A PREDICTIVE MODEL FOR HUMAN-UNMANNED VEHICLE SYSTEMS

FINAL REPORT

J. W. CRANDALL
M. L. CUMMINGS

MASSACHUSETTS INSTITUTE OF TECHNOLOGY*

PREPARED FOR MIT LINCOLN LABORATORY

HAL2008–05

JUNE 2008

http://halab.mit.edu email: halab@mit.edu

*MIT Department of Aeronautics and Astronautics, Cambridge, MA 02139

Abstract

Advances in automation are making it possible for a single operator to control multiple unmanned vehicles (UVs). This capability is desirable in order to reduce the operational costs of human-UV systems (HUVS), extend human capabilities, and improve system effectiveness. However, the high complexity of these systems introduces many significant challenges to system designers. To help understand and overcome these challenges, high-fidelity computational models of the HUVS must be developed. These models should have two capabilities. First, they must be able to describe the behavior of the various entities in the team, including both the human operator and the UVs in the team. Second, these models must have the ability to predict how changes in the HUVS and its mission will alter the performance characteristics of the system. In this report, we describe our work toward developing such a model. Via user studies, we show that our model has the ability to describe the behavior of a HUVS consisting of a single human operator and multiple independent UVs with homogeneous capabilities. We also evaluate the model's ability to predict how changes in the team size, the human-UV interface, the UV's autonomy levels, and operator strategies affect the system's performance.

1 Introduction

For the foreseeable future, unmanned vehicle (UV) technologies will require the assistance of human operators to perform important and challenging tasks. Current UV platforms require multiple operators to control a single UV. However, this need for significant manpower is expensive and often ineffective. As a result, it is desirable to invert this ratio so that a few operators can effectively control many UVs in order to (a) reduce costs, (b) extend human capabilities, and (c) improve human-UV system (HUVS) effectiveness. To achieve this goal, additional research must address many issues related to the human operator, the UVs, and the interactions between them.

For HUVSs consisting of a single operator and multiple UVs to be effective, many questions must be answered, including: How many UVs should there be in the team? What human-UV interaction methodologies are appropriate for the given HUVS and mission? What autonomy levels should the UVs in the team employ, and when should changes in these autonomy levels be made? What aspects of a system should be modified to increase the system's overall effectiveness? The ability to answer these questions will facilitate the development of technologies that can effectively support UV operators in dynamic real-world situations.

High-fidelity computational models of HUVSs are needed to help answer these questions. To be successful, these models should satisfy two capabilities. First, they should adequately describe the behavior of the complete system. Second, these models should have the ability to accurately predict the behavior and performance of the system in conditions that have not previously been observed.

A model with both descriptive and predictive abilities has many important applications. For example, such a model can improve the design and implementation processes of HUVSs. As in any systems engineering process, test and evaluation plays a critical role in fielding new technologies. In systems with significant human-automation interaction, testing with representative users is expensive and time consuming. Thus, the development of a high-fidelity model of a HUVS with both descriptive and predictive capabilities will streamline the test and evaluation cycle since it can both help diagnose the cause of previous system failures and inefficiencies, and indicate how potential design modifications will affect the behavior and performance of the system.

A model with both descriptive and predictive abilities can also, among other things, be used to determine successful combinations of UVs within the team (team composition). The composition of futuristic human-UV teams is likely to dynamically change both in number and type due to changing resource availability and mission assignment. High-fidelity models can be used to ensure that variations in team composition will not cause system performance to drop below acceptable levels. Furthermore, these models can potentially be used to suggest which autonomy levels UVs should employ given the team composition.

This report describes the development and evaluation of a new modeling methodology. This research was executed in three separate three phases. In the first phase, we decomposed HUVSs consisting of a single operator and multiple UVs in order to identify a relevant set of metrics classes. These metric classes form

the theoretical basis upon which our modeling methodology is built. In the second phase, we developed the modeling technology itself. In this work, we showed that models built using our modeling methodology have the ability to describe the behavior of the system and predict how changes in the human-UV interface and in the UVs' autonomy levels alter the system's effectiveness. Finally, in the third phase, we used the modeling methodology to investigate how variations in operator attention allocation strategies affect system effectiveness.

2 Phase 1 – Metric Classes for Human-UV Systems

An understanding of which system processes govern the success of a HUVS is essential to the development of a model capable of answer the questions listed in the introduction. Such an understanding requires the designer to identify a set of metrics that, when combined, can evaluate each of the essential aspects of the system. These mission and system specific metrics are drawn from a set of mission and system generic metric classes. Loosely, a metric class consists of the metrics that measure a particular aspect of a system. Our initial work on metric classes for HUVSs is published in the papers included in Appendices A and B of this report [3, 4]. We briefly summarize this research in this section and refer the reader to the appendices for an in-depth study of the subject.

To ensure that a set of metrics can sufficiently model a HUVS, it should have at least three properties:

1. A set of metrics should identify the limits of both the human operator and the UVs in the team.
2. A set of metrics should contain the key performance parameters of the system.
3. A set of metrics should have predictive power.

Toward this end, we identified a potential set of three metric classes for HUVSs consisting of a single human operator and multiple UVs. These classes were (1) *interaction efficiency*, a set of metrics that measures the effects of human-UV interactions on UV behavior, (2) *neglect efficiency*, a set of metrics that assesses how a UV's behavior changes in the absence of interactions with the operator, and (3) *attention allocation efficiency*, a set of metrics measuring how well the operator allocates her attention among the UVs in the team.

After identifying this set of metric classes, we evaluated various sets of metrics drawn from these metric classes with respect to the three properties just mentioned. To do this, we conducted a user study using RESCU (*R*esearch *E*nvironment for *S*upervisory *C*ontrol of *U*nmanned-Vehicles), a test-bed in which a user controls a team of simulated UVs in a search and rescue mission. Using observational data from the study, we evaluated various sets of metrics with respect to the three desirable properties. We showed that selecting metrics from only two of the three metric classes does not result in a model that satisfies the three properties, particularly that of predictive power. However, we showed that selecting a set of stochastic metrics from each of the metric classes can provide a model with reasonably good predictive power. Specifically, we showed that a set of stochastic metrics can, when combined, give reasonably good predictions of how the system's effectiveness changes as UVs are are added or removed from the team.

This research inspired follow-on work that further expands and develops the concepts and uses of metric classes for HUVSs. This research is published in Pina *et al.* [11] and Cummings *et al.* [6].

3 Phase 2 – A Computational Model for Human-UV Systems

Drawing from the set of metric classes identified in Phase 1, we developed a computational modeling methodology for HUVSs consisting of a single human and multiple independent UVs. In this modeling methodology, four stochastic structures, drawn from the set of metric classes identified in the previous phase, are estimated via observational data. These models describe the behavior of the individual entities in the team, and, when combined together, form a complete description of the system that can be used to predict how changes in

the system will alter its overall effectiveness. We briefly summarize this work in this section, and refer the reader to Appendix C [2] for a more detailed description of the research.

We model a HUVS using four stochastic structures. The first two stochastic structures model the behavior of the individual UVs in the team. These metric structures, drawn from the interaction and neglect efficiency metric classes, are stochastic processes that describe how a UV's state changes in the presence and absence of interactions with the human operator. The second two stochastic structures, drawn from the attention allocation efficiency metric class, model how the human operator allocates her attention among the various UVs in the team.

In our approach, each of the stochastic structures is formed from data obtained from observing the HUVS in a particular condition. Using a discrete event simulation, these models can then be used to describe the behavior of the team in the observed condition. Such a description is useful for identifying the strengths and weaknesses of the HUVS, and for identifying how the system could be modified to improve its performance. Second, the discrete event simulation can be used to predict system efficiency in other unobserved conditions, such as when the human-UV interface changes or when UV autonomy is altered.

To validate the descriptive and predictive ability of our modeling methodology, we conducted a second user study using RESCU. Sixty-four subjects participated in the study. Via data collected in a single condition of the study, we constructed a model of the HUVS. We showed that this model accurately describes the behavior of the system. Furthermore, we showed that this model makes reasonably good predictions of the system's effectiveness in other conditions. Specifically, the model effectively predicts how changes in the human-UV interface and the UVs' autonomy levels affect the system's overall effectiveness. These results represent a significant step toward developing high-fidelity models of HUVSs necessary to the development of flexible and robust HUVSs.

4 Phase 3 – Operator Selection Strategies in Human-UV Systems

In Phase 2, we analyzed the model's ability to predict the effects of changes in the human-UV interface and the UVs' autonomy levels. These particular system alterations primarily affected aspects of the system corresponding to the interaction and neglect efficiency metric classes. In the third and final phase of this research project, we focus on aspects of the system related to the third metric class, that of attention allocation efficiency. Unlike the previous two phases, we describe the main portion of this research in its entirety in the body of this paper since this research is yet to be submitted for publication.

In complex systems such as those in which a human supervises multiple UVs, the human operator must oversee a large number of tasks simultaneously. While the UVs often perform these tasks autonomously for long periods of time, they eventually require human intervention and assistance. In time-critical operations, human-UV interactions must be timely to avoid catastrophic failures and to ensure that the system continues to operate efficiently. Thus, human attention must be carefully and effectively allocated among the UVs in the team.

In the user studies described in the previous phases of this research, determining which UV to service at any given time was left to the judgment of the human operator. While a simple visual alarming system was provided to alert the operator of the UVs' needs, the operator was not told which UV to service, nor was a prioritization scheme provided. Rather, operators were left to create their own priorities, which they developed, and often verbalized, over the course of several practice missions. Different participants developed different prioritization schemes.

That the participants developed different selection strategies raises an interesting set of questions. Given the status of a HUVS and the capabilities of the UVs in the system, which UV should the human attend to? Are the selection strategies learned and employed by operators effective, or should other selection strategies be used? If so, how should the system be designed to improve selection strategies?

Computational models of HUVSs are a critical component in answering these questions. A computational tool capable of predicting the effects of operator selection strategies has a number of useful applications. First, such a tool can help identify when and how operators' selection strategies should be altered to substantially improve the system's effectiveness. For example, the tool can be used to identify an effective selection

strategy. This selection strategy can then be compared to the selection strategy currently being followed by the system to determine which decisions, if any, lead to reduced system effectiveness. System designers can then use this information to determine where to focus system improvements.

Second, the computational tool can be used in decision aids for operator attention allocation. Once the tool identifies an effective selection strategy, this selection strategy can be used to help determine which UV should be serviced at any give time. If done effectively, this could potentially reduce operator workload while increasing system effectiveness, though many human factors concerns must be addressed, including trust [8], automation bias [5], and situation awareness [7].

Analysis of attention allocation strategies in complex systems has been addressed at an abstract level by Sheridan and Tulga [13] and Neth *et al.* [10]. However, a generalized computational method for determining the effects of operator selection strategies in HUVSs remains an open question. In this phase, we analyze the ability of the modeling methodology described in Phase 2 to answer these questions. As in the previous two phases, we validate the the predictions made by the model via user studies performed in RESCU (Appendix B), a test-bed in which users control multiple simulated UVs in a search and rescue mission. For simplicity, we focus primarily on the base instantiation of RESCU (referred to as noDS in Appendix C), with a team size of eight UVs.

Specifically, in Section 4.1, we analyze operators' observed selection strategies in a previous user study. In Section 4.2, we use the model to estimate what operators *should* have done to "maximize" system effectiveness. To determine the correctness of these predictions and to determine how these predictions can be used to help users more effectively allocate their attention among the UVs in the team, we conducted another user study using RESCU. This user study is described in Section 4.3. We report and discuss the results of this study in Section 4.4. In Section 4.5, we summarize and discuss our conclusions, and outline areas of future work.

4.1 Observed Operator Selection Strategies

In all conditions of RESCU studied in the previous two phases, UV selection was the responsibility of the human operator. A visual alerting system was provided which indicated to the user when a UV needed to be assigned a new task, when a UV needed assistance picking up an object, and when the scenario was about to end. In this subsection, we study the users' selection strategies under these conditions.

In RESCU, system effectiveness is measured with respect to number of objects collected and number of UVs lost, which occurs when UVs are left in the maze when time expires. System effectiveness as observed in the noDS condition of the Phase 2 user study is shown in Figure 1. The figure shows that the average number of objects collected in the study peaked at about six UVs. Furthermore, the figure shows that the number of UVs lost also increased with team size. Given the objective function

$$Score = numObjectsCollected - UVsLost \tag{1}$$

which users were asked to maximize, these results indicate that system effectiveness was highest when the team size was between four and six UVs. Adding additional UVs to the team after six UVs did not increase system effectiveness, and, in fact, appears to decrease it. These results show that the operators were, on average, unable to effectively manage more than six UVs.

We note that, at least in RESCU, a decrease in system effectiveness with increasing numbers of UVs need not occur. If the human operator simply chooses not to use excess numbers of UVs, we would expect system effectiveness to plateau after six UVs. However, while some users did follow this strategy, many did not. Thus, the decrease in system effectiveness with large UV teams can be traced, at least in part, to the operators' selection strategies.

Further, consider Figure 2, which shows the average number of interactions per minute for team sizes of six and eight UVs. In the figure, interactions are categorized into three groups: goal assignment, payload operations, and replanning/re-routing. Goal assignment refers to interactions in which the human operator sent UVs into the maze. Payload operations refer to interactions in which the user serviced a UV that was

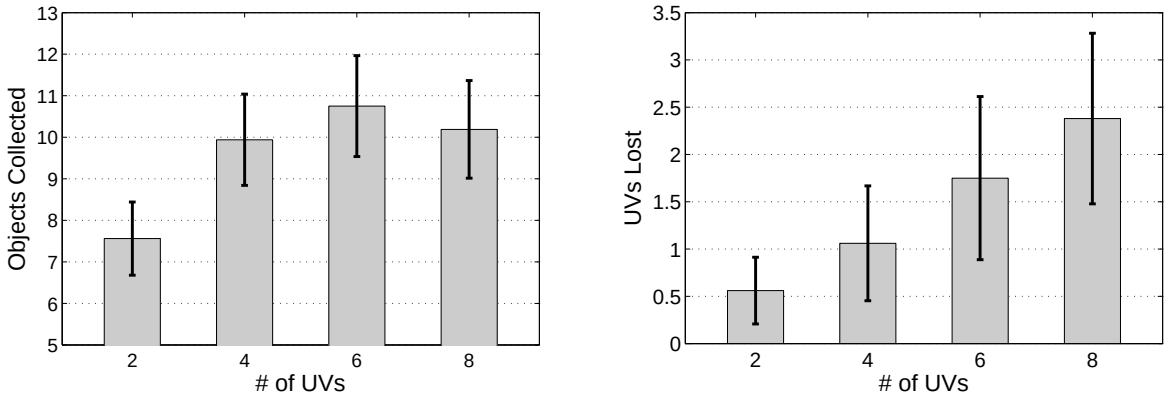


Figure 1: Average number of objects collected (a) and UVs lost (b) in the noDS condition.

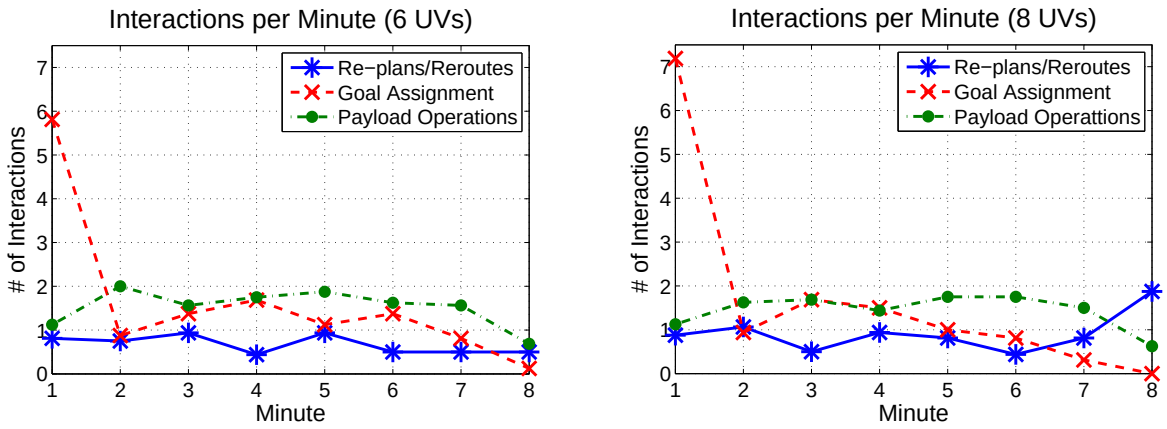


Figure 2: Average number of interactions of each type per minute for (left) six- and (right) eight-UV teams.

ready to pick up an object. Replanning/re-routing refers to all other interactions, which are interactions in which the operator modified the UV's intended path or changed its task.

The strategy profiles, shown in the form of interaction counts per minute in Figure 2, are fairly similar for both in six- and eight-UV teams. However, there are two revealing differences. First, in eight-UV teams, the average user did indeed send more than six UVs into the maze in the first minute. While this is not surprising since there were eight-UVs to send into the maze rather than just six, this result illustrates that the average user was often unable to identify that he could not, on average, effectively control eight UVs.

The second main difference between the strategy profiles shown in Figure 2 occurs in the last minute, where there is a sharp increase in number of interactions involving re-routing and re-planning in the eight-UV condition. In this condition, operators tended to over-estimate the number of tasks they could complete in a given amount of time, as they often sent too many UVs into the maze toward the end of a mission. In the eighth minute, they often realized their mistake, and tried to correct it by removing the UVs from the maze. This resulted in the large increase in interactions involving re-routing and replanning in the eighth minute. However, this reactive behavior often came too late, resulting in lost UVs.

In short, while users often used effective selection strategies, they also made time-critical errors that reduced the system's effectiveness. The visual alerting system provided in the noDS condition was insufficient.

This highlights the need to develop decision aids to help operators better allocate their attention among the UVs in the team. In remainder of this phase, we study how the model can potentially be used in this process.

4.2 Computing an Optimal Selection Strategy

The model constructed in Phase 2 and described in Appendix C can be used to predict how an arbitrary operator selection strategy will affect the system's effectiveness. This capability can be used to estimate an optimal selection strategy with respect to some objective function, which, in the case of RESCU, is the function specified in (1). In this section, we describe this computational methodology. We then discuss the optimal selection strategy computed by the model, and compare it to the average selection strategy employed by human operators in RESCU. In the next subsection, we describe how this selection strategy can be used to create decision support tools designed to improve attention allocation efficiency.

4.2.1 Computing the Effects of Operator Selection Strategies

As in Phase 2, let S be the set of all joint states and let T be the set of mission times. Also, let $S \times T \rightarrow \Sigma$ be the set of possible system states. As before, let $\mathcal{SS}(\sigma)$ be the selection policy used in system state $\sigma \in \Sigma$, which is a probability distribution describing how likely the operator is to service each UV given σ. Then, formally, a *selection strategy* is a specification of $\mathcal{SS}(\sigma)$ for all $\sigma \in \Sigma$. Let Ω denote the set of all possible selection strategies.

We desire to estimate how an arbitrary selection strategy affects system effectiveness given the other behavioral characteristics of the system. Formally, as in Phase 2, let $\mathcal{ST}(\sigma)$ be a probability distribution that describes operator switching time, and let $\mathcal{II}(\sigma)$ and $\mathcal{NI}(\sigma)$ describe the behavior of an individual UV in the presence and absence of human interactions, respectively. Then, let $J(\omega|\mathcal{ST},\mathcal{II},\mathcal{NI})$ denote the expected system effectiveness for using the selection strategy $\omega \in \Omega$ given $\mathcal{ST}$, $\mathcal{II}$, and $\mathcal{NI}$. For brevity, we denote $J(\omega|\mathcal{ST},\mathcal{II},\mathcal{NI})$ as $J(\omega)$.

The model described in Phase 2 provides a technique for computing $\hat{J}(\omega)$, an estimate of $J(\omega)$, for each $\omega \in \Omega$. Let $\mathcal{ST}(\sigma)$, $\mathcal{II}(\sigma)$, and $\mathcal{NI}(\sigma)$ be constructed as in Section III of Appendix C, and let ω define $\mathcal{SS}(\sigma)$ for all $\sigma \in \Sigma$. Then, $J(\omega)$ can be calculated using the discrete event simulation outlined in Algorithm 1 of Appendix C.

4.2.2 Computing an Optimal Operator Selection Strategy

We wish to identify an operator selection strategy $\omega^* \in \Omega$ such that $\hat{J}(\omega^*) \geq \hat{J}(\omega) - \varepsilon$, for all $\omega \in \Omega$ and some small $\varepsilon \geq 0$. We call ω^* an optimal selection strategy if $\hat{J}(\omega) \approx J(\omega)$ for all $\omega \in \Omega$. Since we have no guarantees about the form of the function $\hat{J}(\cdot)$ and since there are an infinite number of operator selection strategies, computing ω^* directly is difficult. However, we can estimate ω^* with respect to a reasonable subset of operator selection strategies, which we denote $\hat{\Omega}$.

The subset of selection strategies we consider for RESCU is derived using three simplifications. In the first simplification, we reduce the time dimension of system state by discretizing mission time T. In practice, a human operator can constantly alter his selection policy over time, meaning that selection policies can change an infinite number of times over the course of a scenario. To avoid this, we allow a user's selection policy to change only at discrete points in time, which effectively divides mission time into a finite set of time periods. In RESCU, we divide mission time into eight discrete time periods, one corresponding to each minute of the eight-minute mission. We chose this coarse discretization for computational purposes. A finer discretization would provide a richer set of selection strategies, but would create a larger state space.

In the second simplification, we reduce the number of possible probability distributions that $\mathcal{SS}(\sigma)$ can take on. While, theoretically, $\mathcal{SS}(\sigma)$ can take on an infinite number of forms, we consider only those probability distributions that place all weight on a particular UV state. These probability distributions can be expressed with a preference ordering over UV states. A preference ordering specifies the order that the operator prefers to (and does) service the UVs in the team.

Minute	Preference Ordering
1^{st}	$s^1 > s^2 > s^3$
2^{nd}	$s^2 > s^1 > s^3$
3^{rd}	$s^2 > s^1 > s^3$
4^{th}	$s^2 > s^3 > s^1$
5^{th}	$s^2 > s^3 > s^1$
6^{th}	$s^3 > s^2 > s^1$
7^{th}	$s^3 > s^2 > s^1$
8^{th}	$s^3 > s^2 > s^1$

Table 1: Example of a simplified selection strategy, expressed as a series of eight preferences orderings. $X > Y$ denotes that state X is preferred to state Y.

These first two simplifications reduce a selection strategy to a sequence of preference orderings over UV states, one preference ordering (second simplification) for each discrete time period (first simplification). Thus, in RESCU, a sequence of eight preference orderings over UV states specifies a selection strategy. As an example, consider Table 1, which shows a simplified selection strategy for a three UV-state system. In the first minute, this strategies specifies that the operator will service UVs in state s^1 first. If no UV is in state s^1, then she services a UV in state s^2, etc. In the second minute of the mission, the operator changes strategies so that she first services UVs in state s^2. Once there are no UVs in state s^2, she services UVs in state s^1, and so on.

While these simplifications reduce the set of selection strategies to a finite set, the size of this set is still, if we use the set of 21 UV states used in Phase 2 (Figure 3 in Appendix C), on the order of 10^{152} in RESCU. Thus, the third simplification is to reduce the number of UV states, which we achieve by grouping the 21 UV states into five categories: (1) assignment states (A), states in which the UV is outside of or nearly outside of the maze, (2) payload states (P), which are states in which the UV is ready or nearly ready to pick up on object, (3) sitting states (S), which are states in which the UV is idle in the maze, but not on an object, (4) replanning/re-routing states (R), which includes states in which the UV is navigating the maze, but is either not taking the likely shortest path to the goal or could benefit from being reassigned to a different task, and (5) good states (G), states in which the UV is effectively and efficiently moving toward its goal destination.

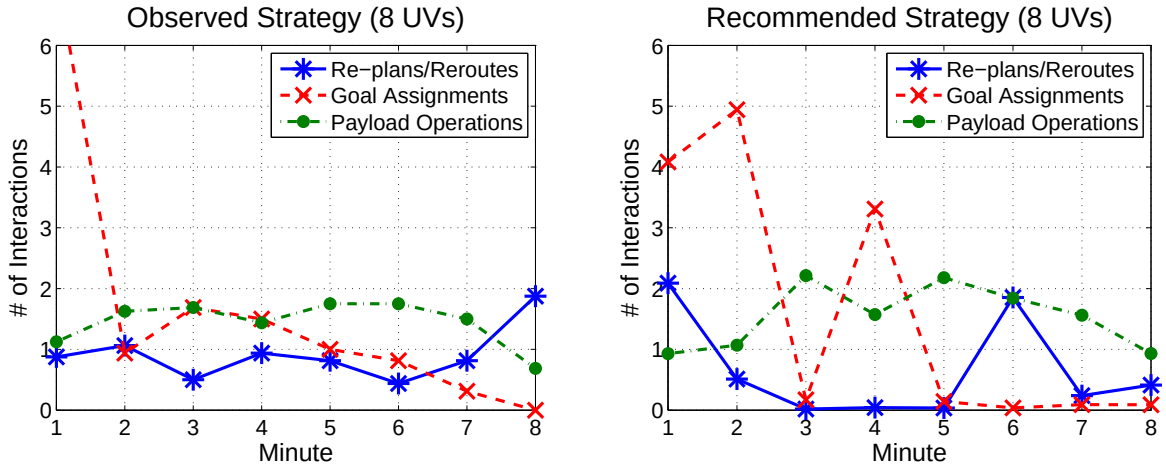
If we assume that UVs in states corresponding to group G always have the lowest priority, then there are 24 possible preference orderings over UV state groupings. This means that these simplifications reduce the number of possible selection strategies to 24^8. While this is still a large set of selection strategies, it is sufficiently small that we can effectively search it using an optimization algorithm. We used a genetic algorithm to find a near optimal selection strategy with respect to the reduced set of selection strategies $\hat{\Omega}$.

We hasten to note that a selection strategy chosen using this technique will only be as good as the assumptions upon which the model is built. We emphasize two of these assumptions. First, we assume that $\hat{\Omega}$ adequately represents Ω. If $\hat{\Omega}$ does not contain an optimal selection strategy with respect to Ω, the model certainly will not identify an optimal selection strategy. However, the model can still identify a successful selection strategy from $\hat{\Omega}$ that could offer a substantial improvement over the selection strategy employed by users in the study. For this reason, we refer to the selection strategy computed by the genetic algorithm as the *recommended* rather than the *optimal* selection strategy throughout the rest of this report.

A second assumption made by our modeling methodology is that $\hat{J}(\omega) \approx J(\omega)$ for all $\omega \in \Omega$. This assumption implies that the structures $\mathcal{II}$, $\mathcal{NI}$, and $\mathcal{ST}$ accurately model the components of the system they represent. However, it is possible that altering a user's selection strategy will cause changes in human-UV interactions, which would lead to changes in interaction impact $\mathcal{II}$ and, possibly, the other structures. These issues must be carefully taken into account as we analyze the ability of the model to predict the effects of selection strategies on system behavior.

Minute	Preference Ordering
1^{st}	$R > A > S > P > G$
2^{nd}	$S > A > P > R > G$
3^{rd}	$P > S > R > A > G$
4^{th}	$A > S > P > R > G$
5^{th}	$S > P > A > R > G$
6^{th}	$R > S > P > A > G$
7^{th}	$S > R > P > A > G$
8^{th}	$S > P > R > A > G$

Table 2: Recommended selection strategy for eight-UV teams, expressed as a set of eight preference orderings. $X > Y$ denotes that state X is preferred to state Y.



(a) (b)

Figure 3: (a) Average number of interactions observed in the noDS condition of the user study. (b) Predicted number of interactions per minute if the recommended selection strategy is followed.

4.2.3 Recommended Selection Strategy

After constructing a model of $\mathcal{II}$, $\mathcal{NI}$, and $\mathcal{SS}$ using data from the noDS condition, we used the methodology in presented in Section 4.2.2 to compute the recommended selection strategy given in Table 2. The model's predicted selection strategy profile for following this recommended selection strategy is displayed in Figure 3b along with the profile observed in the noDS condition of the user study, which is shown in Figure 3a. While the predicted strategy profile for following the recommended strategy is not smooth due to the simplifications described previously, it does provide several interesting insights.

First, the recommended selection strategy gives low priority to sending additional UVs into the maze in the final three minutes of a RESCU mission. This recommendation agrees with our discussion in Section 4.1, in which we noted that users sent too many UVs into the maze in the sixth and seventh minutes, which required users to spend extra time replanning and re-routing in the last minute of the mission. The model predicts that the recommended selection strategy avoids this problem by placing low priority on sending UVs into the maze in the last few minutes of a mission.

Second, the model recommends that users should give less priority to replanning and re-routing in minutes two through five than users actually did in the noDS condition. Rather, the recommended selection strategy suggests that users should spend time performing tasks that the UVs cannot do themselves, such as goal assignment and payload operations. However, by the sixth minute, the model recommends that the human

operator should give high priority to replanning and re-routing so as to ensure that all UVs can pick up an object and carry it out of the maze before time expires.

A third interesting observation about the recommended selection strategy is that it suggests that operators give high priority to re-routing and replanning in the first minute. This is contrary to what users actually did in the noDS condition, as users typically dedicated the first minute to sending UVs into the maze. It is not entirely clear why placing such high priority on replanning and re-routing in the first minute would be effective.

The model predicts that the differences in the recommended selection strategy and the average selection strategy employed by the users in the noDS condition would translate into substantial differences in system effectiveness. Figure 4 shows the predicted effects of the recommended strategy with respect to number of objects collected, UVs lost, and system score. The model predicts that, while the recommended selection strategy would have a relatively small impact on number of objects collected, it would significantly decrease the number of UVs lost, especially for larger team sizes. For comparative purposes, we compare these predicted effects with those of the AVS and MBE enhancements discussed in Phase 2 in Appendix D.

In short, the recommended selection strategy and the subsequent prediction of its impact on system effectiveness give significant insight into the operators' observed attention allocation in the noDS condition. While users implemented reasonably good selection strategies with respect to collecting objects, they did not use good selection strategies for preventing loss of UVs. Thus, the model shows that future design changes

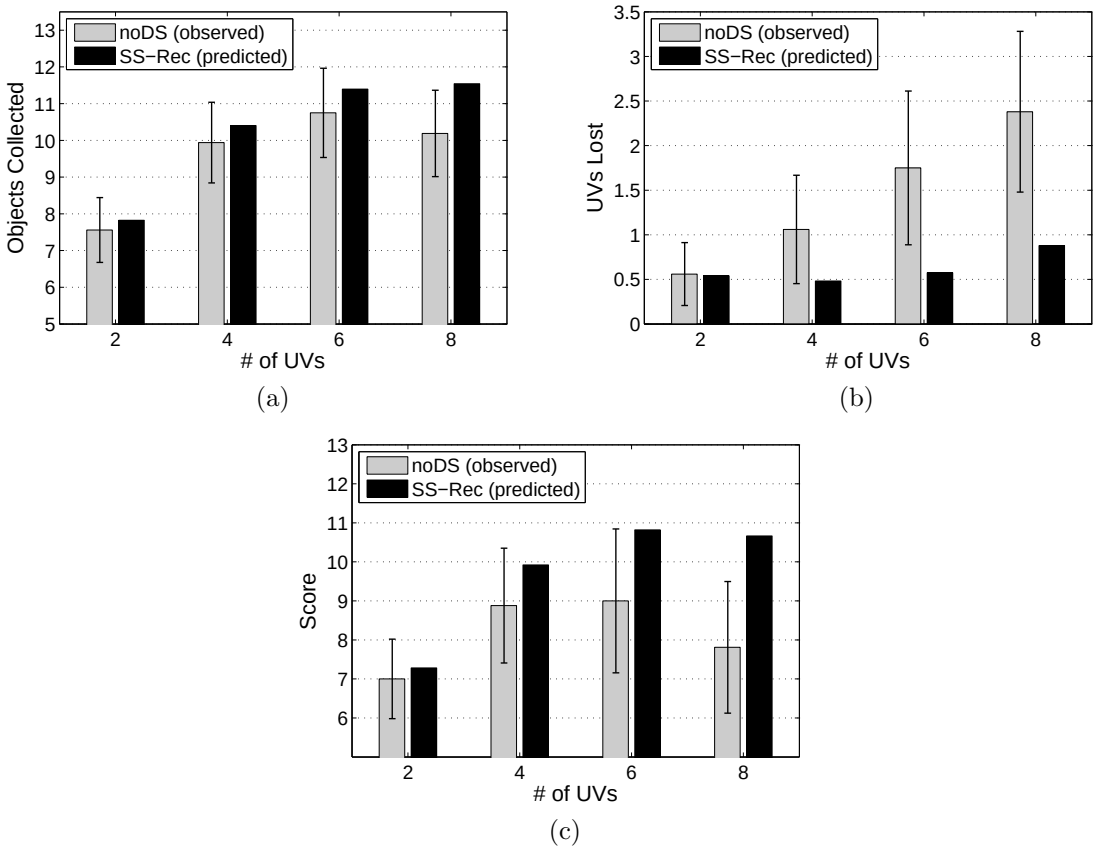


Figure 4: Predicted system effectiveness for following the recommended selection strategy (labeled *SS-Rec*) compared with observed system effectiveness in the user study (labeled *noDS (Observed)*) for (a) objects collected, (b) UVs lost, and (c) system score.

with respect to attention allocation efficiency should aid users in moving UVs out of the maze, rather than helping them to collect more objects. These results show that a predictive model such as the one we have presented can be used to show a cost-benefit analysis approach of where system designers should be spending resources during system improvement.

While the model predicts that system effectiveness would significantly increase if operators followed the recommended selection strategy, two questions remain. First, are the predictions accurate? Second, how can knowledge of the recommended selection strategies be used to improve attention allocation efficiency? In the next subsection, we describe a user study that is designed to answer these two questions.

4.3 User Study

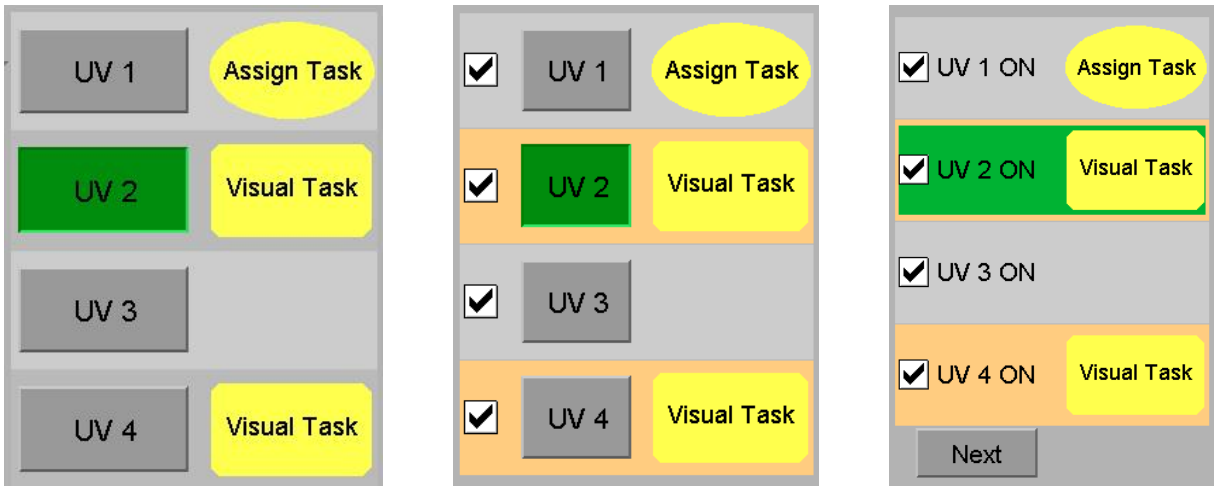
In the user study, which was performed in the RESCU test-bed, we compared three interface modes, which differed from each other in how the system selected UVs for the operator to service. In this subsection, we describe the different interface modes and outline the experimental procedure used in the study.

4.3.1 Decision Support for Operator Selection Strategies

Each of the three different interface modes was implemented in the noDS condition of RESCU, described in Appendix C. In each interface mode, the human was responsible for determining when human-UV interactions began and ended. However, the interface modes differed from each other in the *UV selection process*, which is the process the HUVS follows to select a UV for the user to service once the user decides to service another UV.

The UV selection process in each interface mode was communicated through the control panel of the interface. In past versions of RESCU, the control panel displayed the visual alarming system. It also displayed a button corresponding to each UV in the team, which the participants clicked to select a UV to service. In the new user study, this selection process was altered so that it was different in each interface mode. The control panel for each interface mode is shown in Figure 5. We discuss each in turn.

Manual Mode. The selection process for the Manual Mode was identical to the noDS mode in the previous study. In this mode, users selected the next UV to service by clicking on the button corresponding to that



(a) Manual (b) Guided (c) Auto

Figure 5: Control panels for each interface mode. The UVs recommended by the recommendation system are highlighted in orange.

UV. The Manual Mode was included in the study for control purposes.

Guided Mode. As in the Manual Mode, the Guided Mode was implemented so that the user selected the UV she desired to service by clicking on the button corresponding to that UV. However, in this mode, the system highlighted the UVs that the recommended selection strategy indicated the user should service. (Figure 5b). Thus, in this mode, the user could decide whether or not she desired to follow the recommended strategy. If the user felt that a recommendation was in error, the user could simply ignore the recommendation, or he could temporarily "turn it off" by unchecking the check box next to a UV. Once the button was unchecked, the UV was no longer highlighted, and subsequent recommendations were then displayed in the control panel. Thus, the Guided Mode allowed the human operator and the computer to collaborate with each other using a management-by-consent selection methodology.

Auto Mode. The Auto Mode restricted the set of UVs the operator was allowed to service to the UVs suggested by the recommended selection strategy. Thus, rather than directly select a UV to service, the user simply clicked a button labeled "Next" at the bottom of the control panel (Figure 5c). Once this button was clicked, the computer automatically selected a UV for the operator to service based on the recommended selection strategy. In the event that the user did not want to service a selected UV, he was allowed, as in the Guided Mode, to temporarily "turn-off" a recommendation by unchecking the check box next to the UV. The Auto Mode was included in the study specifically to validate the predictions made by the model since users typically followed the recommended selection strategy in this mode.

4.3.2 Experimental Setup

The user study was a single factor within-subjects study. The independent variable was the interface mode, which had three levels: Manual, Guided, and Auto. Each user controlled a team of eight simulated UVs using each interface mode. The order that the users saw each mode was randomized and counter-balanced to offset ordering effects.

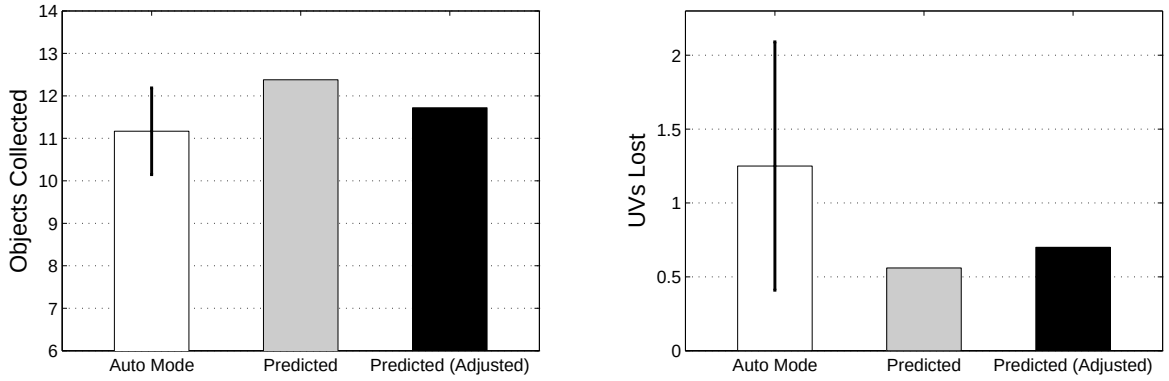
All previous user studies in the RESCU test-bed used UV team size as a within-subjects variable. However, in this study, the within-subjects variable was the interface mode. This required that the experimental procedure be altered somewhat from previous studies. Despite this difference, we attempted to make the experimental procedure used in this study as close as possible to that of previous studies. The following procedure was followed for each subject:

1. The subject was trained on all aspects of RESCU, including the interface, UV behaviors, etc. This was done using a PowerPoint presentation. The subject was allowed to ask questions as she desired.
2. The subject was trained separately on the UV control interface and the city search task.
3. The subject performed a full practice mission.
4. The subject was introduced to one of the three interface modes with a brief PowerPoint presentation.
5. The subject performed a complete practice mission using the new interface.
6. The subject performed a complete test mission using the new interface.
7. The subject answered several subjective questions about their experience.
8. The subject repeated steps 4-7 for the other two interface modes.

Twelve students and postdoctoral associates from the MIT community participated in the experiment. Six of the subjects were females and six were males. The subjects were between the ages of 19 and 32, with a mean age of 23.2 years.

4.4 Results

In this subsection, we present the results of the user study. In so doing, we attempt to answer three questions. First, did the model accurately predict the effects of the recommended selection strategy? Second, how did the different interface modes affect the operators' selection strategies. Third, what were the human operators' subjective responses to each of the interface modes? We answer each of these questions in turn.



(a) (b)

Figure 6: Predicted system effectiveness compared to observed system effectiveness in the Auto Mode. *Predicted* is the model's initial prediction of system effectiveness. *Predicted (Adjusted)* is the model's prediction of system effectiveness when presented with a correct model of $\mathcal{II}$.

4.4.1 Predictive Accuracy

To evaluate the model's ability to predict how changes in operator selection strategies affect system effectiveness, we compare the model's predictions[1] of system effectiveness with the system effectiveness observed in the Auto Mode. These results are shown in Figure 6, where the model's predictions are labeled *Predicted*. The figure shows that the model significantly over-predicted the number of objects the system collected (Figure 6a). Additionally, while the predicted number of UVs lost was just within the 95% confidence interval, the predicted value is still more than double the observed number of UVs lost in the user study (Figure 6b).

The inaccuracy of the predictions can be traced, in part, to an incorrect model of interaction impact $\mathcal{II}$. The model used data from the noDS condition of the previous study to estimate $\mathcal{II}$. However, in the Auto Mode, users typically took about 1.6 seconds longer per interaction than they did in the noDS condition. The longer interaction times in the Auto Mode appear to have been caused by the UV selection process used in this mode. Since users did not decide which UV they serviced, this sometimes caused them to spend more time gaining awareness of the selected UV's situation, which, as predicted by Billings [1], led to longer interaction times than in the noDS condition. Thus, the assumption of a correct estimate of $\mathcal{II}$ as stipulated in Section 4.2.2 was violated, thus compromising the model's predictive ability.

If we change the model's estimate of $\mathcal{II}$ to reflect the observed interaction times in the Auto Mode, we get the predictions labeled *Predicted (Adjusted)* in Figure 6. These predictions for both objects collected and UVs lost fall well within the 95% confidence intervals, which indicates that *if* the model's estimates of interaction impact $\mathcal{II}$, neglect impact $\mathcal{NI}$, and switching times $\mathcal{ST}$ are accurate, the model can adequately predict the effects of operator selection strategies on system effectiveness, as stated in the assumptions in Section 4.2.2. However, when these estimates are incorrect, the model's predictions are likely to be inaccurate.

Our model's reliance on observed data (from the noDS condition) to estimate $\mathcal{II}$ means that it cannot adequately predict the effects of different operator selection strategies when the selection process alters human-UV interactions. This is because, prior to implementing a change in the system, we do not have data to model how $\mathcal{II}$ will vary due to that particular change. In order to improve the model's accuracy, we would need to somehow estimate how the existing interaction data would be altered by the changes in the system.

[1]Due to a system upgrade on the computer used in the study, the mouse scroll wheel was less sensitive in previous user studies than it was in the current user study. This significantly increased the speed at which users were able to locate cities on the map (to pick up objects). In order to still be able to evaluate the predictive ability of the model, these new search times were incorporated into the model. The upgrade did not appear to impact any other aspect of the study.

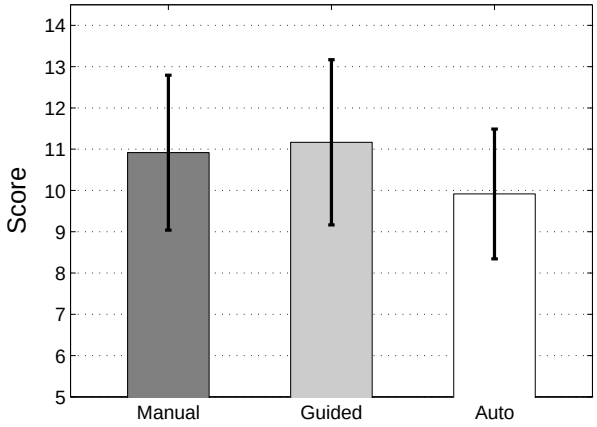


Figure 7: Observed system score in the user study.

In short, since alterations in the selection process can alter interaction efficiency, our current modeling methodology is not sufficiently robust to predict the effects of changes in operator selection strategies to a high degree of fidelity. However, since the model does give reasonably good predictions given correct estimates of interaction efficiency, the model has potential as a high-fidelity predictive tool if we can anticipate the effects of operator selection strategies on interaction efficiency. This is a topic of future work.

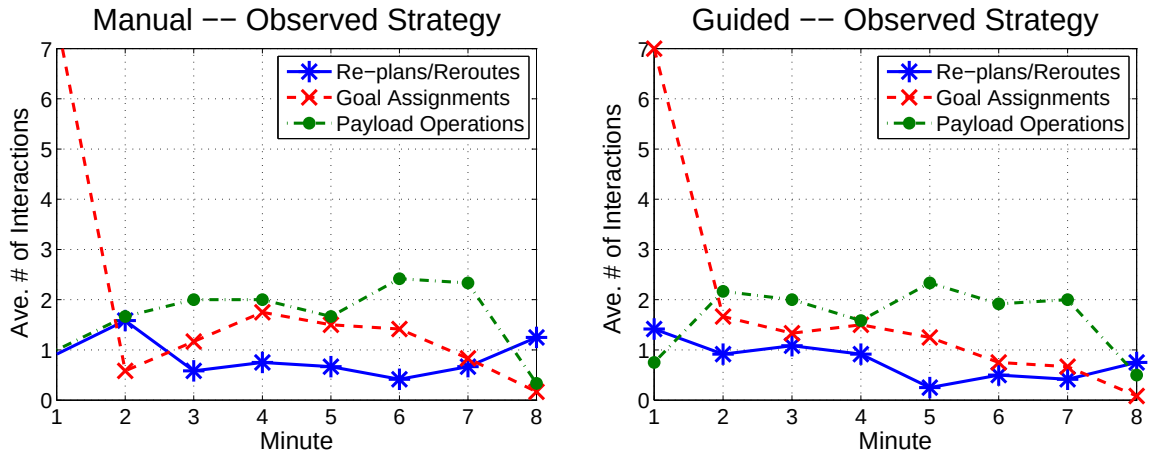
4.4.2 Effects of Decision Support on Operator Selection Strategies

The second question addressed by the user study concerns how the UV selection processes used in the Auto and Guided Modes altered the users' selection strategies. To understand users' reactions to the recommendations, we first discuss the optimality of the recommended selection strategy. While the recommended selection strategy was theoretically optimal with respect to a simplified set of selection strategies and the data observed in the noDS condition of the previous study, it was not optimal with respect to the upgraded system used in the current user study. In fact, users had higher scores in the Manual and Guided Modes, in which users often deviated from the recommended strategy, than in the Auto Mode, in which the recommended selection strategy was typically followed (Figure 7). While this difference was not statistically significant ($F(2, 33) = 0.50, p = 0.609$), it does show that the recommended selection strategy was no better in this user study than the selection strategies the users employed in the Manual Mode.

The average selection strategy profiles observed in each mode and the predicted strategy profile for following the recommended selection strategy are displayed in Figure 8. As expected, the selection strategy used in the Auto Mode most closely mirrors the model-predicted strategy profile. Additionally, the average selection strategy in the Manual Mode was similar to the selection strategy observed in the noDS condition of the previous study (Figure 3a).

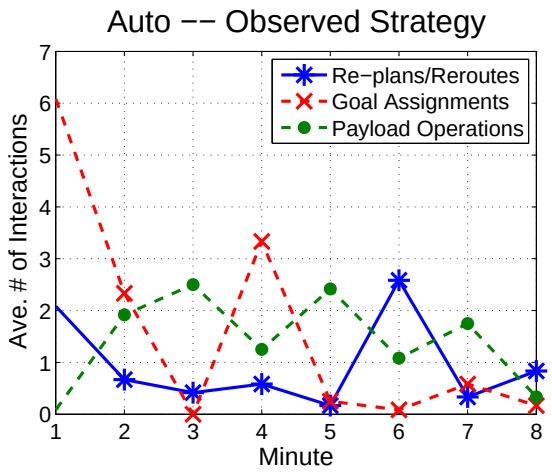
Given the differences between the recommended selection strategy and the observed selection strategies in the Manual Mode, it is interesting to observe selection strategies in the Guided Mode, where users were free to follow or ignore the recommended selections. While users did follow many of the recommendations, they did not always do so. To see this, consider Figure 9a, which shows the percentage of time that users followed the model's recommended selections. In the Manual Mode, users' adhered to the recommended strategy about 50% of the time, which is just higher than random behavior (30% adherence). Meanwhile, as expected users, almost always followed the recommended strategies in the Auto Mode. The percentage of adherence to the recommended selections in the Guided Mode is about 60%. Thus, the user's selections in the Guided Mode more similar to those observed in the Manual Mode than in the Auto Mode.

The effects of highlighting the recommended UVs in the control panel (as was done in the Guided

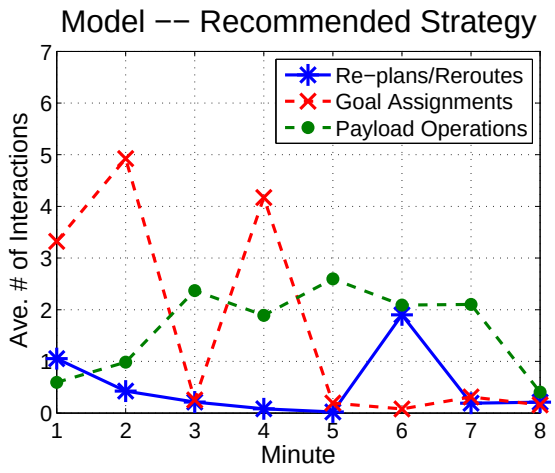


(a) Manual

(b) Guided

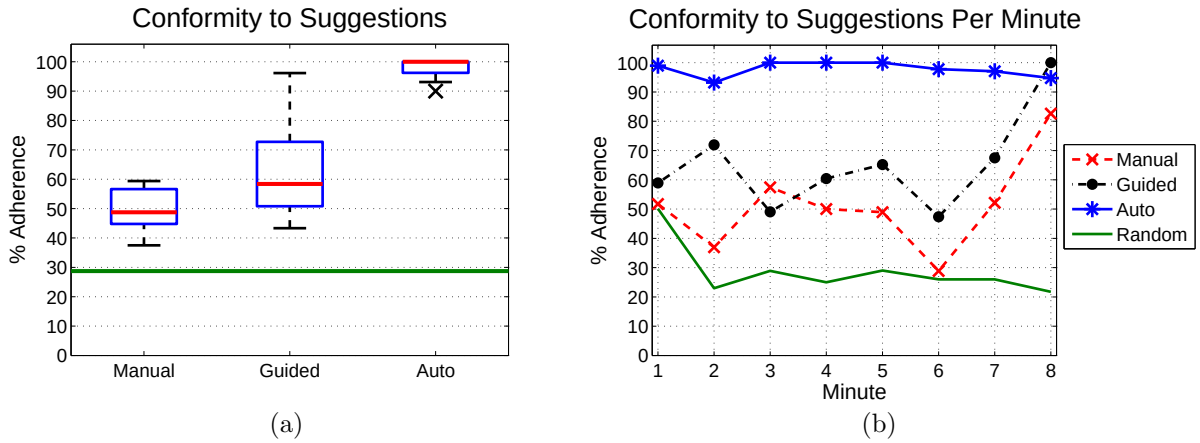


(c) Auto



(d) Model

Figure 8: Average observed selection strategy profiles for the (a) Manual, (b) Guided, and (c) Auto Modes compared to (d) the model's predictions when users adhere to the recommended selection strategy (after incorporating new city search times into the model).



(a) (b)

Figure 9: (a) Percentage of the time users' UV selections were in conformance with the recommended selection strategy averaged over all minutes of the study. The green horizontal line marks the percentage of adherence for purely random UV selections. (b) Percent conformity to the optimal select strategy per minute of the study.

Mode) is further displayed in Figure 9b, which plots the users' average adherence rates per minute of a RESCU mission. Beginning in the fourth minute until the end of the mission, the users' conformance to the recommended strategies in the Guided Mode mirrored that of the Manual Mode, except that the Guided Mode was shifted upward between 10-20 percentage points. This shows that users tended to follow their own strategies, though they were somewhat biased by the recommendations.

Post-experiment comments by the participants give further insight into how the users viewed the recommendations in the Guided Mode. One participant said that he completely ignored the recommendations in the Guided Mode because they were "confusing." Several other users commented that, while they did not always follow the recommendations in the Guided Mode, the recommendations sometimes drew their attention to a UV that required servicing that they otherwise might have missed. Another participant determined that he did not need to follow the recommendations because the penalty for not doing so was not severe.

These comments illustrate that many of the users correctly deduced that the recommended selection strategy was sub-optimal. This is further validated by an analysis of how often users chose to "turn off" recommendations in the Guided and Auto Modes. Recall that the Guided and Auto Modes allowed users to check a box to turn off recommendations for a given UV. Once this was done, a subsequent set of recommendations was provided. In the Guided Mode, none of the users turned off any recommendations. Since users often did not follow the recommendations, this suggests that, in this mode, users preferred to ignore the recommendations when they did not agree with them, rather than receive subsequent recommendations. However, in the Auto Mode, when ignoring the recommendations was impossible without explicitly turning them off, three of the 12 users (or 25%) chose to turn off various recommendations throughout the course of a mission (Figure 10), thus expressing that they did not believe the recommended selections to be desirable. Post-experiment discussions with the participants indicate that more of the users would have used this feature in the Auto Mode if they had remembered how to used it.

4.4.3 User Perceptions

While observed system effectiveness is a crucial metric of any decision support system, one cannot discount the role of user perception. While a system might produce good results, it will not likely become successful unless it gains user acceptance. Thus, we complete the results section with a discussion of the participants' attitudes toward the various recommendation systems.

After completing the user study, participants were asked to rank the different modes according to their

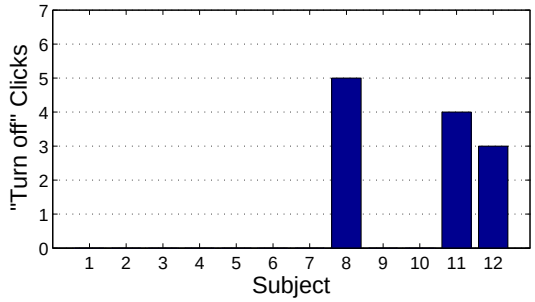


Figure 10: Number of times each subjected "turned off" recommendations in the the Auto Mode.

preferences. Eight of the 12 participants in the study preferred the Guided Mode the most. They liked that the Guided Mode allowed them to service whichever UV they desired. Additionally, several users said that the recommendations alerted them of UVs that needed to be serviced that they otherwise might have missed.

In comparison, eight of the 12 users liked the Auto Mode the least. Many of the users expressed frustration that they were not allowed to select a UV that was not suggested by the recommendation system. On the other hand, several operators appreciated that the Auto Mode relieved some of their workload. This intuition is validated statistically. After each mission in the study, each participant was asked to rank his mental workload during the mission on the scale one to five. An ordered logit model, specifically proportional odds [14], shows a statistical difference in subjective workload measured in this way ($\chi^2(2) = 6.98$, $p = 0.0305$). The odds of having higher subjective workload was higher for the Auto Mode compared to the Guided Mode ($\chi^2(1) = 9.84$, $p = 0.002$) and the Manual Mode ($\chi^2(1) = 5.46$, $p = 0.020$). Thus, while the Auto Mode did frustrate many of the users, it also, as expected, lowered their perceived workload.

4.5 Conclusions and Future Work

The third phase of the project resulted in a number of key insights, which we now summarize and discuss. First, we summarize our findings with regard to our model's ability to predict the effects of operator selection strategies in HUVSs. Second, we discuss our results concerning the role of computational models in decision aids for attention allocation in multi-task environments.

4.5.1 Predicting the Effects of Operator Selection Strategies

In this phase of research, we have developed and evaluated a method for predicting the effects of operator selection strategies on system effectiveness in HUVSs. Our approach has a number of strengths and weaknesses, each of which highlights an area of future work. First, our modeling methodology can be used to show a cost-benefit approach of where system designers should be spend intervention resources. For example, the predictions made by our model showed that, in RESCU, users' selection strategies were effective in achieving the first mission goal, that of collecting as many objects as possible throughout the mission. However, users' selection strategies were not as effective with respect to the second mission goal, that of ensuring that UVs were out of the maze prior to time expiring. Thus, the predictive model indicates that resources should be invested in developing technologies that will help users focus more effectively on this second mission objective, rather than on the first.

Second, under the assumption that we have a correct estimate of the other aspects of the system, including interaction impact $\mathcal{II}$, neglect impact $\mathcal{NI}$, and operator switching time $\mathcal{SS}$, our results indicate that the model gives reasonably good predictions of the effects of operator selection strategies on system effectiveness. While this is a good start, it appears that the assumption is overly strong. In many instances, the act of altering operator selection strategies will induce changes in human-UV interactions, as demonstrated in our

user study in the Auto Mode. Thus, while our modeling methodology provides a framework for developing predictive models capable of predicting the effects of changes in operator selection strategies, it is not sufficiently robust to predict the effects of operator selection strategies to a high-degree of fidelity. To be sufficiently robust, the model must anticipate how changes in one aspect of the system will affect other aspects of the system. This is an area of future work.

Third, while we attempted to use the model to identify an "optimal" selection strategy, the selection strategy was not optimal with respect to the system used in the current user study. This was due to small, unsuspected changes in the system that we, as system designers, did not anticipate, but that had substantial impact on the effectiveness of the system. While it is possible that more robust models can be developed that can mitigate the effects of these unanticipated changes, the real-world is sufficiently complex that these models will eventually fail. In such situations, it is essential that the system be designed so that users can adequately compensate for such failures.

4.5.2 Using Predictive Models to Improve Operator Selection Strategies

Once a predictive model has identified a selection strategy that would improve system effectiveness, it is not clear how it should be implemented into a system. In our user study, we used the recommended selection strategy computed by the model to alter the UV selection process. In the Guided Mode, the recommendations were highlighted on the display, and users were free to follow them or not follow them as they desired. In the Auto Mode, users simply indicated that they wanted to service a new UV. The recommended selection strategy was then used by the system to select a new UV for the operator to service.

While the user study showed no statistically significant difference in system effectiveness between the Auto and Guided Modes, users typically liked the Guided Mode, but they did not like the Auto Mode. This result mirrors the findings of Mitchell *et al.* [9] and Ruff *et al.* [12], in which management-by-consent was the preferred method of human-UV collaboration in supervisory control of multiple UVs. In the Auto Mode, users were often frustrated that they could not service the UV of their choice, as the system sometimes selected a UV they did not want to service. On the other hand, in the Guided Mode, while at least some of the users realized that the suggestions made by the model were sub-optimal, many of the users felt that they were still able to make good use of them. This is significant since, as we mentioned previously, highly robust predictive models are still likely to have moments of failure in complex HUVSs. In such situations, it is essential that operators can determine when to follow and when to not follow the recommendations. Thus, rather than focus on identifying optimal selection strategies, a more desirable approach might be to identify selection strategies that are good enough while promoting behavior in which users judiciously choose whether to follow or not follow system recommendations, thus avoiding the negative effects of automation bias [5] and mis-trust [8].

Acknowledgments

This research was funded by MIT Lincoln Laboratory. We especially thank Gary Condon and Mike Hurley who provided technical oversight and very useful feedback and insightful conversations. We also acknowledge the help of many members of MIT's Human and Automation Laboratory (HAL) for their input and work on the project, of whom we particularly mention Carl Nehme, Mauro Della Pena, Paul de Jong, and Vanessa Esch.

References

[1] C. E. Billings. *Aviation automation: The Search for a human-centered approach.* Mahwah, NJ: Lawrence Erlbaum Associates, 1997.

[2] J. W. Crandall, M. L. Cummings, and C. Nehme. A predictive model for human-unmanned vehicles systems. *Submitted to the Journal of Aerospace Computing, Information, and Communication*, June 2008.

[3] Jacob W. Crandall and M. L. Cummings. Developing performance metrics for the supervisory control of multiple robots. In *Proceeding of the 2nd Annual Conference on Human-robot Interaction*, pages 33–40, 2007.

[4] Jacob W. Crandall and M. L. Cummings. Identifying predictive metrics for supervisory control of multiple robots. *IEEE Transactions on Robotics*, 23(5), Oct 2007.

[5] M. L. Cummings. Automation bias in intelligent time critical decision support systems. In *AIAA 1st Intelligent Systems Technical Conference*, pages 33–40, September 2004.

[6] M. L. Cummings, C. P. Pina, and J. W. Crandall. A metric taxonomy for supervisory control of unmanned vehicles. In *AUVSI Unmanned Systems North America*, 2008.

[7] David B. Kaber and Mica 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*, 5(2):113–153, 2004.

[8] John D. Lee and Neville Moray. Trust, self-confidence, and operators' adaptation to automation. *International Journal of Human-Computer Studies*, 40(1):153–184, January 1994.

[9] Paul J. Mitchell, M. L. Cummings, and Thomas B. Sheridan. Mitigation of human supervisory control wait times through automation strategies. Technical report, Humans and Automation Laboratory, Massachusetts Institute of Technology, June 2003.

[10] H. Neth, S. S. Khemlani, B. Oppermann, and W. D. Gray. Juggling multiple tasks: A rational analysis of multitasking in a synthetic task environment. In *Proceedings of the* 50^{th} *Annual Meeting of the Human Factors and Ergonomics Society*, 2006.

[11] C. P. Pina, M. L. Cummings, and J. W. Crandall. Identifying generalizable metric classes to evaluate human-robot teams. In *HRI Workshop on Performance Metrics for Human-Robot Interaction*, 2008.

[12] Heath A. Ruff, S. Narayanan, and Mark H. Draper. Human interaction with levels of automation and decision-aid fidelity in the supervisory control of multiple simulated unmanned air vehicles. *Presence*, 11(4):335–351, 2002.

[13] T. B. Sheridan and M. K. Tulga. A model for dynamic allocation of human attention among multiple tasks. In *Proceedings of the* 14^{th} *Annual Conference on Manual Control*, 1978.

[14] M. E. Stokes, C. S. Davis, and G. G. Koch. *Categorical Data Analysis Using the SAS System (2nd ed.)*. Cary, NC: SASvPublishing, BBU Press and John Wiley Sons Inc., 2002.

Appendix A

The following is a draft of a paper published in:

The 2nd Annual Conference on Human-Robot Interaction, Washington, D. C., 2007.

Developing Performance Metrics for the Supervisory Control of Multiple Robots

Jacob W. Crandall
Dept. of Aeronautics and Astronautics
Massachusetts Institute of Technology
Cambridge, MA
jcrandal@mit.edu

M. L. Cummings
Dept. of Aeronautics and Astronautics
Massachusetts Institute of Technology
Cambridge, MA
missyc@mit.edu

ABSTRACT

Efforts are underway to make it possible for a single operator to effectively control multiple robots. In these high workload situations, many questions arise including how many robots should be in the team (Fan-out), what level of autonomy should the robots have, and when should this level of autonomy change (i.e., dynamic autonomy). We propose that a set of metric classes should be identified that can adequately answer these questions. Toward this end, we present a potential set of metric classes for human-robot teams consisting of a single human operator and multiple robots. To test the usefulness and appropriateness of this set of metric classes, we conducted a user study with simulated robots. Using the data obtained from this study, we explore the ability of this set of metric classes to answer these questions.

Categories and Subject Descriptors

J.7 [**Computers in Other Systems**]: Command and Control; H.5.2 [**User Interfaces and Presentation**]: Evaluation/methodology

General Terms

Measurement, Performance, Human Factors

Keywords

Multi-robot Teams, Fan-out, Supervisory Control

1. INTRODUCTION

Over the last few years, much research has focused on human-robot teams (HRTs) in which a single operator controls or supervises multiple robots. This is a somewhat daunting task as current technologies (in both air, ground, and water robotics) require multiple humans to control a single robot. However, it is desirable to invert this ratio in order to (a) reduce costs, (b) extend human capabilities, and (c) improve system efficiency. To achieve this goal, additional research must address a multitude of issues related to both the human operator (i.e., human factors issues), the robots (i.e., artificial intelligence capabilities), and the interactions between them.

One important research agenda is determining the effectiveness of a given HRT in accomplishing a mission. To do so, robust and descriptive metrics must be developed. The first conference on Human-Robot Interaction (HRI 2006) included a paper calling for the development of common metrics for human-robot systems [24]. The authors of this paper argued that metrics should be developed that span the range of missions carried out by HRTs. These metrics should relate to both humans and robots in the team as well as the entire human-robot system (HRS). In this paper, we focus on quantitative metrics for HRTs consisting of a single human operator and multiple robots.

Often, a single metric is sought to evaluate an HRT's effectiveness. However, since metrics of overall system effectiveness vary widely across domains [27] and are typically multi-modal, a common metric for overall system effectiveness is unlikely to be found. However, a *set of metric classes* spanning many aspects (and subparts) of a system is likely to be more generalizable. Loosely, a metric class is the set of metrics that measure the effectiveness of a certain aspect of a system. For example, we might consider the metric class of human performance, which includes metrics of reaction time, decision quality, situation awareness, workload, etc.

We propose that a set of metric classes should have the following three attributes to effectively evaluate HRTs:

1. The set of metric classes should contain metrics that *identify the limits of all agents* (both human operator and robots) in the team.

2. The set of metric classes should have *predictive power*. An HRT might be called upon to perform many different kinds of missions in many different kinds of environments. An HRT that performs well in one environment or mission may not perform well in another environment or mission. Additionally, the teams themselves are likely to change (due to casualty, resource availability, mission needs, etc.). Measuring all such circumstances is costly and, ultimately, impossible. Thus, a set of metrics for HRTs should have some power to predict how changes in environment, mission, and team make-up will affect the team's effectiveness.

3. The set of metric classes should contain *key performance parameters* (KPPs). KPPs are the parameters that indicate the overall effectiveness of the system.

HRI'07, March 10–12, 2007, Arlington, Virginia, USA.

Finding a set of metric classes with these three attributes is important for a number of reasons. First, a set of metrics having these attributes can determine the capabilities of a system performing a given mission. In the context of an HRT consisting of a single human operator and multiple robots, such a set of metric classes addresses the question of whether a particular HRT is capable of completing a mission in a satisfactory manner or whether the team's configuration should change. Second, a set of metrics having these three attributes can help determine the levels of autonomy that the robots in the team should employ. This relates to a third reason, which is that such a set of metrics could be used to facilitate dynamic autonomy to a higher degree of fidelity. Fourth, such a set of metrics should identify how changes in system design will impact the system's overall effectiveness. This would both reduce the cost of creating robust HRTs while speeding up their development.

Identifying a set of metrics with these capabilities is a tall order. Nevertheless, we describe initial attempts to do so in this paper. We take the approach of decomposing an HRT into subparts. Measures can be obtained for each of these subparts. Estimates of overall team effectiveness can then potentially be constructed from these measures, even (ideally) when some aspects of the system, environment, or mission change. We demonstrate the potential ability of this set of metric classes via a user study.

The remainder of this paper proceeds as follows. In Section 2, we outline related work. In Section 3, we decompose a single-human multi-robot team into subparts and define metrics for the various subparts. In Section 4, we describe a user study designed to analyze the set of metric classes proposed in Section 3. We present and discuss the results of the user study in Section 5. We offer a concluding remarks and suggest future work in Section 6.

While HRTs of the future will include heterogeneous sets of robots, we focus in this paper only on the homogeneous case. However, the theories developed in this paper appertain to heterogeneous robot teams as well, though additional issues will need to be considered for those teams.

2. RELATED LITERATURE

The work of this paper relates to many topics in the literature. We focus on supervisory control of multiple robots, Fan-out, human-robot metrics, and dynamic autonomy.

2.1 Supervisory Control of Multiple Robots

In supervisory control [21], a human interacts with automation as the automation acts in the world (see Figure 1). When a human supervises multiple robots, care must be taken to ensure that the operator has the capacity to give adequate attention to each robot or group of robots. Adherence to multiple principles are required to make this possible, including offloading low-level control of the robots to the automation [4, 20, 6, 17], ensuring that the automation is reliable [7], and providing effective user interfaces (see [14, 23]). Predictive metrics are necessary to evaluate these technologies in a cost effective manner.

When a human controls multiple robots, the human must necessarily allocate his/her attention between the various robots or groups of robots. This is related to the concept of time-sharing (see [27, 1]). Metrics from the *attention allocation efficiency* (AAE) metric class discussed in Section 3.2 can be used to assess time-sharing capabilities.

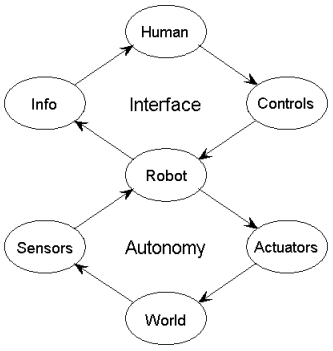


Figure 1: The two control loops of an HRT consisting of a single human operator and a single robot.

2.2 Fan-out

The term Fan-out (FO) refers to the number of (homogeneous) robots that a single operator can effectively control [16]. One line of research on this topic uses measures of interaction times and neglect times to estimate FO [11, 16, 3]. These metrics have been modified to include the use of wait times [14, 5] and extended (in part) to the domain of heterogeneous robot teams [12]. We analyze how effectively these metrics estimate true FO in Section 5.2.2.

2.3 Human-Robot Metrics

Much of the work on metrics for HRTs has focused on the human operator. The most common of these metrics measure situation awareness (SA) (formally defined in [9] and adopted to HRTs in [8]) and operator workload. Various metrics for SA have been devised including SAGAT [9]. Metrics for measuring operator workload include subjective methods (see [27]), secondary task methods, and psychophysiological methods (e.g., [13, 25]). However, metrics for HRTs must go beyond the human operator. Metrics are also needed to evaluate the effectiveness of individual robots in the team as well as the team's overall effectiveness [26].

The work of this paper focuses on combining metrics from various aspects of the HRT to obtain measures of system effectiveness. This is relates to [19], which computes a measure of overall team effectiveness using measures of the individual subtasks performed by the team.

2.4 Dynamic Autonomy

Central to the success of an HRT is the level of automation employed by the robots in the team. Sheridan and Verplank's [22] scale of levels of automation has been widely accepted and adapted for use in system design (e.g., [18]). The level of automation can be varied over time (dynamic autonomy) to manage changing operator workload and mission needs (e.g., [17, 2]). Predictive metrics can be used to determine when autonomy levels should be changed.

3. A SET OF METRIC CLASSES

We can identify a potentially useful set of metric classes by decomposing an HRT consisting of a single human and multiple robots into subparts. We first decompose a single robot team after which we take on the multi-robot case.

3.1 The Single-Robot Case

In the single-robot case, an HRT has the two control loops shown in Figure 1, which is adapted from [3]. These control loops are the control loops of supervisory control [21]. In the upper loop, the human interacts with the robot. The robot sends information about its status and surroundings to the human via the interface. The human synthesizes the information and provides the robot with input via the interface. The lower control-loop depicts the robot's interactions with the world. The robot combines the operator's input with its own sensor data to determine how to act.

The two control loops, though intimately linked, provide a natural decomposition of an HRT of this type. Corresponding to each control loop is a metric class. Metrics that evaluate the effectiveness of human-robot interactions (upper control loop) are in the metric class of *interaction efficiency* (IE). Metrics that evaluate the robot's autonomous capabilities (lower control loop) are in the metric class of *neglect efficiency* (NE). Note, however, that while these two metric classes are separate, they are in no way independent of each other. A failure in one control loop will often cause a failure in the other control loop.

Many metrics in the literature have membership in the IE and NE metric classes. We focus on a small set of these metrics in this paper.

3.1.1 Interaction Efficiency (IE)

Metrics in the IE metric class evaluate the effectiveness of human-robot interactions. That is, they evaluate (a) how well the human can determine the status and needs of the robot, (b) how human inputs affect robot performance, and (c) how much effort these interactions require. One way to estimate IE is by the expected length of a human-robot interaction. This metric is known as *interaction time* (IT), which (for the single-robot case) is the amount of time it takes for the operator to (a) orient to the robot's situation, (b) determine the inputs (s)he should give to the robot, and (c) express those inputs via the interface [15]. Related to IT is the metric WTI (wait times during interactions) [14], which is the expected amount of time during interactions that the robot is in a degraded performance state.

Using IT and/or WTI to capture IE infers that shorter interactions are more efficient than longer ones. Since this is not always the case, we might also want to consider metrics that more explicitly measure the performance benefits of an interaction. These benefits can be determined by observing how the robot's performance changes during human-robot interactions, which can be calculated from the mathematical structure *interaction impact* (II). II is the random process that describes the robot's performance during interactions [3][1]. It is a function of (among other things) the amount of time t since the operator began interacting with the robot. One metric we can derived from II is the robot's average performance during interactions, which is given by

$$\bar{II} = \frac{1}{IT} \int_0^{IT} E[II(t)]dt, \tag{1}$$

where $E[II(t)]$ denotes the robot's expected instantaneous performance at time t ($t = 0$ is when the interaction began).

[1]For descriptive purposes, we have modified the names of some of the terms discussed in this paper.

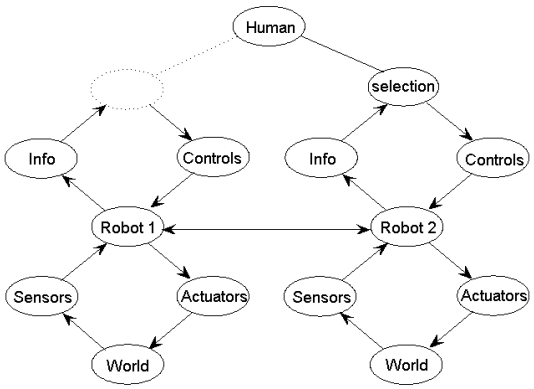


Figure 2: In multi-robot teams, human attention must be distributed between the robots.

3.1.2 Neglect Efficiency (NE)

The NE metric class consists of metrics that evaluate a robot's ability to act when the human's attention is turned elsewhere. *Neglect time* (NT), which is the average amount of time a robot can be ignored before its expected performance falls below a certain threshold [11], is a member of this metric class. One difficulty with this metric is determining the proper performance threshold. Methods for determining the threshold are given in [16, 3]. Like IT and WTI, NT does not completely account for the robot's performance. This additional information can be obtained from the mathematical structure *neglect impact* (NI), which is the random process that describes a single robot's performance when it is ignored by the operator [3]. From NI, we can calculate average robot performance during the time it can be safely neglected using

$$\bar{NI} = \frac{1}{NT} \int_0^{NT} E[NI(t)]dt, \tag{2}$$

where $E[NI(t)]$ denotes the robot's expected instantaneous performance after it has been neglected for time t.

3.2 The Multi-Robot Case

When a human interacts with multiple robots, the nature of interactions between the operator and each robot in the team remains relatively unchanged except for the important exception depicted in Figure 2. The figure shows a separate set of control loops for each robot. However, unlike the single-robot case, the upper loops are not always closed. To close one of the upper loops, the human must attend to the corresponding robot and neglect the others[2]. Thus, the efficiency with which human attention is allocated among the robots is critical to the team's success. Metrics that capture this notion of efficiency have membership in the *attention allocation efficiency* (AAE) metric class.

3.2.1 Attention Allocation Efficiency (AAE)

AAE can be measured in various ways including (a) the time required to decide which robot the operator should service after (s)he has completed an interaction with another

[2]We assume that a human *sequentially* attends to the needs of each robot.

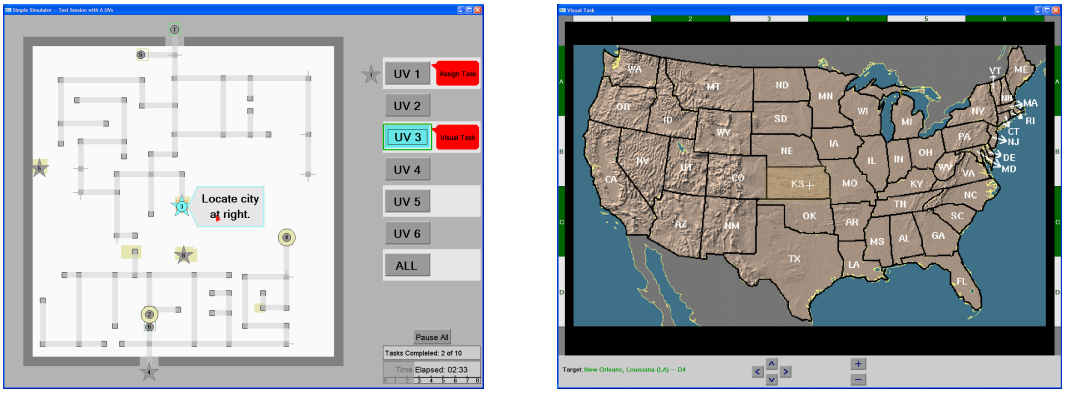


Figure 3: The two displays used in the experiment. Each was displayed on a separate monitor.

robot, and (b) the quality of that decision. The former metric is referred to as *switch times* (*ST*s) and has sometimes been considered part of *IT* [16]. We follow this lead in this paper, though it is of itself an individual metric of *AAE*.

Ideally, a metric evaluating the quality of servicing selections made by the HRT would compare the team's actual decisions with what would have been the "optimal" decisions. However, such a measure is often difficult to obtain given the complexity of the situations encountered by HRTs. One alternative metric is to compute the number of *wait times* (i.e., time in which a robot is in a degraded performance state) *caused by lack of operator SA* (called *WTSA*) [14]. In general, teams with higher *WTSA* have lower *AAE*. However, *WTSA* can also be difficult to measure since they must be distinguished from a third kind of wait time, called *wait times in the queue* (*WTQ*) [14]. *WTQ* occur when the human operator knows that a robot is in a degraded performance state, but does not attend to that robot because (s)he must attend to other robots or tasks. The metric *WTQ* is not exclusively from *IE*, *NE*, or *AAE*, though it is affected by all three system attributes.

Figure 2 also shows a connecting link between robots in the team. This link captures the notion that robots can communicate with each other. The quality of information passed over these links will in turn affect measures of *IE*, *NE*, and *AAE*. This could possibly define a fourth metric class, though we do not consider it in this paper.

4. USER STUDY

To evaluate how effectively sets of metrics drawn from *IE*, *NE*, and *AAE* identify the limits of the agents in the team, predict system characteristics, and provide KPPs, we conducted a user study. In this section, we describe the software test-bed used in the study, the experimental procedure, and the demographics of the participants.

4.1 Software Test-bed

We describe three aspects of the software test-bed: mission, interface, and robot behaviors.

4.1.1 Mission

Across many mission types, an HRT operator assists in performing a set of common tasks including mission planning and re-planning, robot path planning and re-planning, robot monitoring, sensor analysis and scanning, and target designation. These generic tasks apply to HRTs with many different kinds of robots, including unmanned air vehicles (UAVs), unmanned ground vehicles (UGVs), and unmanned underwater vehicles (UUVs). We give two time-critical examples: one with UAVs and the other with UGVs.

A human-UAV team might be assigned various intelligence gathering tasks over a city during the night. The team's mission is to perform as many intelligence gathering tasks before daylight as possible. The operator must assist in assigning the various UAVs to the various intelligence gathering tasks. Once the UAVs are assigned tasks, the UAV operator must assist the UAVs in arriving at the (possibly unknown) locations where these tasks are to be performed. This requires the operator to assist in path planning and the monitoring of UAV progress. As more information becomes available about the various tasks, the intelligence gathering tasks must be reassigned and routes re-planned. Once a UAV arrives at the location where the intelligence must be gathered, the operator must scan the UAV's imagery to identify objects of interest.

A human-UGV team might be tasked with a search and rescue mission in a damaged building. The mission goal would be to remove important objects (such as people) from the building in a timely manner (e.g., before the building collapses). To do this, the operator must assign the UGVs to various places in the building and assist them in getting to these locations. As new information about the building and the objects in it become available, the operator must often reassign the UGVs to other tasks. Once a UGV arrives at the location of an object, it would need the operator's assistance to positively identify and secure the object. This could require the operator to view and analyze imagery from the UGVs video feed. After securing the object, the UGV would then need to exit the building to deliver the object.

We sought to capture each of these generic tasks in our software test-bed, which is shown in Figure 3. In our study, the HRT (which consisted of the participant and multiple simulated robots) was assigned the task of removing objects from an initially unknown maze. The goal was to remove as many objects from the area as possible during an 8-minute session while ensuring that all robots were out of the maze when time expired. An object was removed from the building using a three-step process. First, a robot moved to the location of the object (target designation, mission planning, path planning, and robot monitoring). Second, the robot

"picked up" the object (sensor analysis and scanning). As this action might require the operator to perform a visual task (assist in identifying the object in video data), we simulated this task by asking the user to identify a city on a map of United States using *Google Earth*-style software (the graphical user interface is shown in the right of Figure 3). This locate-a-city task was a primary task and not a secondary task. Third, the robot carried the object out of the maze via one of two exits (one at the top of the maze and the other at the bottom of the maze).

The objects were randomly spread through the maze. The HRT could only see the positions of six of the objects initially. In each minute of the session, the locations of two additional objects were shown. Thus, the total number of objects to collect during a session was 22. Each participant was asked to maximize the following objective function:

$$Score = ObjectsCollected - RobotsLost, \quad (3)$$

where *ObjectsCollected* was the number of objects removed from the area during the session and *RobotsLost* was the number of robots remaining in the area when time expired.

4.1.2 Interface

The human-robot interface used in the study was the two-screen display shown in Figure 3. On the left screen, the maze was displayed along with the positions of the robots and (known) objects in the maze. As the maze was initially unknown to the HRT, only the explored portions of the maze were displayed. The right screen was used to locate cities in the United States.

The user could control only one robot at a time. The user designated which robot (s)he wanted to control by clicking a button on the interface corresponding to the desired robot (labeled UV1, UV2, etc.). Once the user selected the robot, (s)he could control the robot by specifying goal destinations and making path modifications. Goal designation was achieved by dragging the goal icon corresponding to the robot in question to the desired location. Once the robot received a goal command it generated and displayed the path it intended to follow. The user could modify this path using the mouse.

To assist the operator in determining which of the robots needed attention, each robot's status was shown next to its button. This status report indicated if the robot had completed its assigned task, found an object, or needed to exit the maze. If no status report was given, the system determined that the robot was progressing adequately on its assigned task.

4.1.3 Robot Behavior

The robot combined a goal seeking (shortest path) behavior with an exploration behavior to find its way toward its user-specified goal. This behavior, though generally effective, was sometimes frustrating to the users as it often led to seemingly undesirable actions (though, as we mentioned, the user could modify the robot's path if desired).

4.2 Experimental Procedure

After being trained on all aspects of the system and completing a comprehensive practice session, each user participated in six 8-minute sessions. Teams with two, four, six, and eight robots were tested. In each of the first four sessions, a different number of robots were allocated to the team. In the last two sessions, the conditions (i.e., robot team size) from the first two sessions were repeated. Thus, 18 samples were taken for each robot team size[3]. The conditions of the study were counter-balanced. The participants were paid $10 per hour with the highest scorer also receiving a $100 gift certificate.

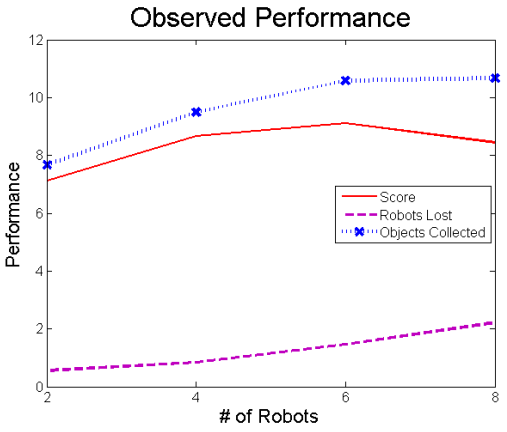


Figure 4: The mean values of number of objects collected, number of robots lost, and overall score.

4.3 Demographics

Twelve people (one professor, ten students, and one other non-academic person) participated in the study; eight were from the United States, two were Canadian, one was Hispanic, and one was Egyptian. Three of these participants were female and nine were male. The mean age was 27.5 years old with a standard deviation of 8.6 years.

5. RESULTS

Data collected from the user study allows us to evaluate sets of metrics (drawn from IE, NE, and AAE) with respect to their ability to identify the limits of the agents in the team, predict system characteristics, and provide KPPs. Before presenting this analysis, we report observations of system effectiveness for each robot team size.

5.1 Observed Team Effectiveness

The dependent variables we consider for HRT effectiveness for this user study are those related to Equation 3: the number of objects collected by the HRT over the course of a scenario and the number of robots lost during a scenario. The mean observations for these dependent variables across the number of vehicles is shown in Figure 4.

Figure 4 shows that a 2-robot HRT collected on average just less than eight objects per 8-minute session. The sample mean steadily increases as team size increases up until 6-robots, at which point it appears to plateau. A repeated measures ANOVA revealed marginal significance across robots, $\alpha = 0.05$, $F = (15, 3) = 2.737$, $p = 0.06$. Pairwise comparisons show that 2-robot teams collect significantly less objects than do 4-, 6-, and 8-robot teams ($p \leq 0.001$), and 4-robot teams collect significantly less objects than 6- and 8-robot teams ($p = 0.057$ and $p = 0.035$,

[3]Only 17 samples are available from the 6-robot condition due to technical difficulties.

	2	4	6	8
IT	18.19	16.86	15.82	15.74
NT	22.26	36.63	44.67	52.22
WT	8.71	26.88	45.03	67.58

Table 1: Estimated values of IT, NT, and WT given in seconds per robot team size (the columns).

respectively). HRTs with six and eight robots are statistically the same.

Figure 4 also shows that the average number of robots lost per session increases as robot team size increases. Robots were lost if they were in the maze when time expired. A clear distinction exists between groupings of 2- and 4-robot teams and 6- and 8-robot teams as demonstrated by a χ^2-test ($\chi^2 = 14.12$, $df = 3$, $p = .033$)[4]. This result is significant as it indicates a performance drop between four and six robots. Thus, while robot teams with six and eight robots collected more objects than smaller robot teams, they also lost more robots.

These results indicate that the HRTs in the user study with the highest performance had, on average, between 4 and 6 robots. Thus, FO for this particular situation appears to be between four and six robots.

5.2 Analysis of Sets of Metrics

We now analyze selected sets of metrics drawn from IE, NE, and AAE with respect to the three attributes listed in the introduction. Namely, we want to determine how well these metrics determine the limits of the agents (both the human and the robots) in the team, predict system characteristics, and provide key performance parameters (KPPs). We analyze each attribute separately.

5.2.1 Limits of the Agents

The observed values of IT, NT, and WT (the average wait time per interaction-neglect cycle) are given in Table 1. We used the following heuristics to calculate them:

- IT was determined by observing clicks on the robot selection buttons as well as other mouse activity. Estimated switch times, which were about 1.7 seconds in each condition, are included in this measure.
- NT was determined to be the time elapsed between the operator's last interaction with the robot and the time at which the operator again interacted with the robot or the robot reached its designated goal location.
- WT was determined to be the average time a robot waited to be serviced after it reached its goal. Thus, both WTQ and $WTSA$ are included in this measure. If a robot did not reach its goal before the operator chose to service it, we assumed that no wait times accrued.

Previous discussions of operator capacity based on the measures IT, NT, and WT are given in [16, 14, 5]. We provide analysis of operator capacity using these measures for our specific study.

In a 2-robot team, Table 1 shows that, on average, a robot was serviced for about 18 seconds (IT), then moved productively toward its goal while being neglected for about 22 seconds (NT), and then waited for operator input for a little less than 9 seconds (WT). Thus, the robot was either actively pursuing its goal or being serviced more than 82% of the time. This indicates that the operator was usually able to provide adequate attention to both robots. However, as the number of robots in the team increased, the amount of time the operator was able to give adequate attention to each robot decreased noticeably. In 8-robot teams, the user was typically unable to attend to the needs of each robot in the team as each robot spent about half of its time waiting for operator input. As a result, as team size increased, the number of objects collected reached a plateau while the number of robots lost continued to increase (see Figure 4).

We can make observations about the limits of the robots by observations of NT. In the 8-robot condition, when interactions with each robot were infrequent, NT was about 53 seconds. Since each robot received little attention from the users in this condition, this value is largely a function of the average time it took for the robots to reach their goals. Thus, it appears that a main limitation of the robots' autonomy was its dependence on user specified goals. Thus, future improvements in robot autonomy could include giving the robots the ability to create their own goals or initiatives.

5.2.2 Predictive Power

In this context, predictive power is the ability to determine how the HRT will perform in unobserved conditions. Thus, metrics are predictive if measures obtained in one condition (e.g., a fixed robot team size) can be used to accurately calculate measures for other (unobserved) conditions (e.g., other robot team sizes). Predictive metrics have two attributes. First, they are *accurate*, meaning that their predictions are close to the actual measures we would have observed in that condition. Second, they are *consistent*, meaning that the predictions are accurate regardless of the observed condition(s). These attributes can be assessed in both *relative* and *absolute* ways [27].

In this subsection, we will analyze the ability of three methods to predict FO and system effectiveness as robot team size changes. Each method uses a different set of metrics drawn from the IE, NE, and AAE metric classes.

Predicting Fan-out. The first method, which was presented in [15], estimates FO using:

$$FO = \frac{NT}{IT} + 1. \tag{4}$$

Thus, this method assumes FO is determined by the number of interactions that can occur with other robots while a robot is being neglect.

The second method, presented in [14], adds wait times to Equation (4) so that FO is computed using:

$$FO = \frac{NT}{IT + WT} + 1, \tag{5}$$

where $WT = WTQ + WTSA$.

The third method is a performance-based method described in [3]. This method uses the metrics IT, $\bar{I}I$, $\bar{N}I$, and NT (though IT and NT are determined in a slightly different fashion than in Table 1). In short, values of IT, $\bar{I}I$, and $\bar{N}I$ are enumerated for all possible values of NT. For each possible tuple (IT, NT) a corresponding average robot performance $\bar{V}$ is calculated using

$$\bar{V} = \frac{1}{IT + NT}\left(IT \cdot \bar{I}I + NT \cdot \bar{N}I\right).$$

[4]The χ^2-test for significance was used in this case since the data violated the assumptions on an ANOVA test.

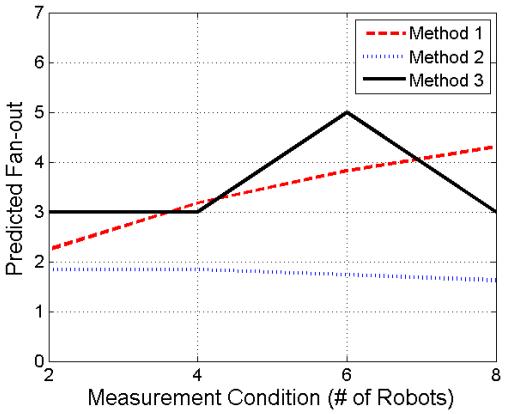


Figure 5: FO predictions using measures obtained from observing 2-, 4-, 6-, or 8-robot teams.

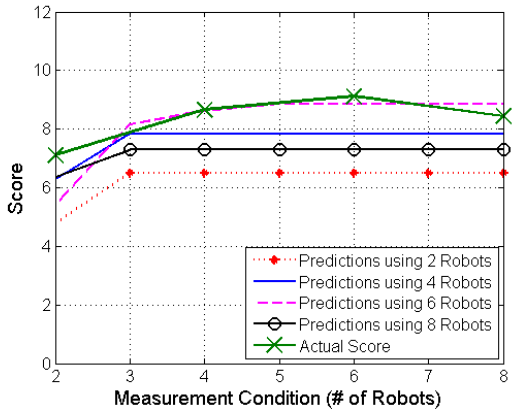


Figure 6: Predictions of system effectiveness based on metrics obtained using 2-, 4-, 6-, and 8-robot teams.

Each robot in the team is then assigned its own (IT, NT) tuple such that the sum of robot performances is maximized given the constraint: $NT_j \geq \sum_{i \neq j} IT_i$ for all j (where NT_i and IT_i are the neglect and interaction times assigned to robot i). This calculation is made for teams of all sizes. FO is the point where performance peaks or plateaus.

FO predictions for each of these methods (using the values shown in Table 1) are shown in Figure 5. In the figure, the x-axis represents the robot team size that was observed and the y-axis shows the resulting FO prediction. None of the methods consistently predicts the true FO (which, as we discussed previously, was between four and six robots). Method 1 predicts FO to be anywhere from 2.45 (when observing two robots) to 4.32 (when observing eight robots). Thus, this method is not consistent due to variations in the estimate of NT. Method 2's FO estimates, though pessimistic, are relatively consistent. This is an interesting result since Method 2 is the only method of the three that uses a metric with (partial) membership in the AAE metric class (other than combining ST with IT). It appears that the variabilities in NT are counteracted by WT. Future work should investigate whether this trend holds in other contexts. Method 3, also provides a pessimistic estimate, though its predictions are consistent except for the 6-robot team condition (at which point it gives a good estimate of FO). We illustrate why this method fails by analyzing its ability to predict system effectiveness.

Predicting System Effectiveness. Methods 1 and 2 use temporally-based methods that only predict the number of robots a team should have. They do not predict what a team's effectiveness will be (for any robot team size). Method 3, however, was designed to predict system effectiveness [3]. These predictions for the HRTs observed in the user study are shown in Figure 6. A set of predictions for each observed robot team size is given. We make several observations.

First, these predictions are not consistent in the absolute sense, though they are in the relative sense. While the predictions follow similar trends they do not always even accurately predict the observed conditions. Second, the figure shows that these predictions are on scale with the actual scores. However, the predictions plateau much sooner than the actual observed scores do. This shortcoming appears to be caused (at least in part) to the method's reliance on average values of IT, NT and robot performance. Future work should investigate this claim. Lastly, though not demonstrated here, this method can make vastly incorrect predictions under certain situations [3]. Some reasons for these failures are addressed in [10].

In closing our discussion on predictive power, we make the following observations. First, it appears that predictive tools that use measures from all three metric classes (IE, NI, and AAE) may be better at providing consistent predictions. Second, performance-based measures seem to be more desirable than time-based measures as they (a) appear to give more accurate predictions and (b) can predict more measures.

5.2.3 *Key Performance Parameters (KPPs)*

The third desirable element of a set of metric classes is that they contain KPPs. Obviously, more than one KPP can exist. However, in the interest of space we discuss just one KPP for this user study, which was the average time it took for a user to to locate a city on the map (part of IT). Several users in the study believed that their performance was driven by how quickly they could perform this primary task. Their claim seems to be somewhat valid as the average time it took a user to find a city on the map was negatively correlated ($r = -717$) with the users' score (from Equation 3). Thus, it appears that an effective way to improve these HRTs' overall effectiveness would be to provide the operator with additional aids in locating the city on the map (or, for a real world example, aids for identify a potential target in video imagery). Such aids could include automated target recognition assistance, etc.

6. DISCUSSION AND FUTURE WORK

We have advocated that sets of metric classes for human-robot teams be developed that indicate the limits of the agents in the team, provide predictive power, and contain key performance parameters. We presented a set of metric classes and analyzed it with respect to these three attributes. While sets of metrics drawn from this set of metric classes show limits of the agents in the team and contain KPPs, they fall short in the category of predictive power. Future

sets of metrics drawn from these classes and other metric classes should improve upon these results.

7. ACKNOWLEDGMENTS

This research was funded by MIT Lincoln Laboratory.

8. REFERENCES

[1] M. H. Ashcraft. *Cognition*. Prentice Hall, third edition, 2002.

[2] J. Brookshire, S. Singh, and R. Simmons. Preliminary results in sliding autonomy for assembly by coordinated teams. In *Proceedings of the International Conference on Robots and Systems*, 2004.

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

[4] M. L. Cummings and S. Guerlain. An interactive decision support tool for real-time in-flight replanning of autonomous vehicles. In *AIAA 3^{rd} "Unmanned Unlimited" Technical Conference, Workshop and Exhibit*, 2004.

[5] M. L. Cummings, C. Nehme, and J. W. Crandall. Predicting operator capacity for supervisory control of multiple UAVs. *Innovations in Intelligent UAVs: Theory and Applications, Ed. L. Jain*, 2006. In press.

[6] S. R. Dixon and C. D. Wickens. Unmanned aerial vehicle flight control: False alarms versus misses. In *Proceedings of the 12^{th} International Symposium on Aviation Psychology*, 2003.

[7] S. R. Dixon, C. D. Wickens, and D. Chang. Unmanned aerial vehicle flight control: False alarms versus misses. In *Proceedings of the Human Factors and Ergonomics Society 48th Annual Meeting*, 2004.

[8] J. Drury, J. Scholtz, and H. A. Yanco. Awareness in human-robot interactions. In *Proceedings of the IEEE Conference on Systems, Man and Cybernetics*, Washington, DC, 2003.

[9] M. R. Endsley. Design and evaluation for situation awareness enhancement. In *Proceedings of the Human Factors Society 32nd Annual Meeting*, Santa Monica, CA, 1988.

[10] M. A. Goodrich, T. W. McLain, J. W. Crandall, J. Johansen, J. Anderson, and J. Sun. Managing autonomy in robot teams: Observations from four experiments. In *Proceedings of the 2nd Annual Conference on Human-Robot Interaction*, 2007.

[11] M. A. Goodrich and D. R. Olsen. Seven principles of efficient human robot interaction. In *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics*, pages 3943–3948, Washington, DC, 2003.

[12] M. A. Goodrich, M. Quigley, and K. Cosenzo. Task switching and multi-robot teams. In *Proceedings of the Third International Multi-Robot Systems Workshop*, 2005.

[13] T. C. Hankins and G. F. Wilson. A comparison of heart rate, eye activity, eeg and subjective measures of pilot mental workload during flight. *Aviation, Space and Environmental Medicine*, 69(4):360–367, 1998.

[14] P. J. Mitchell, M. L. Cummings, and T. B. Sheridan. Mitigation of human supervisory control wait times through automation strategies. Technical report, Humans and Automation Laboratory, Massachusetts Institute of Technology, June 2003.

[15] D. R. Olsen and M. A. Goodrich. Metrics for evaluating human-robot interactions. In *NIST's Performance Metrics for Intelligent Systems Workshop*, Gaithersburg, MA, 2003.

[16] D. R. Olsen and S. B. Wood. Fan-out: Measuring human control of multiple robots. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI)*, 2004.

[17] R. Parasuraman, S. Galster, P. Squire, H. Furukawa, and C. Miller. A flexible delegation-type interfface enhances system performance in human supervison of multiple robots: Empirical studies with roboflag. *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans*, 35(4):481–493, 2005.

[18] R. Parasuraman, T. B. Sheridan, and C. D. Wickens. A model of types and levels of human interaction with automation. *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans*, 30(3):286–297, 2000.

[19] G. Rodriguez and C. R. Weisbin. A new method to evaluate human-robot system performance. *Autonomous Robots*, 14(2-3):165–178, 2003.

[20] H. A. Ruff, G. L. Calhoun, M. H. Draper, J. V. Fontejon, and B. J. Guilfoos. Exploring automation issues in supervisory control of multiple uavs. In *Proceedings of the Human Performance, Situation Awareness, and Automation Technology Conference*, pages 218–222, 2004.

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

[22] T. B. Sheridan and W. L. Verplank. Human and computer control of undersea teleoperators. Technical report, Man-Machine Laboratory, Massachusetts Institute of Technology, Cambridge, MA, 1978.

[23] P. Squire, G. Trafton, and R. Parasuraman. Human control of multiple unmanned vehicles: effects of interface type on execution and task switching times. In *Proceeding of the 1st Annual Conference on Human-robot Interaction*, pages 26–32, New York, NY, USA, 2006. ACM Press.

[24] A. Steinfeld, T. Fong, D. Kaber, M. Lewis, J. Scholtz, A. Schultz, and M. Goodrich. Common metrics for human-robot interaction. In *Proceedings of the 1st Annual Conference on Human-Robot Interaction*, 2006.

[25] J. A. Veltman and A. W. K. Gaillard. Physiological workload reactions to increasing levels of task difficulty. *Ergonomics*, 41(5):656–669, 1998.

[26] E. D. Visser, R. Parasuraman, A. Freedy, E. Freedy, and G. Weltman. A comprehensive methodology for assessing human-robot team performance for use in training and simulation. In *Proceeding of the Human Factors and Ergonomics Society 50th Annual Meeting*, 2006.

[27] C. Wickens and J. G. Hollands. *Engineering Psychology and Human Performance*. Prentice Hall, Upper Saddle River, NJ, third edition, 2000.

Appendix B

The following is a draft of a paper published in:

IEEE Transactions on Robotics, Vol. 23, No. 5, Oct 2007

Identifying Predictive Metrics for Supervisory Control of Multiple Robots

Jacob W. Crandall and M. L. Cummings

***Abstract*— In recent years, much research has focused on making possible single operator control of multiple robots. In these high workload situations, many questions arise including how many robots should be in the team, which autonomy levels should they employ, and when should these autonomy levels change? To answer these questions, sets of metric classes should be identified that capture these aspects of the human-robot team. Such a set of metric classes should have three properties. First, it should contain the key performance parameters of the system. Second, it should identify the limitations of the agents in the system. Third, it should have predictive power. In this paper, we decompose a human-robot team consisting of a single human and multiple robots in an effort to identify such a set of metric classes. We assess the ability of this set of metric classes to (a) predict the number of robots that should be in the team and (b) predict system effectiveness. We do so by comparing predictions with actual data from a user study, which is also described.**

***Index Terms*— Metrics, human-robot teams, supervisory control.**

I. Introduction

While most operational human-robot teams (HRTs) currently require multiple humans to control a single robot, much recent research has focused on a single operator controlling multiple robots. This transition is desirable in many contexts since it will (a) reduce costs, (b) extend human capabilities, and (c) improve system effectiveness. To achieve this goal, additional research must address many issues related to the human operator, the robots, and the interactions between them.

For HRTs consisting of a single operator and multiple robots to be effective, many questions must be answered, including: How many robots should there be in the team? What human-robot interaction methodologies are appropriate for the given human-robot team, mission, and circumstances? What autonomy levels should the robots in the team employ, and when should changes in these autonomy levels be made? What aspects of a system should be modified to increase the team's overall effectiveness?

To answer these questions, generalizable metrics should be identified that span the domain of HRTs [1]. Since metrics of system effectiveness vary widely across domains [2] and are typically multi-modal, it is unlikely that any one metric or set of metrics will suffice. However, a *set of metric classes* that spans the parts (and subparts) of HRTs is likely to be more generalizable. Loosely, a metric class is a set of metrics that measure the effectiveness of a certain aspect of a system. For example, we might consider the metric class of human performance, which includes metrics of reaction time, decision quality, situation awareness, workload, etc.

We claim that a set of metric classes can only answer the previously mentioned questions with high fidelity if it has three properties. A set of metric classes should (a) contain the *key performance parameters* (KPPs) of the HRT, (b) *identify the limits of the agents* in the team, and (c) have *predictive power*.

The first property states the need for metrics that are KPPs. A KPP is a measurable quantity that, while often only measuring a sub-portion of the system, indicates the team's overall effectiveness. Thus, the identification of KPPs helps determine what aspects of the system should be improved to cause the greatest increase in the system's overall effectiveness.

The second property states the need to measure the capacities and limits of both the human operator and the robots in the team. Identifying metrics with this property is necessary to answer questions dealing with the number of robots that should be in the team and what autonomy levels these robots should employ. Additionally, they help identify whether an interaction paradigm is acceptable to a human operator. Failures to adequately measure and identify these limits can lead to catastrophic consequences.

The third property states the need for metrics that have the ability to predict, or generalize, to other situations. Since measures of HRTs are typically only taken over specific conditions, they do not indicate how well a team will perform under untested conditions, many of which are likely to occur when the system is deployed. Conditions can vary in many ways, including variations in the mission type, changes in the environment in which the mission is performed, and variations in the make-up of the team (e.g., number of robots). Thus, without predictive metrics, an extremely large number of user studies must be conducted in order to assess the effectiveness of an HRT. Such a process is expensive, time consuming, and, ultimately, impossible. Thus, sets of metrics should be identified that can, from a small set of measured conditions, adequately estimate the performance characteristics of an HRT under unmeasured conditions.

A set of metrics that can predict a system's overall effectiveness under unmeasured conditions necessarily includes metrics that are KPPs, as well as metrics that demonstrate the limits of the agents in the team. Thus, in this paper, we focus on developing metrics with predictive power. Specifically, we will attempt to identify a set of metrics and their metric classes

Manuscript received October 15, 2006; revised June 7, 2007. This work was funded by MIT Lincoln Laboratory. This paper was recommended by the Guest Editors.

J. W. Crandall is a postdoctoral associate in the Department of Aeronautics & Astronautics at the Massachusetts Institute of Technology. M. L. Cummings is an assistant professor in the Department of Aeronautics & Astronautics at the Massachusetts Institute of Technology.

Digital Object Identifier

that can predict system effectiveness characteristics when the number of robots in the team changes.

The remainder of this paper will proceed as follows. In Section II, we review related work in the literature. In Section III, we decompose an HRT consisting of a single human operator and multiple robots. From this decomposition, we derive a set of metric classes. To validate the usefulness of this set of metric classes, we performed a user study involving multiple simulated robots. We describe the design of the user study in Section IV. In Section V, we present results from the study. Based on measures obtained from this study, we construct predictive tools for various system effectiveness measures. We present these results in Section VI.

While HRTs of the future will include heterogeneous sets of robots, we focus in this paper on the homogeneous case. However, the principles and theories discussed in this paper also apply to heterogeneous robot teams, though additional issues will need to be considered for those teams. We also assume that (a) the robots are remotely located from the operator, and (b) the robots perform independent tasks.

II. Background and Related Work

We now review related work and give relevant definitions.

A. Related Work

The work of this paper relies on and contributes to many topics throughout the literature on human-robot teams. We focus on four topics: supervisory control of multiple robots, Fan-out, metrics for human-robot teams, and adjustable autonomy.

1) Supervisory Control of Multiple Robots: When a human operator supervises multiple robots, care must be taken to ensure that the operator has the capacity to perform all of her/his tasks. Adherence to multiple principles are required to make this possible, including offloading low-level control of the robots to automation [3], [4], [5], [6], ensuring that the automation is reliable [7], and improving interface technologies (e.g. [8], [9]). Predictive metrics provide a means to evaluate these technologies in a cost-effective manner.

When a human controls multiple robots, (s)he must necessarily determine how to allocate his/her attention between the various robots or groups of robots. This is related to the concept of time-sharing of cognitive resources (see [2], [10]). Time-sharing capabilities can be measured by metrics in the *attention allocation efficiency* metric class, which we discuss in the next section.

2) Fan-out: The term Fan-out (FO) refers to the number of (homogeneous) robots that a single operator can effectively control [11]. One line of research on this topic estimates FO using measures of interaction time and neglect time [12], [11]. These metrics have been modified to include the use of wait times [13], [14]. We analyze how effectively these metrics estimate the observed FO in Section VI-A.

3) Metrics for Human-Robot Teams: Much of the work on metrics in HRTs has focused on the human operator. The most common of these metrics are metrics of operator workload and situation awareness (SA). Metrics for measuring operator workload include subjective methods [2], secondary task methods (e.g. [15]), and psychophysiological methods (e.g., [16], [17]). Operator workload is critical in determining operator capacity thresholds [4]. SA, defined formally in [18], is deemed to be critical to human performance in HRTs. Efforts to formalize SA for the human-robot domain include the work of Drury et al. [19], [20]. Despite its popularity, measuring SA effectively in an objective, non-intrusive manner remains an open question, though note [21].

In this paper, we combine metrics from various aspects of the HRT to obtain measures of system effectiveness. This is related to the work of Rodriguez and Weisbin [22], who compute a measure of system effectiveness from measures of the individual subtasks. However, their approach does not address supervisory control of multiple robots.

4) Adjustable Autonomy: Central to the success of an HRT is the level of automation employed by the robots in the team. Sheridan and Verplank's [23] general scale of levels of automation has been widely accepted and adapted for use in system design (e.g., [24], [25]). A system's level of automation need not be static. Due to dynamic changes in operator workload and task complexities, appropriate variations in the level of automation employed by the system are often desirable (e.g., [6], [26]). We believe that predictive metrics such as those discussed in this paper can assist in creating HRTs that use adjustable autonomy more effectively.

B. Definitions

Throughout this paper, we refer to metrics, metric structures, and metric classes. A *metric class* is a set of metrics and metric structures that can be used to measure the effectiveness of a particular system or subsystem. A *metric structure* denotes a mathematical process or distribution that dictates performance characteristics of measurements from within that class. Each metric class has at least one metric structure. For brevity, we often refer to metric structures as metrics.

III. A Set of Metric Classes

In this section, we identify a set of metric classes by decomposing an HRT consisting of a single human operator and multiple (remote) robots. We first decompose an HRT consisting of a human operator and a single (remote) robot. We then consider the multi-robot case.

A. The Single-Robot Case

An HRT consisting of a single robot has the two control loops shown in Fig. 1, which is adapted from [12]. These control loops are the control loops of supervisory control defined in [27]. The upper loop shows the human's interactions with the robot. The robot sends information about its status and surroundings to the human via the interface. The human synthesizes the information and provides the robot with input via the control element of the interface. The lower control-loop depicts the robot's interactions with the world. The robot combines the operator's input with information it gathers from its sensors, and then acts on the world using its actuators.

The two control loops provide a natural decomposition of an HRT with a single robot into two parts. Each part defines a

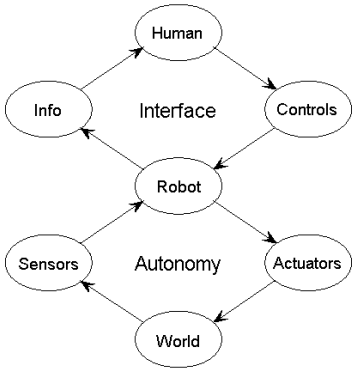


Fig. 1. The two control loops of an HRT consisting of a single human operator and a single (remote) robot. Adapted from [12].

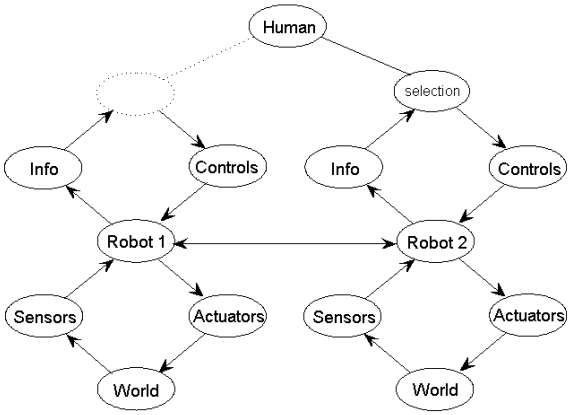


Fig. 2. In HRTs consisting of a single human and multiple robots, the human must determine how to distribute his/her attention between the robots.

metric class. Corresponding to the top control loop are metrics that describe the effectiveness of human-robot interactions. These metrics are members of the *interaction efficiency (IE)* metric class. Corresponding to the bottom control loop are metrics that describe the effectiveness of a single robot when it is ignored by the operator. These metrics are members of the *neglect efficiency (NE)* metric class. However, while these two metric classes are separate, they are not independent from each other. A failure in one control loop is likely to cause a failure in the other control loop.

We now discuss a few metrics in each class.

1) Interaction Efficiency (IE): The IE metric class includes several metrics that have been discussed in the literature. One such metric is *interaction time (IT)*, which (for the single robot case) is the amount of time needed for the operator to (a) orient to the robot's situation, (b) determine the inputs (s)he should give to the robot, and (c) express those inputs to the robot via the interface [28]. Measuring IT can be difficult since doing so requires knowledge of what the operator is thinking. Efforts to estimate IT include [11], [12].

Using IT to capture IE infers that shorter interactions are more efficient than longer ones. Since this is not always the case, we might consider metrics that more fully measure the performance benefits of an interaction. Such metrics can be derived from the metric structure *interaction impact* ($II(t)$), which is the random process that describes a single robot's performance on a particular task as a human interacts with it. This random process is a function of (among other things) *operator time-on-task* t, which is the amount of time since the operator began interacting with the robot. Additional discussion of II can be found in [12]. One metric derived from II is the robot's average performance during interactions:

$$\bar{II} = \frac{1}{IT}\int_0^{IT} E[II(t)]dt, \tag{1}$$

where $E[II(t)]$ denotes the expected value of $II(t)$.

Other metrics in the IE class include wait times during interactions (WTIs) [13] and the operator's SA with respect to that particular robot (SAr).

2) Neglect Efficiency (NE): The NE metric class consists of metrics that describe the robot's performance when the human's attention is turned elsewhere. *Neglect time* (NT), the average amount of time a robot can be ignored by the operator before its expected performance falls below a certain threshold [28], is a member of this metric class. Like IT, NT does not completely account for the robot's performance. This additional information can be obtained from the metric structure *neglect impact* NI, which is the random process that describes a single robot's performances when it is ignored by the operator. Additional information on NI can be found in [12]. From NI, we can calculate the average performance of the robot when it is neglected:

$$\bar{NI} = \frac{1}{NT}\int_0^{NT} E[NI(t)]dt, \tag{2}$$

where $E[NI(t)]$ denotes the expected value of $NI(t)$.

B. The Multi-Robot Case

When a human interacts with multiple robots, the nature of each human-robot interaction is similar to the single-robot case with the important exception depicted in Fig. 2. The figure shows two separate sets of control loops, one for each robot. However, unlike the single-robot case, the upper loop for each robot is not always closed. To close the loop, the human must attend to the corresponding robot and neglect the others. Thus, critical to the system's effectiveness is the efficiency with which the human allocates his/her time between the robots. Metrics that seek to capture this efficiency have membership in the *attention allocation efficiency (AAE)* metric class.

1) Attention Allocation Efficiency (AAE): Several metrics in the AAE metric class have been studied in the literature. These metrics include SA of the entire HRT (denoted SAg, for *g*lobal SA, to distinguish it from SAr), wait times due to loss of SA (WTSA) (times in which a robot is in a degraded performance state due to a lack of operator SA [13]), and switching times (STs) (the amount of time it takes for the operator to decide which robot to interact with). Additional metrics with membership in AAE can be determined from estimates of the operator's robot selection strategy SS (a metric structure).

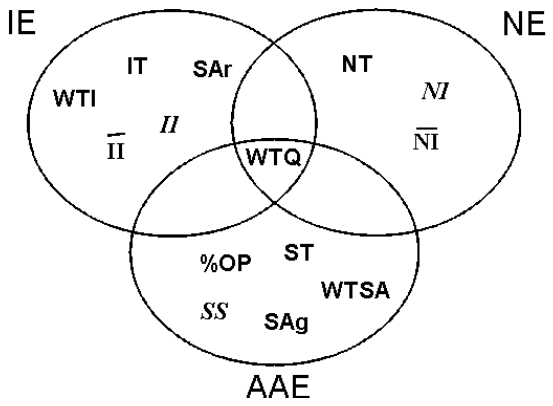


Fig. 3. A set of metric classes ($\{IE, NE, AAE\}$) and various metrics drawn from those classes.

One such metric could be the probability that an operator's selection corresponds to the optimal policy (i.e., selection strategy). We denote this metric as $\%OP$ (percent optimal policy). We note that the optimal policy might ultimately be impossible to know, though it can be approximated in some domains using the metric structures II and NI (via dynamic programming or some other optimization technique).

Fig. 2 also shows a connecting link between robots in the team. This link captures the notion that interactions between robots can have a significant impact on the team. This impact could be made manifest in measures of IE, NE, and AAE, or it could potentially be defined by a fourth metric class. However, when robots perform independent tasks (as we assume in this paper), this link has no effect on the behavior of the team.

C. Summary of Set of Metric Classes

The set of metric classes we have discussed is summarized by Fig. 3. Note the intentional overlap of the metric classes as some metrics span multiple classes. For example, the metric WTQ (wait times in the queue [13]) is a metric dependent on the interplay between all three metric classes.

IV. A Case Study

We conducted a user study to evaluate the predictive power of sets of metrics drawn from the previously described set of metric classes. The user study was performed using a software test-bed designed to capture the abstract tasks performed by HRTs. In this section, we describe the software test-bed and the experimental procedure of the user study.

A. Software Test-bed

We describe the software test-bed in three parts: the HRT's mission, the human-robot interface, and the robots' behaviors.

1) Mission: Across many mission types, an HRT operator commonly assists in performing a set of abstract tasks. These abstract tasks include mission planning and re-planning, robot path planning and re-planning, robot monitoring, sensor analysis and scanning, and target designation. Each of these tasks can be performed using various levels of automation [23].

In designing this test-bed, we sought to capture each of these tasks in a time-critical situation. The HRT (which consisted of the participant and multiple simulated robots) was assigned the task of removing as many objects as possible from the maze in an 8-minute time period. At the end of 8-minutes, the maze "blew up," destroying all robots and objects that remained in it. Thus, in addition to collecting as many objects as possible, users needed to ensure that all robots were out of the maze when time expired.

An object was removed from the maze (i.e., collected) using a three-step process. First, a robot moved to the location of the object (i.e., target designation, mission planning, path planning, and robot monitoring). Second, the robot "picked up" the object (i.e., sensor analysis and scanning). In the real world, performing such an action might require the human operator to assist in identifying the object with video or laser data. To simulate this task, we asked users to identify a city on a map of the mainland United States using $Google^{\mathrm{TM}}$ $Earth$-style software. Third, the robot carried the object out of the maze via one of two exits.

The mission also had the follow details:

- At the beginning of the session, the robots were positioned outside of the maze next to one of two entrances.
- The form of the maze was initially unknown. As each robot moved in the maze, it created a map which it shared with the participant and the other robots.
- The objects were randomly spread through the maze. The HRT could only see the positions of six of the objects initially. In each minute of the session, the locations of two additional objects were shown. Thus, there were 22 possible objects to collect during a session.
- The participant was asked to maximize the following objective function:

$$Score = ObjectsCollected - RobotsLost, \qquad (3)$$

where $ObjectsCollected$ was the number of objects removed from the area during the session and $RobotsLost$ was the number of robots remaining in the area when time expired.

2) Interface: The human-robot interface was the two-screen display shown in Fig. 4. On the left screen, the map of the maze was displayed, along with the positions of the robots and (known) objects in the maze. The right screen was used to locate the cities.

The participant could only control one robot at a time. When a user desired to control a certain robot, (s)he clicked a button on the interface corresponding to that robot (labeled UV1, UV2, etc.). Once the participant selected the robot, (s)he could direct the robot by designating a goal location and modifying the robot's intended path to that goal. Designating a goal for the robot was done by dragging the goal icon corresponding to the robot in question to the desired location. Once the robot received a goal command, it generated and displayed the path it intended to follow. The participant was allowed to modify this path using the mouse.

To assist the operator in determining which robots needed input, warning indicators related to a particular robot were

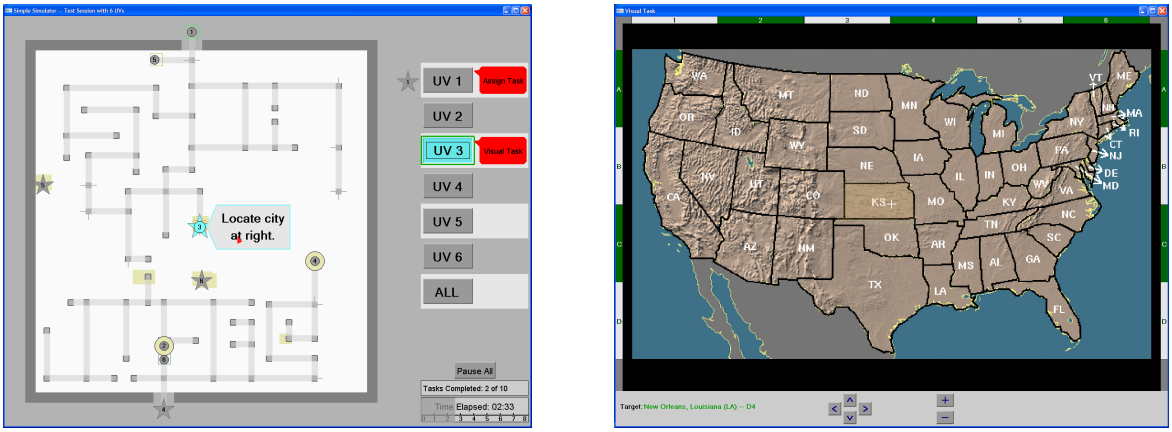


Fig. 4. Two displays used in the experiment (on separate monitors). The display at left shows the map of the maze, the locations of the robots, their destinations, the locations of the objects to be gathered, etc. The display at right shows the interface for the abstract visual task.

displayed next to its corresponding button. There were four kinds of warning indicators:

- Assign Task Indicator – The robot had reached its goal position.
- Visual Task Indicator – The robot had reached the object it was designated to collect.
- Time Warning – In the last minute of a session, the robot was still in the maze and had not been told to leave it.
- Deliver Object Indicator – The robot was carrying an object, but had not been told to leave the maze.

If no status or warning was reported, the system determined that the robot was satisfactorily progressing on its task.

3) Robot Behavior: The robots' map of the maze took the form of an undirected graph. Each edge of the graph was an ordered pair (u, v) representing a connection between vertices u and v in the graph. Associated with each edge was a weight indicating the cost for a robot to move along that edge. Since the maze was not fully known, a robot had to choose between (a) moving along the shortest path of the known maze to its user-specified goal and (b) exploring the unknown portions of the maze in hopes of finding a shorter path. To make this decision, a robot assumed that an unmapped edge from a known vertex v led directly to the goal position with a cost equal to the Manhattan distance from v to the robot's goal, plus some cost of exploration (C_E). The robot used Dijkstra's algorithm on the resulting graph to determine the path it intended to follow.

Using this approach, the constant C_E determines the degree to which the robots explore the unknown maze. Higher values of C_E result in less exploration. We used a small value of C_E for a robot that was searching for an object, and a higher value for a robot that was carrying an object. Since users sometimes felt that the resulting behavior was undesirable, they were allowed to modify a robot's path if they desired.

B. Experimental Procedure

Following training on all functions of the system and after completing a comprehensive practice session, each user participated in six eight-minute sessions. In each of the first four sessions, a different number of robots (2, 4, 6, or 8) were allocated to the team. In the last two sessions, the experimental conditions (i.e., the robot team size) of the first two session were repeated. The conditions of the study were counterbalanced and randomized. The participants were paid $10 per hour; the highest scorer also received a $100 gift certificate.

Twelve people (one professor, ten students, and one other person from the community) between the ages of 19 and 44 years old (mean of 27.5) participated in the study. Of these twelve participants, eight were U.S. citizens, two were Canadian, one was Hispanic, and one was Egyptian. Three of the participants were female and nine were male.

C. Note on Simulation

While simulated environments make it possible to evaluate metric technologies in a cost-effective manner, simulated robots often behave differently than real robots. For example, our simulated robots have errorless localization capabilities, but real robots typically do not. Thus, measures of system performance characteristics of a human, real-robot team will be different than those of a human, simulated-robot team (see, for example, [12]). However, in both situations, we believe that measures of AAE, IE, and NE are necessary to (a) thoroughly evaluate the effectiveness of the HRT and (b) predict how the HRT will behave in unmeasured conditions. All of the metrics and metric classes we discuss in this paper can be used to measure the performance of HRTs with both simulated and real robots. Thus, while the results of this user study do not generalize to HRTs with real robots, they are a demonstration of the usefulness of these proposed metric classes.

V. Results – Empirical Observations

The user study allows us to address two distinct questions related to the HRT in question. First, how does the number of robots in the team affect the system's effectiveness? Second, how does the number of robots in the team affect measures drawn from the IE, NE, and AAE metric classes?

A. System Effectiveness Measures

The dependent variables we consider for system effectiveness are those related to Eq. (3): the number of objects

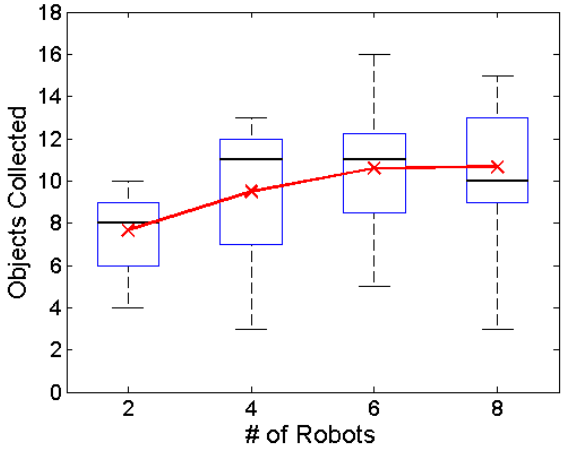


Fig. 5. Means and distributions of number of objects collected for each robot team size.

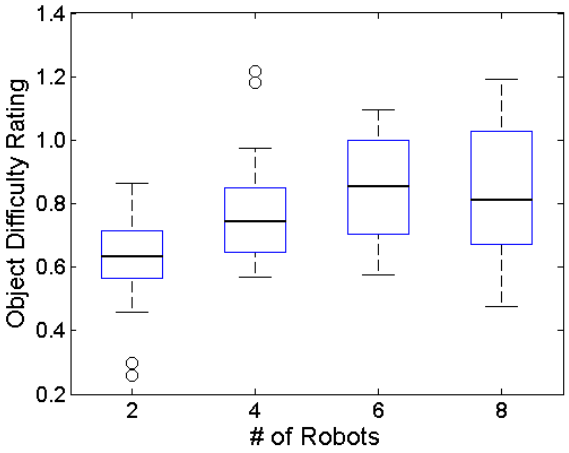


Fig. 6. Box plot showing difficulty of the objects collected under each robot team size.

collected by the HRT over the course of a scenario and the number of robots lost during a scenario. We analyze each variable separately.

1) Objects Collected: Fig. 5 shows the means and distributions of number of objects collected for each robot team size. The figure shows that the number of objects collected steadily increases as the number of robots in the team increases up to six robots, at which point effectiveness plateaus. A repeated measure ANOVA revealed a statistically significant difference in number of objects collected across team sizes, $\alpha = 0.05$ ($F(3, 15) = 24.44$, $p < 0.001$). Pairwise comparisons show that 2-robot teams collected significantly less objects than did 4-, 6-, and 8-robot teams ($p \leq 0.001$), and 4-robot teams collected less objects than 6-robot teams (marginal statistical significance; $p = 0.057$) and 8-robot teams ($p = 0.035$).

Teams with six and eight robots collected only about 3 more objects than teams with two robots. This relatively small performance increase appears to be a bit deceiving, since objects were weighted equally, regardless of how far into the maze a robot had to travel to reach them. While both smaller and larger robots teams collected the objects closest to the exits, larger teams tended to collect more objects that were deeper in the maze. This trend is illustrated by Fig. 6, which shows the distributions of average (for each session) *object difficulty weightings* of the collected objects for each team size. Formally, each object i's difficulty weight (denoted w_i) was defined by $w_i = \frac{d_i}{E[d_i]}$, where d_i was the shortest path from the object to one of the two maze exits and $E[d_i]$ is the average distance from an exit to an object. Thus, an average difficulty weight (w_i) was equal to one, and objects with lower weights were generally easier to collect. Thus, the difference between the amount of work done by larger and smaller robot teams is greater than Fig. 5 seems to indicate.

2) Robots Lost: Robots were lost if they were still in the maze when time expired. Operators failed to help robots leave the area for a number of reasons, including incorrectly estimating the speed at which the robots moved, underestimating the amount of time it took to locate a city on the map, and employing too many robots toward the end of the session.

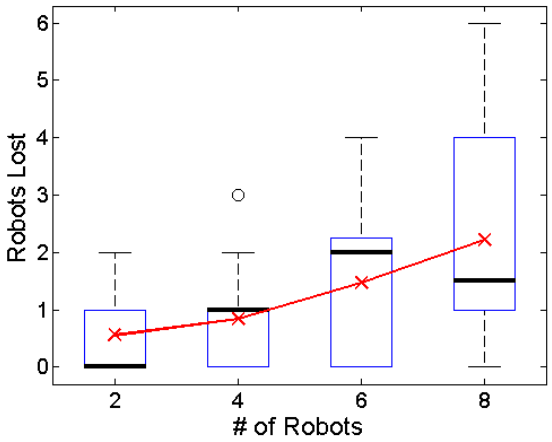


Fig. 7. Means and distributions of number of robots lost for each robot team size.

Fig. 7 shows the number of robots lost for each team size. A clear, statistically significant, distinction exists between groupings of 2- and 4-robot teams and 6- and 8-robot teams ($\chi^2 = 13.71, df = 6, p = 0.033$). This result indicates a performance drop between four and six robots. Thus, while robot teams with six and eight robots collected more objects than smaller robot teams, they also lost more robots.

These results show that the HRTs in the user study with the highest effectiveness had, on average, between four and six robots. The "optimal" robot team size depends on the ratio between the values of the objects and the robots.

B. Effects of Team Size on Measurements of IE, NE, and AAE

In this section, we discuss how metrics from the three metric classes varied across conditions (i.e., numbers of robots). We begin with the IE metric class.

1) Effects on Interaction Efficiency: For the IE metric class, we consider interaction time IT. Distributions of ITs are shown in Fig. 8. A repeated measures ANOVA shows a statistical difference between ITs for different robot team sizes

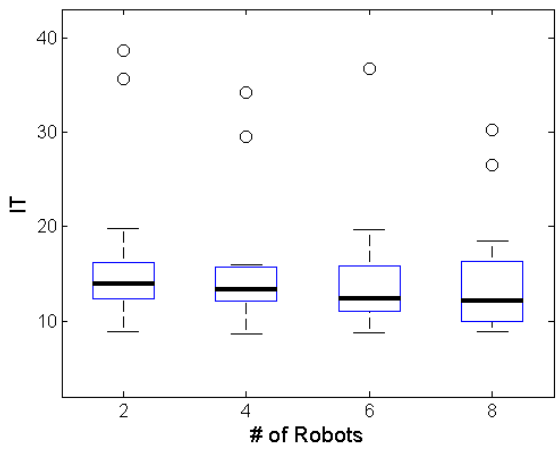


Fig. 8. Distributions of interaction times for different team sizes.

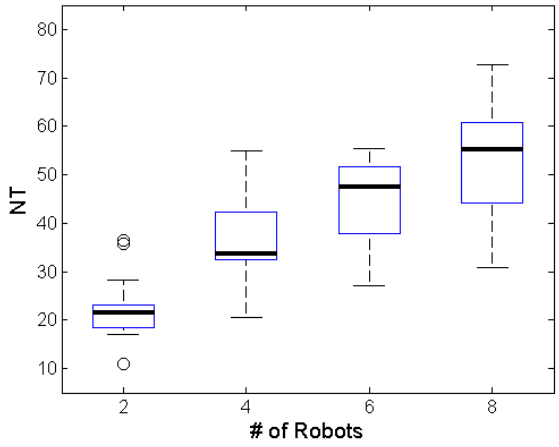


Fig. 9. Distributions of neglect times for different team sizes.

$(F(3,15) = 3.29, p = 0.049)$. Average IT was slightly shorter for larger team sizes, though the difference was relatively small (just 2.34 second difference between 2- and 8-robot teams). Thus, robot team size had little impact on IT.

2) Effects on Neglect Efficiency: As an indicator of the NE metric class, we consider neglect time NT. For this user study, we calculated NT as the time between when the operator finished servicing a robot until the time that either (a) the robot arrived at its goal, or (b) the operator again decided to service that robot. Distributions of NTs are shown in Fig. 9.

Measures of NT differed significantly and drastically across team sizes $(F(3,15) = 47.21, p < 0.001)$. This trend can be attributed to two different reasons. First, in the conditions with less robots, operators had less to do. As such, they tended to micro-manage the robots, changing the robots' goals and routes when they appeared to behave erratically. This meant that the users' decisions to interact often ended the neglect period prematurely. On the other hand, when operators had more to do (with larger robot teams), they tended to focus less on local robot movements and more on global control strategies. Thus, neglect periods were longer since they often lasted until the robot reached its goal. A second reason that

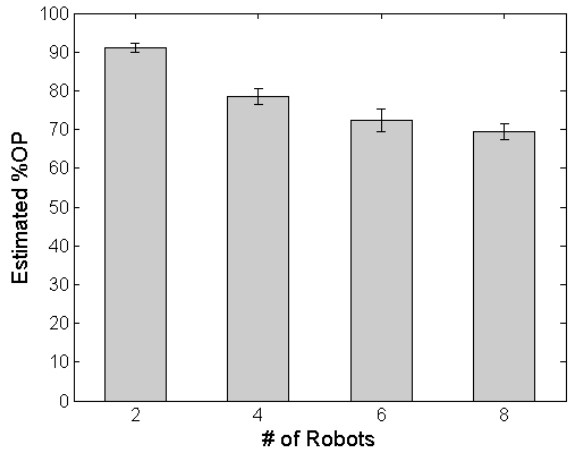


Fig. 10. Estimated percentage of optimal robot selections by the operators.

NT was higher for larger robot teams is due to differences in the distances robots traveled to reach their goals (Fig. 6). In larger teams, it took robots longer to reach their goals since they were assigned goals deeper in the maze.

3) Effects on Attention Allocation Efficiency: As an indicator of AAE, we use an estimate of $\%OP$. Recall from Section III that $\%OP$ is the percentage of time the operator serviced the "right" robot. Via a discrete event simulation, models of robotic behavior in the presence and absence of human attention (i.e., II and NI, respectively) can be used to estimate how various robot selection strategies would affect the system's effectiveness. In this way, we can estimate the (near) optimal robot selection strategies and then compare these strategies with actual operator selections to determine $\%OP$. The resulting estimates of $\%OP$ from our user study are shown in Fig. 10. The figure shows that the users' ability to determine which robot should be serviced decreased as the number of robots in the team increased.

VI. Results – Predictive Power

We now turn to the task of extrapolating measures from a single observed condition to unmeasured conditions. We assume that we can observe the system in only a single condition, which we refer to as the *measured condition*. Thus, we must predict measures for the other desired conditions (the *unmeasured conditions*) based on the measurements from the measured condition. In this case, we seek to make predictions for different robot team sizes.

The effectiveness of a predictive metric is determined by two attributes: *accuracy* and *consistency*. Accuracy refers to how close the predictions are to reality. Consistency refers to the degree to which the metric predicts the same quantity from different measured conditions. For example, a consistent prediction algorithm would predict the same quantity for a particular robot team size regardless of the whether the measured condition had two or four robots.

In this paper, we consider predicting two different system characteristics: FO and overall system effectiveness.

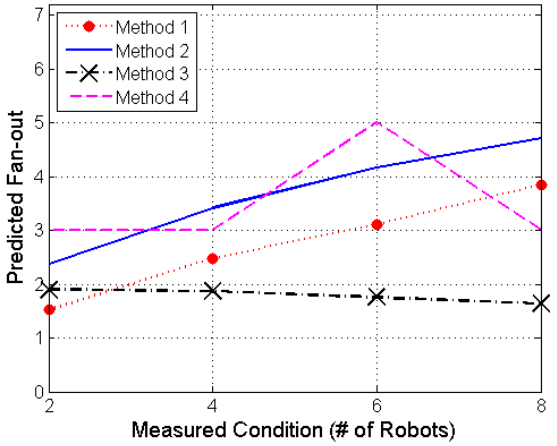


Fig. 11. Fan-out predictions of four different methods for four measured conditions (x-axis).

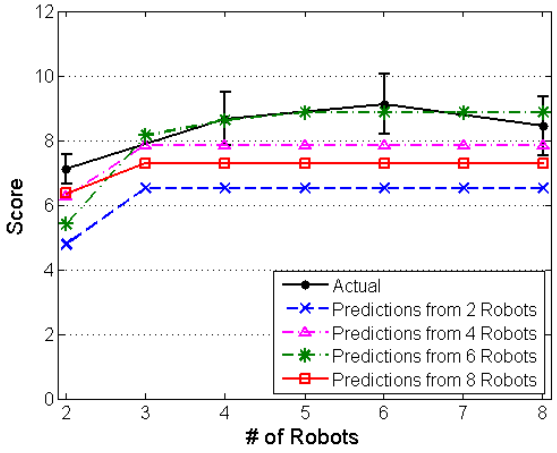


Fig. 12. Predictions of overall system effectiveness using method 4 [12]. *Actual* refers to the mean (and standard error) of observed scores in the user study and *Predictions from N Robots* shows the predictions (for all team sizes shown along the x-axis) from the N-robot measured condition.

A. Predicting Fan-out

Predicting FO consists of predicting the point at which the system's effectiveness peaks or plateaus [11]. We consider four methods for predicting FO found in the literature. The FO predictions made by each method for each measured condition are shown in Fig. 11. In the figure, the x-axis designates the measured condition (i.e., robot team size), and the y-axis gives the corresponding estimate of FO. Recall that we observed in Section V-A that FO was between four and six robots. We discuss the results from each predictive method in turn.

1) Method 1: This method, described in [11], predicts FO to be the average number of robots that are active (called activity time). Thus, this measure does not consider whether or not a robot is gainfully employed, but just if it is doing something. The method relies on the assumption that the operator has as many robots at his/her disposal as (s)he desires. When this assumption does not hold, the prediction fails, as demonstrated in Fig. 11. The figure shows that the estimate of FO increases as the number of robots in the measured condition increases. Thus, this predictive method is not *consistent*. It does, however, make a reasonable estimate of $FO \approx 4$ from the 8-robot measured condition.

2) Method 2: Olsen and Goodrich [29], [28] proposed that FO could be estimated using the equation

$$FO = \frac{NT}{IT} + 1. \tag{4}$$

Thus, this method uses metrics drawn from the IE and NE metric classes, but not AAE. To obtain predictions using this method, we estimated IT and NT as discussed in the previous section. The resulting FO predictions are shown in Fig. 11. Like method 1, these predictions increase nearly linearly with the number of robots in the measured condition. Thus, this method also fails to be consistent in this case (due to variations in measures of NT for different team sizes). The FO predictions from the 6- and 8-robot conditions, however, do fall into the range of 4-6 robots. Thus, like method 1, this second method might require that measures be extracted from measured conditions with many robots to be accurate.

3) Method 3: Cummings et al. [13] modified Eq. (4) to include wait times (WT). Thus, this method considers metrics from all three metric classes discussed in Section III. The resulting FO equation is

$$FO = \frac{NT}{IT + WT} + 1. \tag{5}$$

Fig. 11 shows that the resulting predictions are relatively consistent, though they are lower than the observed FO. At least in this case, the inclusion of wait times counteracts variations in NT. This makes an argument that predictive tools should use metrics from IE, NE, and AAE.

4) Method 4: The previous methods we considered used temporal-based measures to estimate FO. The fourth method, described in [12], considers both temporal and performance-based measures, including IT, $\bar{I}I$, and $\bar{N}I$ (see Eqs. (1) and (2)), but no measure of AAE. Using these quantities (determined from the measured condition), it estimates the system's effectiveness for each potential robot team size (Fig. 12) and then reports FO as the point at which performance is maximized. Fig. 11 shows the resulting predictions. From the 2-, 4-, and 8-robot measured conditions, this method predicts that $FO = 3$. From the 6-robot condition, it estimates $FO = 5$. Thus, this method has a semblance of consistency, though its predictions still vary and tend to be pessimistic.

5) Summary: None of the methods we analyzed consistently predicts the observed FO (between four and six robots). Methods 1 and 2 appear to require that the measured condition include many robots. Method 3's predictions were consistent, though low, suggesting that using metrics from all three metric classes are needed for robust predictive power. Method 4 made, perhaps, the closest predictions on average, though its predictions are also low and lacked some consistency. Thus, while each of these metrics might have descriptive power, they are unable to consistently predict the observed FO.

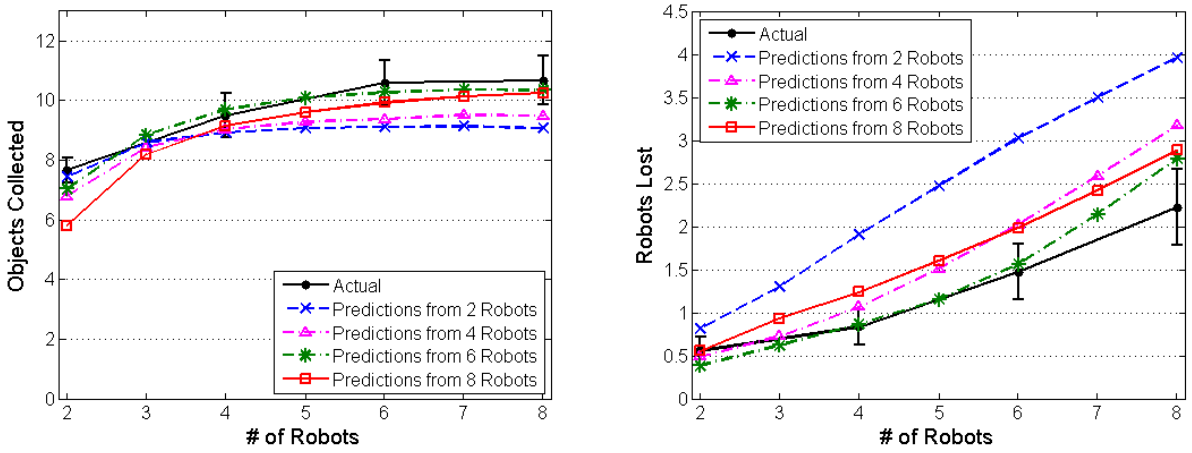


Fig. 13. Predictions of objects collected (left) and robots lost (right) compared to the sample means obtained in the user study. *Actual* refers to the mean (and standard error) of observed scores in the user study and *Predictions from N Robots* shows the predictions (for all team sizes shown along the x-axis) from the N-robot measured condition. Each prediction is the average of 10,000 samples.

B. Predicting System Effectiveness

Method 4 was designed to predict an HRT's overall effectiveness [12]. Such predictions for the HRTs discussed in this paper are shown in Fig. 12. The figure shows four sets of predictions of HRT scores (Eq. (3)). Each set of predictions estimates the HRT's score for all team sizes (the x-axis) for a single measured condition (specified in the legend). The figure also shows the actual average scores (labeled *Actual*) in the user study for each team size.

The general trend of each set of predictions in Fig. 12 is similar to the actual average scores from the users study, especially those predictions made from the 6-robot measured condition. However, a few noticeable differences between the predictions and actual results are present. First, this method assumes that predictions plateau once performance peaks, which may not be the case, as it appears that HRTs with more than six robots have degraded scores. To predict such a trend, it is likely that a predictive algorithm must use measures of AAE. Second, as was shown in the previous subsection, this method predicts that overall effectiveness peaks sooner (i.e., with smaller team sizes) than it actually does. This seems to be due to the reliance of the algorithm on the means of the random processes and temporal variables rather than the complete distributions. Third, Fig. 12 shows that this predictive method is not as consistent as it otherwise might be.

We sought to improve these results by creating a new predictive tool. This predictive tool uses stochastic metric structures from each of the metric classes. As in method 4, II and NI are modeled from data gathered from the measured condition (i.e., robot team size) in the user study. Models of SS (the operator's strategy for choosing which robots to service) and ST (the amount of time it takes the operator to select a robot) are also constructed from this data in order to represent metrics from the AAE metric class. If we assume that these metric structures describe how the human operator and each robot in the team would behave for each robot team size, we can run a discrete event simulation using these models for different robot team sizes to estimate how the number of robots in the team will affect system effectiveness.

The average (out of 10,000 data samples) predictions generated by the discrete event simulations are shown in Fig. 13. On the left are predictions of number of objects collected, and on the right are predictions of number of robots lost. The predictions give reasonably accurate estimates of the conditions from which the metrics were modeled, especially for objects collected. For example, from the 2-robot measured condition, predictions of the number of objects collected for 2-robot teams are within the standard error of the actual mean value. This result is important, as it suggests a certain robustness in the set of metric structures used to obtain the predictions. We note, however, that the predictions tend to be slightly pessimistic, as they tend to estimate that the HRTs would collect slightly less objects and lose slightly more robots than they actually did.

The predictions also follow the trend of the actual observed results. However, predictions tend to be less accurate when the distance between the team size in the measured condition and the team size for which we want to make estimates is high. This is particularly true of predictions made from the 2-robot measured condition. This is likely caused by a number of issues, not the least of which is that, like NT and $\%OP$ (Fig. 9), NI and SS vary depending on the number of robots in the measured condition. Predicting how these metrics change would allow for more accurate predictions. This could potentially be achieved by using multiple measurement conditions, though this would require larger user studies.

VII. Summary and Future Work

The goal of this research is to identify sets of metrics that (a) have predictive power, (b) identify the limits of the agents in the team, and (c) are KPPs. In this paper, we focused on constructing predictive metrics from a particular set of metric classes, which we identified by decomposing a human-robot team consisting of a single human and multiple robots. We assessed the ability of predictive algorithms to predict Fan-out and overall system effectiveness by conducting a user study in

which participants controlled multiple simulated robots. From the data collected in this study, we constructed models of human and robotic behavior. We then used those models to estimate Fan-out and system effectiveness in unmeasured conditions. We compared these predictions to the actual results.

Though these results are encouraging, future work is needed. Improvements should be made to the metrics discussed in this paper, and other important metrics and metric classes should be identified. Future work should also consider extrapolating predictions from multiple measured conditions rather than a single condition in order to obtain more robust predictions. Other future research directions in this area should address HRTs with multiple human operators and robots that perform dependent tasks.

REFERENCES

[1] A. Steinfeld, T. Fong, D. Kaber, M. Lewis, J. Scholtz, A. Schultz, and M. Goodrich, "Common metrics for human-robot interaction," in *Proc. of the ACM/IEEE Int. Conf. on Human-Robot Interaction*, Salt Lake City, UT, Mar. 2006, pp. 33–40.

[2] C. Wickens and J. G. Hollands, *Engineering Psychology and Human Performance*, 3rd ed. Upper Saddle River, NJ: Prentice Hall, 2000.

[3] M. L. Cummings and P. J. Mitchell, "Operator scheduling strategies in supervisory control of multiple UAVs," *Aerospace Science and Technology*, 2007.

[4] M. L. Cummings and S. Guerlain, "An interactive decision support tool for real-time in-flight replanning of autonomous vehicles," in *AIAA 3^{rd} "Unmanned Unlimited" Technical Conference, Workshop and Exhibit*, 2004.

[5] H. A. Ruff, S. Narayanan, and M. H. Draper, "Human interaction with levels of automation and decision-aid fidelity in the supervisory control of multiple simulated unmanned air vehicles," *Presence*, vol. 11, no. 4, pp. 335–351, Aug. 2002.

[6] R. Parasuraman, S. Galster, P. Squire, H. Furukawa, and C. Miller, "A flexible delegation-type interfface enhances system performance in human supervison of multiple robots: Empirical studies with roboflag," *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans*, vol. 35, no. 4, pp. 481–493, Jul. 2005.

[7] S. R. Dixon, C. D. Wickens, and D. Chang, "Unmanned aerial vehicle flight control: False alarms versus misses," in *Proc. of the Human Factors and Ergonomics Society 48th Annual Meeting*, New Orleans, LA, Sep. 2004.

[8] C. D. Wickens, S. Dixon, and D. Chang, "Using interference models to predict performance in a multiple-task UAV environment," Aviation Human Factors Division, Institute of Aviation, University of Illinois at Urbanna-Champaign, Tech. Rep. AHFD-03-9/MAAD-03-1, Apr. 2003.

[9] M. Quigley, M. A. Goodrich, and R. W. Beard, "Semi-autonomous human-UAV interfaces for fixed-wing mini-UAVs," in *Proc. of the Int. Conf. on Intelligent Robots and Systems*, Sendai, Japan, Sep. 2004, pp. 2457–2462.

[10] M. H. Ashcraft, *Cognition*, 3rd ed. Prentice Hall, 2002.

[11] D. R. Olsen and S. B. Wood, "Fan-out: Measuring human control of multiple robots," in *Proc. of the SIGCHI Conf. on Human Factors in Computing Systems*, Vienna, Austria, Apr. 2004, pp. 231–238.

[12] J. W. Crandall, M. A. Goodrich, D. R. O. Jr., and C. W. Nielsen, "Validating human-robot systems in multi-tasking environments," *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans*, vol. 35, no. 4, pp. 438–449, Jul. 2005.

[13] M. L. Cummings and P. J. Mitchell, "Predicting controller capacity in remote supervision of multiple unmanned vehicles," *IEEE Transactions on Systems, Man, and Cybernetics – Part A Systems and Humans*, 2007, In press.

[14] M. L. Cummings, C. Nehme, and J. W. Crandall, "Predicting operator capacity for supervisory control of multiple UAVs," *Innovations in Intelligent UAVs: Theory and Applications,* Ed. L. Jain, 2007. In press.

[15] M. L. Cummings and S. Guerlain, "Using a chat interface as an embedded secondary task tool," in *Proc. of the 2nd Annual Conf. on Human Performance, Situation Awareness and Automation*, Mar. 2004.

[16] T. C. Hankins and G. F. Wilson, "A comparison of heart rate, eye activity, eeg and subjective measures of pilot mental workload during flight," *Aviation, Space and Environmental Medicine*, vol. 69, no. 4, pp. 360–367, Apr. 1998.

[17] J. A. Veltman and A. W. K. Gaillard, "Physiological workload reactions to increasing levels of task difficulty," *Ergonomics*, vol. 41, no. 5, pp. 656–669, May 1998.

[18] M. R. Endsley, "Automation and situation awareness," in R. Parasuraman and M. Mouloua (Eds.)*, Automation and human performance: Theory and applications*. Mahwah, NJ: Lawrence Erlbaum, 1996, pp. 163–181.

[19] J. Drury, J. Scholtz, and H. A. Yanco, "Awareness in human-robot interactions," in *Proc. of the IEEE Int. Conf. on Systems, Man and Cybernetics*, Washington, DC, Oct. 2003, pp. 912–918.

[20] J. Drury, L. Riek, and N. Ratcliffe, "A decomposition of UAV-related situation awareness," in *Proc. of the ACM/IEEE Int. Conf. on Human-Robot Interaction*, Salt Lake City, UT, Mar. 2006, pp. 88–94.

[21] M. R. Endsley, "Design and evaluation for situation awareness enhancement," in *Proc. of the Human Factors Society 32nd Annual Meeting*, Santa Monica, CA, Oct. 1988, pp. 97–101.

[22] G. Rodriguez and C. R. Weisbin, "A new method to evaluate human-robot system performance," *Autonomous Robots*, vol. 14, no. 2-3, pp. 165–178, Mar.–May 2003.

[23] T. B. Sheridan and W. L. Verplank, "Human and computer control of undersea teleoperators," Man-Machine Laboratory, Massachusetts Institute of Technology, Cambridge, MA, Tech. Rep. ADA057655, 1978.

[24] R. Parasuraman, T. B. Sheridan, and C. D. Wickens, "A model of types and levels of human interaction with automation," *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans*, vol. 30, no. 3, pp. 286–297, May 2000.

[25] M. R. Endsley and D. B. Kaber, "Level of automation effects on performance, situation awareness and workload in a dynamic control task," *Ergonomics*, vol. 42, no. 3, pp. 462–492, Mar. 1999.

[26] B. P. Sellner, F. Heger, L. Hiatt, R. Simmons, and S. Singh, "Coordinated multi-agent teams and sliding autonomy for large-scale assembly," *Proceedings of the IEEE - Special Issue on Multi-Robot Systems*, vol. 94, no. 7, pp. 1425 – 1444, Jul. 2006.

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

[28] D. R. Olsen and M. A. Goodrich, "Metrics for evaluating human-robot interactions," in *Proc. of Workshop on Performance Metrics for Intelligent Systems*, Gaithersburg, MA, Sep. 2003.

[29] M. A. Goodrich and D. R. Olsen, "Seven principles of efficient human robot interaction," in *Proc. of IEEE Int. Conf. on Systems, Man, and Cybernetics*, Washington, DC, Oct. 2003, pp. 3943–3948.

Jacob W. Crandall received the B.S., M.S., and Ph.D. degrees in Computer Science from Brigham Young University, Provo, UT, in 2001, 2004, and 2006, respectively.

He is currently a postdoctoral associate in the Department of Aeronautics & Astronautics at the Massachusetts Institute of Technology. His research interests include multi-agent learning, human-machine interaction, decision theory, and human supervisory control.

Mary L. Cummings (M'03) received her B.S. in Mathematics from the United States Naval Academy in 1988, her M.S. in Space Systems Engineering from the Naval Postgraduate School in 1994, and her Ph.D. in Systems Engineering from the University of Virginia in 2003.

A naval officer and military pilot from 1988-1999, she was one of the Navy's first female fighter pilots. She is currently an assistant professor in the Aeronautics & Astronautics Department at the Massachusetts Institute of Technology. Her research interests include human supervisory control, human-uninhabited vehicle interaction, bounded collaborative human-computer decision making, decision support, information complexity in displays, and the ethical and social impact of technology.

Appendix C

The following is a draft of a paper submitted to:

The Journal of Aerospace Computing, Information, and Communication, June 2008

A Predictive Model for Human-Unmanned Vehicle Systems

Jacob W. Crandall, * M. L. Cummings,† and Carl E. Nehme‡
Massachusetts Institute of Technology, Cambridge MA 02139

Advances in automation are making it possible for a single operator to control multiple unmanned vehicles (UVs). However, the complex nature of these teams presents a difficult and exciting challenge for designers of human-UV systems. To build such systems effectively, models must be developed that *describe* the behavior of the human-UV team and that *predict* how alterations in team composition and system design will affect the system's overall effectiveness. In this paper, we describe a methodology for modeling human-UV systems consisting of a single operator and multiple independent UVs. Via a case study, we show that this modeling methodology yields an accurate description of the observed human-UV system. Additionally, results show that the model is also able to accurately predict how changes in the human-UV interface and the UVs' autonomy levels will alter the system's effectiveness.

I. Introduction

Many important missions, including search and rescue, border security, and military operations, require human reasoning to be combined with automated unmanned vehicle (UV) capabilities to form a synergistic human-UV team. However, the design and implementation of such systems remains a difficult and challenging task. Challenges related to the human operators, the UVs, and the interactions between them must be solved before human-UV systems will realize their full potentials.

To understand and address these issues more fully, comprehensive models of human-UV systems should be developed. These models should have two important capabilities. First, they should adequately *describe* the behavior and performance of the team and the system as a whole. Second, these models should be able to accurately *predict* the behavior and performance of the team as the environment, mission, or human-UV system changes.

A model with both descriptive and predictive abilities has a number of important applications. For example, such a model can improve the design and implementation processes of human-UV systems. As in any systems engineering process, test and evaluation plays a critical role in fielding new technologies. In systems with significant human-automation interaction, testing with representative users is expensive and time consuming. Thus, the development of a high-fidelity model of a human-UV system with both descriptive and predictive capabilities will streamline the test and evaluation cycle since it can both help diagnose the cause of previous system failures and inefficiencies, and indicate how potential design modifications will affect the behavior and performance of the system.

A model with both descriptive and predictive abilities can also, among other things, be used to determine successful combinations of UVs within the team (team composition). The composition of futuristic human-UV teams is likely to dynamically change both in number and type due to changing mission assignments and resource availability. High-fidelity models can be used to ensure that changes in team composition will not cause system performance to drop below acceptable levels. Furthermore, given the team composition, these models can suggest which autonomy levels are appropriate for the UVs to employ.

As a step toward developing such high-fidelity and comprehensive models, we propose a methodology for modeling human-UV systems consisting of a single human operator and a team of homogeneous and

*Postdoctoral Associate, Department of Aeronautics and Astronautics, Cambridge MA 02139. AIAA Member
†Assistant Professor, Department of Aeronautics and Astronautics, Cambridge MA 02139. AIAA Associate Fellow
‡Ph.D. Candidate, Department of Aeronautics and Astronautics, Cambridge MA 02139

independent UVs. In this modeling methodology, stochastic models of both the human operator and the UVs in the team are formed from observational data. Combined, these models can successfully predict how changes in the UVs' autonomy and the human-UV interface affect the performance of the human-UV system.

The remainder of this paper proceeds as follows. In Section II, we define a stochastic model of human-UV systems consisting of a single operator and multiple independent UVs. This stochastic model is constructed using observational data as described in Section III. In Section IV, we describe a user study in which users controlled a simulated UV team to perform a search and rescue mission. We use data observed from this user study to model the human-UV team and make predictions about how design changes will affect the system's effectiveness. In Section V, we compare these predictions with observed results from the user study to validate the effectiveness of the model. We conclude and discuss future work in Section VI.

II. Modeling Human-UV Systems

In this section, we define the specific kinds of human-UV systems we consider in this paper. We then describe a modeling methodology for such systems.

II.A. Human-UV Teams

While current UV technologies typically require multiple human operators to control a single remote UV, it is anticipated that continued advances in technology will make it possible for a few operators to supervise many UVs. This capability is particularly desirable given a report that operators of unmanned aerial vehicles (UAVs) are overworked due to an insufficient numbers of crew members.[1] As such, in this paper, we consider systems with reduced manning. In particular, we focus on systems consisting of a single operator and multiple UVs.

A number of design decisions affect the nature of the interactions between the operator and the UVs in these systems. One of these design decisions is the level of automation at which the UVs operate, ranging between complete human control and complete system autonomy.[2] The effects of levels of automation on human-UV systems have been analyzed in many contexts, including work by Mitchell *et al.*,[3] Kaber and Endsley,[4] and Wang and Lewis.[5] Our modeling methodology can be applied to teams employing any level or combined levels of automation.

Another design decision that affects human-UV interactions, and, ultimately, the system's effectiveness, is the level of teaming used in the system. Goodrich *et al.*[6] discussed two teaming paradigms. In sequential control, the human operator attends to each UV individually. This approach is often necessary with independent UVs. On the other hand, a human may desire to direct multiple UVs simultaneously through goal-based commands. An example of such control is the *Playbook*$^{\text{TM}}$ methodology.[7] The model we present in this paper assumes sequential control.

Futuristic human-UV teams will often be composed of UVs of multiple types (i.e., heterogeneous UV teams). For example, teams could be composed of both unmanned aerial vehicles and unmanned ground vehicles. While the model we present in this paper can be extended to teams with heterogeneous UV capabilities, we consider only the homogeneous case in this paper. Nehme *et al.*[8] and Wang and Lewis[9] present models that explicitly consider the heterogeneous UV case. Our model also makes the assumption that the UVs perform independent tasks, though it can be extended to include the collaborative case.

We now describe a methodology for modeling these human-UV systems.

II.B. A Stochastic Model

Our modeling methodology requires that stochastic models be constructed for various aspects of the system, including the behaviors of both the human operator and the UVs. These separate models are then combined to form a complete model of the human-UV system. In general, high-fidelity models of the separate aspects of a system do not necessarily equate to a high-fidelity model of the complete system, particularly in the case of complex interactions between team members. However, in our methodology, the separate stochastic models are joined by the common theme of *system state*, which makes it possible to capture the interactions between these individual models. Thus, joined together, the individual models form a complete model of the human-UV system.

We first define system state as it pertains to the human-UV teams we consider. We then describe individual stochastic models used to describe the behavior of the various members of the human-UV team.

II.B.1. System State

Let the (factorized) system state be a vector of m features $\sigma = (f_1, \ldots, f_m) \in \Sigma$, where Σ is the set of system states, and where each feature f_i can take on either a discrete or continuous set of values. A critical challenge in modeling human-UV systems, then, is to identify the features that determine the behavior and performance of the team at any given time.

In this paper, we define system state with two features, indicated with the tuple $\sigma = (\bar{\mathbf{s}}, \tau)$. The first component of system state consists of the states of the individual UVs in the team. Formally, let s_i be the state of UV i. Then, the *joint state* of the UVs in the team is given by the vector $\bar{\mathbf{s}} = (s_1, \ldots, s_n)$. Second, in time-critical missions, system state is also determined by *mission time* τ, which is defined as the time elapsed since the mission began. Other features, including the human operator's cognitive state and world complexity, could affect the behavior of the human-UV team. However, for simplicity, we do not explicitly consider these features in this paper, though doing so is a subject of future work.

System state can be used to define various models of human-UV systems. We now describe our models, beginning with models of individual UV behavior.

II.B.2. Modeling UV Behavior

A UV's behavior is dependent on its automated capabilities as well as the frequency and duration of human-UV interactions.[10] Thus, one method for modeling UV behavior is via the temporal metrics of *neglect time* and *interaction time*.[10–12] A UV's neglect time is the expected amount of time the UV can maintain acceptable performance levels in the absence of interactions with the human operator. Interaction time is the average amount of time an operator must interact with the UV to restore or maintain desirable performance levels. Paired together, these metrics have been used to estimate a number of UV properties, including *UV attention demand*,[13] which refers to the amount of time an operator that must be devote to a single UV, and *Fan-out*,[10,12] which refers to the number of UVs that a single operator can effectively control.

While neglect time and interaction time are valuable and informative metrics, they often do not provide a sufficiently detailed model of UV behavior to accurately describe and predict many relevant aspects of UV behavior.[14] An alternate modeling methodology is to describe a UV's behavior with random processes.[10,14] In this methodology, two sets of random processes are constructed, one which describes the UV's performance during human-UV interactions, and the other which describes the UV's performance in the absence of human-UV interactions. We call these random processes *interaction impact* ($\mathcal{II}$) and *neglect impact* ($\mathcal{NI}$), respectively. Our modeling methodology follows this approach, though the random processes we construct describe a UV's state transitions rather than measured performance.

Formally, let $\sigma \in \Sigma$ be the system's state when the human began interacting with UV i. Then, the random process $\mathcal{II}(\sigma)$ stochastically models UV i's states throughout the duration of the interaction. The time variable of the process takes on all values in the interval $[0, l]$, where l is the length of the human-UV interaction and $t = 0$ corresponds to the time that the human-UV interaction began. Thus, for each $t \in [0, l]$, $\mathcal{II}(\sigma; t)$ is a random variable that specifies a probability distribution over UV states.

Likewise, we model the behavior of a UV in the absence of human attention with a random process. Let $\sigma \in \Sigma$ be the system's state when the UV's last interaction with the operator ended. Then, the random process $\mathcal{NI}(\sigma)$ describes the UV's state transitions over time in the absence of human attention. Hence, for each $t \in [0, \infty)$, where $t = 0$ corresponds to the time that the UV's last interaction with the operator ended, $\mathcal{NI}(\sigma; t)$ is a random variable that specifies a probability distribution over UV states.

The structures $\mathcal{II}(\sigma)$ and $\mathcal{NI}(\sigma)$ assume the Markov property that UV behavior is dependent only on the system state σ at the beginning and end of the human-UV interaction, respectively. For the situations we consider in this paper, this assumption does not appear to significantly detract from the predictive ability of the model. However, if needed, the first-order assumption could be exchanged for an n-order Markov assumption, though doing so drastically increases the amount of data needed to model these processes. In large part, the accuracy of the Markov assumption is dependent on the set of UV states considered.

Finally, we note that when the UVs perform independent tasks, $\mathcal{II}(\sigma)$ and $\mathcal{NI}(\sigma)$ need not consider the full joint state $\bar{\mathbf{s}}$ of the UVs in the team. Rather, it is only necessary to consider the state of the UV in question. Since we assume independent UVs in this paper, we use the simplified system state $\sigma = (s_i, \tau)$ in these structures.

II.B.3. Modeling Operator Behavior

The human operator plays a crucial role in the success of the human-UV system. Thus, any high-fidelity model of human-UV systems must accurately model the human operator. Since $\mathcal{II}(\sigma)$ and $\mathcal{NI}(\sigma)$ are driven by human input, they implicitly model human behavior during interactions with an individual UV. However, these structures do not account for how the human operator allocates attention to the various UVs in the team, a process which we call *attention allocation*. Thus, our model must include other structures to model operator attention allocation.

Previous work has identified two important aspects of attention allocation. The first aspect of attention allocation involves how the operator prioritizes multiple tasks,[15–17] which we define as the operator's *selection strategy*. A second important aspect of attention allocation is *switching time*, which is the time it takes the operator to determine which UV to service.[18,19]

As is the case with modeling UV behavior, temporal metrics can be derived to measure attention allocation. One such set of metrics involves the concept of *wait times*, or times that UVs spend in degraded performance states.[16,20,21] Higher wait times typically indicate less effective prioritization schemes and longer switching times. Such metrics have been used to augment Fan-out predictions.[20,21]

However, as in modeling UV behavior, we choose to model operator selection times stochastically with two separate structures. Formally, let $\sigma \in \Sigma$ be the system state at the time that the previous human-UV interaction ended. Then, $\mathcal{ST}(\sigma)$ is a random variable describing operator *switching time* given the system state σ. Similarly, given σ, the operator's *selection strategy* is defined by the random variable $\mathcal{SS}(\sigma)$, which specifies a probability distribution over the UVs in the team.

In this model, the switching time $\mathcal{ST}(\sigma)$ is a combination of two kinds of time periods. First, it consists of the time it takes for the operator to (a) orient to the circumstances of the UVs in the team, (b) select a UV to service, and (c) carry out the necessary steps to select that UV. Second, it consists of any time in which the operator chooses not to service any of the UVs in the team when he believes none of the UVs need servicing. In such situations, the operator simply monitors the UVs' progress, etc. In many situations, it is desirable to distinguish between these time periods, though doing so requires knowledge of operator intentions. We leave further investigation of this extension to future work.

We note that operator behavior in human-UV systems is driven by a number of important cognitive processes and limitations. These processes and limitations include, among others, operator workload,[22] operator utilization,[23–25] operator trust in automation,[26] operator situation awareness,[27,28] and automation bias.[29] Because our stochastic structures directly model human behavior, they implicitly capture the results of each of these cognitive effects, though they do not do so explicitly.

II.B.4. Model Summary

In summary, we have identified four stochastic structures that, when combined, form a model of the human-UV system. The random processes of interaction impact $\mathcal{II}(\sigma)$ and neglect impact $\mathcal{NI}(\sigma)$ describe the behavior of each of the UVs in the team in the presence and absence of interactions with the human operator. Meanwhile, operator switching time $\mathcal{ST}(\sigma)$ and operator selection strategy $\mathcal{SS}(\sigma)$ describe how the human operator allocates his attention among the UVs in the team. Taken together, these structures describe the behavior of all members of the team at any given time.

In the next section, we describe how these stochastic structures can be constructed from data obtained from observing the human-UV system.

III. Constructing the Model from Data

The mathematical structures $\mathcal{II}(\sigma)$, $\mathcal{NI}(\sigma)$, $\mathcal{ST}(\sigma)$, and $\mathcal{SS}(\sigma)$ can be constructed in a number of ways. In the pre-implementation phase of human-UV system design, these structures can be handcrafted to represent a hypothetical team from which inexpensive, yet powerful evaluations of the target system can be made. However, once a human-UV system is implemented and observed in operations, data from these observations can be used to construct a high-fidelity model of the system. This model can then, in turn, be used to (a) analyze the shortcomings of the human-UV team, (b) estimate the effectiveness of various design improvements, and (c) predict how the human-UV team will behave in previously unobserved situations. In this section, we focus on situations in which the stochastic structures are constructed from observed data.

Time	τ_0	τ_1	τ_2	τ_3	τ_4	τ_5	τ_6	τ_7	τ_8	τ_9	τ_{10}	τ_{11}	τ_{12}	τ_{13}	τ_{14}	τ_{15}	τ_{16}	τ_{17}	τ_{18}
Human	1	1	1			3	3			1	1	1	1				2	2	
UV 1	s^1	s^2	s^2	s^2	s^2	s^2	s^3	s^3	s^3	s^3	s^4	s^4	s^2	s^2	s^2	s^2	s^3	s^4	s^4
UV 2	s^1	s^1	s^1	s^1	s^1	s^1	s^1	s^1	s^1	s^1	s^1	s^1	s^1	s^1	s^1	s^1	s^1	s^3	s^3
UV 3	s^1	s^1	s^1	s^1	s^1	s^1	s^2	s^2	s^2	s^3	s^3	s^3	s^3	s^4	s^4	s^3	s^3	s^3	s^4

(a) Hypothetical data from a single human, 3-UV team from time t_0 through time t_{18}.

UV	**System State**	**Sequence**
1	(τ_0, s^1)	s^1, s^2, s^2
3	(τ_5, s^1)	s^1, s^2
1	(τ_9, s^3)	s^3, s^4, s^4, s^2
2	(τ_{16}, s^1)	s^1, s^3

(b) The set $\Theta^{\mathcal{II}}$

UV	**System State**	**Sequence**
1	(τ_3, s^2)	s^2, s^2, s^2, s^3, s^3, s^3, ??
3	(τ_7, s^2)	s^2, s^2, s^3, s^3, s^3, s^3, s^4, s^4, s^3, s^3, s^3, s^4, $\cdots$
1	(τ_{13}, s^2)	s^2, s^2, s^2, s^3, s^4, s^4, $\cdots$
2	(τ_{18}, s^3)	s^3, $\cdots$

(c) The set $\Theta^{\mathcal{NI}}$

System State	**Switch Time**
$(\tau_3, \bar{\mathbf{s}} = (s^2, s^1, s^1))$	2
$(\tau_7, \bar{\mathbf{s}} = (s^3, s^1, s^2))$	2
$(\tau_{13}, \bar{\mathbf{s}} = (s^2, s^1, s^4))$	3
$(\tau_{18}, \bar{\mathbf{s}} = (s^3, s^1, s^3))$	?

(d) The set $\Theta^{\mathcal{ST}}$

System State	**State Selected (x_o)**
$(\tau_0, \bar{\mathbf{s}} = (s^1, s^1, s^1))$	s^1
$(\tau_5, \bar{\mathbf{s}} = (s^2, s^1, s^1))$	s^1
$(\tau_9, \bar{\mathbf{s}} = (s^3, s^1, s^3))$	s^3
$(\tau_{16}, \bar{\mathbf{s}} = (s^3, s^1, s^3))$	s^1

(e) The set $\Theta^{\mathcal{SS}}$

Figure 1. Data logged from observations of a human-UV system is organized into four sets of samples. (a) Hypothetical data of a human-UV team with 3-UVs. (b)-(e) Sets of samples derived from the hypothetical data.

III.A. Constructing the Individual Behavioral Models

To model $\mathcal{II}(\sigma)$, $\mathcal{NI}(\sigma)$, $\mathcal{ST}(\sigma)$, and $\mathcal{SS}(\sigma)$, the data is first organized into four sets of data samples, which we denote $\Theta^{\mathcal{II}}$, $\Theta^{\mathcal{NI}}$, $\Theta^{\mathcal{ST}}$, and $\Theta^{\mathcal{SS}}$. We then implicitly model $\mathcal{II}(\sigma)$, $\mathcal{NI}(\sigma)$, $\mathcal{ST}(\sigma)$, and $\mathcal{SS}(\sigma)$ for each $\sigma \in \Sigma$ by forming probability distributions over the samples in these sets. These probability distributions over samples can then be used to model the human-UV system via a discrete event simulation.

III.A.1. Extracting Data Samples

To see how data is organized into the sets of samples $\Theta^{\mathcal{II}}$, $\Theta^{\mathcal{NI}}$, $\Theta^{\mathcal{ST}}$, and $\Theta^{\mathcal{SS}}$, consider Figure 1a, which shows a hypothetical data log for a 3-UV team sampled at the discrete mission times τ_0 through τ_{18}. In the figure, the row labeled "Human" indicates the UV that the operator attended to at each mission time, with empty cells indicating that the operator was not servicing any UV. For example, at mission time τ_6, the operator was attending to UV 3. The figure also shows the state of each UV at each mission time. For example, at mission time τ_6, the joint state was $\bar{\mathbf{s}} = (s^3, s^1, s^2)$, meaning that UV 1 was in state s^3, UV 2 was in state s^1, and UV 3 was in state s^2.

Figure 1b-e shows the various sets of data samples derived from this set of hypothetical data. The set $\Theta^{\mathcal{II}}$ contains four data samples (Figure 1b), one corresponding to each human-UV interaction. A data sample in $\Theta^{\mathcal{II}}$ consists of two pieces of information: the system state at the beginning of the interaction and the UV's sequence of states throughout the duration of the interaction. Thus, the second entry in Figure 1b corresponds to the operator's second interaction in the data segment, in which the operator serviced UV 3 at times τ_5 through τ_6.

Corresponding to each sample $x \in \Theta^{\mathcal{II}}$ is a sample in $\Theta^{\mathcal{NI}}$. These samples, which are shown for our hypothetical example in Figure 1c, contain the same kinds of information as those in $\Theta^{\mathcal{II}}$, except that the sequences of UV states in these samples are theoretically infinite (since they represent UV behavior in the absence of UV interactions). This is problematic, since the data log in Figure 1a does not provide such information. For example, the sequence of UV states for the first entry of Figure 1c is incomplete since this sequence was trumped by human interaction at time τ_9. Thus, UV behavior in the absence of human-UV interactions after this time is unknown and must be estimated.

The sets $\Theta^{\mathcal{ST}}$ and $\Theta^{\mathcal{SS}}$ formed from the example data log are shown in Figures 1d and 1e, respectively. Each of these samples is composed of the system state at the time the sample occurred and the outcome

of that particular sample. The outcomes take on the form of a switching time in $\Theta^{\mathcal{ST}}$ and the state of the selected UV in $\Theta^{\mathcal{SS}}$.

III.A.2. Constructing Probability Distributions over Samples

The structures $\mathcal{II}(\sigma)$, $\mathcal{NI}(\sigma)$, $\mathcal{ST}(\sigma)$, and $\mathcal{SS}(\sigma)$ can be estimated for all $\sigma \in \Sigma$ by forming probability distributions over the sets of samples just described. We now formally define these probability distributions for each structure.

$\boldsymbol{\mathcal{II}(\sigma)}$. The behavior of a UV during a human-UV interaction can be estimated using the state sequences of the samples in $\Theta^{\mathcal{II}}$ that are close matches to the target system state σ. Specifically, the probability that sample $x \in \Theta^{\mathcal{II}}$ is chosen to model a UV's behavior during an interaction given the target system state σ is

$$Pr(x|\sigma) = \frac{w_x^{II}(\sigma)}{\sum_{y \in \Theta^{\mathcal{II}}} w_y^{II}(\sigma)}. \tag{1}$$

The value $w_z^{\mathcal{II}}(\sigma)$, where $z \in \{x, y\}$, is a weight determined by the distance of the sample's system state $z_\sigma = (z_s, z_\tau)$ to $\sigma = (s_i, \tau)$. Formally,

$$w_z^{\mathcal{II}}(\sigma) = \begin{cases} f(z_\tau - \tau) & \text{if } s_i = z_s \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where $f(\cdot)$ is a time-weighting function that gives higher weight to samples in which z_τ is close to τ. In words, the weight of sample z is given by the proximity of the target mission time τ to the sample's mission time z_τ, provided that the target UV state s_i is equal to the sample's UV state z_s.

$\boldsymbol{\mathcal{NI}(\sigma)}$. Likewise, the stochastic behavior of a UV while being neglected can be estimated with a probability distribution over the samples in $\Theta^{\mathcal{NI}}$. However, in practice this is not necessary, since the behavior of the UV when it is neglected follows from the previous human-UV interaction. Thus, the sample $y \in \Theta^{\mathcal{NI}}$ that succeeds the selected sample from $\Theta^{\mathcal{II}}$ can be used to describe UV behavior after the interaction is completed. For example, if the first entry of Figure 1b were selected to define a UV's behavior during a human-UV interaction, then the first sample of Figure 1c would be selected to define the behavior of the UV when the interaction terminated.

$\boldsymbol{\mathcal{ST}(\sigma)}$. Operator switching time is modeled as a probability distribution over the samples in $\Theta^{\mathcal{ST}}$. However, unlike the samples in $\Theta^{\mathcal{II}}$ and $\Theta^{\mathcal{NI}}$, samples in $\Theta^{\mathcal{ST}}$ consider the complete joint state $\bar{\mathbf{s}}$ rather than just the individual UV's state s_i. Thus, determining the proximity of a sample's system state $x_\sigma = (x_{\bar{\mathbf{s}}}, x_\tau)$ to the target system state $\sigma = (\bar{\mathbf{s}}, \tau)$ becomes more complicated than in Equation (2).

To define the proximity between $\bar{\mathbf{s}}$ and x_σ, let V_x be the set of permutation vectors of $x_{\bar{\mathbf{s}}}$ and let v_x^i be the i^{th} element of the vector $\mathbf{v_x} \in V_x$. Then, the similarity between the joint states $\bar{\mathbf{s}}$ and $x_{\bar{\mathbf{s}}}$, denoted $\Phi(\bar{\mathbf{s}}, x_{\bar{\mathbf{s}}})$, is given by

$$\Phi(\bar{\mathbf{s}}, x_{\bar{\mathbf{s}}}) = \min_{\mathbf{v_x} \in V_x} \prod_{i=1}^{n} \phi(s_i, v_x^i), \tag{3}$$

where $\phi(s_i, v_x^i) \in [0, 1]$ is the similarity between the individual UV states s_i and v_x^i, and where $\phi(s_i, v_x^i) = 1$ when $s_i = v_x^i$. In words, the similarity between the joint states $\bar{\mathbf{s}}$ and $x_{\bar{\mathbf{s}}}$ is the product of the similarities between the individual UV states when the states in the two vectors are reordered to form the closest match between the two vectors.

Then, the weight of a sample $x \in \Theta^{\mathcal{ST}}$ is a combination of $\Phi(\bar{\mathbf{s}}, x_{\bar{\mathbf{s}}})$ and the difference between the sample mission time x_τ and the target mission time τ. Formally, let

$$w_x^{ST}(S, T) = \Phi(\bar{\mathbf{s}}, x_{\bar{\mathbf{s}}}) \cdot f(x_\tau - \tau), \tag{4}$$

where $f(\cdot)$ is defined as before. Thus, the probability that a sample $x \in \Theta^{\mathcal{ST}}$ given the system state σ is:

$$Pr(x|\sigma) = \frac{w_x^{ST}(\sigma)}{\sum_{y \in \Theta^{ST}} w_y^{ST}(\sigma)}. \tag{5}$$

1. Set $\tau = 0$, determine an initial sample from $\Theta^{\mathcal{NI}}$ for each UV
2. Repeat
   (a) $\tau = \tau + x_o$, where x_o is the outcome of some sample $x \in \Theta^{\mathcal{ST}}$
   (b) Update the joint state $\bar{\mathbf{s}}$
   (c) Select a UV (denoted UV k) to service using sample $x \in \Theta^{\mathcal{SS}}$
   (d) Select a sample $x \in \Theta^{\mathcal{TI}}$ for UV k
   (e) $\tau = \tau + l$ (l is the length of the state sequence of sample x chosen in (d))
   (f) Update the joint state $\bar{\mathbf{s}}$
   (g) Select a sample $x \in \Theta^{\mathcal{NI}}$ for UV k

Algorithm 1: Outline of the discrete event simulation. Samples are selected as described in Section III.A.2.

$\boldsymbol{\mathcal{SS}(\sigma)}$. The operator's switching strategy in the target system state σ is modeled with a probability distribution over the samples in $\Theta^{\mathcal{SS}}$. This probability distribution is defined in the same manner as $\mathcal{ST}(\sigma)$, except that $w_x^{\mathcal{SS}}(\sigma)$ is defined as

$$w_x^{\mathcal{SS}}(\sigma) = \begin{cases} \Phi(\bar{\mathbf{s}}, x_{\bar{\mathbf{s}}}) \cdot f(x_\tau - \tau) & \text{if } \exists s^i \in \{\bar{\mathbf{s}}\} : s^i = x_o \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where x_o is the outcome (or selected state) in sample $x \in \Theta^{\mathcal{SS}}$ and $\{\bar{\mathbf{s}}\}$ denotes the set of individual UV states contained in $\bar{\mathbf{s}}$. In words, $w_x^{\mathcal{SS}}(\sigma)$ is defined identically to $w_x^{\mathcal{ST}}(\sigma)$ except that we consider only samples with an outcome x_o that matches the state of one of the UVs in the target joint state $\bar{\mathbf{s}}$.

III.B. Combining the Behavioral Models

Once constructed, these four stochastic structures can be used to simulate the behavior of a human-UV system using a discrete event simulation, outlined in Algorithm 1. In step 2a, the human operator's switching time is determined, during which time we observe the UVs' state transitions defined by each UV's chosen sample in $\Theta^{\mathcal{NI}}$. In steps 2c-d, UV k is selected for servicing and a sample is drawn from $\Theta^{\mathcal{TI}}$. UV k acts according to this sample's state sequence for the next l time units, while the other UVs continue to act according to their samples of $\Theta^{\mathcal{NI}}$. When the interaction is completed, another sample is chosen to simulate the behavior of UV k from $\Theta^{\mathcal{NI}}$. The process then repeats.

III.C. Implementation Specific Modeling Parameters

The model we have described can be used to model a wide variety of human-UV systems performing a wide variety of missions. However, the model requires several mission and team-specific definitions. First, the model requires that a set of individual UV states be identified. As in any mathematical model that uses the concept of state, a good set of individual UV states balances two objectives. First, if two situations cause different behaviors from either the human operator or a UV in the team, they should be marked as different states. Second, a good set of individual UV states should be as small as possible in order to make the model efficient. In general, a smaller state space is necessary for smaller data sets, while a larger state space can be used for larger data sets.

In addition to an enumeration of individual UV states, the model requires two other definitions. First, definitions of similarities between the individual UV states are necessary (see (3)). Second, in time-critical missions, the model requires that the time-weighting function $f(\cdot)$ be defined. Ideally, the model should select only samples taken in similar mission times to the target mission time. However, when the data set is small, this restriction must often be relaxed.

In the next section, we describe a user study that illustrates the descriptive and predictive abilities of this modeling methodology.

IV. Experimental Case Study

To validate the modeling methodology, we conducted a user study in which a human operator directed a simulated UV team in a search and rescue mission. In this section, we describe this user study, including

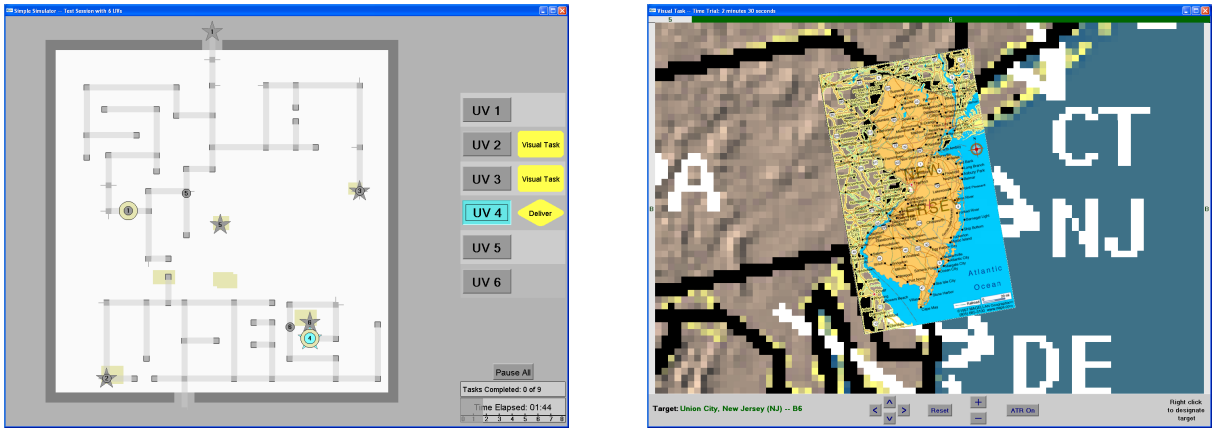


Figure 2. The human-UV interface used in the experiment. The display at left shows the map of the maze, the locations of the UVs, their destinations, the locations of the objects to be gathered, etc. The display at right shows the interface for locating cities.

the the study's software test-bed and experimental procedure. We also define system- and mission-specific modeling parameters needed to model the human-UV systems in the study. In the next section, we demonstrate the descriptive and predictive power of our modeling methodology using data obtained from this user study.

IV.A. Software Test-bed

In the user study, participants supervised simulated UVs in RESCU (*R*esearch *E*nvironment for *S*upervisory *C*ontrol of *U*nmanned-Vehicles). A simulated UV environment was chosen since current UV capabilities do not allow for rapid prototyping of systems that allow a single human to control multiple UVs simultaneously. Furthermore, while simulated UVs obviously behave differently than real UVs in many respects, the modeling methodology described in this paper can be used to model teams with both simulated and real UVs.

We now describe the software test-bed used in the user study in three parts: the human-UV team mission, the human-UV interface, and the UVs' behaviors.

IV.A.1. Mission

Each participant was tasked with using simulated UVs to collect as many objects as possible from a maze in an eight-minute time period, while ensuring that all UVs were out of the maze when time expired. The objects were randomly spread through the initially unknown maze. However, as each UV moved about the maze, it created a map that it shared with the participant and the other UVs in the team. Initially, only the positions of six of the objects were shown to the team. The locations of two additional objects were shown to the team in each minute of the scenario, so there were 22 possible objects for the team to collect.

An object was collected from the maze using a three-step process. First, a UV moved to the location of the object in the maze. Second, the UV picked up the object. In the real world, performing such an action might require the human operator to assist in identifying the object with video or laser data. To simulate this task, we asked users to identify a city on a map of the mainland United States using Google Earth-style software. Third, the UV carried the object out of the maze via one of two exits.

The subjects were told to maximize the following objective function:

$$Score = ObjectsCollected - UVsLost, \tag{7}$$

where *ObjectsCollected* was the number of objects removed from the area during the session, and *UVsLost* was the number of UVs remaining in the area when time expired.

IV.A.2. Interface

The human-UV interface was the two-screen display shown in Figure 2. The map of the maze was displayed on the left screen, along with the positions of the UVs and the known objects in the maze. The participant

used the right screen to identify cities on the map. The participant could only control one UV at a time, which he selected by clicking a button on the interface corresponding to that UV. Once the participant selected the UV, she could direct the UV by designating a goal location and modifying the UV's intended path to that goal. To designate a goal, the user dragged the goal icon corresponding to the UV in question to the desired location. The UV then generated and displayed the path it intended to follow. The participant could modify this path using the mouse.

Two different interface modes were used in the user study. In the first mode, the operator was provided no assistance in identifying the cities on the map. In the second mode, the operator was assisted by an automated visioning system (AVS). The AVS suggested two candidate cities on the map to the user. These suggestions were imposed on the map as blinking red boxes around the suggested cities. The system was designed so that one of these suggestions was correct about 70-75% of the time.

IV.A.3. UV Behavior

The UVs used Dijkstra's algorithm to plan their movements through the maze toward their goal destinations. However, as the maze was incomplete, the UVs had to decide between exploring the unknown maze and taking a known, possibly longer, path. Further details on how the UVs made these decisions are documented in previous work.[14] The participant was allowed to modify a UV's path if she desired.

Two different UV autonomy modes were used in the study. In the first mode, goal generation was solely the task of the participant. If the participant did not provide a goal for the UV, the UV did not act. In the second mode, each UV automatically selected a new goal when it was left idle. Specifically, a management-by-exception (MBE) level of automation was used in which a UV left idle at its goal destination, but not on an object in the maze, waited 15 seconds for the user to intervene. If the user did not intervene, the UV automatically derived its own goal. In the case that the UV was searching for an object, it made the nearest unassigned object its new goal. On the other hand, if the UV was already carrying an object, it set the nearest perceived exit as its new goal. Additionally, when the user did not intervene, the UVs automatically chose to exit the maze via the nearest perceived exit in the final 45 seconds of a session.

IV.B. Experimental Procedure

The experiment was a 4 (decision support) x 4 (UV team size) mixed design study. Decision support (DS) was a between-subjects factor, with the four levels of decision support being (1) *noDS* (no AVS and no MBE), (2) *AVS* (AVS but no MBE), (3) *MBE* (MBE but no AVS), and (4) *AVS-MBE* (both AVS and MBE). UV team size was a within-subjects factor; each participant performed the search and rescue mission for team sizes of two, four, six, and eight UVs. The order in which the participants used each team size was randomized and counter-balanced throughout the study.

Each participant was randomly assigned to a DS condition and trained on all aspects of that system. The participant then completed three comprehensive practice sessions. Following these practice sessions, each participant performed four test sessions, each with a different team size. After the session, a retrospective verbal protocol was conducted in which participants answered questions about their behavior in the study. Each participant was paid $10 per hour; the highest scorer also received a $100 gift certificate. Sixty-four participants between the ages of 18 and 49 participated in the study, 16 in each condition.

IV.C. Model Parameters

As outlined in Section III.C, our modeling methodology requires a set of individual UV states, a similarity metric, and a time-weighting function $f(\cdot)$. We now give these definitions for the human-UV system used in the user study.

IV.C.1. A Set of UV States

Recall that a set of UV states should distinguish among situations that evoke or should evoke different behaviors from the human operator or the UVs in the team. We used statistical analysis of participants' selection strategies during the user study as well as user's post-experiment comments to identify such a set of states in RESCU. From this process, we identified 21 distinct UV states. The decision tree for determining a UV's state is shown in Figure 3.

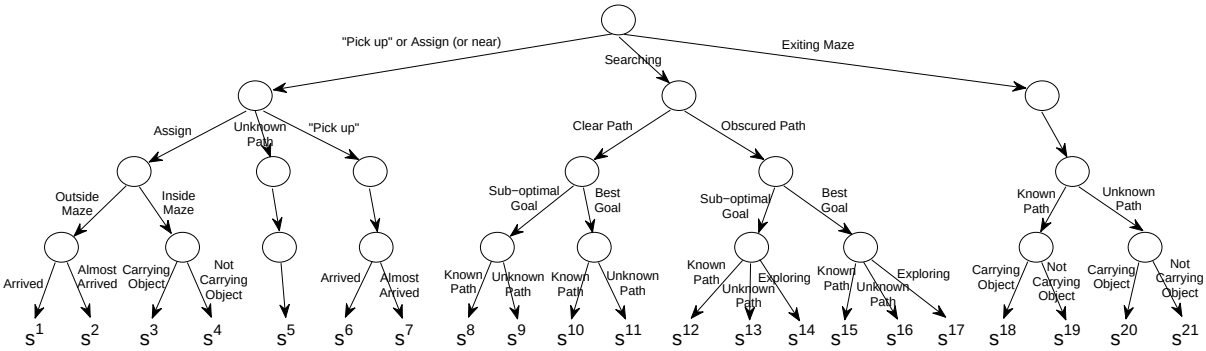


Figure 3. The decision tree used to define a UV's state at any given time in the user study.

As shown in Figure 3, a number of features were used to determine a UV's state. These features included whether a UV was searching for an object, exiting the maze, waiting for the human to help pick up an object, or waiting for a goal assignment from the human. The state of a UV searching for an object or exiting the maze was further distinguished by features such as whether or not the UV was following a known path to the goal, and if another unassigned object was closer to the UV than its currently assigned destination.

IV.C.2. A State Similarity Metric

A similarity function $\phi(s^i, s^j)$, which defines the similarity of state s^i to state s^j, for RESCU can also be derived from the decision tree shown in Figure 3. In the decision tree, features that have a higher impact on human and UV behavior are placed higher in the tree. Thus, states in the same subtree tend to have a higher similarity than those that are not in the same subtree. This point can be exploited to construct a similarity metric.

Formally, let $g(s^i, s^j)$ denote the length of the path in the decision tree from s^i to the nearest common ancestor of s^i and s^j. For example, $g(s^1, s^2) = 1$ since the states s^1 and s^2 share the same parent, whereas, $g(s^1, s^5) = 3$ since the nearest common ancestor is three levels up the tree. Then, $\phi(s^i, s^j)$ is given by

$$\phi(s^i, s^j) = \frac{1}{g(s^i, s^j)^c}, \tag{8}$$

where c is some positive integer that controls the sensitivity of the metric. Increasing c decreases the similarities between states.

IV.C.3. A Time-Weighting Function $f(\cdot)$

Recall that the time-weighting function $f(\cdot)$ is used to weight each sample x based on how closely a sample's mission time (x_τ) matches the target mission time τ. We use a function proportional to a truncated Gaussian, namely

$$f(x_\tau - \tau) = \begin{cases} exp\left(-\frac{(x_\tau - \tau)^2}{2\nu^2}\right) & \text{if } (x_\tau - \tau) < W \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

where ν and W are positive constants. Due to the time-critical nature of RESCU, we chose to truncate the Gaussian function (with W) so that a sample's weight was positive only if x_τ was close to τ.

V. Results

In this section, we discuss the results of the user study, beginning with a summary of the system effectiveness achieved by the human-UV systems in each condition of the study. We then analyze the model's ability to describe human-UV systems and predict how changes in the system will alter its effectiveness.

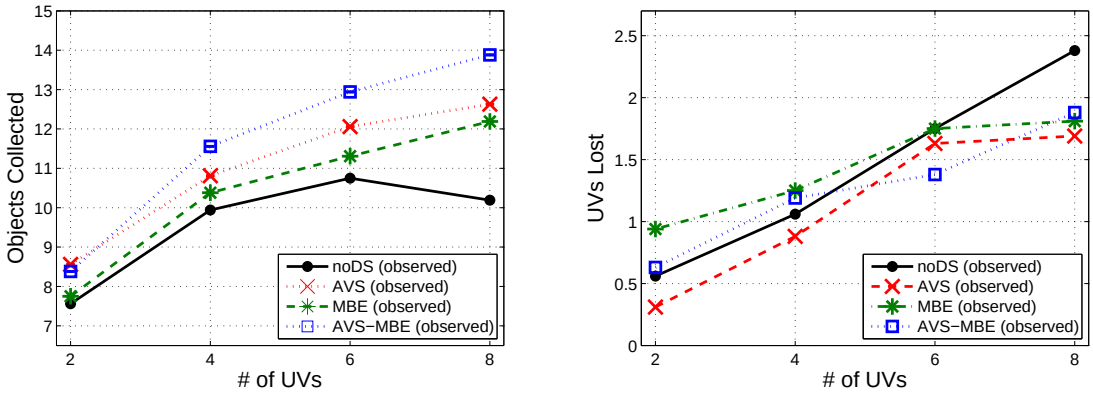


Figure 4. Mean number of objects collected (left) and UVs lost (right) observed in each experimental condition.

V.A. Observed System Effectiveness

In the user study, system effectiveness was determined by the number of objects collected and the number of UVs lost. Figure 4 shows the average number of objects collected and UVs lost in each condition of the study. The figure shows several trends. First, a repeated measures ANOVA shows that team size had a main effect on the number of objects collected ($F(3, 180) = 92.65$, $p < 0.001$). Pairwise comparisons show a significant difference in objects collected for all team sizes ($p < 0.001$ in each case) except six- and eight-UV teams ($p = 0.663$). The number of objects collected increased with team size up to six UVs.

A second trend visible in Figure 4 is that the number of UVs lost also increased with team size ($F(3, 180) = 19.27$, $p < 0.001$). Pairwise comparisons show similar trends to those seen in number of objects collected. Four-UV teams lost more UVs than two-UV teams ($p = 0.003$), six- and eight-UV teams lost more UVs than four-UV teams ($p = 0.033$ and $p = 0.001$, respectively), but there was not a statistical difference in UVs lost between six- and eight-UV teams ($p = 0.663$).

Analysis of decision support type showed a main effect on the number of objects collected ($F(3, 60) = 3.616$, $p = 0.018$), but not on the number of UVs lost ($F(3, 60) = 0.54$, $p = 0.655$). Pairwise comparisons show a statistical difference in objects collected between the noDS and AVS-MBE decision support types ($p = 0.013$), but not in any of the other pairings. However, these results, coupled with the trends shown in the figure, suggest that the AVS and the MBE enhancements both had a positive effect on the number of objects collected by the system, though only the combined effect showed a statistical difference from the noDS condition.

However, in practice, a human-UV system and its enhancements are typically deployed in series and not in parallel. Thus, system designers do not often have the benefit of statistical comparisons between the base system and its enhancements. As an example, consider the situation in which the noDS human-UV system is implemented and deployed. After observations of this system during deployment, various system improvements could be considered if the noDS system was not deemed sufficiently effective. High-fidelity models can be used to accurately evaluate these enhancements in a cost-effective manner.

In the remainder of this section, we analyze the ability of the modeling methodology described in this paper to provide such capabilities. In so doing, we use data from the noDS condition of the user study to model the noDS system for each team size. We use the resulting models to predict the system effectiveness of AVS-, MBE-, and AVS-MBE-enhanced systems, without the benefit of data obtained from observing these systems. We compare these predictions to the results observed in the user study.

V.B. Modeling Observed Results

We constructed four separate models, one corresponding to each team size, from observational data taken from the user study in the noDS conditions. Using the least mean squared error criteria, we set the parameters found in Equations (8) and (9) so that $c = 10$, $\nu = 10$ seconds, and $W = 100$ seconds, though these parameter values had relatively little affect on the accuracy of the models. The resulting models were then used to simulate the noDS human-UV system using Algorithm 1.

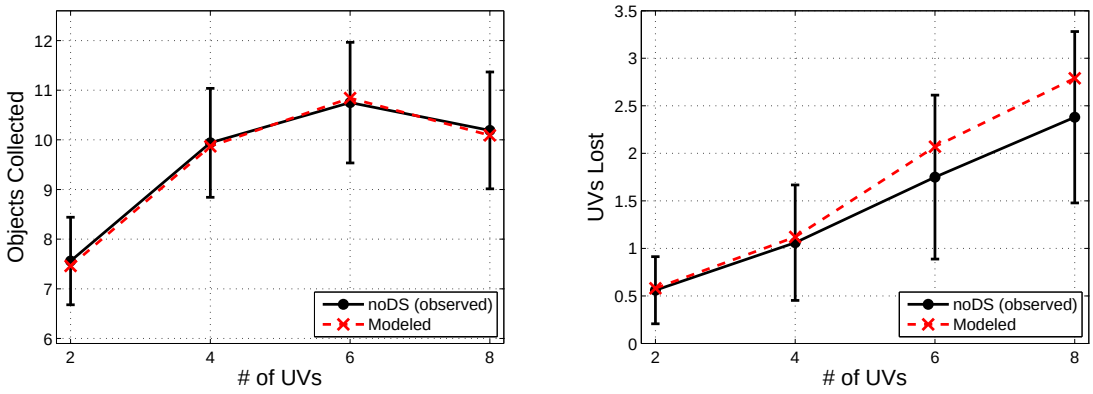


Figure 5. Comparison of observed system effectiveness in the user study in the noDS conditions to the models' estimates of objects collected (left) and UVs lost (right). The models' estimates are the mean of 5,000 simulations of Algorithm 1. Error bars represent a 95% confidence interval on the mean.

Figure 5 compares the average number of objects collected and UVs lost in the noDS condition of the user study to the estimates made by our models. The figure shows that the models' estimates of objects collected are nearly identical to those observed in the user study. Furthermore, the models' estimates of UVs lost are also reasonably good. For two- and four-UV teams, the estimates are almost identical to the observed values. However, for six- and eight-UV teams, the model slightly over-estimates the number of UVs lost, though estimates are still well within the 95% confidence intervals.

The reason for the slight inaccuracies in estimating UVs lost for larger teams appears to be tied to small inaccuracies in the modeling of operator selection strategies. To avoid losing UVs in RESCU, an operator is sometimes required to make precise time-critical decisions in order to ensure that UVs leave the maze in time, whereas the same time-critical precision is not required to collect objects. As the joint state space becomes larger with larger teams, the relatively small number of data samples compared to the size of the joint state space makes it difficult to model operator strategies with sufficient precision. As a result, the model slightly over-estimates the number of UVs lost in larger UV teams.

Despite these small inaccuracies, the estimates are, overall, reasonably good. They demonstrate that this modeling methodology is able to describe the performance of the system in the noDS condition. However, these results do not demonstrate predictive ability since the model is only duplicating observed results. To be predictive, the model must have the capability of predicting the effectiveness of the system under alternate conditions. Such alternate conditions include changes in the system design itself, including the AVS, MBE, and AVS-MBE enhancements.

V.C. Predicting the Effects of System Design Modifications

We now assess the models' ability to predict the effectiveness of the system with the AVS, MBE, and AVS-MBE enhancements. We discuss each set of predictions separately.

V.C.1. Predicting the Effects of the AVS Enhancement

To predict how the AVS enhancement would change the human-UV system's effectiveness, we must determine which aspects of the team will be affected and in what way they will change. These anticipated changes in the system must then be reflected in the individual samples contained in the model, which entails either deriving a new set of samples, or editing the existing samples. We use the latter approach in this paper.

Since the AVS assists in human-UV interactions, it will mostly affect $\mathcal{II}$; we assume that the other structures are left unchanged. To capture the change in $\mathcal{II}$ induced by the AVS enhancement, we edit the samples in $\Theta^{\mathcal{II}}$ in which the operator identified a city on the map. In these samples, the amount of time taken to identify the city should be altered to reflect the change in search times caused by the AVS. We estimated the city search times with the AVS enhancement using data from a different user study, and substituted these new search lengths into the samples of the model. On average, the AVS decision support tool reduced city search times by approximately five seconds.

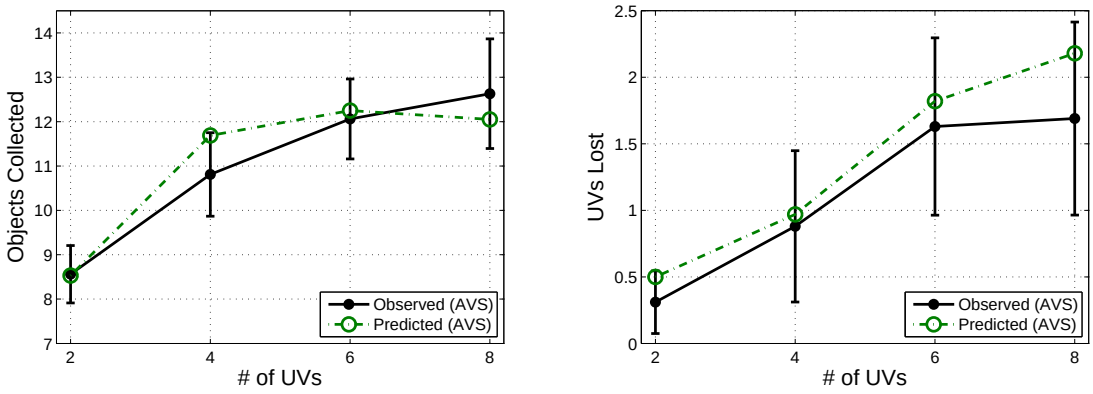


Figure 6. Comparison of model predictions to observed data from the AVS condition. The predictions are the mean of 5,000 simulations of Algorithm 1. Error bars represent a 95% confidence interval on the mean.

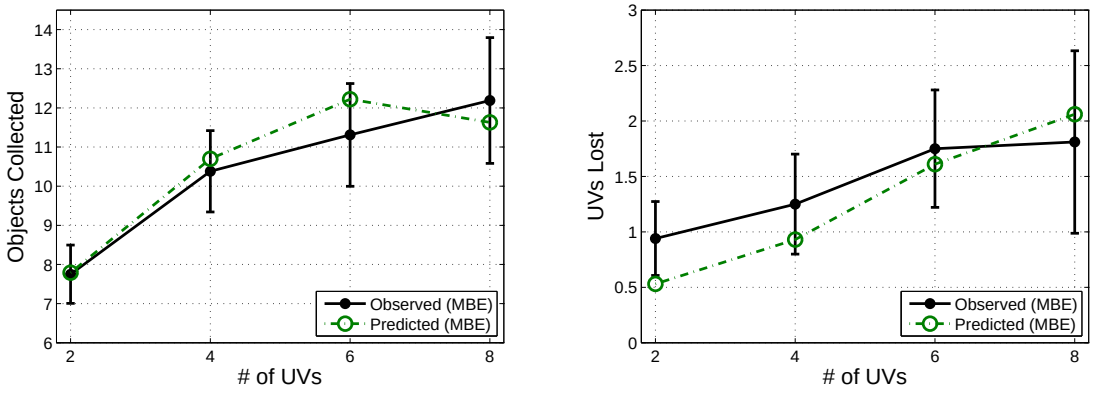


Figure 7. Comparison of model predictions to observed data from the MBE condition. The predictions are the mean of 5,000 simulations of Algorithm 1. Error bars represent a 95% confidence interval on the mean.

After modifying the samples in $\Theta^{\mathcal{II}}$ from the noDS condition to model the effects of the AVS enhancement, we simulated the AVS-enhanced team using the discrete event simulation outlined in Algorithm 1. We then adjusted the simulation's predictions of system effectiveness to account for the initial errors present in the model. Specifically, we multiply the simulation's predictions by the ratio of the observed values to the modeled values in Figure 5. The resulting predictions are shown in Figure 6 along with the observed effectiveness of the AVS-enhanced system in the user study. For each team size, the models' predictions are reasonably accurate for both objects collected and UVs lost, as all predictions fall within the 95% confidence intervals. Thus, for this case, the model was able to predict how changes in the human-UV interface affect the system's effectiveness.

V.C.2. Predicting the Effects of the MBE Enhancement

The MBE enhancement alters a UV's behavior in the absence of human-UV interactions. Thus, whereas the AVS enhancement primarily affected $\mathcal{II}$, the MBE enhancement primarily affects $\mathcal{NI}$. As such, to simulate the MBE enhancement, we must modify the samples in $\Theta^{\mathcal{NI}}$ to reflect this new behavior.

To estimate how each sample $x \in \Theta^{\mathcal{NI}}$ would change due to the MBE enhancement, we must first identify when the MBE enhancement would cause changes in each sample's state transition sequence. Second, we must determine new state transition sequences for these samples. We used MBE's rules for automated goal assignment to determine when state transition sequences would change. To estimate UV behavior at these times, we assumed that a UV's behavior would be similar to a UV's state transition sequence after the operator assigned the UV a goal in the noDS condition. Thus, state transition sequences in $\Theta^{\mathcal{NI}}$ that would be altered by the MBE enhancement can be edited with state transition sequences in $\Theta^{\mathcal{NI}}$ that came directly after goal assignments.

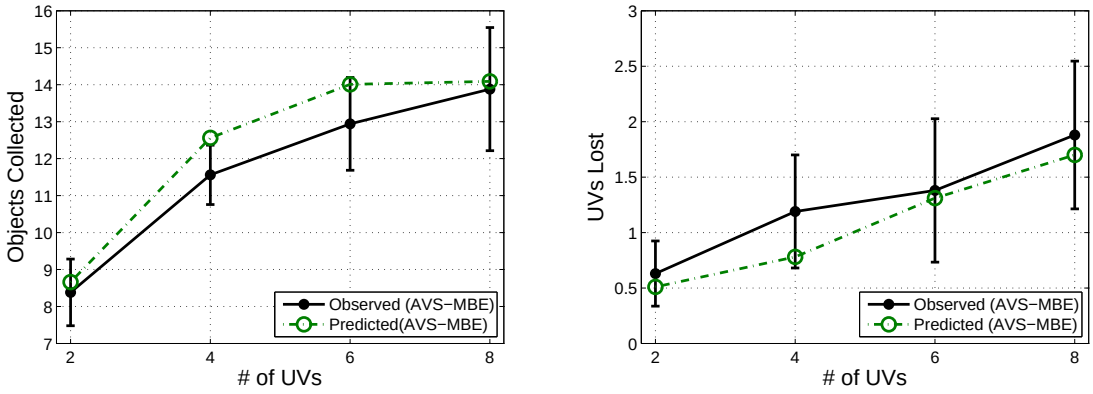


Figure 8. Comparison of model predictions to observed data from the AVS-MBE condition. The predictions are the mean of 5,000 simulations of Algorithm 1. Error bars represent a 95% confidence interval on the mean.

After modifying each sample $x \in \Theta^{\mathcal{NI}}$ as described, we predicted the effectiveness of the MBE-enhanced system using the same process as we used for the AVS condition. Figure 7 compares these predictions to the observed performance of the MBE-enhanced system in the user study. For number of objects collected, the figure shows that the model makes reasonably good predictions for each team size, as all predictions fall within the 95% confidence interval. However, the predictions for UVs lost, though accurate for six- and eight-UV teams, are low for smaller UV teams. This is particularly true of the two-UV case in which the prediction is outside of the 95% confidence interval. While the model predicted a slight drop in number of UVs lost from the noDS condition for small teams, the number of UVs lost in these conditions actually increased. This trend, however, is not statistically significant. The increase in UVs lost could be due to operators' over-trust in the UVs' ability to remove themselves from the maze in sufficient time. Since the model does not explicitly account for operator trust, we leave further analysis and inclusion of this factor to future work.

V.C.3. Predicting the Effects of the AVS-MBE Enhancement

The AVS-MBE enhanced system can be simulated by combining the changes in the models used to simulate the AVS- and MBE-enhancements individually. Figure 8 compares predictions of system effectiveness made by the models against those observed in the user study. Once again, the predictions made by the model are reasonable, though not perfect. For each team size, the predictions of objects collected are higher than the observed results, while the predictions of UVs lost are lower than the observed results. However, with the exception of the prediction of objects collected in the four-UV condition, all predictions are with the 95% confidence interval. These results demonstrate that the model is capable of giving reasonably good predictions even for multiple design changes.

V.D. Implications for Human-UV System Design

In this section, we have shown that our models can both describe and predict the effectiveness of human-UV systems consisting of a single operator and multiple UVs. Thus, the modeling methodology described in this paper could be of value to designers of human-UV systems. For example, consider the situation in which the noDS system has been designed, implemented, and deployed. Since the noDS system often loses a substantial number of UVs and collects, on average, no more than half of the total possible objects (Figure 4), it would be desirable to alter the system to improve its effectiveness. Our modeling methodology can evaluate the effectiveness of various potential system alterations, including the AVS, MBE, and AVS-MBE enhancements, without the benefit of implementing them.

The predicted effects of the AVS, MBE, and AVS-MBE enhancements on the system's effectiveness are shown together in Figure 9. The figure shows predictions for objects collected (Figure 9a), UVs lost (Figure 9b), and system score (Figure 9), which is the objective function given in Equation (7) that subjects were asked to maximize. The predictions show several key insights into the AVS, MBE, and AVS-MBE enhancements. First, the predictions indicate that each of the design enhancements would increase the

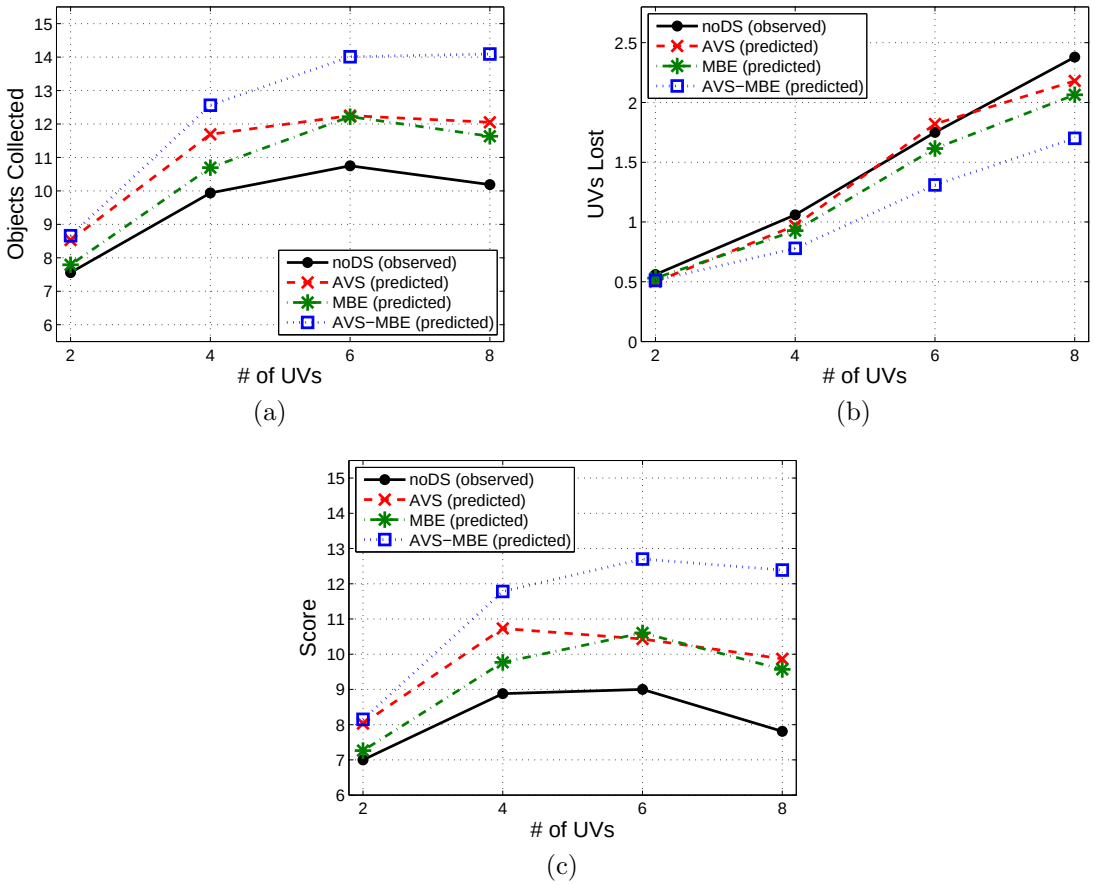


Figure 9. Comparison of the predicted effectiveness of various design improvements with respect to (a) objects collected, (b) UVs lost, and (c) system score.

effectiveness of the system. Implemented alone, our models predict that the AVS and MBE enhancements would produce moderate improvements for each team size. The model predicts that the AVS enhancement will increase the system score by 15%, 21%, 16%, and 26% for two-, four-, six-, and eight-UV teams, respectively. Similarly, the MBE enhancement is predicted to increase the system score by 4%, 10%, 18%, and 23%, respectively. These results indicate that, while the MBE and AVS enhancements are predicted to have similar impact on the system's effectiveness in larger teams, the AVS enhancement would be more effective for two- and four-UV teams. This trend is also present in the observed data in the user study (Figure 4).

Predictions indicate that the combined AVS and MBE enhancements would substantially improve the system's effectiveness. In fact, the AVS-MBE enhancement is predicted to improve the system score by 16%, 33%, 41%, and 59% in two-, four-, six-, and eight-UV teams, respectively. These improvements are due to both increases in number of objects collected and decreases in number of UVs lost.

However, these predicted increases in system score are potentially misleading, as system designers must also consider the costs of performance increases. Since increases in UV capabilities would likely increase the UVs' cost, a system designer must also alter the system objective function from Equation (7) to:

$$Score = ObjectsCollected - (UVCostRatio \cdot UVsLost), \tag{10}$$

where $UVCostRatio$ is the ratio of the cost of a UV in the original implementation (noDS) to the cost of a UV equipped with AVS and MBE capabilities.

Figure 10 shows the effects that different values of $UVCostRatio$ have on the predicted system scores of the AVS-MBE-enhanced system. Even when the cost of a UV is doubled, the model predicts that the AVS-MBE-enhancement would still increase the system score by 9%, 24%, 27%, and 36% for two-, four-, six-, and eight-UV teams, respectively.

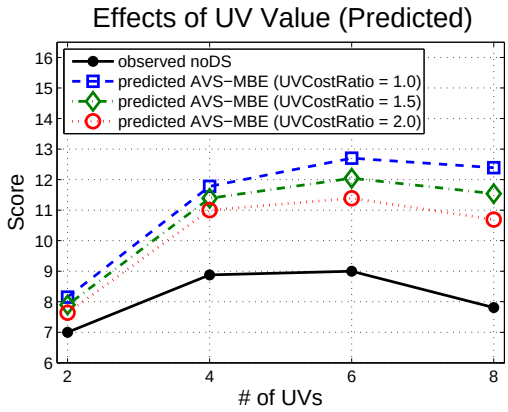


Figure 10. Predicted system score of the AVS-MBE enhanced system for various $UVCostRatio$'s.

VI. Conclusions and Future Work

In this paper, we have described a methodology for modeling human-UV systems consisting of a single operator and multiple UVs. In this model, the stochastic behavior of both the human operator and UVs is constructed using data obtained from observing the human-UV system. These structures can then be used to describe and predict the behavior of the human-UV system in previously unobserved situations, such as when the human-UV interface or UV autonomy levels are altered. Via a user study, we have shown that the model can adequately predict the effects of changes in UV autonomy and the human-UV interface. These results have significant implications for the design and implementation of future human-UV systems.

While these results are encouraging, the model has a number of limitations that should be addressed in future work. First, it is limited to UV teams performing independent tasks. Future work should consider how this model can be extended to collaborative UV teams. Second, the modeling methodology we have proposed in this paper does not explicitly consider the cognitive state of the human operator. While our models gave reasonably good predictions for the scenarios discussed in this paper, we anticipate that we would need to explicitly model the human operator's cognitive state in order to generate accurate predictions in some situations, such as when fatigue could significantly hamper performance.

Acknowledgments

This work was funded by MIT Lincoln Laboratory.

References

[1]A. P. Tvaryanas, W. Platte, C. Swigart, J. Colebank, and N. L. Miller. A resurvey of shift work-related fatigue in MQ-1 Predator unmanned aircraft system crewmembers. Technical Report NPS-OR-08-001, Naval Postgraduate School, Monterey, CA, March 2008.

[2]T. B. Sheridan and W. L. Verplank. Human and computer control of undersea teleoperators. Technical report, Man-Machine Laboratory, Massachusetts Institute of Technology, Cambridge, MA, 1978.

[3]P. J. Mitchell, M. L. Cummings, and T. B. Sheridan. Mitigation of human supervisory control wait times through automation strategies. Technical report, Humans and Automation Laboratory, Massachusetts Institute of Technology, June 2003.

[4]D. B. Kaber 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*, 5(2):113–153, 2004.

[5]J. Wang and M. Lewis. Human control for cooperating robot teams. In *Proceeding of the 2nd Annual Conference on Human-robot Interaction*, pages 9–16, 2007.

[6]M. A. Goodrich, T. W. McLain, J. D. Anderson, J. Sun, and J. W. Crandall. Managing autonomy in robot teams: observations from four experiments. In *Proceeding of the 2nd Annual Conference on Human-robot Interaction*, pages 25–32, 2007.

[7]C. Miller, H. Funk, P. Wu, R. Goldman, J. Meisner, and M. Chapman. The playbook approach to adaptive automation. In *Proceedings of the Human Factors and Ergonomics Society 49th Annual Meeting*, 2005.

[8]C. E. Nehme, B. Mekdeci, J. W. Crandall, and M. L. Cummings. The impact of heterogeneity on operator performance in futuristic unmanned vehicle systems. Submitted to *the C2 International Journal*, December 2007.

[9]J. Wang and M. Lewis. Assessing cooperation in human control of heterogeneous robots. In *Proceeding of the 3rd Annual Conference on Human-robot Interaction*, pages 9–16, 2008.

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

[11]M. A. Goodrich, D. R. Olsen Jr, J. W. Crandall, and T. J. Palmer. Experiments in adjustable autonomy. In *Proceedings of IJCAI Workshop on Autonomy, Delegation and Control: Interacting with Intelligent Agents*, 2001.

[12]D. R. Olsen Jr. and S. B. Wood. Fan-out: Measuring human control of multiple robots. In *Proceedings of the Conference on Human Factors in Computing Systems*, 2004.

[13]D. R. Olsen Jr. and M. A. Goodrich. Metrics for evaluating human-robot interactions. In *NIST's Performance Metrics for Intelligent Systems Workshop*, Gaithersburg, MA, 2003.

[14]J. W. Crandall and M. L. Cummings. Identifying predictive metrics for supervisory control of multiple robots. *IEEE Transactions on Robotics*, 23(5), October 2007.

[15]T. B. Sheridan and M. K. Tulga. A model for dynamic allocation of human attention among multiple tasks. In *Proceedings of the* 14^{th} *Annual Conference on Manual Control*, 1978.

[16]S. Mau and J. Dolan. Scheduling to minimize downtime in human-multirobot supervisory control. In *Workshop on Planning and Scheduling for Space*, 2006.

[17]H. Neth, S. S. Khemlani, B. Oppermann, and W. D. Gray. Juggling multiple tasks: A rational analysis of multitasking in a synthetic task environment. In *Proceedings of the* 50^{th} *Annual Meeting of the Human Factors and Ergonomics Society*, 2006.

[18]P. Squire, G. Trafton, and R. Parasuraman. Human control of multiple unmanned vehicles: effects of interface type on execution and task switching times. In *Proceeding of the* 1^{st} *Annual Conference on Human-robot Interaction*, pages 26–32, New York, NY, USA, 2006. ACM Press.

[19]M. A. Goodrich, M. Quigley, and K. Cosenzo. Task switching and multi-robot teams. In *Proceedings of the Third International Multi-Robot Systems Workshop*, 2005.

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

[21]M. L. Cummings, C. Nehme, J. W. Crandall, and P. J. Mitchell. Predicting operator capacity for supervisory control of multiple UAVs. *Innovations in Intelligent UAVs: Theory and Applications,* Ed. L. Jain, 2007.

[22]R. M. Yerkes and J. D. Dodson. The relation of strength of stimulus to rapidity of habit-formation. *Journal of Comparative Neurology and Psychology*, 18:459–482, 1908.

[23]D. K. Schmidt. A queuing analysis of the air traffic controller's workload. *IEEE Transactions on Systems, Man, and Cybernetics*, 8(6):492–498, 1978.

[24]W. B. Rouse. *Systems Engineering Models of Human-Machine Interaction.* New York: North Holland, 1983.

[25]M. L. Cummings and S. Guerlain. Developing operator capacity estimates for supervisory control of autonomous vehicles. *Human Factors*, 49(1):1–15, 2007.

[26]J. D. Lee and N. Moray. Trust, self-confidence, and operators' adaptation to automation. *International Journal of Human-Computer Studies*, 40(1):153–184, January 1994.

[27]M. R. Endsley. Design and evaluation for situation awareness enhancement. *Proceedings of the Human Factors Society 32nd Annual Meeting*, pages 97–101, 1988.

[28]J. Drury, J. Scholtz, and H. A. Yanco. Awareness in human-robot interactions. In *Proceedings of the IEEE Conference on Systems, Man and Cybernetics*, Washington, DC, 2003.

[29]M. L. Cummings. Automation bias in intelligent time critical decision support systems. In *AIAA* 1^{st} *Intelligent Systems Technical Conference*, pages 33–40, September 2004.

Appendix D

Comparing the Predicted Effects of System Enhancements

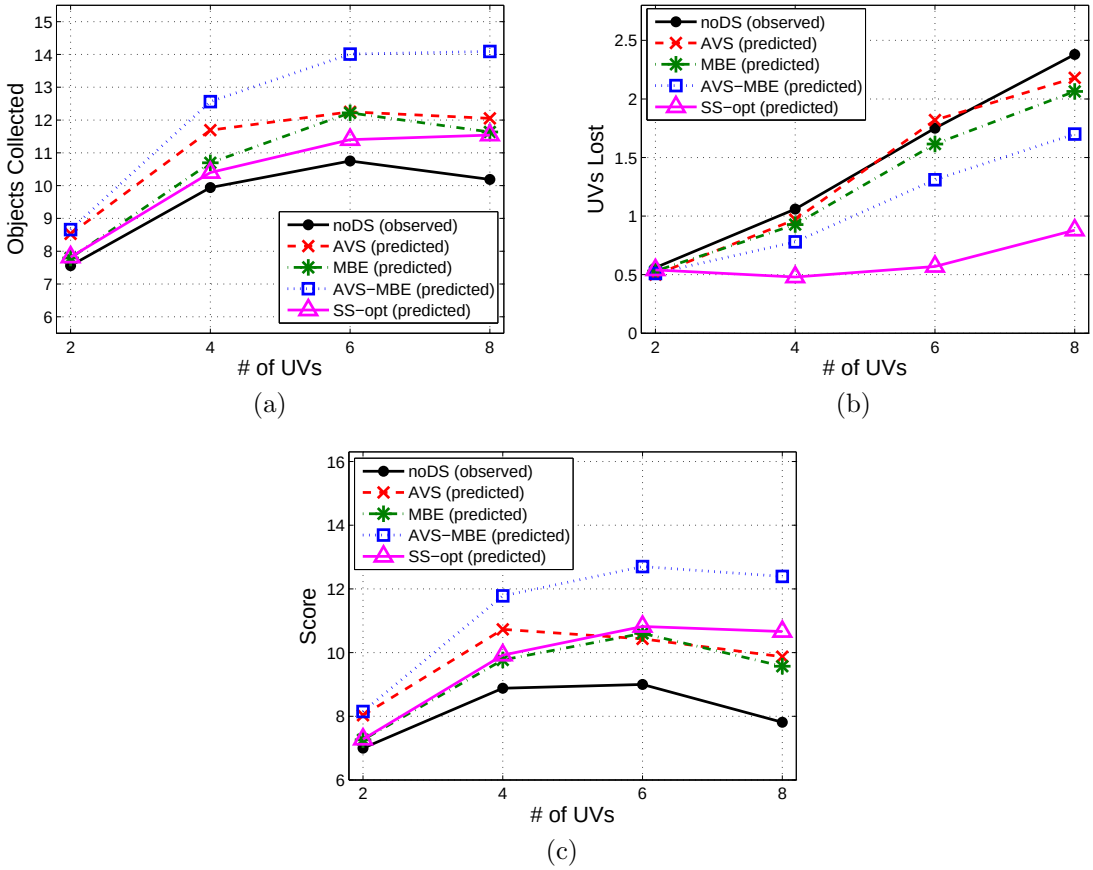


Figure 1: Comparison of the predicted effectiveness of various design improvements with respect to (a) objects collected, (b) UVs lost, and (c) system score. *SS-Opt* refers to the predicted effectiveness of using the recommended selection strategy. The other labels are equivalent to those used in Phase 2.