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

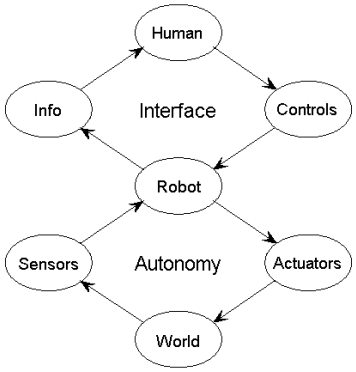

Fig. 1. The two control loops of an HRT consisting of a single human operator and a single (remote) robot. Adapted from [12].

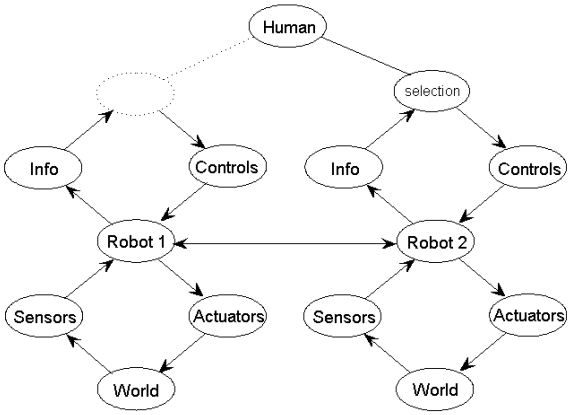

Fig. 2. In HRTs consisting of a single human and multiple robots, the human must determine how to distribute his/her attention between the robots.

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).

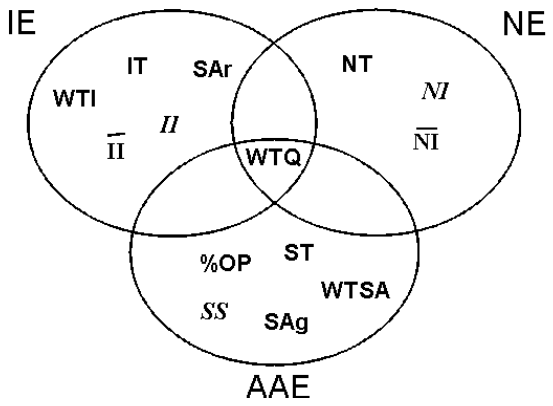

Fig. 3. A set of metric classes ($\{IE, NE, AAE\}$) and various metrics drawn from those classes.

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

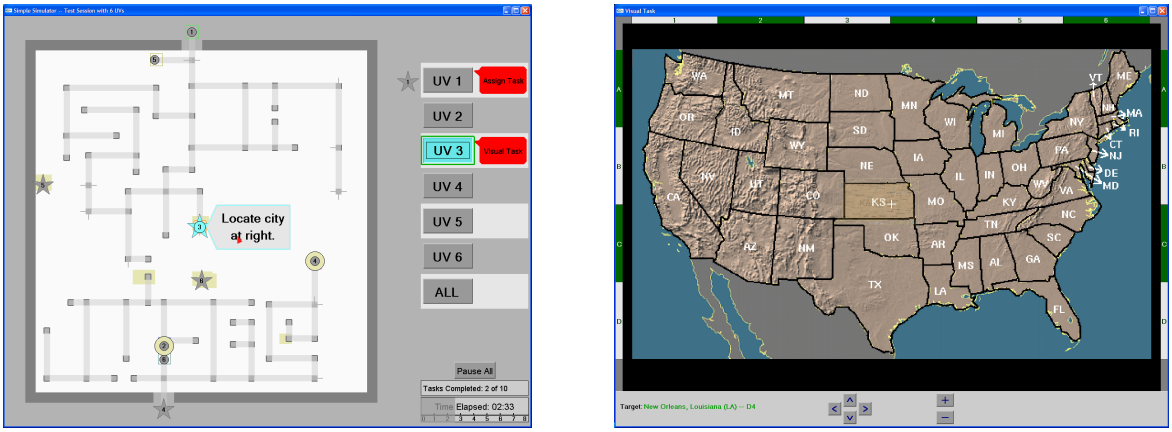

Fig. 4. Two displays used in the experiment (on separate monitors). The display at left shows the map of the maze, the locations of the robots, their destinations, the locations of the objects to be gathered, etc. The display at right shows the interface for the abstract visual task.

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

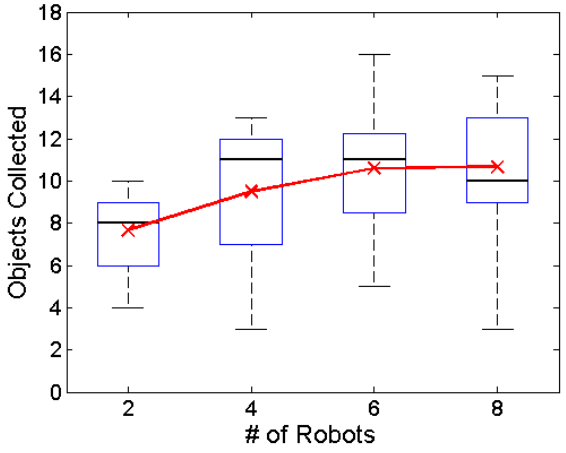

Fig. 5. Means and distributions of number of objects collected for each robot team size.

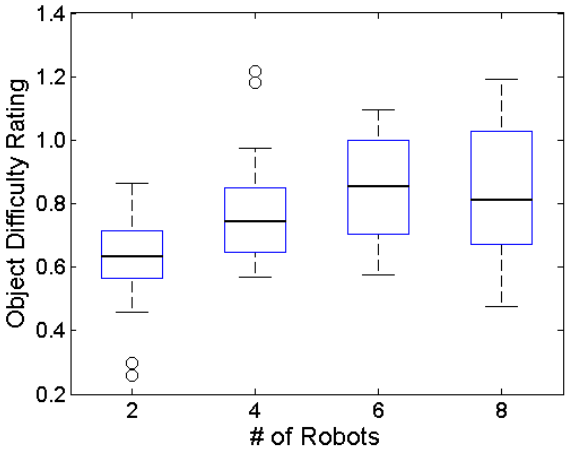

Fig. 6. Box plot showing difficulty of the objects collected under each robot team size.

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.

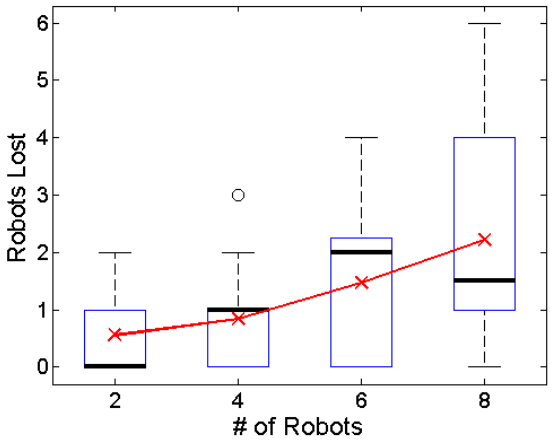

Fig. 7. Means and distributions of number of robots lost for each robot team size.

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

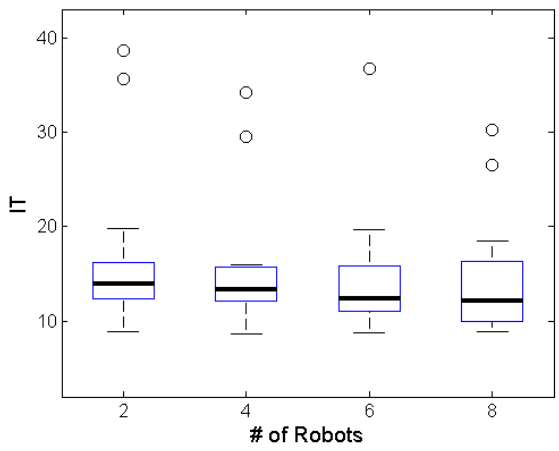

Fig. 8. Distributions of interaction times for different team sizes.

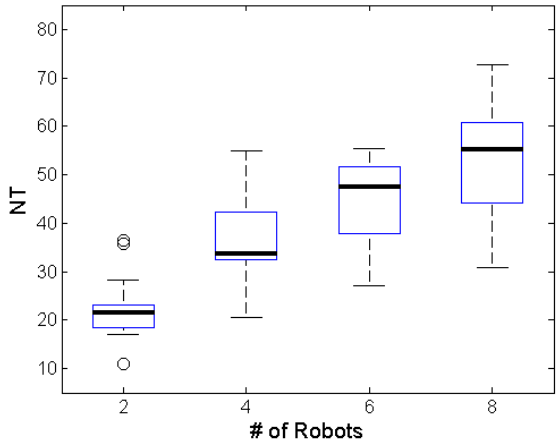

Fig. 9. Distributions of neglect times for different team sizes.

$(F(3,15) = 3.29, p = 0.049)$. 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

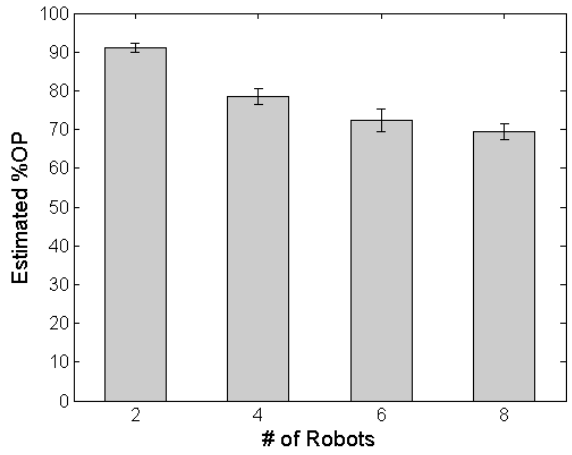

Fig. 10. Estimated percentage of optimal robot selections by the operators.

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.

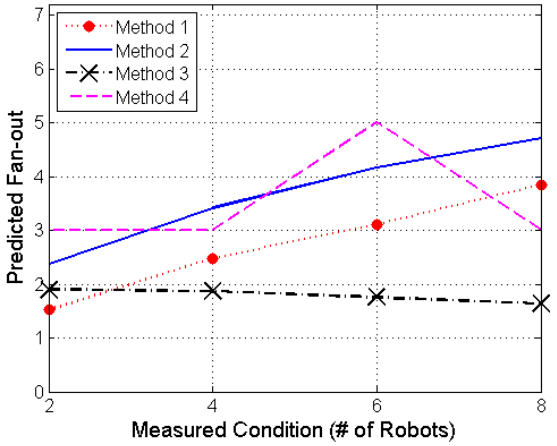

Fig. 11. Fan-out predictions of four different methods for four measured conditions (x-axis).

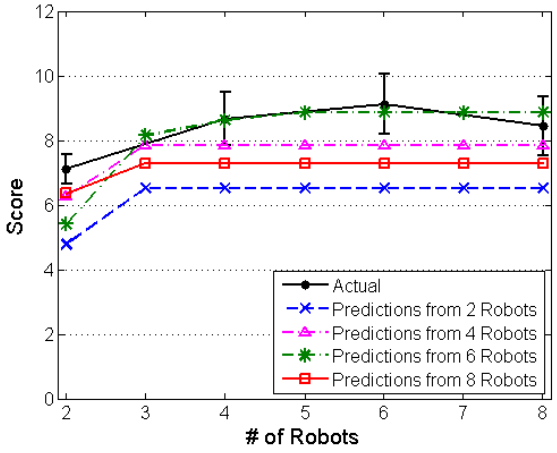

Fig. 12. Predictions of overall system effectiveness using method 4 [12]. *Actual* refers to the mean (and standard error) of observed scores in the user study and *Predictions from N Robots* shows the predictions (for all team sizes shown along the x-axis) from the N-robot measured condition.

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.

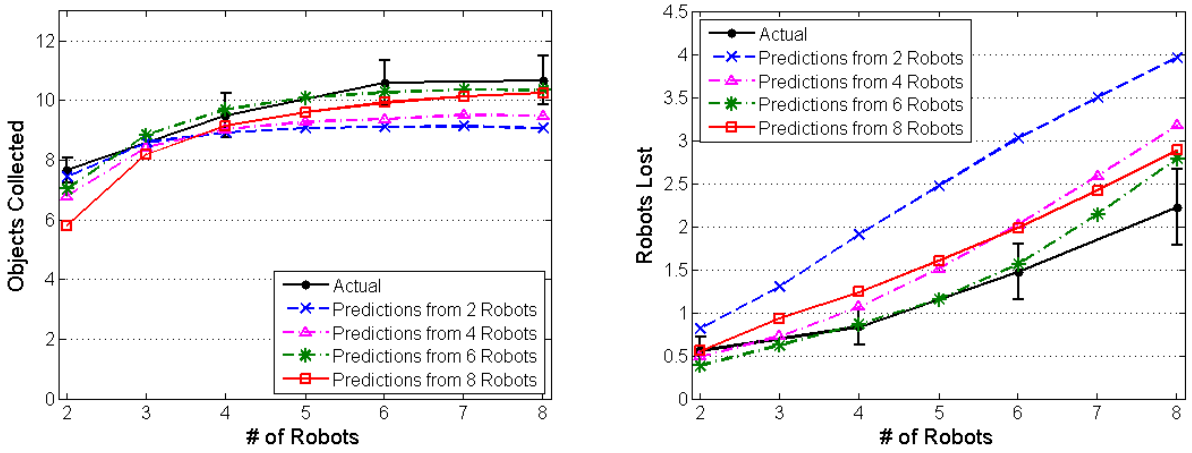

Fig. 13. Predictions of objects collected (left) and robots lost (right) compared to the sample means obtained in the user study. *Actual* refers to the mean (and standard error) of observed scores in the user study and *Predictions from N Robots* shows the predictions (for all team sizes shown along 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.