the $IT_T = \sum(FT+OT) + IT$ mentioned above. However, IEP allows us to address more complex and general situations. Consider the example of two robots (e.g., one transporter and one lifter) that must cooperate to move an object. The IEP includes moving the transporter to the lifter, loading the object onto the transporter, and moving the transporter to the destination. That is, $IEP = IT_{trans}^{move1} + FT_{trans} + IT_{lift} + FT_{lift} + IT_{trans}^{move2}$, which includes two different ITs for the single transporter. When a robot is used in a search task to navigate and perceive a local region, the IEP will be $IEP = IT^{move} + FT^{move} + IT^{camera} + FT^{camera}$.

In Crandall and Goodrich's evaluation [19], an IT includes at least four components: sub-task selection, context acquisition, solution planning, and expression of robot directives. In IEP, in addition to these four components, there is an evaluation component in which the operator evaluates whether he is able to move to the next interaction episode with a satisfied performance level. Once the operator satisfies the performance level, he moves to the next interaction episode and ignores (neglects) the current one. Otherwise, the operator adjusts the current IEP by applying more ITs and FTs. Hence, IEP can be defined more precisely as the period of time in which the operator interacts with the robotic system to achieve the satisfied performance level for a sub-task while paying attention to a subset of the system under the same context. The final outcome of an IEP is always improved performance if a correct evaluation has occurred. However, at some points in the IEP, the performance may be worse because of issues such as conflict and sacrifice. The evaluation component plays a very important role in multi-robot control during complex tasks because it requires a high level of SA as well as a high mental demand and because it significantly impacts effectiveness. The linear relationship between the number of switches and the number of found victims in our pilot experiment reflects the impact of evaluation in robot control. Prompt and correct evaluation leads to shorter IEPs (more switches) and therefore more found victims.

There are two main patterns in IEP. The first is the action pattern (P_A), which can be represented as $\sum(w*A)$, where w is the weight and A is the action. This pattern reveals the action scheme for the given task, world, and robotic system. The second is the pattern of robots, which can be represented as $P_R = \sum(w*R)$, where w is the weight and R is a type of robot. Finding P_R is important in improving HRI in MRS. Although system and task analysis can guide us in deciding P_R, we still need to verify it when the human operator intervenes. On the other hand, P_R can help us to construct the robot team. For example, for a team of N communication robots and M search robots, we need to know what the best combination of N and M is for a human operator who controls the team.

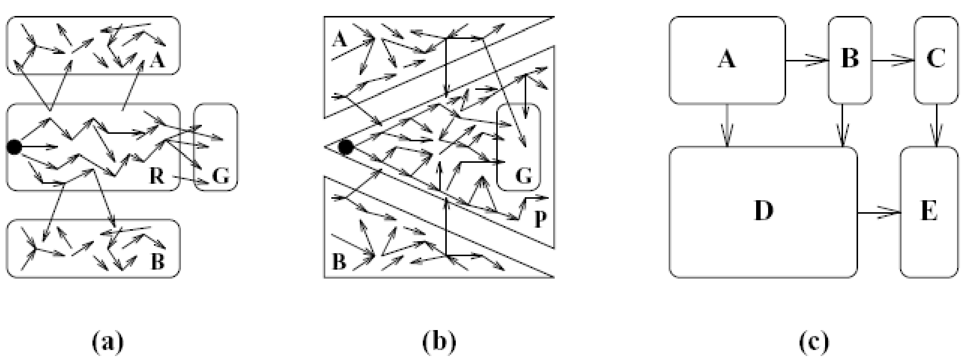


Figure 35. Reachability and serial problem decomposition (copied from [6]).

An IEP is a heuristic decomposition that is very similar to the serial problem decomposition[14] (Figure 35) described in [6]. Here, however, the decomposition is performed from the human's perspective, which in turn is based on the human's perception, prediction, trust and other related affective issues, and the given interaction scheme. This may be an approximate decomposition that has a weak correlation with other components. For instance, in the N communicators and M searchers example, an IEP may be a dedicated search that ignores all the communicators.

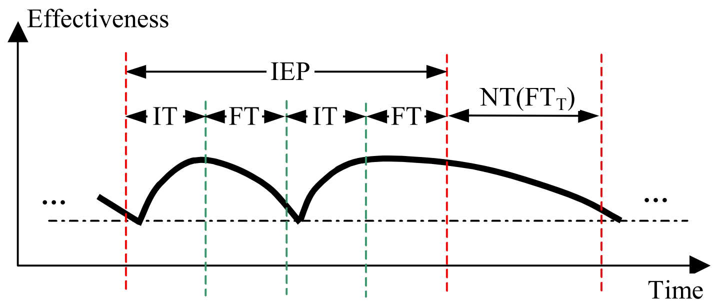


Figure 36. An example of IEP of a strong cooperative system.

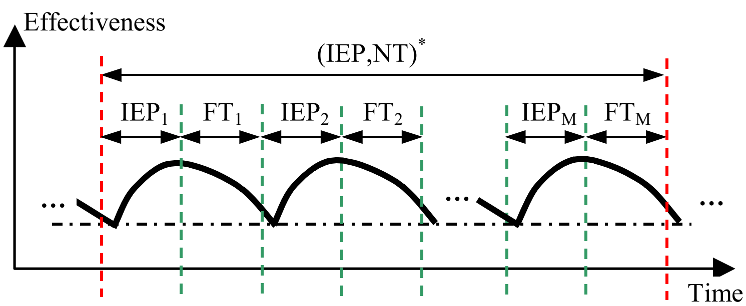


Figure 37. An example of IEPs of a weak cooperative system.

For a strong cooperation system, only one type of IEP exists because one IEP includes all the required cooperation actions. We can use a similar equation of IT and NT to evaluate HRI via using IEP as the count unit (Figure 36). For example, FO = (IEP+NT)/IEP; RAD = IEP/(IEP+NT). If we decompose IEP into a series of ITs and FTs, then we are able to compute cooperation demand and cooperation effort, which were introduced in the previous

[14] For the N people and M robots problem, parallel decomposition will be involved.

section. However, this interaction is very general and can be either an individual robot or a robot team control.

For weak cooperation problems, as shown in the previous example, multiple types of IEP exist. The distribution of the IEPs, that is, $m\sum_{i=1}^{N_{Type}}\left(w_i\left(IEP_i, NT_i\right)\right)$, reveals the weak cooperation. The N_{Type} is the number of types, and w_i is the correlation weight that measures the correlation among the sub-teams' sub-tasks. By treating ∑(w(IEP,NT)) as an (IEP,NT)*, we are able to apply the approach used in strong cooperation settings to evaluate HRI (Figure 37). Here, however, there is no obvious NT* that allows us to compute the metrics. Because of the dependent nature of the task, each NT theoretically will be affected by the previous control interactions. In terms of the IEPs, the heuristic task decomposition makes these (IEP_i,NT_i)s weakly dependent on each other such that the order of (IEP_i,NT_i) has no significant impact to the overall performance. Therefore, we can simply treat NT* as the mean of NTs[15], i.e. $NT^* = \frac{1}{N_{Type}}\sum_i w_i NT_i$ and calculate the HRI evaluation metrics as:

$$CE_j = \frac{\sum_{\substack{i=1 \\ i\neq j}}^{N} w_i IEP_i}{w_j IEP_j}$$

$$CD_j = \frac{\sum OT}{NT_j} = \frac{\sum_{\substack{i=1 \\ i\neq j}}^{N} w_i IEP_i}{NT_j} = \frac{\sum_{\substack{i=1 \\ i\neq j}}^{N} w_i IEP_i}{\sum_{\substack{i=1 \\ i\neq j}}^{N} w_i IEP_i + NT^*} = \frac{\sum_{\substack{i=1 \\ i\neq j}}^{N} w_i IEP_i}{T_{(IEP,NT)^*} - w_j IEP_j}$$

[15] The NTs may overlap with each other. They should roughly equal each other within one (IEP,NT)*. Therefore, we use the mean value here.

$$TAD = \frac{\sum_{i=1}^{N} w_i IEP_i}{\sum_{i=1}^{N} w_i IEP_i + NT^*}$$

$$FO = \frac{\sum_{i=1}^{N} w_i IEP_i + NT^*}{\sum_{i=1}^{N} w_i IEP_i} = 1 + \frac{NT^*}{\sum_{i=1}^{N} w_i IEP_i},$$

where FO is the number of cooperating control systems that one operator can simultaneously maintain. If this system involves controlling m robots, then the maximum number of robots an operator is able to simultaneously control is mFO.

Furthermore, for a particular (IEP,NT) that comprises ITs and FTs, we can compute the CD with respect to a control action (corresponding to an IT) to measure the cooperation demand among the control actions for a type of IEP.

5.1.3 The measurement

The same type (IEP,NT) may have different time durations in IT and NT, so we need to use the mean in computation. In summary, we define

$\overline{IEP} = Mean(IEP_j)$, where the sample is the population of a type of IEP,

$\overline{NT} = Mean(NT_j)$, where the sample is the population of a type of IEP, and

$(IEP, NT)^* = \sum_{i=1}^{N_{Type}} \left(w_i \left(\overline{IEP_i}, \overline{NT_i}\right)\right)$, where N_{Type} is the number of types,

where w_i is the weight that can have the following definitions.

Assuming the whole control process is $\sum_{i=1}^{N_{Type}}\left(n_i\left(IEP_i, NT_i\right)\right)$, where n_i is the number of type-i (IEP,NT):

Def1: w_i is the integer that makes $m*w_i=n_i$ with the maximum integer m for types of IEP.

Def2: $w_i = \frac{n_i}{\sum n_i}$, $w_i<1.0$.

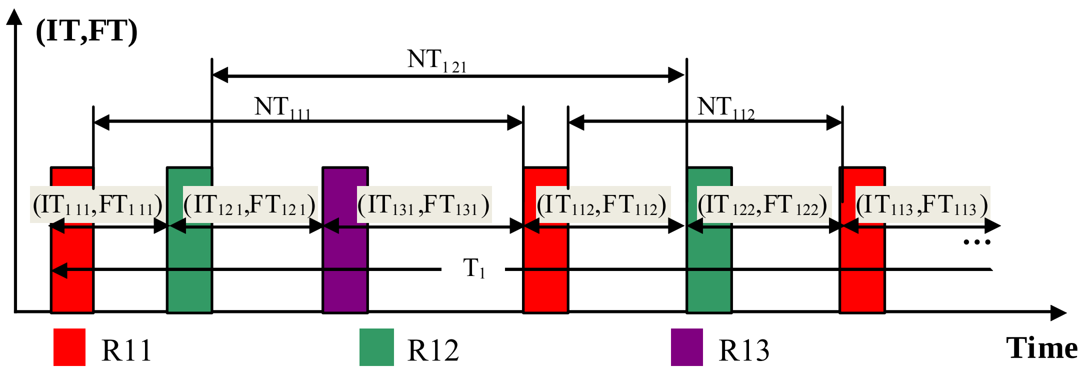


Figure 38. Distribution of (IT,FT)

In the more general case, requirements for cooperation can be relaxed to allow the operator to choose which subteams of robots will be operated in a cooperative manner as well as which robot will be operated next. To accommodate this case, the Neglect Tolerance model must be further extended to measure coordination between different robot types.

We describe this form of heterogeneous MRS as a MN system with M robots that belong to N robot types. For robot type i, there are m_i robots, that is, $M = \sum_{i=1}^{N} m_i$. Thus, we can denote a robot in this system as R_{ij} , where $i = [1,N]$, $j = [1,m_i]$. If we assume that the operator serially controls the robots for time T and that each robot R_{ij} is interacted with l_{ij} times, then we can represent each interaction as IT_{ijk}, where $i = [1,N]$, $j = [1,m_i]$, $k = [1,l_{ij}]$, and the following free time as FT_{ijk}, where $i = [1,N]$, $j = [1,m_i]$, $k = [1,l_{ij}]$. The total control time T_i for type i robot should then be $T_i = \sum_{j,k} \left(IT_{ijk} + FT_{ijk} \right)$. Because robots that are of the same robot type are identical, and substitution may cause uneven demand, we are only interested in measuring the average coordination demand CD_i, i=[1,N].

Given identical robots R_{ij}, j=[1,m_i], there are OT_i^* and NT_i^* such that for each robot R_{ij} we have $CD_{ij} = \frac{1}{l_{ij}} \sum_{k=1}^{l_{ij}} \frac{OT_{ijk}}{NT_{ijk}} = \frac{l_{ij} OT_i^*}{l_{ij} NT_i^*}$. Therefore, the CD_i for type i robot is

$$CD_i = \frac{1}{m_i} \sum_{j=1}^{m_i} CD_{ij} = \frac{1}{m_i} \sum_{j=1}^{m_i} \frac{l_{ij} OT_i^*}{l_{ij} NT_i^*} = \frac{OT_i^* \sum_{j=1}^{m_j} l_{ij}}{NT_i^* \sum_{j=1}^{m_j} l_{ij}}$$

If we assume all the other types of robots are dependent with the current type of robots, then the numerator $OT_i^* \sum_{j=1}^{m_j} l_{ij}$ is the total interaction time of all the other types of robots, i.e.,

$OT_i^* \sum_{j=1}^{m_j} l_{ij} = \sum_{\substack{type=1 \\ type \neq i}}^{N} IT$.

For the denominator, it is difficult to directly measure NT_i^* because the system performance depends on multiple types of robots and because an individual robot may cooperate with different team members over time. Because of this dependency, we cannot use an individual robot's active time to approximate NT. On the other hand, the robots may be unevenly controlled. For example, a robot might be controlled only once and then ignored because there is another identical robot that is available, which means that we cannot simply use the time interval between two interactions of an individual robot as NT. Considering all of the robots that belong to a robot type, the distribution of (IT,FT)s reveals the NT for a type of robot. Figure 38 shows an example of an (IT,FT) distribution. When each robot is evenly controlled, $(IT_i,NT_i)^*$ should be $m_i * (IT_i,FT_i)$ where (IT_i,FT_i) is the average (IT,FT) for type i robot, $\left(IT_i, FT_i\right) = \frac{T_i}{\sum_{j=1}^{m_i} l_{ij}}$. When only one robot is controlled, $(IT_i,NT_i)^*$ will be (IT_i,FT_i). In summary, $(IT_i,NT_i)^*$ should be in the range $\left(IT_i, FT_i\right) \le \left(IT_i, FT_i\right)^* \le m_i\left(IT_i, FT_i\right)$. Here, we introduce weight $w_i = \frac{\sum_{j=1}^{m_i} l_{ij}}{\max_{j=1}^{m_i}\left(l_{ij}\right)}$ to measure how evenly the robots are controlled. With the weight, we can approximate $(IT_i,NT_i)^*$ as:

$$\left(IT_i, NT_i\right)^* = w_i\left(IT_i, NT_i\right) = \frac{\sum_{j=1}^{m_i} l_{ij}}{\max_{j=1}^{m_i}\left(l_{ij}\right)} \times \frac{T_i}{\sum_{j=1}^{m_i} l_{ij}} = \frac{T_i}{\max_{j=1}^{m_i}\left(l_{ij}\right)}$$

Thus, the denominator in CD_i can be calculated as:

$$NT_i^* \sum_{j=1}^{m_i} l_{ij} = \left(\frac{T_i}{\max_{j=1}^{m_i}\left(l_{ij}\right)} - IT_i^* \right) \sum_{j=1}^{m_i} l_{ij} = \frac{\sum_{j=1}^{m_i} l_{ij}}{\max_{j=1}^{m_i}\left(l_{ij}\right)} T_i - IT_i^* \sum_{j=1}^{m_i} l_{ij} = \frac{\sum_{j=1}^{m_i} l_{ij}}{\max_{j=1}^{m_i}\left(l_{ij}\right)} T_i - \sum_{type=i} IT \, ,$$

where $\sum_{type=i} IT$ is the total interaction time for all the type i robots.

In summary, we can compute CD_i as:

$$CD_i = \frac{\sum_{type \neq i} IT}{\frac{\sum_{j=1}^{m_i} l_{ij}}{\max_{j=1}^{m_i}\left(l_{ij}\right)} T_i - \sum_{type=i} IT}$$

5.1.4 Maximum effectiveness

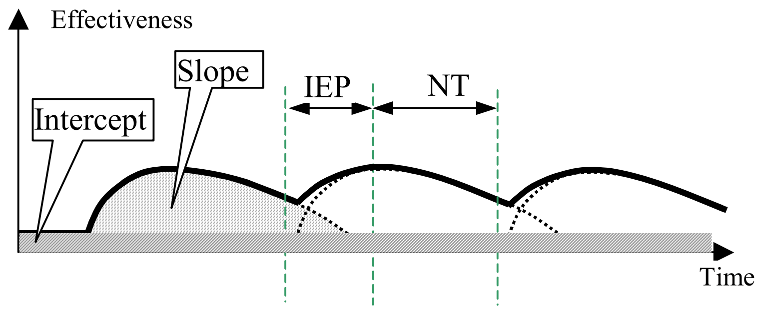


Figure 39. Intercept and slope in the trend line represented as an effectiveness curve.

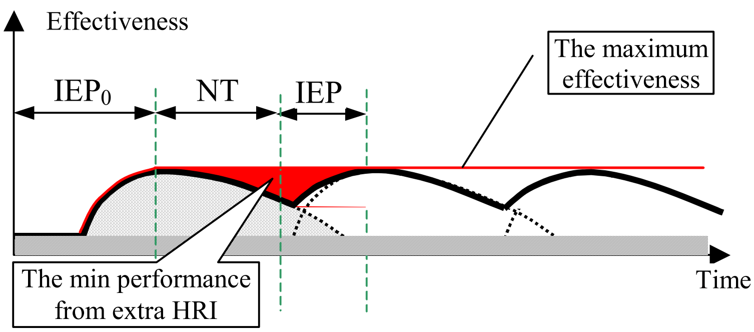


Figure 40. The minimum required HRI effort in multi-robot control.

As an application of the interaction episode methodology, in this section we discuss maximum effectiveness instead of satisfied effectiveness. In Figure 34, we show the individual robot's effectiveness with respect to the task. In Figure 39, we show the sub-team's effectiveness with respect to the task. When multiple individuals or sub-teams work on the same task, the team performance will be the "sum" of the individuals' or sub-teams' performances. The red line in Figure 40 shows the maximum possible effectiveness if the operator dedicates attention to one particular sub-team. When the operator sacrifices interaction time to control other sub-teams, the benefit from the extra HRI should be greater than the lost performance (represented by the red area in Figure 40) and will benefit the task overall. Figure 41 shows the four basic effectiveness curves where II/GI indicates an immediate/gradual increase in IT and ID/GD indicates an immediate/gradual decrease in NT.

A Type A curve gives the same performance in dedicated single-robot control and multi-robot control. A Type B curve gives worse performance in multi-robot control. A Type C curve always gives better performance in multi-robot control. A Type D curve, when the sacrifice (represented by the red area) is smaller than the benefit (represented by the green area), gives better performance in multi-robot control. In general, quick performance increments in IT and slow performance decrements in NT will benefit us in multi-robot control. From the effectiveness curve, we can calculate the critical interaction frequency when multi-robot control is better than single-robot control:

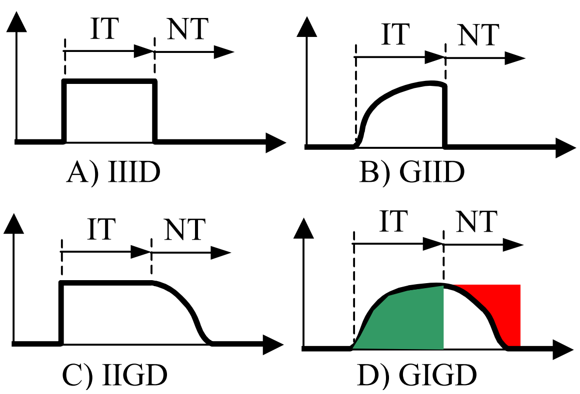


Figure 41. Four basic effectiveness curves.

$Js(\pi,C,t_{on}) > (Js(\pi,C,t_{on}+t_{off}) - (Js(\pi,C,t_{on})+J_N(\pi,C,t_{off},t_{on})))$ assumes a linear relationship in performance overlap, where $Js(\pi,C,t)$ is the serving performance, where π is the interaction scheme, C denotes the world complexity, and t measures the interaction time; $J_N(\pi,C,t_1,t_2)$ is the neglect performance that occurs after previously interaction for t_2 and then neglects for t_1 duration; t_{on} is the time spent on-task; and t_{off} is the time spent off-task.

In general, given (1) the interaction function $f_I(\pi,C,t,f_0)$[16], (2) the neglect function $f_N(\pi,C,t,f_0)$[17], (3) the relaxation level (i.e., the free time occupation rate needed by the

[16] f_0 is the initial performance before interaction with the robotic system.
[17] f_0 is the initial performance before neglect of the robotic system.

operator to maintain a level of comfort), and (4) the sequential control style (i.e., no overlap in IT), we can compute the number of robots (or sub-teams, if we replace individual robot control with IEP) that the operator needs to control in order to maintain maximum effectiveness. If we assume that performance increase and decrease is irrelevant to the initial performance, and that the relationship in performance operation is linear, the problem can be simplified as shown in Figure 42:

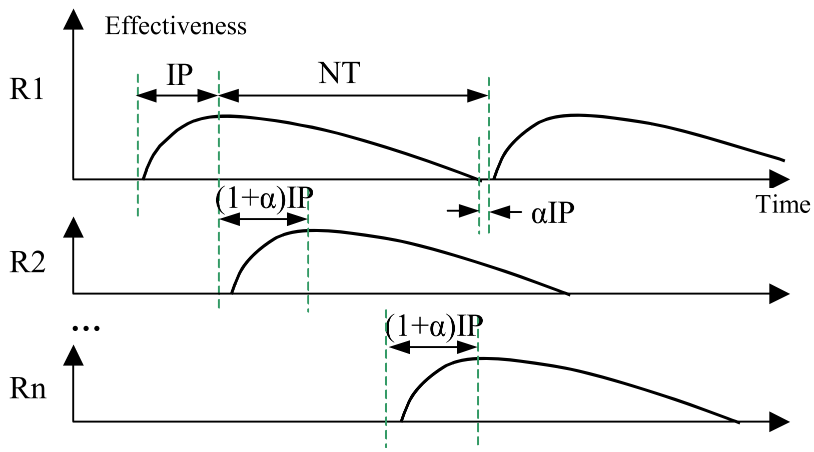


Figure 42. n robots control.

$Max\{n[f_I(n(\alpha+1)t_I) - f_D((n-1)(\alpha+1)t_I + \alpha t_I)]\}$, where $f_I(t)$ is the performance increase function, $f_D(t)$ is the performance decrease function, α is the relaxation level that stands for *free time = α* interaction time*, and n is the number of robots or sub-teams. When n=1 and α=0, we find $f_I(t_I)$, which means that achieving maximum effectiveness in single-robot control requires consistent control ($t_I=\infty$). When α=0, $f_I(t)=f_D(t)$ (the Type A effectiveness curve) and $f_I(a+b)=f_I(a)+f_I(b)$, we find $n*f_I(t_I)$, which means either $t_I=\infty$ or $n=\infty$ will yield maximum effectiveness. This case corresponds to teleoperating multiple robots: $t_I=\infty$ means consistently controlling a robot to reach maximum effectiveness, and $n=\infty$ means that, when spending limited time on a robot, achieving maximum effectives requires shifting to another robot immediately if we are not able to control the first robot. For the Type A effectiveness curve, this is identical to consistently controlling one robot.

5.2 TIGHT COOPERATION EXPERIMENT

One of the main potential contributions of the proposed methodology is that we are able to use neglect tolerance to study dependent tasks that require cooperating robots. Cooperation demand (CD) and team attention demand (TAD) are the metrics that measure how tightly the coordination must be for a task, given a robotic system. Since there is no well-defined and accepted definition of CD and TAD, this experiment implicitly validates our definitions of CD and TAD via comparison (i.e., the change of the measured CDs). TADs should hold the correct trend under different conditions.

5.2.1 Experiment design

In this experiment, we investigated CD and TAD by comparing performance across three conditions selected to differ substantially in their coordination demands. We selected box pushing, a typical cooperative task that requires the robots to coordinate. When an operator teleoperates the robots one by one to push the box forward, he must continuously interact with one of the robots because neglecting both would immediately stop the movement of the box. Because the task allows no free time (FT) we expect CD to be 1. However, when the user is able to issue waypoints to both robots, the operator may have FT before he must coordinate these robots again because the robots can be instructed to move simultaneously. In this case

CD should be less than 1. Intermediate levels of CD can be found in comparing the control of homogeneous robots with that of heterogeneous robots. A higher CD can be found in the heterogeneous group since the unbalanced pushes from the robots would require more frequent coordination.

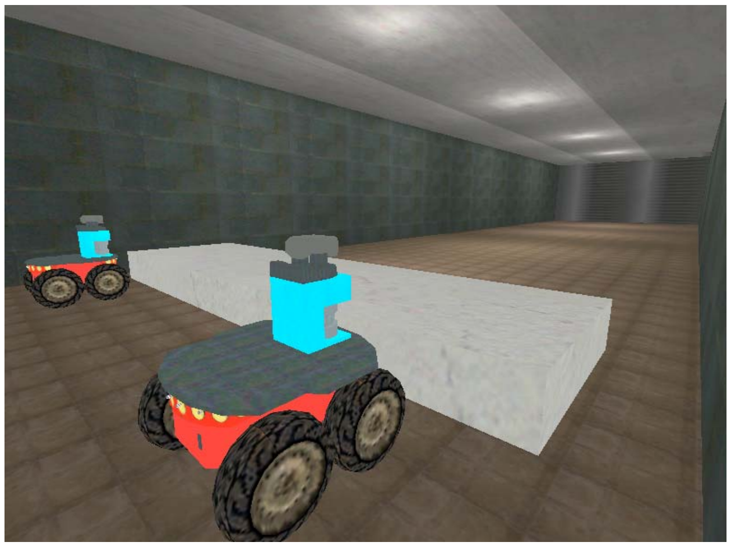


Figure 43. Box pushing task

Figure 43 shows our experiment setting simulated in USARSim. The controlled robots were either two Pioneer P2AT robots or one Pioneer P2AT robot and one Pioneer P2DX robot. Each robot was equipped with a GPS, a laser scanner, and an RFID reader. On the box, we mounted two RFID tags to enable the robots to sense the box's position and orientation. When one of the robots pushes the box, both the box's and the robot's orientation and speed will change. Furthermore, because of irregularities in initial conditions and in the accuracy of the physical simulation, the robot and the box are unlikely to move precisely as the operator expected. In addition, delays in receiving sensor data and executing commands were modeled, presenting participants with a problem very similar to coordinating physical robots.

We introduced a simple matching task as a secondary task in order to estimate the FT available to the operator. Participants were asked to perform this secondary task as often as possible when they were not occupied with controlling a robot. Every operator action and periodic timestamped sample of the box's moving speed were recorded for computing CD. In this experiment, the CD for the left and right robots was calculated as $CD_L = IT_R/(T-IT_L)$; $CD_R = IT_L/(T-IT_R)$, respectively, where T is the total control time and IT_L, IT_R are the total interaction times of the left and right robots.

A within-subject design was used to control for individual differences in operators' control skills and abilities to use the interface. To avoid biasing the CD comparison because of abnormal control behavior, such as a robot bypassing the box, we added safeguards to the control system to stop the robot when it tilted the box.

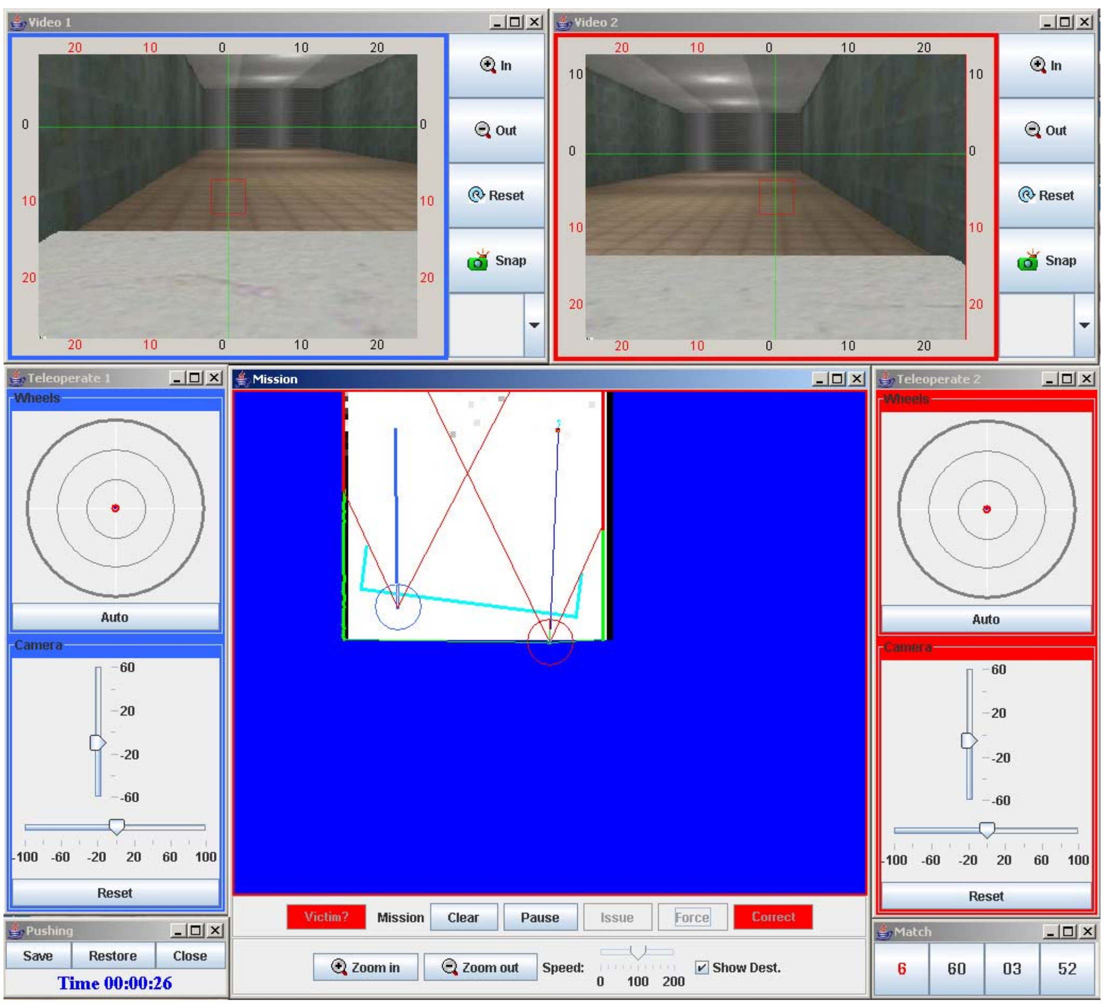


Figure 44. GUI for box pushing

The operator controlled the robots using the distributed multi-robot control system (MrCS) shown in Figure 44. On the left and right sides are the teleoperation widgets that control the left and right robots separately. At the bottom center of the screen is a map-based control panel that allows the user to monitor the robots and issue waypoint commands on the map. In the bottom right corner is the secondary task window where the participants were asked to perform the matching task when possible.

5.2.2 Participants

Fourteen paid participants who ranged from 18 to 57 years old were recruited from the University of Pittsburgh community. None had prior experience with robot control although most were frequent computer users. The participants' demographic information and experience are summarized in Table 5.

Table 5. Sample demographics and experiences.

	Age		Gender		Education	
	18~35	>35	Male	Female	Currently/Complete Undergraduate	Currently /Complete Graduate
Participants	11	3	11	3	2	12

	Computer Usage (hours/week)				Game Playing (hours/week)			
	<1	1-5	5-10	>10	<1	1-5	5-10	>10
Participants	0	1	2	11	8	4	2	0

	Mouse Usage for Game Playing		
	Frequently	Occasionally	Never
Participants	9	4	1

5.2.3 Procedure

The experiment started with collection of the participant's demographic data and computer experience. The participant then read standard instructions on how to control robots using the MrCS. In the following eight-minute training session, the participant practiced each control operation and tried to push the box forward under the guidance of the experimenter. Participants then performed three testing sessions in counterbalanced order. In the first two sessions, the participants controlled two P2AT robots using teleoperation alone or a mixture of teleoperation and waypoint control. In the third session, the participants were asked to control heterogeneous robots (one P2AT and one P2DX) using a mixture of teleoperation and waypoint control. The participants were allowed eight minutes to push the box to the destination in each session. At the conclusion of the experiment participants completed a questionnaire about their experiences.

5.2.4 Results

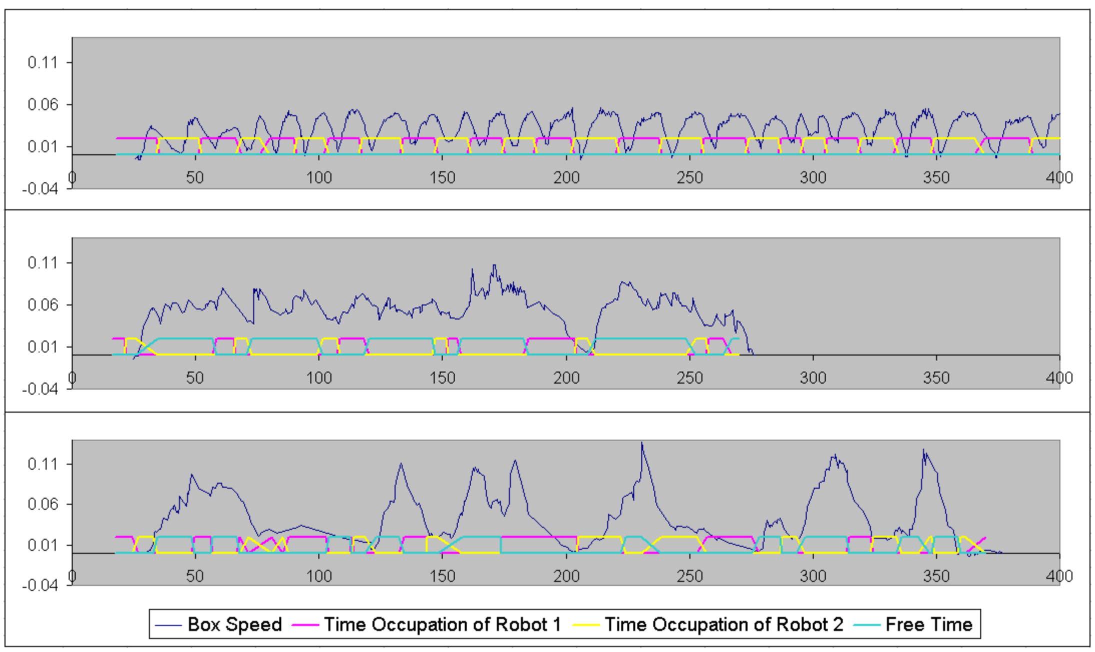


Figure 45. The time distribution and effectiveness curves for teleoperation (upper) and waypoint control (middle) for homogeneous robots and waypoint control (bottom) for heterogeneous robots

Figure 45 shows a time distribution of robot control commands recorded in the experiment. As we expected, no free time was recorded for robots in the teleoperation condition. The longest free times were found in controlling homogeneous robots with waypoints. The box speed shown in Figure 45 is the moving speed along the hallway that reflects the interaction effectiveness (IE) of the control mode. The IE curves in this picture show the delay effect and the frequent bumping that occurred in controlling heterogeneous robots revealing the poorest cooperation performance.

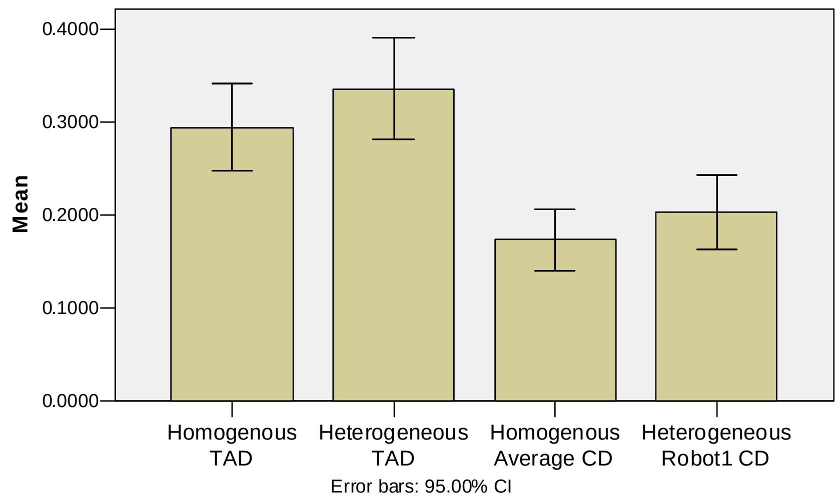


Figure 46. Team task demand (TAD) and Cooperation demand (CD)

None of the 14 participants was able to perform the secondary task while teleoperating the robots. Hence, we uniformly find TAD=1 and CD=1 for both robots under this condition. Within-participant comparison found that under waypoint control the team attention demand in heterogeneous robots is significantly higher than the demand in controlling homogeneous robots, $t(13)=2.213$, $p=0.045$ (Figure 46). No significant differences were found between the homogeneous P2AT robots in terms of the individual cooperation demand ($P=0.2$). Since the robots are identical, we compared the average CD of the left and right robots[18] with the CDs measured under heterogeneous condition. A two-tailed t-test shows that when a participant controlled a P2AT robot, a lower CD was required in the homogeneous condition than that in

[18] The CD of homogeneous robots refers to the average individual CD of the robot group.

the heterogeneous condition, $t(13)=-2.365$, $p=0.034$. The CD required in controlling the P2DX in the heterogeneous condition is marginally higher than the CD required in controlling homogenous P2ATs, $t(13)=-1.868$, $p=0.084$ (Figure 46). Surprisingly, no significant difference was found in CDs between controlling P2AT and P2DX in the heterogeneous condition ($p=0.79$). This can be explained by three observed robot control strategies: (1) the participant always issued new waypoints to both robots when adjusting the box's movement; therefore, similar CDs were found between the robots; (2) the participant tried to give short paths to the faster robot (P2DX) to balance the different speeds of the two robots; therefore, we found a higher CD in P2AT; and (3) the participant gave the same length paths to both robots and the slower robot needed more interactions because it tended to lag behind the faster robot; therefore, we found a lower CD for the P2AT. Among the 14 participants, five of them (36%) showed a higher CD for the P2DX, contrary to our expectations.

5.2.5 Discussion

Although we expected a uniformly higher CD for the P2AT robot in the heterogeneous condition, three exceptions were found in the experiment. The first exception occurred for a participant who commented on having problems in using the control interface. This was confirmed by the recorded irregular time distribution (Figure 47). The close CDs (0.23 and 0.22 for the P2AT robot in the homogeneous and heterogeneous conditions, respectively) demonstrate that the participant's lack of operational skill overwhelmed the impact of the task and the robotic system. In the second exception, we observed that an abnormally long time (41.25 sec) in controlling homogeneous P2ATs was spent in figuring out and recovering from a mistake. Because of the short task completion time (380 sec), this mistake led to a relatively high CD. The last exception occurred when the participant changed his control strategy, specifically the satisfying level of performance, between the homogeneous and heterogeneous conditions. While controlling the homogeneous robots, she paid more attention to keeping the box in the center of the hallway and made many more adjustments to the robots, which led to a total lateral offset of 0.28 meters. However, in the following heterogeneous robots trial, she lowered her criteria for accuracy and finished this session with a total lateral offset of 0.54 meter (Figure 48). The higher CD for homogeneous robots (0.17 and 0.11 under homogeneous and heterogeneous conditions, respectively) reflects the impact of this change in criteria.

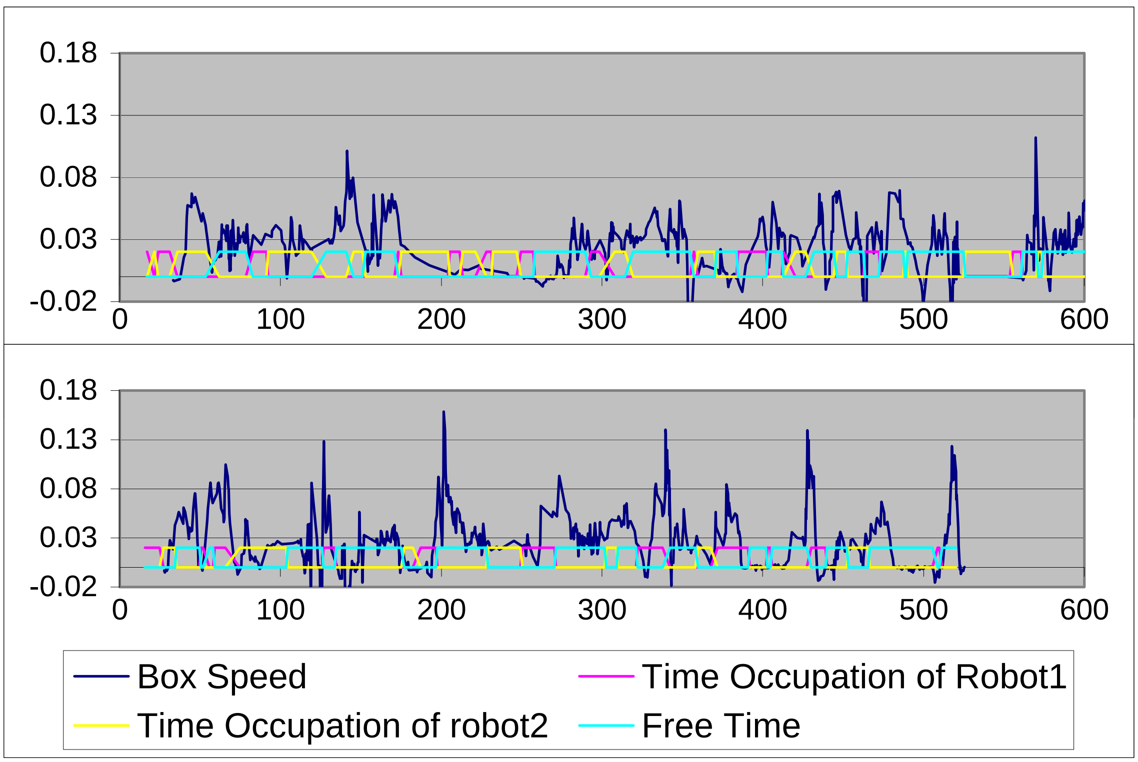


Figure 47. Exception I: Homogeneous (upper) and heterogeneous (bottom) robots control

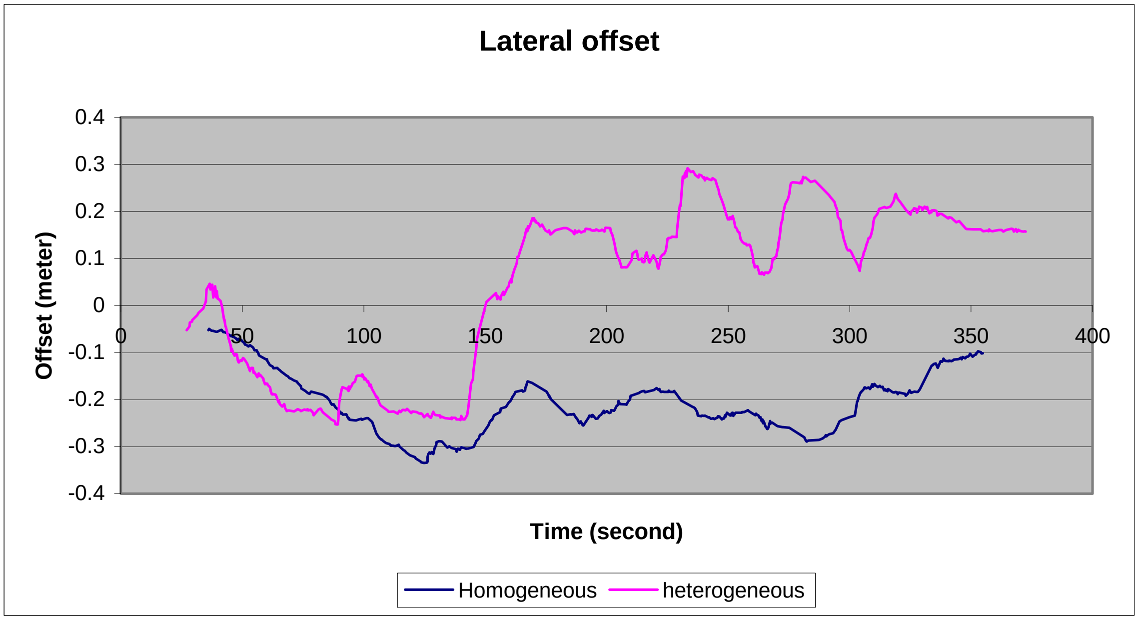


Figure 48. Exception III: Different satisfying levels

This experiment demonstrates that, as a generic HRI metric in a tight cooperation situation, CD is able to account for the various factors that affect HRI and can be used in HRI evaluation and analysis. Although only 14 participants were involved in this experiment, using measured CDs, we were able to quickly identify three aberrant robot control modes. On the other hand, the generality of the measure required us to design the experiment carefully to control target factors. As demonstrated in this experiment, individual differences can easily overwhelm other factors at control tasks of this sort, making within-subject comparisons desirable for smaller samples.

5.3 WEAK COOPERATION EXPERIMENT

Most MRS research has investigated homogeneous robot teams where additional robots provide redundant (independent) capabilities. Differences in capabilities such as mobility or payload, however, may lead to more advantageous opportunities for cooperation among heterogeneous robots. These differences in roles and other characteristics affecting IT, NT, and OT introduce additional complexity to assessing CD. To test the usefulness of the CD measurement for a weakly cooperative MRS, we conducted an experiment to assess CD using an USAR task requiring high human involvement [75] and a level of complexity suitable to exercise heterogeneous robot control. In the experiment, participants were asked to control *explorer* robots equipped with a laser range finder but no camera and *inspector* robots equipped with only cameras. Finding and marking a victim on the map required using the *inspector's* camera to find a victim that could then be marked on the map generated by the *explorer*. The capability of the robots and the cooperation autonomy level were used to adjust the CD of the task.

5.3.1 Experiment design

The experiment was conducted in USARSim with MrCS. Three simulated Pioneer P2AT robots and three Zergs [5], a type of small experimental robot, were used. Each P2AT was equipped with a front laser scanner with a 180-degree FOV and a resolution of one degree. The Zerg was mounted with a pan-tilt camera with a 45-degree FOV. The robots were capable of localization and able to communicate with other robots and the control station. The P2AT served as an *explorer* to build the map while the Zerg could be used as an *inspector* to find victims using its camera. To accomplish the task, the participant must coordinate these two types of robots to ensure that when an *inspector* robot finds a victim, it is within a region mapped by an *explorer* robot so the position can be marked.

Figure 49. The robots and the environment

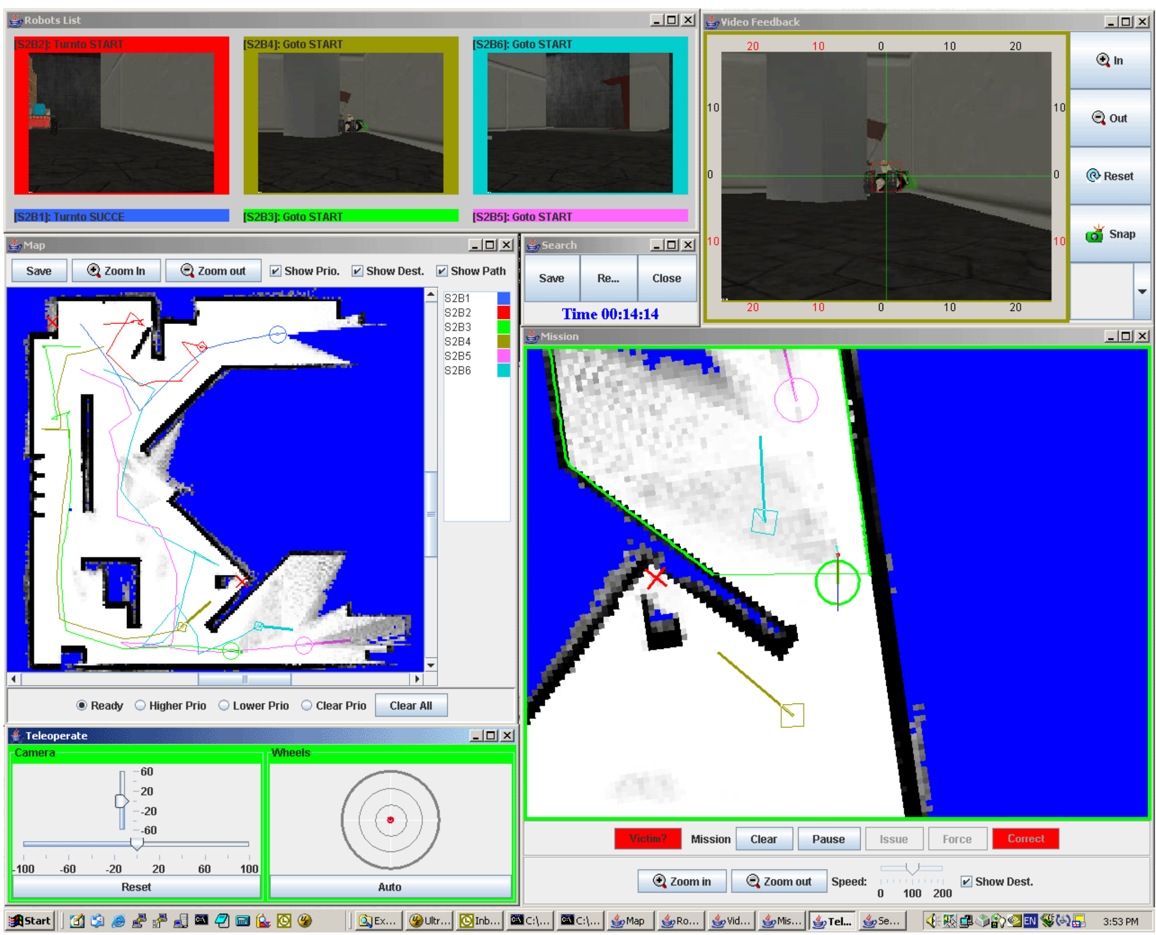


Figure 50. The GUI

Three conditions were designed to vary the coordination demand on the operator. Under the first condition, the *explorer* had a 20-meter detection range, allowing *inspector* robots considerable latitude in their search. Under the second condition, the scanner range was reduced to five meters requiring closer proximity to keep the *inspector* within mapped areas. Under the third condition, the *explorer* and *inspector* robots were paired as sub-teams in which the *explorer* robot with a sensor range of five meters followed its *inspector* robot to map areas being searched. We hypothesized that CDs for *explorer* and *inspector* robots would be more evenly distributed under the second condition (short-range sensor) because *explorers* would need to move more frequently in response to *inspectors*' searches than in the first condition in which the CD should be more asymmetric with *explorers* exerting greater demand on *inspectors*. We also hypothesized that a lower CD would lead to higher team performance. Three equivalent damaged buildings were constructed from the same elements using different layouts. Each environment was a maze-like building with obstacles, such as chairs, desks, cabinets, and bricks with ten evenly distributed victims. A fourth environment was constructed for training. Figure 49 shows the simulated robots and environment.

A within-subject design with counter-balanced presentation was used to compare the cooperative performance across the three conditions. The same control interface shown in Figure 50which allowed participants to control robots through waypoints or teleoperation was used in all conditions.

5.3.2 Participants

Nineteen paid participants, ranging from 19 to 33 years of age, were recruited from the University of Pittsburgh community. None had prior experience with robot control, although most were frequent computer users. Six of the participants (31.5%) reported playing computer games for more than one hour per week. The participants' demographic information and experience are summarized in Table 6.

Table 6. Sample demographics and experiences.

<table>
<tr><th rowspan="2"></th><th colspan="2">Age</th><th colspan="2">Gender</th><th colspan="4">Education</th></tr>
<tr><td>19~29</td><td>30~33</td><td>Male</td><td>Female</td><td colspan="2">Currently/Complete Undergraduate</td><td colspan="2">Currently /Complete Graduate</td></tr>
<tr><td>Participants</td><td>18</td><td>1</td><td>7</td><td>12</td><td colspan="2">11</td><td colspan="2">8</td></tr>
<tr><td rowspan="2"></td><td colspan="4">Computer Usage (hours/week)</td><td colspan="4">Game Playing (hours/week)</td></tr>
<tr><td><1</td><td>1-5</td><td>5-10</td><td>>10</td><td><1</td><td>1-5</td><td>5-10</td><td>>10</td></tr>
<tr><td>Participants</td><td>0</td><td>1</td><td>5</td><td>13</td><td>13</td><td>4</td><td>1</td><td>1</td></tr>
<tr><td rowspan="2"></td><td colspan="8">Mouse Usage for Game Playing</td></tr>
<tr><td colspan="3">Frequently</td><td colspan="3">Occasionally</td><td colspan="2">Never</td></tr>
<tr><td>Participants</td><td colspan="3">14</td><td colspan="3">2</td><td colspan="2">3</td></tr>
</table>

5.3.3 Procedure

After collecting demographic data the participant read standard instructions on how to control robots via MrCS. In the following 15- to 20-minute training session, the participant practiced each control operation and tried to find at least one victim in the training arena under the guidance of the experimenter. Participants then began three testing sessions in counter-balanced order with each session lasting 15 minutes. At the conclusion of the experiment, participants completed a questionnaire about their experiences.

5.3.4 Results

Overall performance was measured by the number of victims found, the explored areas, and the participants' self-assessments. To examine cooperative behavior in finer detail, CDs were computed from logged data for each type of robot under the three conditions. We compared the measured CDs between the first condition (20-meter sensing range) and the second condition (5-meter sensing range) as well as between the second and third conditions (sub-teams). To further analyze the cooperation behaviors, we evaluated the total attention demand in robot control and control action pattern. Finally, we introduce control episodes to show how CDs can be used to identify and diagnose abnormal control behaviors.

5.3.4.1 Overall performance

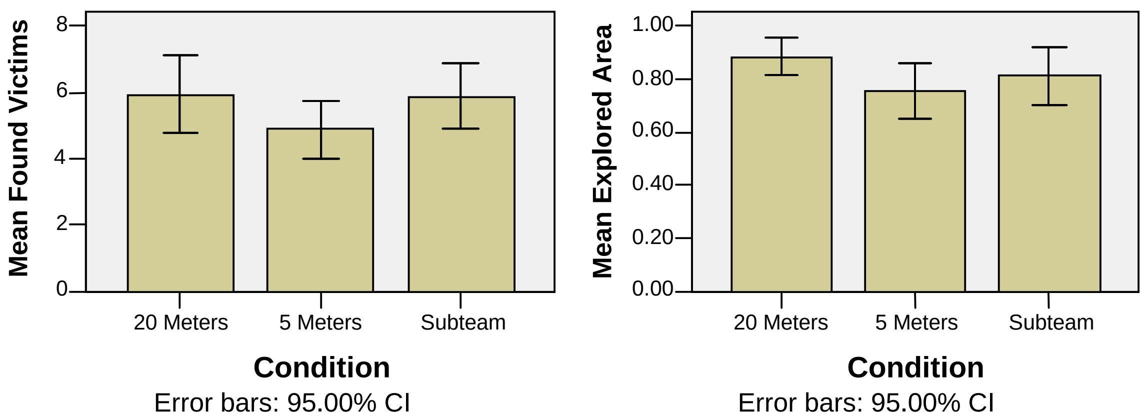


Figure 51. Found victims (left) and explored areas (right) by mode

Examination of the data showed that two participants failed to perform the task satisfactorily. One commented during debriefing that she thought she was supposed to mark inspector robots rather than victims. After removing these participants a paired t-test shows that in the first condition (20-meter range scanner) participants explored more regions, $t(16) = 3.097$, $p = 0.007$, as well as found more victims, $t(16) = 3.364$, $p = 0.004$, than under the second condition (short-range scanner) (Figure 51). In the third condition (automated subteam) participants found marginally more victims, $t(16) = 1.944$, $p = 0.07$, than in the second condition (controlled cooperation) but no difference was found for the extent of regions explored (Figure 51).

In the posttest survey, 12 of the 19 (63%) participants reported they were able to control the robots although they had problems in handling some interface components. Six of the 19

(32%) participants thought they used the interface very well. Only one participant reported it being hard to handle all the components on the user interface, but still maintained she was able to control the robots. Most participants (74%) thought it was easier to coordinate inspectors with explorers with long range scanner. Twelve of the 19 (63%) participants rated auto-cooperation between inspector and explorer robots (the sub-team condition) as improving their performance, and five (26%) participants thought auto-cooperation made no difference. Only two (11%) participants judged team autonomy to make things worse.

5.3.4.2 Coordination effort

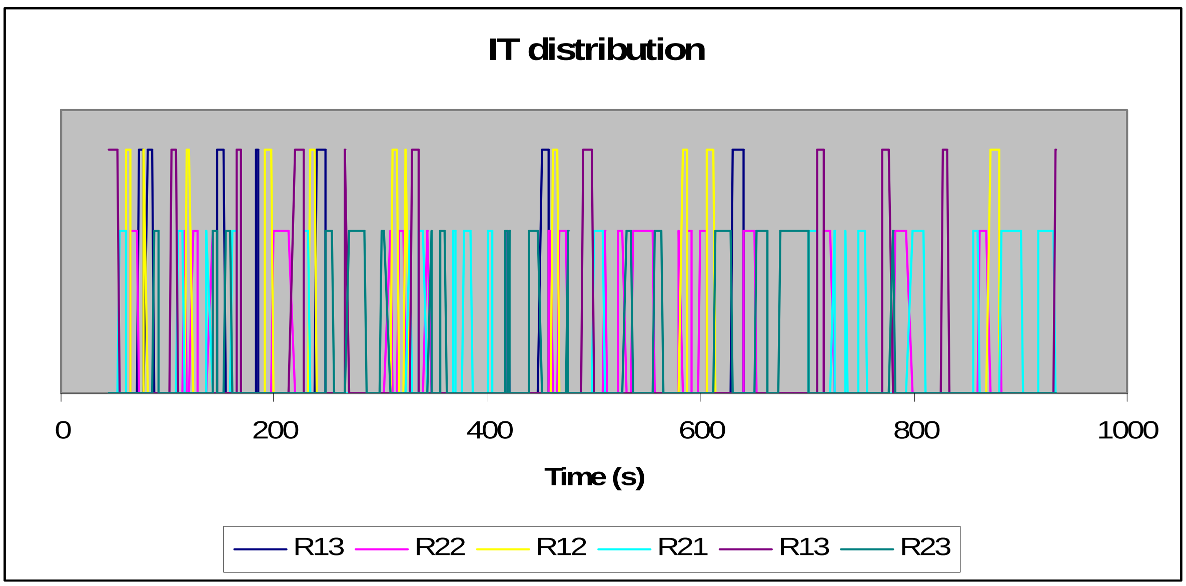


Figure 52. Typical (IT,FT) distribution (higher line indicates the interactions of explorers).

During the experiment we logged all the control operations with time-stamps. From the log file CDs were computed for each type robot according to the equation in section 5.1.3. Figure 52 shows a typical (IT,FT) distribution under the first condition (20-meter sensing range) in the experiment with a calculated CD of 0.185 for the *explorer* and of 0.06 for the *inspector*. The low CDs reflect that, in trying to control six robots, the participant ignored some robots while attending to others. The CD for *explorers* is roughly twice the CD for *inspector*s. After the participant controlled an *explorer*, he needed to control an *inspector* multiple times or multiple *inspector*s since the *explorer* has a long detection range and large FOV. In contrast, after controlling an *inspector*, the participant needed less effort to coordinate *explorers*.

Figure 53 shows the mean of measured CDs. We predicted that when the *explorer* has a longer detection range, operators need to control the *inspector*s more frequently to cover the mapped area. Therefore a longer detection range should lead to higher CD for *explorers*. This was confirmed by a two-tailed t-test that found a higher CD, $t(18) = 2.476$, $p = 0.023$, when participants controlled *explorers* with large (20-meter) sensing range.

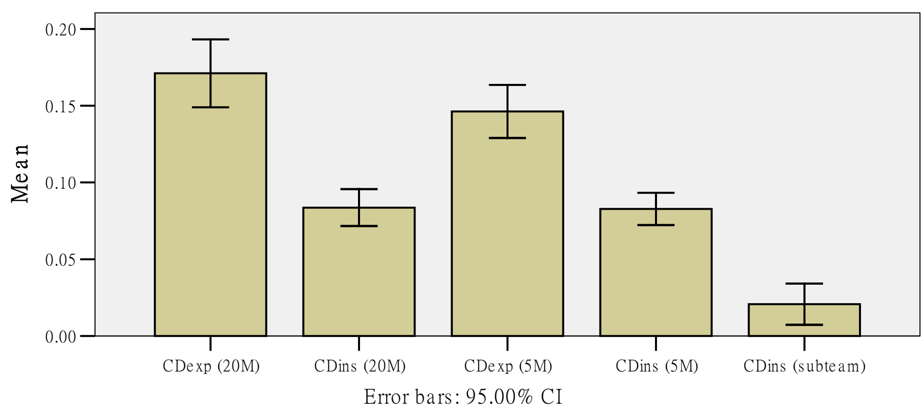


Figure 53. CDs for each robot type

We did not find a corresponding difference, $t(18)=.149$, $p=0.884$, between long- and short-range conditions for the CD for *inspector*s. This may have occurred because under these two conditions the *inspector*s have exactly the same capabilities and the difference in *explorer* detection range was not large enough to impact *inspector*s' CD for *explorers*. Under the subteam condition, the automatic cooperation within a subteam decreased or eliminated the coordination requirement when a participant controlled an *inspector*. Within-participant comparisons show that the measured CD of *inspector*s under this condition is significantly lower than the CD under the second condition (independent control with a five-meter detection range), $t(18) = 6.957$, $p < 0.001$. Because the *explorer* always tries to automatically follow an *inspector*, we do not report CD of *explorers* in this condition.

As auxiliary parameters, we evaluated the total attention demand, which is the occupation rate of total interaction time in the whole control period, and the action pattern, which is the ratio of control times between *inspector* and *explorer*. Total attention demand measures the team task demand, that is, the difficulty of the task. A paired t-test shows that, under long-sensing conditions, participants spent more time controlling robots than under the short-sensing condition, $t(18)=2.059$, $p=0.054$. This is opposite to our hypothesis that searching for victims using a shorter-sensing range should be more difficult because the robot would need to be controlled more often. Noting that the hypothesis was based on the number of times a robot was controlled rather than the time spent controlling a robot, we examined the number of control episodes. Under long- and short-sensing range conditions, two-tailed t-tests found that participants controlled explorers more often with short-sensing explorers, $t(18)=2.464$, $p=.024$ with no differences found in frequency of inspector control, $p=.97$. We believe that with longer-sensing explorers, participants tend to issue longer paths in order to build larger maps. Because the sensing range in the first condition is five times longer than the range in the second condition, the increased control time under the long-sensing condition may overwhelm the increased explorer control times. Thus, we found a higher total attention demand under the first condition. This is partially confirmed by a paired t-test that found longer average-control-time for explorers and inspectors under the long detection condition, $t(18)=3.139$, $p=.006$ and $t(18)=2.244$, $p=.038$, respectively. On average participants spent 1.5s and 1.0s more time in explorer and inspector control in the long-range condition. The mean action patterns under long and short-range scanner conditions were 2.31 and 1.9, respectively. This means that, with 20- and 5-meter scanning ranges, participants controlled *inspector*s 2.31 and 1.9 times, respectively, after an *explorer* interaction. Within-participant comparisons show that the ratio is significantly larger under long-sensing condition than under short-range scanner condition, $t(18) = 2.193$, $p = 0.042$.

5.3.4.3 Analyzing Performance

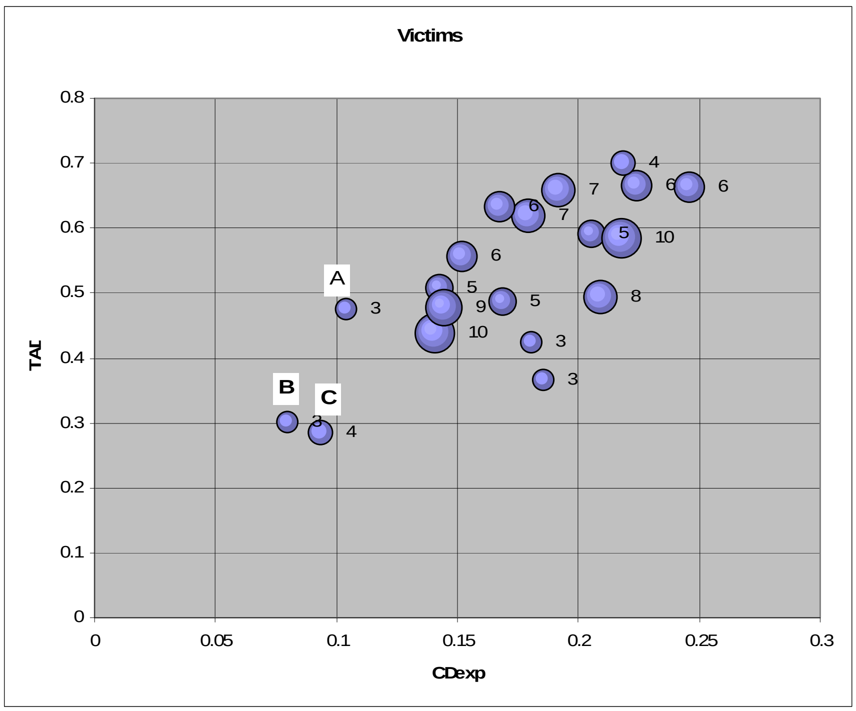
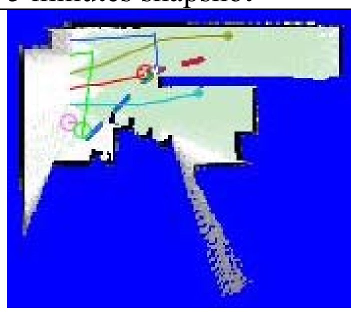
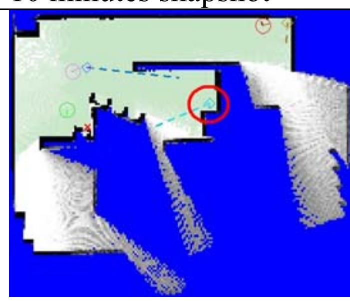
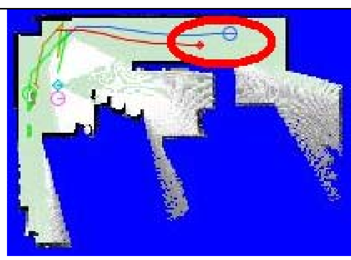
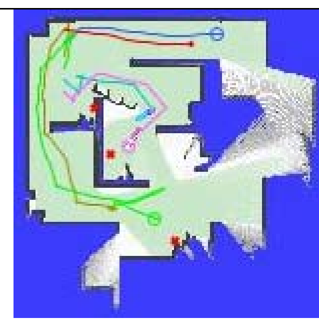
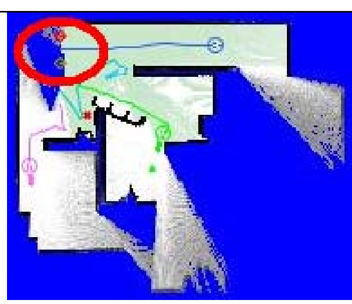
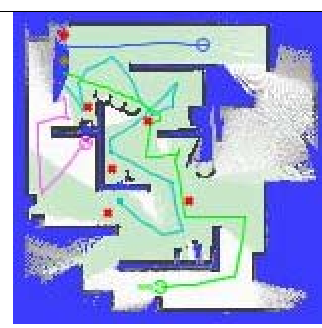


Figure 54 Found victims distribution over CDexp and TAD (total attention demand).

As an example of applying CDs to analyze coordination behavior, Figure 54 shows the performance over *explorer* CD and total attention demand under the 20-meter sensing range condition. We use the number of found victims, which is represented as the size of a bubble in the figure, to measure the overall performance. Figure 54 shows that when the participants ignored a particular explorer robot, most of them spent 13~25% of the neglect time in coordinating inspector robots, and the total time spent in controlling all the robots is 35~70% of the total task time. Character B and C marked in Figure 54 indicates two abnormal cases in which both the CD of explorer robot and the total attention demand are relatively low (less than 10% and 30% respectively), and the participants found only 3 or 4 victims. The total attention demand of case A marked in Figure 54 is within the range of 35~70%, however it is marginally abnormal because of the low CD of explorer robot and the small number of found victims. Although analyzing other cases, such as bad performance cases, may be interesting as well, here we only pick up case B, C and A to do the detailed performance analysis.

Table 7. Map snapshots of abnormal control behaviors

	5 minutes snapshot	10 minutes snapshot	15 minutes snapshot
A		 the robot on the center of the map was stuck	
B	 the two robots on the upper map were never controlled since then		
C	 the two robots on the upper left corner were totally ignored		

Associating these cases with recorded map snapshots, we observed that in case A, one robot was entangled by a desk and stuck after five minutes; in case B, two robots were controlled in the first five minutes and afterwards ignored; and in case C, the participant ignored two *inspector*s throughout the entire trial (Table 7). Comparing with case B and C, in case A only one robot didn't function properly after five minutes. This explains our observation in Figure 54 that A is closer to the normal cases than B and C.

5.3.5 Discussion

We proposed an extended Neglect Tolerance model to allow us to evaluate cooperation effort in applications where an operator must coordinate multiple robots to perform dependent tasks. The previous experiment validated CD measurement for an extended model under tight cooperation, such as box pushing. However, most target applications such as construction or search and rescue are likely to require weaker cooperation among heterogeneous platforms. The present experiment validated our NT extension under such weak cooperation conditions. Upon initial examination, our findings on CD for sensor ranges may seem counter-intuitive because inspectors would be expected to exert greater CD on explorers with short sensor range. Our data show, however, that this effect is not substantial and provide an argument for focused metrics like this that measure constituents of the human-robot system directly. Moreover, this experiment also shows how CD can be used to guide us identify and analyzing aberrant control behaviors.

We anticipated a correlation between found victims and the measured CDs. However, we did not find the expected relationship in this experiment. From observation of participants during the experiment we believe that high-level strategies, such as choosing areas to be searched and planning paths, have significant impact on the overall performance. The participants had few problems in learning to jointly control explorers and inspectors but they needed time to figure out effective strategies for performing the task. Because CD measures control behaviors rather than strategies, these effects were not captured. On the other hand, because the NT methodology is domain- and task-independent, our CD measurement could be used to characterize any dependent system. For use in performance analysis, however, it must be associated with additional domain- and task-dependent information. As shown in our examples, combined with generated maps and traces, CD provides an excellent diagnostic tool for examining performance in detail.

In the present experiment, we examined the action pattern under long- and short-sensing range conditions. The results reveal that it can be used as an evaluation parameter and, more importantly, it may guide us in the design of multi-robot systems. For instance, the observation that one explorer control action was followed on average by two inspector control actions may imply that the MRS should be constructed by n explorers and 2n inspectors.

6.0 CONCLUSIONS AND FUTURE WORK

Advances in robotic technologies and artificial intelligence allow robots to emerge from research laboratories into our everyday lives. However, experience in field applications such as homeland security, search and rescue, health care, personal assistance, and entertainment show that we have underestimated the importance of human-robot interaction (HRI) and that new problems arise in HRI as robotic technologies expand. This thesis classifies HRI along four dimensions—human, robot, task, and world—and illustrates that previous HRI classifications can be successfully interpreted as one of these elements or as the relationship between two or more of these elements. This perspective was used to review current HRI studies of single-operator single-robot (SOSR) control and single-operator multiple-robots (SOMR) control.

Human control of multiple robots has been suggested as a way to improve effectiveness in robot control. However, multiple robots substantially increase the complexity of the operator's task because attention must be continually shifted among robots, and human supervision will be needed to supply the perhaps changing goals that direct multirobot system activity. In addition, humans are likely to be called upon to assist with a variety of low-level problems such as sensor failures or obstacles that robots cannot solve on their own. One approach to increasing human capacity for control is to allow robots to cooperate, reducing the need to control them independently. Unlike the previous studies that investigate human-robot interaction either in low-fidelity simulations or for simple tasks, this thesis studies human interaction with cooperating robot teams within a realistically complex environment.

USARSim, a high-fidelity game engine-based robot simulator, and MrCS, a distributed multirobot control system, were developed to serve as an HMRI study testing bed. In the pilot experiment, we compared the control of small robot teams in which cooperating robots explored autonomously and were controlled independently by an operator or through mixed initiative as a cooperating team. Mixed-initiative teams found more victims and searched wider areas than either fully autonomous or manually controlled teams. Operators who switched attention between robots more frequently were found to perform better in both manual and mixed-initiative conditions. The control episode observed in this experiment reveals that, for a multi-robot system, decomposing the team goal into independent or weakly related sub-goals and allowing the human operator to intervene in the sub-goals are potential ways to improve and analyze human-multi-robot performance.

To avoid limitations to particular application fields, the present thesis focuses on common HRI evaluations that enable us to analyze HRI effectiveness and guide HRI design independently of the robotic system or application domain. Theories based on Neglect Tolerance (NT) evaluate HRI in this way. This thesis introduces the interaction episode (IEP), which was inspired by our pilot human-multirobot control experiment, to extend NT to support general robot or multi-robot control for complex tasks. Cooperation Effort (CE), Cooperation Demand (CD), and Team Attention Demand (TAD) are defined to measure the cooperation in SOMR control. Other HRI metrics introduced in NT are extended as well. Finally, two experiments were conducted to validate the proposed NT model under tight and weak cooperation conditions, respectively. The results show that, as a generic HRI metric, CD is able to account for the various factors that affect HRI and could be used in HRI evaluation and analysis. With the measured CD, we were able to quickly identify and analyze abnormal control behaviors under both experiments.

6.1 FUTURE RESEARCH WORK

The thesis proposed an extended Neglect Tolerance model that can be generally used to evaluate cooperation effort in applications where an operator must coordinate multiple robots to perform dependent tasks. The reported two experiments validated the CD and TAD measurements under tight and weak cooperation conditions. Fan-out (FO), which is the maximum number of robots that a single operator is capable of controlling, is another important index that measures the efficiency of the human-robot system. Previous studies [19, 20, 22] examined FO for independent tasks. However, in some applications, such as construction or search and rescue, humans may be required to coordinate heterogeneous robots to perform a task. Finding the maximum number of robot groups an operator is able to control is an important yet difficult problem because of the cooperation constraints among the robots. With the extended NT model, this thesis proposes the FO measurement for dependent tasks. In the future, we could extend the FO measurement to weak cooperation conditions to help us find the robot pattern and the maximum number of robot patterns one operator can manipulate. A validation experiment, of course, will be needed.

In the weak cooperation experiment, the time-based assessment showed higher coordination demand under a longer sensing condition. The control times evaluation reported more control times, which implies a higher coordination demand in the shorter sensing condition. This difference illustrates how the measurement unit, control time, or control times may impact the HRI evaluation. Usually, the time-consuming operations such as teleoperation are suited to time-based assessment. In contrast, control times may provide more accurate evaluation to the one-time style operations such as command issuing. Modifying the NT model to suit control time-based evaluation should be an area for future work.

In the pilot experiment, we found an inverted "U" relationship between performance and human interaction. Too much or too little human intervention led to poor human-robot performance. Furthermore, the flatness of the curve seems to reflect how sensitive the system is to human control. For example, our experiment shows that a mixed-initiative system was associated with a flatter curve, and it was indeed less sensitive to human interaction than the manually controlled system. Evaluating the performance and human intervention relationship can be another domain-independent HRI assessment methodology. It measures how the system tolerates variations in human interaction. A good system should be insensitive to human intervention, so it has fewer requirements in user training. Proposing and validating sensitivity metrics to measure the relationship between performance and human interaction can be another area for future research.

Increasing the autonomy level in robotic systems is supposed to be able to reduce cognitive demand when a single operator controls multiple robots. The current work focuses on evaluating HRI for a single operator controlling a cooperating robot team. However, in terms of improving human-robot performance, previous robot teleoperation studies [9-11] in USAR shows that multiple operators can significantly improve the performance as well. For example, [10] reports that the performance of two operators is nine times better than one operator's performance. Multiple operators controlling cooperating robot teams seems to be a promising approach that allows us to pursue both effectiveness and efficiency. Our MrCS is a distributed system that can be easily extended to support multiple operators (human agents with a user interface). Applying our extended NT model to study the cooperation effort in multiple-operator multiple-robot control will be very interesting. It deserves more work in the future.

BIBLIOGRAPHY

[1] Adams, J.A., *Critical Considerations for Human-Robot Interface Development*, in *Proceedings of the 2002 AAAI Fall Symposium: Human Robot Interaction*. 2002: Menlo Park, CA. p. 1-8.

[2] Albrecht, S., et al., *Device Level Simulation of Kurt3D Rescue Robots*, in *Third International Workshop on Synthetic Simulation and Robotics to Mitigate Earthquake Disaster*. 2006.

[3] Albus, J., *4-D/RCS Reference Model Architecture For Unmanned Ground Vehicles*, in *Proceedings of the SPIE AeroSense Session on Unmanned Ground Vehicle Technology*, G. Gerhart, Gunderson, R., and Shoemaker, C, Editor. 2000: Orlando, FL. p. 11-20.

[4] Baker, M., et al., *Improved Interfaces for Human-Robot Interaction in Urban Search and Rescue*, in *Proceedings of the IEEE Conference on Systems, Man and Cybernetics*. 2004. p. 2960-2965.

[5] Balakirsky, S., et al., *Towards heterogeneous robot teams for disaster mitigation: Results and Performance Metrics from RoboCup Rescue.* Journal of Field Robotics, 2007, in press.

[6] Boutilier, C., *Planning, learning and coordination in multiagent decision processes*, in *Proceedings of the 6th conference on Theoretical aspects of rationality and knowledge*. 1996, Morgan Kaufmann Publishers Inc: The Netherlands. p. 195 - 210.

[7] Browning, B. and E. Tryzelaar, *UberSim: A Realistic Simulation Engine for Robot Soccer*, in *Proceedings of Autonomous Agents and Multi-Agent Systems, AAMAS'03*. 2003: Australia.

[8] Bruce, A., I. Nourbakhsh, and R. Simmons, *The Role of Expressiveness and Attention in Human-Robot Interact ion*, in *Proceedings of the 2002 IEEE International Conference on Robotics 8 Automation*. 2002: Washington, DC. p. 4138-4142.

[9] Bruemmer, D.J., et al., *Shared Understanding for Collaborative Control.* IEEE Transactions On Systems, Man, And Cybernetics-Part A: Systems And Humans, 2005. **35**(4): p. 494-504.

[10] Burke, J.L. and R.R. Murphy, *Human-robot interaction in usar technical search: Two heads are better than one*, in *Proceedings of the IEEE International Workshop on Robot and Human Interactive Communication*. 2004.

[11] Burke, J.L., et al., *Moonlight in Miami: Field Study of Human-Robot Interaction in the Context of an Urban Search and Rescue Disaster Response Training Exercise.* Human Ccompter Interaction, 2004. **19**(1-2): p. 85-116.

[12] Carpin, S., et al., *Quantitative assessments of USARSim accuracy*, in *Proceedings of PerMIS 2006*. 2006.

[13] Carpin, S., et al., *High fidelity tools for rescue robotics: Results and perspectives*, in *Robocup 2005: Robot Soccer World Cup IX*. 2005. p. 301-311.

[14] Casper, J. 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, 2003. **33**(3): p. 367- 385.

[15] Chaimowicz, L., et al., *Deploying Air-Ground Multi-Robot Teams in Urban Environments*, in *Proceedings of the 2005 International Workshop on Multi-Robot Systems*. 2005: Washington, DC. p. 223-234.

[16] Chapanis, A., et al., *Human Engineering for An Effective Air Navigation and Traffic Control System.*, ed. P.M. Fitts. 1951, Washington, D.C.: National Research Council.

[17] Craighead, J., R.R. Murphy, and J. Burke. *A Survey of Commercial and Open Source Unmanned Vehicle Simulators*. Sumitted to Proceedings of ICRA 2007.

[18] Crandall, J. and M. Goodrich, *Characterizing Efficiency of Human Robot Interaction: A Case Study of Shared-Control Teleoperation*, in *Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2002.

[19] Crandall, J.W., et al., *Validating human-robot interaction schemes in multitasking environments.* IEEE Transactions on Systems, Man, and Cybernetics, Part A, 2005. **35**(4): p. 438-449.

[20] Cummings, M.L. and P.M. Mitchell, *Management of Multiple Dynamic Human Supervisory Control Tasks for UAVs*, in *Human Computer Interaction International Human Systems Integration Conference*. 2005: Las Vegas, Nevada.

[21] Cummings, M.L. and P.M. Mitchell, *Managing Multiple UAVs through a Timeline Display*, in *Proceedings of AIAA Info Tech*. 2005: Arlington, VA.

[22] Cummings, M.L., C. Nehme, and J. Crandall, *Predicting Operator Capacity for Supervisory Control of Multiple UAVs*, in *Innovations in Intelligent UAVs: Theory and Applications*, L. Jain, Editor. 2006, In Press.

[23] Cyberbotics Ltd. (2006). *Webots*. Retrieved December 07, 2006, from http://www.cyberbotics.com/

[24] Drury, J.L., J. Scholtz, and H.A. Yanco, *Awareness in human-robot interactions*, in *Proceedings of the IEEE Conference on Systems, Man and Cybernetics*. 2003. p. 912-918.

[25] Drury, J.L., J. Scholtz, and H.A. Yanco, *Applying CSCW and HCI techniques to human-robot interaction*, in *CHI 2004 Workshop on Shaping Human-Robot Interaction*. 2004: Vienna. p. 13-16.

[26] Dudek, G., M.R.M. Jenkin, and E. Milios, *A Taxonomy of Multirobot Systems*, in *Robot Teams: From Diversity to Polymorphism*, T. Balch and L.E. Parker, Editors. 2002, AK Peters: Wellesley,MA. p. 3-22.

[27] Dudek, G., et al., *A Taxonomy for Multi-Agent Robotics.* Autonomous Robots, 1996. **3**(4): p. 375-397.

[28] Endsley, M.R., *Design and Evaluation for Situation Awareness Enhancement*, in *Proceedings of the Human Factors Society 32nd Annual Meeting*. 1988, Human Factors Society: Santa Monica, CA. p. 97-101.

[29] Endsley, M.R., *Toward a theory of situation awareness in dynamic systems.* Human factors, 1995. **37**(11): p. 32-64.

[30] Endsley, M.R., *Automation and situation awareness*, in *Automation and human performance: Theory and applications*, R. Parasuraman and M. Mouloua, Editors. 1996, Erlbaum: Mahwah, NJ. p. 163-181.

[31] Endsley, M.R., et al., *A comparative analysis of SAGAT and SART for evaluations of situation awareness*, in *Proceedings of the 42nd annual meeting of the Human Factors and Ergonomics Society*. 1998: Chicago, IL.

[32] Envarli, I.C. and J.A. Adams, *Task Lists for Human-Multiple Robot Interaction*, in *Proceedings of the 14th IEEE International Workshop on Robot and Human Interactive Communication*. 2005.

[33] Epic Games Inc. (2006). *Unreal Engine 2*. Retrieved December 07, 2006, from http://www.unrealtechnology.com/html/technology/ue2.shtml

[34] Farinelli, A., L. Iocchi, and D. Nardi, *Multirobot Systems: A Classification Focused on Coordination.* IEEE Transactions On Systems, Man, And Cybernetics-Part B, 2004. **34**(5): p. 2015-2028.

[35] Fong, T., C. Thorpe, and C. Baur, *Multi-Robot Remote Driving With Collaborative Control.* IEEE Transactions on Industrial Electronics, 2003. **50**(4): p. 699-704.

[36] Fong, T.W. and C. Thorpe, *Vehicle Teleoperation Interfaces.* Autonomous Robots, 2001. **11**(1): p. 09-18.

[37] Fong, T.W., C. Thorpe, and C. Baur, *Collaboration, Dialogue, and Human-Robot Interaction*, in *Proceedings of the 10th International Symposium of Robotics Research*. 2001, Springer-Verlag: Lorne, Victoria, Australia.

[38] Gerkey, B., R. Vaughan, and A. Howard, *The Player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems*, in *Proceedings of the International Conference on Advanced Robotics (ICAR 2003)*. 2003: Coimbra, Portugal. p. 317-323.

[39] Goodrich, M., M. Quigley, and K. Cosenzo., *Switching and multi-robot teams*, in *Proceedings of the Third International Multi-Robot Systems Workshop*. 2005.

[40] Gutwin, C. and S. Greenberg, *Workspace Awareness for Groupware*, in *Proceedings of the Conference on Human Factors in Computing Systems*. 1996: Vancouver, British. p. 208-209.

[41] Gutwin, C. and S. Greenberg, *A Descriptive Framework of Workspace Awareness for Real-Time Groupware.* Computer Supported Cooperative Work, 2002. **11**(3): p. 411-446.

[42] Gutwin, C. and S. Greenberg, *The Importance of Awareness for Team Cognition in Distributed Collaboration*, in *Team Cognition: Understanding the Factors that Drive Process and Performance*, E. Salas and S.M. Fiore, Editors. 2004, APA Press: Washington. p. 177-201.

[43] Hart, S.G. and L.E. Staveland, *Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research*, in *Human mental workload*, Hancock and N. Meshkati, Editors. 1988, American Helicopter Society: Washington, DC. p. 657-672.

[44] Hughes, S. and M. Lewis, *Task-Driven Camera Operations for Robotic Exploration.* Transactions on Systems, Man and Cybernetics Part A, IEEE, 2005. **35**(4): p. 513-522.

[45] Humphrey, C.M., S.M. Gordon, and J.A. Adams, *Visualization of Multiple Robots During Team Activities*, in *Proceedings of the 2006 Human Factors and Ergonomics Society 50th Annual Meeting*. 2006: San Francisco, CA.

[46] Humphrey, C.M., et al., *Evaluating a scaleable Multiple Robot Interface based on the UTFARSim Platform*. 2006, Human-Machine Teaming Laboratory.

[47] Hunn, B.P., *The Human Challenges Of Command And Control With Multiple Unmanned Aerial Vehicles*, in *Proceedings Of The Human Factors And Ergonomics Society 49th Annual Meeting*. 2005. p. 20-24.

[48] Jacobson, J. and M. Lewis, *Game engine virtual reality with CaveUT.* Computer, 2005. **38**(4): p. 79-82.

[49] Jacoff, A., E. Messina, and J. Evans, *Experiences in deploying test arenas for autonomous mobile robots*, in *Proceedings of the 2001 Performance Metrics for Intelligent Systems (PerMIS) Workshop*. 2001: Mexico City, Mexico.

[50] Kaber, D.B. and M.R. Endsley, *The effects of level of automation and adaptive automation on human performance, situation awareness and workload in a dynamic control task.* Theoretical Issues in Ergonomics Science, 2004. **5**(2): p. 113-153(41).

[51] Kaminka, G.A. and Y. Elmaliach, *Experiments with an Ecological Interface for Monitoring Tightly-Coordinated Robot Teams*, in *Proceedings of the IEEE International Conference on Robotics and Automation*. 2006.

[52] Kaminka, G.A., et al., *GameBots: a flexible test bed for multiagent team research.* Communications of the ACM, 2002. **45**(1): p. 43-45.

[53] Keyes, B., et al., *Camera Placement and Multi-Camera Fusion for Remote Robot Operation*, in *Proceedings of the IEEE International Workshop on Safety, Security and Rescue Robotics*. 2006.

[54] Kiesler, S., *Fostering common ground in human-robot interaction*, in *Proceedings of 2005 IEEE International Workshop on Robot and Human Interactive Communication*. 2005. p. 729-734.

[55] Konolige, K. and K. Myers, *The Saphira Architecture for Autonomous Mobile Robots*, in *Artificial intelligence and mobile robots: case studies of successful robot systems*, D. Kortenkamp, R.P. Bonasso, and R. Murphy, Editors. 1998, MIT Press: Cambridge, MA. p. 211-242.

[56] Lee, J.D. and K.A. See, *Trust in Automation: Designing for Appropriate Reliance.* Human Factors, 2004. **46**(1): p. 50-80.

[57] Lee, P., D. Ruspini, and O. Khatib, *Dynamic simulation of interactive robotic environment*, in *Proceedings of International Conference on Robotics and Automation*. 1994. p. 1147-1152.

[58] Lewis, M. and J. Jacobson, *Game Engines in Research.* Communications of the Association for Computing Machinery, 2002. **45**(1): p. 27-48.

[59] Lewis, M., et al., *Experiments with attitude: attitude displays for teleoperation*, in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*. 2003. p. 1345-1349.

[60] Lindeman, R.W., et al., *Effectiveness of Directional Vibrotactile Cuing on a Building-Clearing Task*, in *Proceedings of ACM CHI 2005*. 2005: Portland, Oregon, USA.

[61] Lorenz, B., et al., *The effects of level of automation on the out-of-the-loop unfamiliarity in a complex dynamic fault-management task during simulated spaceflight operations*, in

Proceedings of the Human Factors and Ergonomics Society 45th Annual Meeting. 2001: Santa Monica, CA. p. 44-48.

[62] Magerko, B., et al., *AI Characters and Directors for Interactive Computer Games*, in *Proceedings of the 2004 Innovative Applications of Artificial Intelligence Conference*. 2004.

[63] Marble, J.L., D.J. Bruemmer, and D.A. Few, *Lessons learned from usability tests with a collaborative cognitive workspace for human-robot teams*, in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*. 2003. p. 448-453.

[64] Marble, J.L., et al., *Evaluation of supervisory vs. peer-peer interaction with human-robot teams*, in *Proceedings of the 37th Annual Hawaii International Conference on System Sciences*. 2004.

[65] Mataric, M.J., *Coordination and learning in multirobot systems.* IEEE Intelligent Systems, 1998: p. 6-8.

[66] Mathengine, *Mathengine Karma User Guide*. 2002.

[67] McFarlane, D.C., *Interruption of People in Human-Computer Interaction*. 1998, George Washington University: Washington, DC.

[68] McFarlane, D.C. and K.A. Latorella, *The Scope and Importance of Human Interruption in Human-Computer Interaction Design.* Human-Computer Interaction, 2002. **17**(1): p. 1-61.

[69] McGovern, D.E., *Experience and results in teleoperation of land vehicles*, in *Pictorial Communications in Virtual and Real Environments*, S. Eills, Editor. 1991, Taylor and Francis: New York. p. 182-195.

[70] Meier, R., et al., *A Sensor Fusion Based User Interface for Vehicle Teleoperation*, in *Proceedings of the International Conference on Field and Service Robotics (FSR '99)*. 1999.

[71] Meiran, N., Z. Chorev, and A. Sapir, *Component processes in task switching.* Cognitive Psychology, 2000. **41**(4): p. 211-253.

[72] Miller, C.A. and R. Parasuraman, *Beyond Levels Of Automation: An Architecture For More Flexible Human-Automation Collaboration*, in *Proceedings of the Human Factors and Ergonomics Society 47th Annual Meeting*. 2003. p. 182-186.

[73] Molino, V., et al., *Traversability Metrics For Urban Search and Rescue Robots On Rough Terrain*, in *Proceedings of the Performance Metrics for Intelligent Systems*. 2006: Gaithersburg, MD.

[74] Moray, N., T. Inagaki, and M. Itoh, *Adaptive automation, trust and self-confidence in fault management of time-critical tasks.* Journal of Experimental Psychology: Applied, 2000. **6**: p. 44-58.

[75] Murphy, R.R. and J.L. Burke, *Up from the Rubble: Lessons Learned about HRI from Search and Rescue*, in *Proceedings of the 49th Annual Meetings of the Human Factors and Ergonomics Society*. 2005: Orlando.

[76] Nass, C., *Etiquette equality: exhibitions and expectations of computer politeness.* Communications of the ACM, 2004. **47**(4): p. 35-37.

[77] National Institute of Standards and Technology.*Urban Search and Rescue Robot Competitions Rules*. Retrieved November 14, 2006, from http://www.isd.mel.nist.gov/projects/USAR/rules.htm

[78] Nickerson, J.V. and S.S. Skiena, *Attention and Communication: Decision Scenarios for Teleoperating Robots*, in *Proceedings of the 38th Annual Hawaii International Conference on System Sciences*. 2005.

[79] Nielsen, C.N., M.A. Goodrich, and J.W. Crandall, *Experiments in Human-Robot Teams*, in *Proceedings of the 2002 NRL Workshop on Multi-Robot Systems*. 2003.

[80] Nielsen, C.W. and M.A. Goodrich, *Comparing the Usefulness of Video and Map Information in Navigation Tasks*, in *Proceedings of the 2006 Human-Robot Interaction Conference*. 2006: Salt Lake City, Utah.

[81] Nielsen, C.W., M.A. Goodrich, and R.J. Rupper, *Towards facilitating the use of a pan-tilt camera on a mobile robot*, in *Proceedings of the 2005 IEEE International Workshop on Robots and Human Interactive Communication*. 2005. p. 568-573.

[82] Olsen, D.R. and M.A. Goodrich, *Metrics for evaluating human-robot interactions*, in *Proceedings of PERMIS 2003*. 2003.

[83] Olsen, D.R. and S.B. Wood, *Fan-out: measuring human control of multiple robots*, in *Proceedings of the SIGCHI conference on Human factors in computing systems*. 2004, ACM Press: Vienna, Austria. p. 231-238.

[84] Olsen, D.R., S.B. Wood, and J. Turner, *Metrics for human driving of multiple robots*, in *Proceedings of IEEE International Conference on Robotics and Automation 2004*. 2004. p. 2315-2320.

[85] Parasuraman, R., et al., *A Flexible Delegation-Type Interface Enhances System Performance in Human Supervision of Multiple Robots: Empirical Studies with RoboFlag.* IEEE Systems, Man and Cybernetics-Part A, Special Issue on Human-Robot Interactions, 2005. **33**(4): p. 481-493.

[86] Parasuraman, R. and C.A. Miller, *Trust and etiquette in high-criticality automated systems.* Communications of the ACM, 2004. **47**(4): p. 51-55.

[87] Parasuraman, R., T.B. Sheridan, and C.D. Wickens, *A Model for Types and Levels of Human Interaction with Autonomation.* IEEE Transactions On Systems, Man, And Cybernetics-Part A: Systems And Humans, 2000. **30**(3): p. 286-297.

[88] Parker, L.E., *Lifelong Adaptation in Heterogeneous Multi-Robot Teams: Response to Continual Variation in Individual Robot Performance.* Autonomous Robots, 2000. **8**(3): p. 239-267.

[89] Reid, G.B. and T.E. Nygren, *The subjective workload assessment technique: A scaling procedure for measuring mental workload*, in *Human mental workload*, P.A. Hancock and N. Meshkati, Editors. 1988, Elsevier: Amsterdam. p. 185-218.

[90] Rodden, T., *Populating the application: A model of awareness for cooperative applications*, in *Proceedings of the Conference on Computer-Supported Cooperative Work*. 1996, ACM press: New York, N.Y. p. 87-96.

[91] Rogers, E. and R.R. Murphy. (2001). *Human-robot interaction*. Retrieved September 12, 2006, from http://www.csc.calpoly.edu/~erogers/HRI/HRI-report-final.html

[92] Rubio, S., et al., *Evaluation of Subjective Mental Workload: A Comparison of SWAT, NASA-TLX, and Workload Profile Methods.* Applied Psychology: An International Review, 2004. **53**(1): p. 61-86.

[93] Scerri, P., et al., *Allocating tasks in extreme teams*, in *Proc. of the fourth international joint conference on Autonomous agents and multiagent systems*. 2005.

[94] Scerri, P., et al., *Coordinating large groups of wide area search munitions*, in *Recent Developments in Cooperative Control and Optimization*, D. Grundel, R. Murphey, and P. Pandalos, Editors. 2004, Singapore: World Scientific. p. 451-480.

[95] Schmidt, K., *The Problem with 'Awareness': Introductory Remarks on 'Awareness in CSCW'.* The Journal of Collaborative Computing, 2002. **11**(3-4): p. 285-298.

[96] Scholtz, J., *Theory and Evaluation of Human Robot Interactions*, in *Proceedings of the 36th Annual Hawaii International Conference on System Sciences (HICSS'03)*. 2003: Hawaii.

[97] Scholtz, J., B. Antonishek, and J. Young, *Evaluation of a human-robot interface: development of a situational awareness methodology*, in *Proceedings of the 37th Annual Hawaii International Conference on System Sciences*. 2004.

[98] Scholtz, J., B. Antonishek, and J. Young, *A field study of two techniques for situation awareness for robot navigation in urban search and rescue*, in *Proceedings of the IEEE International Workshop on Robot and Human Interactive Communication*. 2005: Roman. p. 131-136.

[99] Scholtz, J., et al., *Evaluation of human-robot interaction awareness in search and rescue*, in *Proceedings of the 2004 International Conference on Robotics and Automation*. 2004.

[100] Schurr, N., et al., *The Future of Disaster Response: Humans Working with Multiagent Teams using DEFACTO*, in *Proceedings of AAAI Spring Symposium on AI Technologies for Homeland Security*. 2005.

[101] Scott, R.G. and R.C. Richardson, *Issues in Debris Field Characterization and USAR Robot Interaction*, in *Proceedings of the IEEE International Workshop on Safety, Security and Rescue Robotics*. 2006: Gaithersburg, MD.

[102] Sibert, J., et al., *Vibrotactile Feedback for Enhanced Control of Urban Search and Rescue Robots*, in *Proceedings of the IEEE International Workshop on Safety, Security and Rescue Robotics*. 2006: Gaithersburg, MD.

[103] Simone, C. and S. Bandini, *Integrating Awareness in Cooperative Applications through the Reaction-Diffusion Metaphor.* The Journal of Collaborative Computing, 2002. **11**(3-4): p. 495-530.

[104] Squire, P., G. Trafton, and R. Parasuraman, *Human control of multiple unmanned vehicles: effects of interface type on execution and task switching times*, in *Proceedings of the 2006 Human-Robot Interaction Conference*. 2003: Salt Lake City, Utah. p. 26-32.

[105] Steinfeld, A., et al., *Common Metrics for Human-Robot Interaction*, in *Proceedings of the 2006 Human-Robot Interaction Conference*. 2006: Salt Lake City, Utah.

[106] Sugimoto, M., et al., *Time Follower's Vision: a teleoperation interface with past images.* IEEE Computer Graphics and Applications, 2005. **25**(1): p. 54-63.

[107] Sukthankar, G., et al., *Modeling Physical Variability for Synthetic MOUT Agents*, in *Proceedings of the 2004 Conference on Behavior Representation in Modeling and Simulation*. 2004.

[108] Tambe, M., *Towards Flexible Teamwork.* Journal of Artificial Intelligence Research, 1997. **7**: p. 88-124.

[109] Trafton, J.G., et al., *Enabling effective human-robot interaction using perspective-taking in robots.* IEEE Transactions on Systems, Man and Cybernetics, Part A, 2005. **35**(4): p. 460-470.

[110] Trouvain, B., C. Schlick, and M. Mevert, *Comparison of a map- vs. camera-based user interface in a multi-robot navigation task*, in *Proceedings of the 2003 International Conference on Robotics and Automation*. 2003. p. 3224-3231.

[111] Trouvain, B. and H.L. Wolf, *Evaluation of multi-robot control and monitoring performance*, in *Proceedings of the 2002 IEEE Int. Workshop on Robot and Human Interactive Communication*. 2002. p. 111-116.

[112] Tsang, P.S. and V.L. Velazquez, *Diagnosticity and multidimensional subjective workload ratings.* Ergonomics, 1996. **39**(3): p. 358-381.

[113] Wang, J. (2003). *USARSim User Manual*. Retrieved December 07, 2006, from http://downloads.sourceforge.net/usarsim/USARsim-manual2.1.0.pdf?modtime=1160068541&big_mirror=0

[114] Wang, J., M. Lewis, and S. Hughes, *Gravity-Referenced Attitude Display for Teleoperation of Mobile Robots*, in *Proceedings of the Human Factors and Ergonomics Society 48th Annual Meeting (HFES'05)*. 2004. p. 2662-2666.

[115] Wang, J., et al., *Validating USARsim for use in HRI Research*, in *Proceedings of the Human Factors and Ergonomics Society 49th Annual Meeting*. 2005. p. 457-461.

[116] Wang, J., M. Lewis, and P. Scerri, *Cooperating robots for search and rescue*, in *Proceedings of the AAMAS 1st International Workshop on Agent Technology for Disaster Management*. 2006. p. 92-99.

[117] Woods, D.D., et al., *Envisioning human-robot coordination in future operations.* IEEE Transactions on Systems, Man & Cybernetics, 2004. **34**(2): p. 210-218.

[118] Yanco, H.A. and J. Drury, *"Where Am I?" Acquiring Situation Awareness Using a Remote Robot Platform*, in *Proceedings of the IEEE Conference on Systems, Man and Cybernetics*. 2004.

[119] Yanco, H.A. and J.L. Drury, *A Taxonomy for Human-Robot Interaction*, in *Proceedings of the AAAI Fall Symposium on Human-Robot Interaction*. 2002: Falmouth, Massachusetts. p. 111-119.

[120] Yanco, H.A. and J.L. Drury, *Classifying Human-Robot Interaction: An Updated Taxonomy*, in *Proceedings of the IEEE Conference on Systems, Man and Cybernetics*. 2004.

[121] Yanco, H.A., J.L. Drury, and J. Scholtz, *Beyond Usability Evaluation: Analysis of Human-Robot Interaction at a Major Robotics Competition.* Journal of Human-Computer Interaction, 2004. **19**(1-2): p. 117-149.

[122] Zalud, L., *ARGOS – User Interface for Multiple Mobile Robot Teleoperation*, in *Proceedings of the IEEE International Workshop on Safety, Security and Rescue Robotics*. 2006: Gaithersburg, MD.

[123] Zaratti, M., M. Fratarcangeli, and L. Iocchi, *A 3D Simulator of Multiple Legged Robots based on USARSim*, in *Proc. of RoboCup Symposium 2006*. 2006.