Faculty and Researchers | Faculty and Researchers' Publications

2012

An Agent-Based Simulation Model of Human-Robot Team Performance in Military Environments

Giachetti, Ronald E.; Marcelli, Veronica; Cifuentes, José; Rojas, José A.

An Agent-Based Simulation Model of Human-Robot Team Performance in Military Environments

Ronald E. Giachetti,[1,]* Veronica Marcelli,[2] José Cifuentes,[2] and José A. Rojas[2]

[1]Naval Postgraduate School, Monterey, CA 93943
[2]Florida International University, Miami, FL 33174

ABSTRACT

Prior to deploying human-robot teams on military missions, system designers need to understand how design decisions affect team performance. This paper describes a multiagent simulation model that captures both team coordination and human-robot interaction. The purpose of the model is to evaluate proposed team designs in uncertain Military Operations in Urban Terrain (MOUT) scenarios and determine which design factors are most critical to team performance. The simulation model is intended to be a tool in the systems engineering iterations of proposing designs, testing them, and then evaluating them during the conceptual design phase. To illustrate the model's usefulness for this purpose, a fractional factorial design of experiments is conducted to evaluate team design factors and the two-factor interaction between controllable factors and noise factors that described the environment and robot reliability. The experimental results suggest that (1) larger teams have more robust performance over the noise factors, (2) robot reliability is critical to the formation of human-robot teams, and (3) high centralization of decision-making authority created communication bottlenecks at the commander in large teams. This work contributes to the agent-based modeling of teams, and to understanding how the U.S. Army can attain its goal of greater utilization of robots in future military operations. © 2012 Wiley Periodicals, Inc. Syst Eng 15: 000–000, 2012

Key words: agent-based simulation; teams; military operations; shared mental model; system test and evaluation

1. INTRODUCTION

The U.S. Army continues to advocate for greater utilization of robots including unmanned ground vehicles (UGVs) and unmanned aerial vehicles (UAVs) for tasks that have proven dangerous for soldiers. The U.S. military already makes significant use of semiautonomous robots in military operations including operations in Afghanistan and Iraq, and it has an overall goal to further develop and deploy more autonomous robots [Davies, 2009]. However, significant scientific hurdles remain, so that the deployment of robotic systems increases performance of the military without degrading the perform-

ance of the soldiers in the unit [Cosenzo and Barnes, 2010]. It is becoming clear that the robots should be deployed in human-robot teams that capitalize on the strengths of both humans and robots [Cosenzo and Barnes, 2010; Parasuraman, Barnes, and Cosenzo, 2007]. One potential application for mixed human-robot teams is military operations in urban terrain (MOUT). Normally, MOUT teams are only composed of humans. However, advances in technology are such that the U.S. Army now envisions how semiautonomous robots can be included in the team composition. These robots will provide logistical support as well operational support. For example, a team for Urban Search and Rescue might be formed of various human search experts as well as one or more robots designed to enter small crevices in the rubble inaccessible to humans [Murphy, 2004]. Thus, robots can be assigned tasks that are risky, difficult, or dangerous to the human team members. In these scenarios, robots are no longer regarded as just another piece of technology; instead, as robot autonomy increases, then the robot should be considered as a team member [Hoeft, Kochan, and Jensch, 2007].

The formation of human-robot teams for the military is a system-of-systems (SoS) engineering effort because it requires the integration of many technologies including sensors, displays, and robots, as well as the command and control architecture to coordinate the team's activities. Many of these component systems exist, and it their integration into an SoS that provides the desired capability. An SoS is a set of separately acquired systems, often managed under different authorities that through integrated efforts provide capabilities in fulfillment of a mission need [Maier, 1998; Sage and Cuppen, 2001]. Acquisition of new systems by the U.S. Department of Defense (DoD) employs the Joint Capabilities Integrated Development System (JCIDS) to determine the capabilities needed by combatant commands, and to support the decision of whether to pursue a materiel solution or a nonmateriel solution and enter the Defense Acquisition System to design and develop the system. To design a system for this environment requires many iterations of design, analysis, and evaluation to arrive at a suitable system design [Blanchard and Fabrycky, 1998; Giachetti et al., 1997]. Early analysis mostly focuses on the operational performance of the SoS in executing mission threads. This paper deals with the development of an agent-based simulation tool to support the analysis, testing, and evaluation of the human-robot teaming structures. Simulation as an evaluation tool is preferred during the early design stages because it can be used to evaluate many possible system configurations, in a short period of time, with less expense than alternative evaluation methods such as physical prototyping or field testing [Hill, Miller, and McIntyre, 2001]. The U.S. military has long seen simulation as playing an essential role in the domain of system acquisition because it allows military planners and system designers to analyze and understand system performance [National Research Council, 2006]. Moreover, agent-based simulation is especially appropriate for modeling and analyzing SoS because of the fidelity of modeling each component system as an agent with its own behavior.

The simulation tool alone does not evaluate system designs, but needs to be part of an experimental approach to derive useful results from the simulation experiments. We see the tool being used in a design of experiments in two contexts: first, to conduct screening experiments to identify significant design factors from a list of many factors and then, second, to evaluate the performance of system designs under operational conditions. In the paper, we illustrate how this can be done with the simulation model. Moreover, it is likely that one or more technologies used in the human-robot arrangement will change, or how they are employed will change during its operational life. The model may also be useful for evaluation of these changes. As an agent-based model, it is possible to update the agent representation for just one component, such as the robot, without needing to change other components in the simulation model.

The contribution of this research is the development of a simulation model that can be used by military planners to help design human-robot teams by understanding command and control. One of the many challenges is the coordination of human actions with robot actions. In this paper we describe a multiagent simulation model of human-robot team behavior with a focus on coordination processes for MOUT missions. The underlying concept is that by modeling individual team member behavior the aggregate team level behavior can be replicated. The model integrates two separate streams of research: one in team design and one in human-robot interaction. We have identified the main team factors and relationships from the literature and used this to build the model (See Rojas and Giachetti [2009] for further details). The simulation model is used to evaluate team design in an uncertain MOUT environment. The long-term goal of our research is to develop theories, models, and tools for human-robot team coordination in complex, dynamic, and demanding work environments exemplified by MOUT.

The paper is organized as follows. Section 2 reviews the literature on simulation of human-robot interaction and of robotic systems for military applications. Section 3 presents our human-robot team model and explains how the agent-based simulation model is constructed. Section 3.4 describes the simulation scenario and Section 3.5 describes the verification and validation performed on the model. Section 4 presents our experimental analysis. Section 5 describes and discusses the results. Section 6 concludes the paper and discusses the model's limitations and need for continuing research.

2. LITERATURE REVIEW

Our research deals with simulating human-robot teams and can be positioned in relation to two areas of literature: simulation of teams and simulation of robots. Simulation of teams is a subset of the broader research in simulation of organizations to allow researchers to generate and test hypothesis of how organizations work [Carley, 1999; Rouse and Boff, 2005]. The research in simulation of robotic systems can be further divided into that research that investigates teams of robots and the research focused on human-robot interaction. Of the various simulation models, agent-based modeling is especially good for representing the behavior of the individual team members and then understanding how it affects overall team performance [Fan and Yen, 2004].

The Virtual Design Team (VDT) project was one of the earliest simulation models developed to study team performance. VDT is an agent-based simulation model developed to study project teams such as found in the construction industry [Kunz, Levitt, and Jin, 1998]. VDT is built on Galbraith's information processing theory and uses discrete event simulation of the team member's actions to understand project performance. It models a project as a predefined, unchanging activity network with predefined task assignments for each team member. Since the original work, extensions to VDT include adding goal incongruence between team members [Thomsen, Levitt, and Nass, 2005] and modeling cultural differences of team members [Horii, Jin, and Levitt, 2005], and continues to be used [MacKinnon et al., 2007].

The Team-RUP model was developed to study the effects of team behavior on the performance of software development teams [Yilmaz and Philips, 2007]. This simulation model considers a dynamic job environment and provides flexibility in terms of the organizational structure and size of the organization being model. Team-RUP is used to investigate the relationship of autonomy and concurrency in coordination on the performance of the software development team. It is developed using the REPAST agent-simulation toolkit. Dong and Hu [2008] also use REPAST, but to model a team and conduct studies of a contract net protocol for task allocation to team members through a bidding process. Their model incorporates a measure of relationship between team members in terms of friendship. The team model is limited to centralized teams with static structures.

Acquisti et al. [2002] describe ongoing work using an agent-based modeling system called Brahms to model the work practices aboard the international space station so that insights as to how work is actually performed can be incorporated into the planning of future crew expeditions. Loper and Presnell [2005] used agent-based simulation to evaluate the performance at individual and at aggregated level for the Georgia Emergency Management Agency (GEMA). Martínez-Miranda and Pavón [2009] address the problem of team configuration by modeling the human characteristics of actual team members such as their creativity, emotional behavior, and trust.

In military modeling, there is great interest in the application of agent-based modeling [Crino, 2001]. CAST is an agent-based model used to model military teams where the environment and actions are certain but how the system behavior emerges is unknown [Yen et al., 2001]. CAST implements a shared mental model for the team as a shared database that all agents have access to. Woodaman [2000] finds agent-based simulation useful to evaluate operational policies of soldiers performing riot control duties. The work on simulating teams by Brennen et al. [2007] is notable as one of the few that do not use agent-based modeling. They developed a discrete-event simulation model to analyze team performance for the British navy.

A stream of research is on the modeling, design, and performance of teams comprised of two or more robots. Young and Kott [2009] identify some of the issues and progress against those issues for small teams of robots for military operations. These issues include situation awareness and communication within the robot team. A great deal of research has been on task assignment in multirobot teams [Mataric, Sukhatme, and Ostergaard, 2003] and coordination of team actions [Grabowski and Christiansen, 2005].

One stream of research is simulation of strictly robot teams to allow system designers to address a broad range of design problems that arise while developing robot systems. Friedmann, Petersen, and von Stryk [2008] develop a simulation framework to model the physical movement of humanoid robots so that designers can evaluate the robot kinematics, dynamics, and interaction with the environment. The simulation model's behavior is based on a physical model of the robots mechanical design. Balakirsky, Messina, and Albus [2002] describe a simulation framework to study multirobot teams. The USARSim simulation model developed by Balaguer et al. [2008] is distinguished because it combines simulation of the physical motion and also the issued related to human-robot interaction.

Wang and Lewis [2007] describe the use of the USARSim simulation model to investigate different robot control of teleoperation, waypoint/heterogeneous robots, waypoint control/homogeneous robots. Wang and Lewis demonstrate how the USARSim model can be used to study different methods of robot control. Researchers have studied various aspects of search and rescue type missions using agent-based modeling [Kitano et al., 1999]. Ortiz et al. [2009] use a simulation model to investigate team performance when replacing a human team member with a robot for military operations.

The agent-based simulation model in this research is distinguished from previous research in two main ways. First, we model a team that has both human and robot members. Previous research has modeled human teams, robot teams, or human-robot interaction. Modeling human-robot teams includes elements of all these aspects in a single model. Second, the agent-based model we present includes uncertainty in many parameters. Yen and Fan [2006] model the shared mental model as a common data repository that all agents have access to. In actual teams, such a construct does not exist. The mental model is shared to the degree that there is congruence between the team members' mental models, which may be less than perfect congruence. We represent this less than perfect sharing through probabilities. Most of the previous simulation models of team processes consider teams that are faced with defined and static activity structures. By static activity structure, the network of activities is known ahead of time, and no changes to the structure can occur while the simulation unfolds. In our model, we use a dynamic activity structure where activities and events unfold probabilistically such that in each simulation run it is possible that a different set of activities were performed. Representing uncertainty in the tasks to be performed is important to the modeling of military operations. In the next section, we describe our human-robot team model that builds on the theory of teams and human-robot interaction.

3. HUMAN-ROBOT TEAM MODEL

A team is a collection of individuals who are interdependent in their tasks, who share responsibility for outcomes, who see themselves, and are seen by others, as an intact social-entity

embedded in a larger social system [Sundstrom, DeMeuse, and Futrell, 1990]. There is a large body of research on teams and team performance (see Paris, Salas, and Cannon-Bowers [2000] and Kendall and Salas [2004] for general reviews). The reason for selecting an agent-based architecture for the simulation model is the close correspondence to how we view team composition and team processes. Teams are composed of members, each with different skills, knowledge, and motivations. Team members act autonomously and must coordinate their work with the work of other team members. Coordination is primarily via communication between members. The agent-based simulation allows the implementation of this view by the modeling of each team member as an autonomous agent, and the agents coordinate by sending/receiving messages.

The simulation is built using the agent infrastructure CybelePro written in Java. In CybelePro, agents are defined as "a group of event-driven activities that share data, thread, and execution concurrency structure."zaq;1 Activities are internal to the agent, and act on internal data in response to incoming events. Other agents are incapable of accessing or manipulating the internal data, which reinforces the notion of agent autonomy. In CybelePro, agents generate and deliver events that may or may not trigger the desired response in other agents of the multiagent simulation.

The MOUT Team Simulation model has a single task agent and multiple team member agents as shown in Figure 1. The task agent defines the mission the team needs to complete. The mission is divided into activities. Activities are represented in an activity-on-node network. There are two types of arcs between activities: Either control arcs that define prerequisites such as the Insert Robot activity must occur prior to the Robot Search activity, or information arcs that define information inputs to the activities.

A team member agent represents each team member, whether a human or robot. The team member agent contains task processing activities, coordination activities, and decision-making activities. Task activities are responsible for the execution of the task or portions of the task assigned to the agent. Coordination activities execute all the communication activities performed by the agent. Decision activities contain the decision rules of how the agent allocates his/her/its attention to activities and messages. The decision activities serve as a link between task and coordination activities. For example, if a discrepancy or an error event occurs, a decision activity will evaluate whether to proceed with the rework or send a communication to a supervisor or another team member. The model parameters are shown in Table I. The means of agent communication, coordination, and the human-robot interaction are described in the following subsections.

3.1. Communication

In MOUT, the primary means of communication is via radio transmissions that are monitored by the entire team. For this reason we model only synchronous communications. When a message is sent, it can either be broadcast to the group or directed toward an individual. When directed toward an individual, the sender first hails the receiver, waits for confirmation, and then, when confirmation is received, will convey the message. The message can be heard by all team members within range, but the sender designates that the message is for a particular individual; for example, "Sergeant Major, Sergeant Major, come in please ...." The length of time a sender will wait to send a communication is a model parameter. Messages are ordered by importance of the sender. A receiver makes a decision whether to receive the message or to finish the task the receiver is engaged in. This probability depends
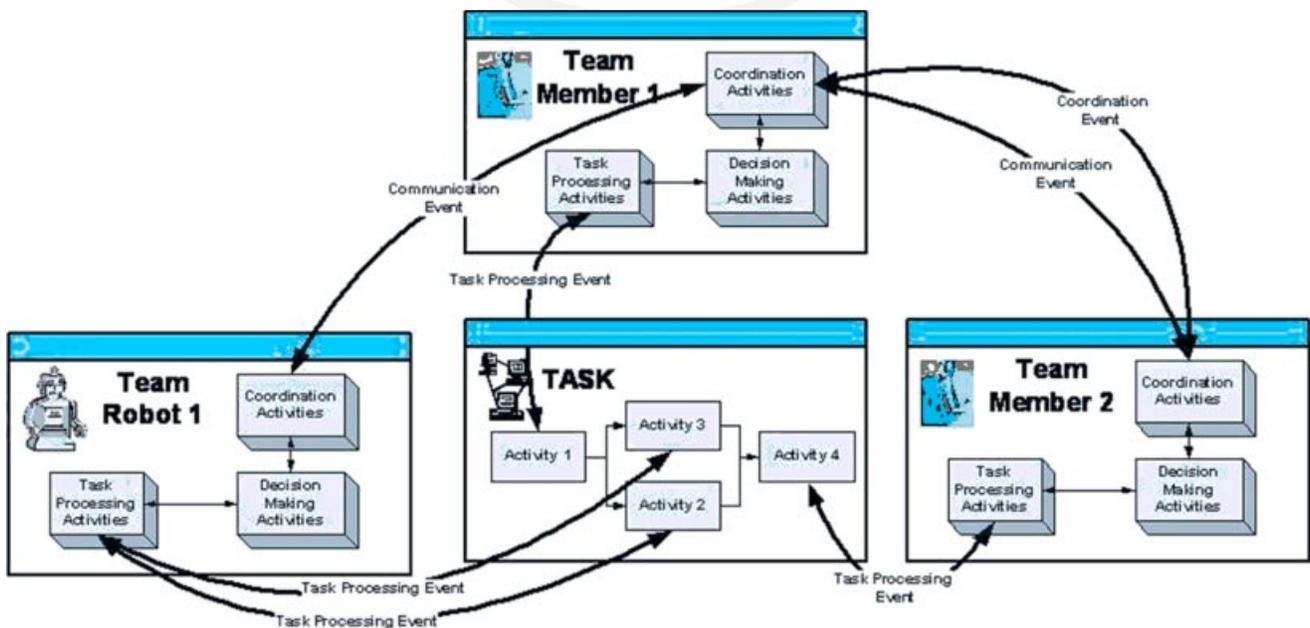

Figure 1. Agent model framework. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

Table I. Team Model Parameters

Team Level Parameters
Team centralization
Team formalization
Shared mental model agreement
Probability the leader makes decision
Probability of unsolicited communication
Probability of task error
Probability communication is relevant to recipient
Maximum time agent waits for task input requirements
Agent Level Parameters
Experience level
Teamwork skill level
Speed
Communication Parameters
Communication channel reliability
Robot Parameters
Neglect time
Interaction time
Mean time between failures (MTBF)
Mean time to repair (MTTR)
Robot task-failure probability
Failure detection probability
Recover-from-failure probability
Robot speed
Environmental Parameters
Probability containing target
Danger level

on the message importance. To illustrate, a soldier might acknowledge immediately a message from the commander whereas in the same situation the soldier might spend a minute completing his/her task before acknowledging a message from a mechanic. Consequently, messages may or may not preempt the receiver. If the receiver does not respond, and the sender's wait time is exceeded, then the sender will keep the message in its outbound message queue to reattempt sending the message after a random delay, which is an input parameter.

3.2. Coordination

There are two types of coordination among teams: planning prior to the mission start, and communication during the mission. Researchers have reported that planning prior to a mission, during a mission, or both can enhance team performance [Stout, Cannon-Bowers, and Salas, 1999]. For example, the team can set goals, create an open environment, share information related to task requirements (e.g., discuss the consequences of errors and discuss prepared information), and clarify each team member's roles and responsibilities. In addition, teams can discuss relevant environmental characteristics and constraints (e.g., how high workload affects performance, how the team will manage this constraint, and how they will deal with unexpected events), prioritize tasks, determine what types of information all team members have access to and what types of information are held by only certain members, and discuss their expectations, such as how they will back each other up or self-correct. (For more information on team planning, see Zaccaro, Gualtieri, and Minionis [1995] and Hackman [1987]). The military is well aware of the benefits of preliminary team meetings, and in the simulation model we model a task planning phase that outputs a task assignment plan. The task assignment plan is the

agreement of the task structure and which agents do which tasks, although the commander can change task assignments as events unfold.

Team effectiveness appears to be enhanced when team members provide information before they are requested to do so [e.g., Entin and Serfaty, 1999; Volpe, Cannon-Bowers, and Salas, 1996]. Further, providing information in advance appears to be particularly beneficial in situations characterized by increased workload. For example, Orasanu [1990] found that when more effective aircrews encountered high-workload conditions, copilots (i.e., nonleaders) increased the amount of information that they provided in advance, whereas pilots or captains (i.e., leaders) decreased the number of requests for information. Less effective crews/teams showed the reverse trend. Considering that Johnston and Briggs [1968] theorized that communications are restricted in high-workload conditions, it appears that, in such cases, effective teams contain at least one member who continues to provide information so that others do not need to explicitly request it.

A Shared Mental Model (SMM) is a theoretical concept that provides an explanation of how effective teams are able to utilize this efficient communication strategy whereas ineffective teams are not [Cannon-Bowers, Salas, and Converse, 1993]. In the simulation model, the SMM is implemented by manipulating two probabilities. First is the probability of agreement between team members of task assignments during the mission planning phase. Each team member creates a mental model of which team member should be assigned to each task. Second is the probability of unsolicited information being sent by a team agent, one of the observed behaviors of teams with a strong SMM. For the first probability, a value of 1 implies all team members are in agreement for task assignments, and a value of 1 for the second probability implies that information is pushed to the correct team member when

required in all cases. For example, with a high value for the SMM, then a soldier who observes a change in his/her environment will send the information to the correct recipient without waiting for a request for that information.

Team centralization describes where in the team structure decision authority rests where in centralized teams decision authority rests with a team leader and in decentralized teams decision authority is granted to team members who can all make autonomous decisions for themselves. In centralized structures, the leader tells each team member what actions they should or should not take, or alternatively, waits for team members to make requests for permission to take various actions. These requests are approved, or denied or amended, but, in the end, the single person serving as leader has ultimate authority for what actions are taken. In large organizations, there may be several layers of leadership and orders and/or requests may move up and down several layers of hierarchical management. In decentralized organizational structures, the team members can act on their own, without prior orders or having been granted hierarchical permission. Each individual team member has authority to make their own decisions, and the role of the team leader is to help support those individuals. Most organizational structures are neither totally centralized (where all decisions are made by the top leader) nor totally decentralized (where team members are totally autonomous), but instead lie on a continuum where these are the extreme endpoints.

3.3. Human-Robot Interaction

Human-robot interaction models the workload placed on the human operator while interacting with the robot, and it models the performance of the robot. In the simulation model, robot team members are semiautonomous, which means they cycle between periods of teleoperation by a human team member and autonomous action. We assume the robot interaction scheme is scripted wherein the operator provides the robot with a series of waypoints, and the robot navigates the route defined by the waypoints while avoiding obstacles autonomously. We adopt the model of Crandall et al. [2005] by defining neglect time and interaction time for the robot performance. This model was also used by Balaguer et al. [2008] in their simulation of individual robots. Crandall et al. [2005] proposed and tested a model of robot effectiveness that cycles between autonomous operations during which effectiveness degrades and human teleoperation which restores peak robot effectiveness. The time periods during which the robot acts autonomously and under human intervention are termed the neglect time and interaction time, respectively. Neglect time (NT) is a function of the robot autonomy or ability to act independent of the operator and the task complexity. When the human operator turns his/her attention back to the robot, there is a time lag before peak performance is attained again; this time is called the interaction time. Interaction time is a function of the interface. Interaction time (IT) consists of the time to switch attention from the secondary task to the primary task, time required to reestablish context, time to plan, and time to communicate the plan to the robot. The two metrics can be used to estimate the amount of time the human operator must dedicate to the robot, called the robot attention demand.

Given some performance threshold, the robot attention demand (RAD) is $IT/(IT + NT)$. The proportion of free time available to an operator to dedicate to other activities is simply $1 - RAD$. In the simulation model, once the robot is autonomous or in neglect time, then its performance will degrade as a function of its neglect time. This means that the robot's movement speed decreases and the probability of robot failure increases until the operator returns their attention to the robot, defined as the start of interaction time.

In addition to the autonomous performance of the robot, we also model the reliability of the robot. Robot reliability is measured as Mean Time Between Failure ($MTBF$) and Mean Time To Repair ($MTTR$). The robot availability is $MTBF/(MTBF + MTTR)$. Robot reliability remains a concern with some research indicating high downtimes. Carlson and Murphy [2003] report for their group of robots, used in the field, a MTBF of 8.3 h and in a given hour each mobile robot had a 5.5% probability of failure. A robot failure may not be detected immediately due to operator complacency and lack of situation awareness [Parasuraman, Sheridan, and Wickens, 2000]. We model this as a robot failure detection probability. When a robot fails, then a human agent is tasked to investigate the failure and repair the failure if possible. Collectively, robot neglect time, interaction time, and reliability capture the workload placed on the human team members to interact with the robot.

3.4. Human-Robot Simulation Scenario

The MOUT simulation scenario presented here is one in which a human-robot team must search for a target in an urban area. We assume that the search area has already been partitioned into 25 sectors, and that, using previously gathered intelligence, the team has classified each sector by the probability of containing the target and the danger level. The probability of containing the target is high, medium, or low. The sector's danger level is also high, medium, or low, and the danger level aggregates terrain, hostile activity, and other environmental factors that would affect movement and task performance. The target is assigned to one of the 25 sectors. The mission ends when the target is found.

The team composition is shown in Table II. The commander is the team leader, and creates the initial task assignments, assigns search subteams to sectors, and communicates with the higher-level commander. The search subteams have at least two members on each subteam: either a soldier and a robot or two soldiers.

The simulation logic flow for the mission is divided into two phases: Phase one is the mission planning phase, and phase two is the actual search mission.

3.4.1. Planning Phase

During the planning phase, the team meets, the commander briefs the team and makes the initial task assignments, and the robots are inserted into the first sector. The heuristic algorithm for the assignment of subteams to sectors is shown in Figure 2. The algorithm seeks to assign subteams so as to minimize the time of finding the target by searching in decreasing order of probability of target in sector, and it seeks to minimize the exposure of human subteams to high danger sectors. During

Table II. Team Composition

Team member	Number	Role
Commander	1	Supervises and leads team; tells subteams the sector to search.
Robot	2 or 4	Each robot is assigned to an operator. Robots search sectors.
Soldier-operator	2 or 3	Supervises robot and conducts search when not supervising.
Soldier-mechanic	1 or 2	Fixes robot failures and otherwise searches.
Soldier	2 or 6	Together form a search subteam and also investigate robot finds.

the planning phase, there are no random events that change the process flow. The planning phase ends once the task assignment plan is generated.

Figure 2. Algorithm to assign subteams to sectors.

3.4.2. Search Phase

During the search phase, the following activities are done by each search subteam:

1. The subteam searches their assigned sector. We assume the same search pattern is used in every sector. The search pattern is defined by its waypoints, and the robot or operator moves sequentially from waypoint to waypoint. Given the average speed s of the robot and distance d, the expected search time for the sector can be estimated as, $t = d/s$. The speed is modified by the sector's danger-level, such that in a medium danger-level sector the speed is s, in a low danger-level sector the speed is $1.2s$, and in a high danger-level sector the speed is $0.8s$.
2. For a human subteam, the outcomes of a sector search are either they find the target or they do not find the target. For a robot, upon reaching a waypoint, the outcomes listed in Table III are possible.
3. After searching the entire sector, the search subteam either continues to their next assigned sector or requests assignment from the commander to the next sector. The commander makes the assignment, and the process repeats for the new sector until the target is found.
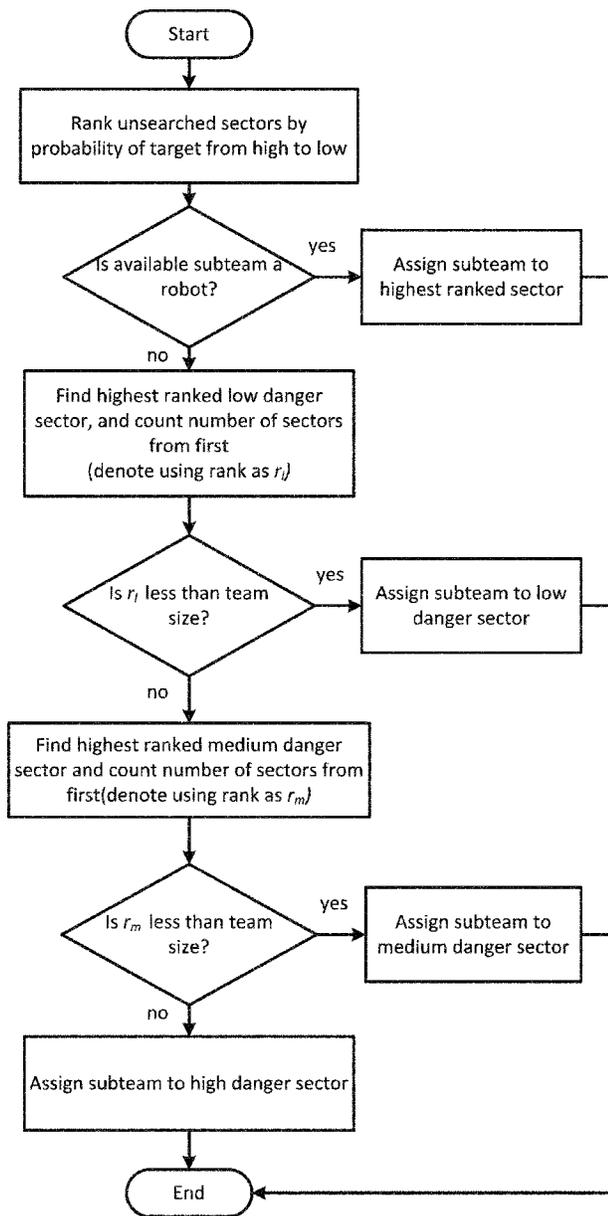
It is possible that due to the uncertain and dynamic events unfolding as the simulation progresses that the initial assignment of subteams to sectors no longer makes sense (e.g., if one subteam completes a sector earlier than expected). In this case the commander needs to make a dynamic task assignment, and uses the same logic as presented in Algorithm 1 on those sectors still not searched.

3.5. Verification and Validation

Ideally, validation would involve comparing the model's response surface for each factor examined with results from actual teams. There are two problems with this approach in the current context. First, simulations of military operations are particularly difficult to validate because combat is singular in nature and expensive to emulate with actual humans [Champagne and Hill, 2007]. Second, for agent-based simulation models, this is an immense task given the number of factors examined. Generally, most models of human behavior face this same difficulty. Validation of human-behavior models is difficult because these models are not mathematically based on physical systems with quantifiable, predictable behaviors [Goerger, McGinnis, and Darkin, 2005]. Human behavior is characterized by the nonlinear response of human cognition, the large number of interdependent variables, and

Table III. Waypoint Outcomes

outcome
The robot reports the target is found. The target identification may be a false positive, so a soldier is always dispatched to investigate and confirm the report.
The robot reports an equipment failure. The operator relays a request to the commander to dispatch a mechanic. The probability of repairing the equipment failure is a model input.
The robot reports a task failure such as being blocked or stuck. Task failures are a function of the danger level and terrain level for the sector. The operator relays the message to the commander. A soldier is dispatched to investigate the report, and take corrective action if possible. The probability of repair or recovering from the task failure is a model input.
The robot does not report after reaching a waypoint. This indicates a robot failure. The probability of robot failure is given as before. The operator will relay the information to the commander, and a soldier will be dispatched to investigate.
The robot reports, "nothing to report" and continues the search to the next waypoint.

the difficulty in obtaining sufficiently diverse data for traditional validation.

Acknowledging these difficulties does not mean validation is not possible. Robinson [1997] argues absolute validity of a model does not exist. The goal is the model should be demonstrated to be valid for the context in which it will be used and for the intended purpose of the model [Thompsen et al., 1999; Burton and Obel, 1995]. Goerger, McGinnis, and Darkin [2005] studied subject matter expert (SME) validation of a MOUT simulation model, and recommend techniques to reduce SME bias and improve consistency. Sterman [2000] presents an extensive list of evaluations that can be done to increase confidence of the simulation model.

In this research, the model purpose is to compare the performance and robustness of different team designs in a MOUT scenario. We validated the model based on the validation procedures described by the aforementioned authors [Sterman, 2000; Goerger, McGinnis, and Darkin, 2005]. Specifically we first verified the simulation code. We checked the boundaries of the model and tested extreme values of the input parameters. Face validity was established by having the model and scenario reviewed by four subject matter experts, two with military operational experience. We did experiments by setting input parameters to extreme values of low and high within their range, and then recorded the output measures and did a paired t-test to determine whether the output measures were statistically different based on the input parameter values. We did experiments when all the input parameters were deterministic. Rojas and Giachetti [2009] describe a replication experiment on the agent-based simulation model comparing the simulated results against actual data for a sailing race committee. The race committee is a team of people who run sailboat races. Their job is to setup the race course, run the race, and then score the results. The race committee team communicates verbally face-to-face, with VHF radios, and with visual signals such as flags and hand signals, which is similar to the communication found in MOUT teams. The race committee team validation accomplishes two things. First, validation in that scenario suggests the way communication is modeled is adequate for the MOUT scenario. Second, it shows that we were able to calibrate the simulation model to capture the essential team behavior. What is lacking from the race committee scenario is the human-robot interaction. As a result of the series of verification and validation tests, we claim the model performs as designed, and the model

has sufficient external validity to examine differences in team performance when changing team input parameters within the ranges studied.

4. EXPERIMENTAL ANALYSIS

To demonstrate how the human-robot team agent-based simulation can be used to evaluate system designs, we conducted a design of experiments. We identified a subset of simulation model factors, shown in Table IV, and partitioned them into those factors we can control and those factors we cannot control, called the noise factors [Sanchez, 2002]. The factors were chosen to illustrate how team design affects team coordination when robots are integral part of the team. Centralization is an important element of team decision-making and coordination. In general, centralization is more efficient in achieving coordination and performs well in stable environments, whereas decentralized teams performance better in task environments that demand speed [Hollenbeck et al., 2011]. Research findings on team size are mixed, but many studies suggest that larger teams suffer coordination problems that detract from performance Stewart [2006]. However, in the MOUT scenario, it would be expected that a larger team could search more sectors and consequently perform better than a smaller team. Research in human-robot interaction identifies issues in remote sensing, loss if situation awareness, and a demand for attention that limits most work to investigate dyadic relationships between human and robot [Murphy, 2004; Woods et al., 2004]. However, future operations will likely require humans to supervise multiple robots, and how this effects overall performance needs to be studied [Dudenhoeffer, Bruemmer, and Davis, 2001]. Robot supervision also depends on the reliability of the robot such that lower reliability translates into greater demands for command and control

Table IV. Team Design Factors

Team design factor	Levels
Controllable factors	
Number of robots	{2, 4}
Team size	{6, 12}
Team centralization	{low, high}
Noise factors	
Danger level	{30%, 70%}
Robot reliability (MTBF)	{6, 10} hours

[Crandall et al., 2005]. Lastly, the final factor selected was the danger level as the aggregate measure of the environment, and was selected because human-robot performance should be assessed in its environment [Burke et al., 2004].

The experimental design was a 2^{5-1} fractional factorial design that has five factors, and each factor was manipulated at two levels as shown in Table IV. We were interested in both main effects and two-factor interactions so that we could examine performance over the noise variables. For this reason we selected an experimental design of resolution *V* [Montgomery, 1999; Kleijnen et al., 2005]. A resolution *V* design organizes the treatments so that we were able to estimate all the main effects and all two-factor interactions cleanly—without worrying about confounding. Therefore, the initial model had 16 terms—the intercept term, the 5 main effects, and the 10 two-factor interactions. The experimental design required 16 runs, and we did four replications for each run, for a total of 64 separate simulation experiments.

In the simulation model, changing the team centralization changes the probability the leader makes the decision and the time an agent will wait for task input requirements. To manipulate the environment danger level, we set it at two levels of 30% and 70%, where the percentage is of the number of sectors that are rated as high danger, which for the 25 sectors was 8 high-danger sectors for low and 18 high-danger sectors for high. Other factors not listed in Table IV were held constant for all experiments. Initially we had planned to randomly place the target in a sector; however, this would introduce unnecessary randomness into the mission which could skew the results for our purposes. Consequently, we decided to put the target in sector 18.

In the experiments we had measures of performance (MOP) and measures of effectiveness (MOE). MOP is defined as observable measures of the team's skills, strategies, and processes. The simulation model allowed us to measure the time spent by each agent doing different activities. We measured the communication time, the waiting time, the rework time, the robot supervision time, and the waiting time for all agents. Additionally, we measured the number of communications, and the number of communication failures. Collectively, these measures indicated how well the team coordinates their activities. MOE is the degree to which the team completed their mission. In the experiments the MOE was how long it took the team to find the target, called the total mission time. Additionally, data were collected on the number of robot failures, which is uncontrollable by the team, but causes additional work.

5. RESULTS

The experimental data is analyzed using the Statistical Package for the Social Sciences (SPSS). We conduct separate ANOVA analysis for each of the MOPs and for the MOE. We are interested in the interaction effects between controllable factors and noise factors. First we describe the main effects and then we describe the interaction effects. The results are shown graphically, with less emphasis on the actual values but greater emphasis on the differences because we are concerned with the relative performance of different team designs.
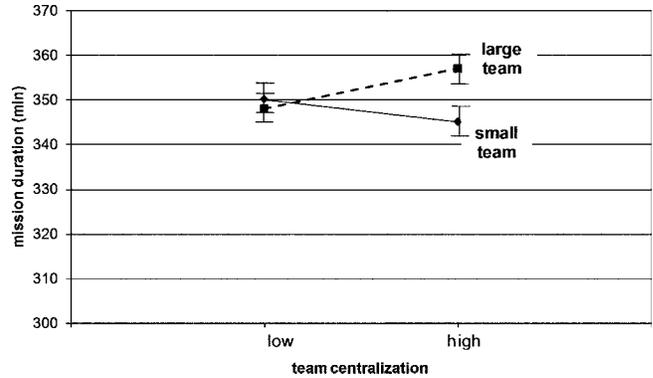

Figure 3. Team size and centralization.

The main effects for both controllable and noise factors are significant at the 0.05 level for the mission duration. A notable interaction is observed between the control factors team size and team centralization [$F(1, 12) = 9.18, p = 0.010$]. Figure 3 shows that, for high centralization, small teams perform significantly better than large teams. However, for low centralization there is no statistical difference in the performance of large versus small teams because the least significant difference bars, set at a 95% confidence interval, overlap. Intuitively, we expect large teams to outperform small teams because they can search more sectors in a given time. To explain the counterintuitive result, we analyze the MOP for waiting time and number of communications. Table V shows that large teams spend more time waiting and make more communications for high centralization than for low. Examination of the simulation output logs shows that, in many instances, with high centralization, the subteams are waiting on the commander who has become a bottleneck. For small teams, the commander does not become a bottleneck because there are fewer subteams to coordinate. Also, in low centralization there are instances of multiple subteams searching the same sector because they did the search on their own initiative; it should be remembered that the subteams have local information only. This helps explain why small teams do slightly worse under low centralization than high centralization.

As mentioned, it is expected that the large teams should outperform the small teams because they can search more sectors simultaneously. It is also expected that the larger number of robots would reduce the total mission time. The experiments confirm both expectations, except as noted above that small teams do better when centralization is high. Figure 4 shows that four robots are more effective than two robots

Table V. Mean Wait Times and Number of Messages

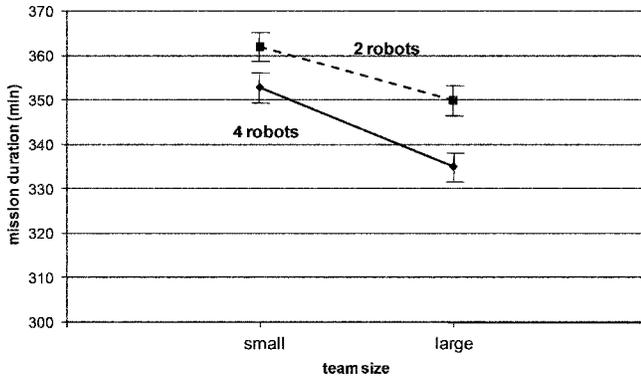|	Small team		Large team	
Centralization	Low	High	Low	High
Waiting time (min)	24.7	28.1	22.5	33.6
Number of messages	75	67	109	89

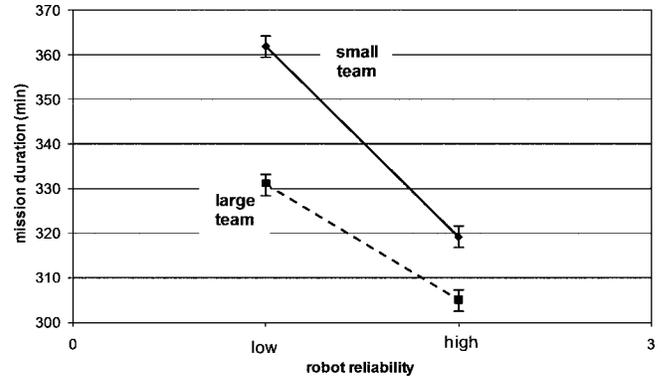**Figure 4.** Number robots and team size.

Figure 6. Team size and robot reliability.

$[F(1, 12) = 44.88, p < 0.001]$, and larger teams are more effective than smaller teams $[F(1, 12) = 70.13, p < 0.001]$. The lines are not parallel, which suggest that smaller teams cannot make as effective use of more robots as large teams. In small teams with four robots, each operator had to control two robots, whereas in large teams two operators controlled a single robot and the third operator controlled two robots. The robot supervision time for small teams with four robots is 78 min, whereas for large teams it is 57 min. Additionally, with a single operator controlling two robots, the robot that is not being controlled would reach its neglect time, and its speed would decrease until the operator could return his attention to the robot. Many human-robot interaction studies investigate the workload of robots on humans; the experiment results suggest that given the modeled values of robot neglect time and interaction time, the operator can effectively control only a single robot. The results are consistent with findings in experiments conducted by Barnes et al. [2006], who found that the addition of a second UGV was at best marginally useful.

Figure 5 shows that the interaction effects between the controllable factor, the number of robots, and the noise factor, danger level, is significant $[F(12, 1) = 28.66, p < 0.001]$.
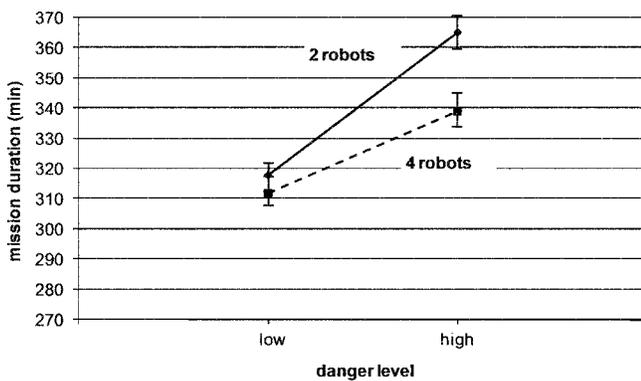
Having four robots is more effective than two robots for high danger levels. At low danger levels the difference between two and four robots is insignificant. The interaction between team centralization and danger level is not significant (p-value = 0.067).

The experiments show, as expected, that high-reliability robots do better than low-reliability robots. However, our assumption is that robot reliability is uncontrollable, so what is of interest is the robustness of the control factors over robot reliability. Large teams did better with low robot reliability than small teams as shown in Figure 6 $[F(12, 1) = 22.35, p < 0.001]$. These results suggest that having extra human resources such as mechanics or soldiers can help mitigate against low robot reliability. Figure 7 shows that having more robots was better, but the difference is small $[F(12, 1) = 9.24, p = 0.010]$.

5.1. Limitations

The agent-based model described has several limitations in representing actual MOUT missions. First, the physical workload on a soldier has been shown to affect performance of cognitive tasks [Perry et al., 2008; Mastroianni, Chuba, and Zupan, 2003]. MOUT is an example of a complex, dynamic
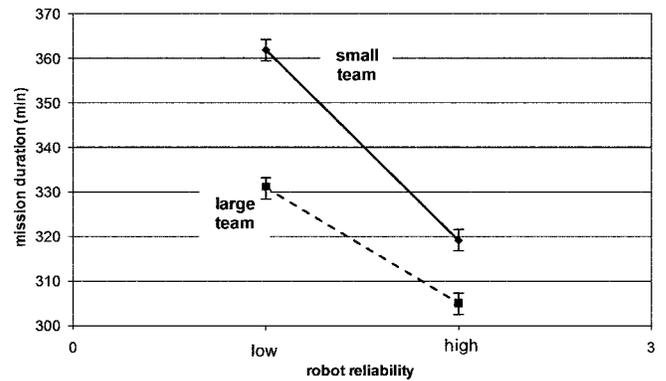

Figure 5. Number robots and danger level.

Figure 7. Number of robots and robot reliability.

task that places both physical as well as cognitive loads on the soldiers. Our model ignores the physical effort required and the negative consequences for situation awareness. Also, we collect all terrain effects into a single danger parameter. The nature of terrain is more complex; terrain affects mobility, field of fire, observability, and cover/concealment. Greater fidelity could be achieved if each of these terrain factors were operationalized individually as factors affecting robot mobility, robot failure, communications, and soldier mobility. Finally, our model does not consider any effects on robot reliability other than the intrinsic reliability of the robot itself. So, for example, operator experience does not contribute to or detract from robot reliability. This assumption is made due to the limited data available; however, it is possible that inexperienced robot operators might contribute to poor robot reliability by how they program waypoints and otherwise supervise the robot. The model did not include the travel time to get to a sector. We did not analyze all the possible factors and interactions in the model.

6. CONCLUSIONS

This paper presents an agent-based simulation model that makes several contributions. First, the model integrates team performance theory with human-robot interaction theory. Second, the model includes complexity due to individual agent-behavior, how they interact, and due to the uncertainty present in the environment. Consequently, we are able to model human-robot teams to understand how they would perform in MOUT.

To analyze the team performance we conducted a 2^{5-1} fractional factorial design of experiments. The experiments confirm some expectations and provide other results that are worth investigating further. The experiments indicate there are limits to the number of robots that a team can effectively manage. The larger teams performed better with four robots whereas the smaller teams had difficulty operating four robots as indicated by the supervision time. The effectiveness of a team increases with size, but it is limited by the amount of coordination possible. Highly centralized, large teams experienced bottlenecks at the commander as was indicated by the waiting time difference between low and high centralization. However, with less centralization, the large teams did better, even when there was some duplication of searching by the subteams. Larger teams also performed more robustly with regard to robot reliability such that when reliability was low, larger teams performed better than smaller teams.

As Champagne and Hill [2007] note, the value of agent-based simulation is often in the ability to generate insight and understanding of why one tactic is preferred to another. The paper demonstrates the feasibility and utility of analyzing human-robot interaction using agent-based simulation. While discussed primarily as a tool for robot team configuration in this paper, the simulation model also provides system designers with valuable insight into the development of individual and group behaviors for the team of robots. We foresee the simulation model being used to analyze system and operational design concepts, to evaluate the many tradeoffs made during system design, and also with modification to the agent descriptions to evaluate proposed changes to robot technology and/or operational requirements.

The conclusions drawn are only valid to the extent the simulation model is consistent with actual systems and scenarios. Future work on model validation is needed and we also need to improve the modeling of the environment, critical to understanding MOUT. Moreover, future work should incorporate not just the cognitive aspects of individual agents but also the physical aspects such as fatigue, reduction in performance under stress, and other physical constraints.

ACKNOWLEDGMENT

REFERENCES

A. Acquisti, M. Sierhuis, W.J. Clancey, and J.M. Bradshaw, Agent-based modeling of collaboration and work practices onboard the international space station, 11th Comput Generated Forces Behavior Representation Conf, Orlando, FL, 2002.zaq;2

B. Balaguer, S. Balakirsky, S. Carpin, M. Lewis, and C. Scrapper, USARSim: A validated simulator for research in robotics and automation, Workshop Robot Simulators Avail Software Sci Appl Future Trends, 2008.zaq;2

S. Balakirsky, E. Messina, and J. Albus, "Architecting a simulation and development environment for multi-robot teams," International Workshop on Multi-Robot System, Kluwer Academic, Dordrecht, The Netherlands, 2002, pp. 161–168.

M.J. Barnes, K.A. Cosenzo, F. Jentsch, J.Y. Chen, and P. McDermott, "Understanding soldier robot teams in virtual environments," Virtual media for military applications, 2006zaq;3, pp. 1–14.

B.S. Blanchard and W.J. Fabrycky, Systems engineering and analysis, 3rd edition, Prentice-Hall, Englewood Cliffs, NJ, 1998.

S.D. Brennen, R.J. Strong, C.J. Ryder, C. Blendell, and J.J. Molloy, Teams, computer modeling, and design, IEEE Trans Syst Man Cybernet Part C 37 (2007), 995–1005.

J.L. Burke, R.R. Murphy, M.D. Coovert, and D.L. Riddle, Moonlight in Miami: Field study of human-robot interaction in the context of an urban search and rescue disaster response training exercise, Hum Comput Interaction 19(1–2) (2004), 85–116.

R.M. Burton and B. Obel, The validity of computational models in organizational science: From model realism to purpose of the model, Comput Math Org Theory 1 (1995), 57–72.

J.A. Cannon-Bowers, E. Salas, and S. Converse, "Individual and group decision making,"zaq;4 Shared mental models in expert team decision making, Lawrence Erlbaum, Hillsdale, NJ, 1993, pp. 221–246.

K.M. Carley, On generating hypotheses using computer simulations, Syst Eng 2(2) (1999), 69–77.

J. Carlson and R.R. Murphy, Reliability analysis of mobile robots, Proc IEEE Int Conf Robot Automat, 2003, pp. 274–281.

L.E. Champagne and R.R. Hill, Agent-model validation based on historical data, Proc 2007 Winter Simul Conf, Washington, DC, 2007, pp. 1223–1231.

K.A. Cosenzo and M. J. Barnes, "Human-robot interaction research for current and future military applications: From the laboratory to the field," Proceedings of SPIE Unmanned Systems Technology XII, G.R. Gerhart, D.W. Gage, and C.M. Shoemaker (Editors)zaq;3, Orlando, FL, 2010.zaq;2

J.W. Crandall, M.A. Goodrich, D.R. Olsen, and C.W. Nielsen, Validating human-robot interaction schemes in multitasking environments, IEEE Trans Syst Man Cybernet Part A 35(2005), 438–449.

S.T. Crino, "Representation of urban operations in military models and simulations," Proceedings of the 2001 Winter Simulation Conference, D.J. Medeiros, B.A. Peters, J.S. Smith and M.W. Rohrer (Editors)zaq;3, Arlington, VA, 2001, pp. 715–720.

S. Davies, It's war—but not as we know it, Eng Technol 4(9) (2009), 40–44.

S. Dong and B. Hu, Multi-agent based simulation of team effectiveness in team's task process: A member-task interaction perspective, Int J Simul Proc Model 4 (2008), 54–68.

D.D. Dudenhoeffer, D.J. Bruemmer, and M.L. Davis, "Modeling and simulation for exploring human-robot team interaction requirements", Proceedings of the 2001 Winter Simulation Conference, D. J. Medeiros, B. A. Peters, J. S. Smith and M. W. Rohrer (Editors)zaq;3, Arlington, VA, 2001, pp. 730–740.

E.E. Entin and D. Serfaty, Adaptive team coordination, Hum Factors 41 (1999), 312–325.

X. Fan and J. Yen, Modeling and simulating human teamwork behaviors using intelligent agents, Phys Life Rev 1 (2004), 173–201.

M. Friedmann, K. Petersen, and O. von Stryk, "RoboCup 2007: Robot Soccer World Cup XI," Tailored real-time simulation for teams of humanoid robots, Springer, Heidelberg, 2008, pp. 425–432.

R.E. Giachetti, R.E. Young, A. Roggatz, and W. Eversheim, A methodology for the reduction of imprecision in the engineering process, Eur J Oper Res 100(2) (1997), 277–292.

S.R. Goerger, M.L. McGinnis, and R.P. Darkin, A validation methodology for human behavior representation models, J Def Mil Syst 2 (2005), 5–17.

R. Grabowski and A. Christiansen, A simplified taxonomy of command and control structures for robotteams,10th Int Comm Contr Res Technol Symp, 2005, pp. 2–27.

J.R. Hackman, Handbook of Organizational Behavior, The design of work teams, Prentice-Hall, Englewood Cliffs, NJ, 1987, pp. 315–342.

R.R. Hill, J.O. Miller, and G.A. McIntyre, "Applications of discrete event simulation modeling to military problems," Proceedings of the 2001 Winter Simulation Conference, D.J. Medeiros, B. A. Peters, J.S. Smith, and M.W. Rohrer (Editors)zaq;3, Arlington, VA, 2001, pp. 780–788.

R.M. Hoeft, J.A. Kochan, and F. Jentsch, Automated systems in the cockpit: Is the autopilot, "George", a team member? Making effective work teams with people, machines, and networks, Lawrence Erlbaum, Mahwah, NJ, 2007.

J.R. Hollenbeck, A.P.J. Ellis, S.E. Humphrey, A.S. Garza, and D.R. Ilgen, Asymmetry in structural adaptation: The differential impact of centralizing versus decentralizing team decision-making structures, Organ Behav Hum Dec 114(1) (2011), 64–74.

T. Horii, Y. Jin, and R. Levitt, Modeling and analyzing cultural influences on project team performance, Comput Math Organ Theory 10 (2005), 305–321.

W.A. Johnston and G.E. Briggs, Team performance as a function of team arrangement and work load, J Appl Psychol 52 (1968), 89–94.

D.L. Kendall and E. Salas, "Measuring team performance: Review of current methods and consideration of future needs," Advances in human performance and cognitive engineering researchzaq;5, Elsevier, London, 2004, pp. 307–326.

H. Kitano, S. Tadokoro, I. Noda, H. Matsubara, T. Takahashi, A. Shinjou, and S. Shimada, Robocup rescue: Search and rescue in large-scale disasters as a domain for autonomous agents research, Proc IEEE Conf Syst Man Cybernet, 1999, pp. 739–743.

J.P.C. Kleijnen, S.M. Sanchez, T.W. Lucas, and T.M. Cioppa, A user's guide to the brave new world of designing simulation experiments, INFORMS J Comput 17(3) (2005), 263–289.

J.C. Kunz, R.E. Levitt, and Y. Jin, The virtual design team: A computational simulation model of project organizations, Commun ACM 41 (1998), 84–92.

M. Loper and B. Presnell, Modeling an emergency operations center with agents, Proc 2005 Winter Simul Conf, Orlando, FL, 2005, pp. 895–903.

D.J. MacKinnon, M. Ramsey, R.E. Levitt, and M.E. Nissen, Hypothesis testing of edge organizations: empirically calibrating an organizational model for experimentation, Proc 12th Int Command Control Res Technol Symp, Newport, RI, 2007.zaq;2

M.W. Maier, Architecting principles for systems-of-systems, Syst Eng 1(4) (1998), 267–284.

J Martínez-Miranda and J Pavón, "Modeling trust into an agent-based simulation tool to support work teams formation," 7th International Conference on Practical Applications of Agents and Multi-Agent Systems (PAAMS 2009), Springer, Heidelberg, 2009, pp. 80–89.

G.R. Mastroianni, D.M. Chuba, and M.O. Zupan, Self-pacing and cognitive performance while walking, Appl Ergon 34 (2003), 131–139.

M.J. Mataric, G.S. Sukhatme, and E.H. Ostergaard, Multi-robot task allocation in uncertain environments, Auton Robot 14 (2003), 255–263.

D. Montgomery, Design and analysis of experiments, Wiley, New York, 1999.

R.R. Murphy, Human-robot interaction in rescue robotics, IEEE Trans Syst Man Cybernet Part C 34 (2004), 138–154.

National Research Council (NRC), Defense modeling, simulation, and analysis: Meeting the challenge, The National Academy Press, Washington, DC, 2006.

J.M. Orasanuu, Shared mental models and crew decision making, Cognitive Science Laboratory Report No. 46, Princeton University Press, Princeton NJ, 1990.

E. Ortiz, D. Barber, J. Stevens, and N. Finkelstein, Simulation to assess an unmanned system's effect on team performance, Interservice Indust Train Simul Ed Conf (I/ITSEC), Orlando, FL, 2009.zaq;2

R. Parasuraman, M. Barnes, and K. Cosenzo, Adaptive automation for human-robot teaming in future command and control systems, Int Command Control J 1(2) (2007), 43–68.

R. Parasuraman, T.B. Sheridan, and C.D. Wickens, A model for types and levels of human interaction with automation, IEEE Trans Syst Man Cybernet Part A 30(2000), 286–297.

C.R. Paris, E. Salas, and J.A. Cannon-Bowers, Teamwork in multiperson systems: A review and analysis, Ergonomics 43 (2000), 1052–1075.

C.M. Perry, M.A. Sheik-Nainar, N. Segall, R. Ma, and D.B Kaber, Effects of physical workload on cognitive task performance and situation awareness, Theor Issues Ergon Sci 9 (2008), 95–113.

S. Robinson, Simulation model verification and validation: Increasing the user's confidence, Proc 1997 Winter Simul Conf, Atlanta, GA, 1997, pp. 53–59.

J.A. Rojas and R.E. Giachetti, An agent-based simulation model to analyze team performance on jobs with a stochastic structure, 1st Int Conf Adv Syst Simul (SIMUL 2009), 2009.zaq;2

W.B. Rouse and K.R. Boff, Organizational simulation, Wiley, New York, 2005.

A.P. Sage and C.D. Cuppen, On the systems engineering and management of systems of systems and federations of systems. Inform-Knowl-Syst Management 2(4) (2001), 325–345.

S.M. Sanchez, Robust design: Seeking the best of all possible worlds, Proc 2000 Winter Simul Conf, Orlando, FL, 2002, pp. 69–76.

J.D. Sterman, Business dynamics, McGraw-Hill, Boston, 2000.

G.L. Stewart, A meta-analytic review of relationships between team design features and team performance. J Management 32 (2006), 29–55.

R.J. Stout, J.A. Cannon-Bowers, and E. Salas, Planning, shared mental models, and coordinated performance: An empirical link is established, Hum Factors 41 (1999), 61–71.

E.K. Sundstrom, K.P. DeMeuse, and D. Futrell, Work teams: Applications and effectiveness, Am Psychol 42 (1990), 120–133.

J. Thompsen, R.E. Levitt, J.C. Kunz, C.I.. Nass, and D.B. Fridsma, A trajectory for validating computational emulation models of organizations, Comput Math Org Theory 5 (1999), 385–401.

J. Thomsen, R.E. Levitt, and C.I. Nass, The virtual team alliance: Extending Galbraith's information processing model to account for goal incongruency, Comput Math Org Theory 10(4) (2005), 349–372.

C.E. Volpe, J.A. Cannon-Bowers, and E. Salas, The impact of cross-training on team functioning: An empirical investigation, Hum Factors 38 (1996), 87–100.

J. Wang and M. Lewis, Human control for cooperating robot teams, Proc ACM/IEEE Int Conf Human-Robot Interaction, Arlington, VA, 2007, pp. 9–16.

R.F.A. Woodaman, Agent-based simulation of military operations other than war, Ph.D. thesis, Naval Postgraduate School, Monterey, CA, 2000.

D.D. Woods, J. Tittle, M. Feil, and A. Roesler, Envisioning human-robot coordination in future operations, IEEE Trans Syst Man Cybernet Part C 34(2004), 210–218.

J. Yen and X. Fan, Agents with shared mental models for enhancing team decision makings, Decision Support Syst 41 (2006), 634–653.

J. Yen, J. Yin, T.R. Ioerger, M.S. Miller, D. Xu, and R.A. Volz, CAST: Collaborative agents for simulating teamwork, Proc 17th Int Conf AI, 2001, pp. 1135–1142.

L. Yilmaz and J. Philips, Team-RUP: Agent-based simulation of team behavior in software development organizations, Int J Simul Process Model 3 (2007), 170–179.

S. Young and A. Kott, Control of small robot squads in complex adversarial environments: A review, Proc 14th Int Commun Control Res Technol Symp (ICCRTS), Washington, DC, 2009.zaq;2

S.J. Zaccaro, J. Gualtieri, and D Minionis, Task cohesion as a facilitator of team decision making under temporal urgency, Mil Psychol 7 (1995), 77–93.

Ronald E. Giachetti, Ph.D., is a Professor of Systems Engineering at the Naval Postgraduate School (NPS) in Monterey, CA. He teaches and conducts research in the design of enterprise systems, systems modeling, and system architecture. He has published over 50 technical articles on these topics including a textbook titled *Design of Enterprise Systems: Theory, Methods, and Architecture*.zaq;6 He is a member of the DoDAF DM2 Working Group, a member of INCOSE, and associate editor of the *Journal of Enterprise Transformation*. Prior to joining NPS, he was an Associate Professor of Industrial and Systems Engineering at Florida International University in Miami, FL. At FIU he developed and led the MS in Information Systems program and actively taught in external programs in Jamaica, Mexico, Colombia, and Peru. He has conducted $1M in externally funded research for the NSF, US Army, US Air Force, Royal Caribbean Cruise Lines, Carnival Cruise Lines, and other Florida companies. He was a National Research Council postdoctoral researcher in the Manufacturing Systems Integration Division at the National Institute of Standards and Technology (NIST) in Gaithersburg, MD. He earned a Ph.D. in Industrial Engineering from North Carolina State University in Raleigh, NC in 1996; an M.S. in Manufacturing Engineering from Polytechnic University in Brooklyn, NY in 1993; and a B.S. in Mechanical Engineering from Rensselaer Polytechnic Institute in Troy, NY in 1990.

Veronica Marcelli is an engineer at General Electric Healthcare. She earned a B.S. in Industrial & Systems Engineering from Florida International University in Miami, FL.

José Cifuentes is a supplier development leader at Rolls-Royce North America. He earned a B.S. in Industrial & Systems Engineering from Florida International University in Miami, FL

José A. Rojas is an assistant professor in the Department of Industrial and Management Engineering at the Universidad del Turabo. He holds degrees from the University of Puerto Rico (B.S. IE, 1996), University of Michigan (M.S. IOE, 1997), and Florida International University (Ph.D. I&SE, 2010). His research interests include social and organizational simulation, agent-based modeling and simulation, and engineering education.

<enote>AQ1: Reference for this quotation?
<enote>AQ2: Please give page range
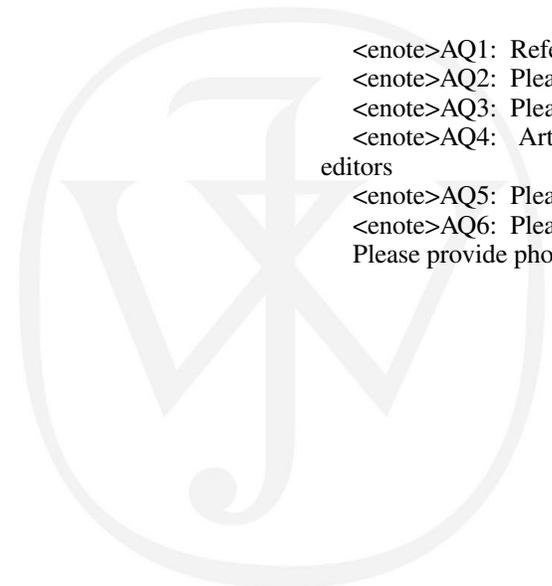<enote>AQ3: Please give publisher and its city
<enote>AQ4: Article in edited book? If so, please list editors
<enote>AQ5: Please list editors
<enote>AQ6: Please give publisher and year
Please provide photo of third author