Task Lists for Human-Multiple Robot Interaction*

Izzet Can Envarli
EECS Department
Vanderbilt University
Nashville, TN
izzet.c.envarli@vanderbilt.edu

Julie A. Adams
EECS Department
Vanderbilt University
Nashville, TN
julie.a.adams@vanderbilt.edu

Abstract – **Interest in developing large multiple robot systems continues to grow, however, the autonomous technology to permit such systems to work on their own is lacking. Therefore, it is necessary to develop interaction capabilities that will permit a small number of humans to supervise a larger number of robots. The objective was to develop a preliminary interface that provided an ability to monitor and interact with teams of robots rather than individual robots. Specifically, this work focused on understanding how a task list would affect the human's interaction with the system. This paper describes the preliminary interface design, the user evaluation and the associated results. The interface has tools for creating robot teams for new tasks, managing teams, and attending to problems in teams. The primary interface feature is a task list that collects all relevant tasks and information from the environment in a single window, sorted by priority. Twenty novice users evaluated the system.**

Index Terms – human-robot interaction, multiple robots, task list, task priority

I. INTRODUCTION

Many current robotic systems are single robot systems that require heavy user interaction in the form of teleoperation [1, 2]. These systems provide limited autonomous capabilities. There has been an interest in expanding these systems to include more robots but such systems still rely on humans' teleoperating the robots and limited automation. As the number of robots increases, so do the number of humans working with the robots. Very little progress has been made in developing interfaces that permit a small number of humans to support a large team of robots. A good example of the progress that has been made is the work by Parasuraman et al. [3] in the development of delegation-based interfaces. They have incorporated the Playbook^{TM} mechanism in order to permit a single user to interact with up to eight robots.

Fundamental work is required to understand how to effectively develop interfaces for multiple robot systems that have limited autonomy. The desire is to provide interfaces that support the human's ability to interact with a large number of robots. In particular, visualization techniques are required to permit the operator to understand how the system's robots are performing.

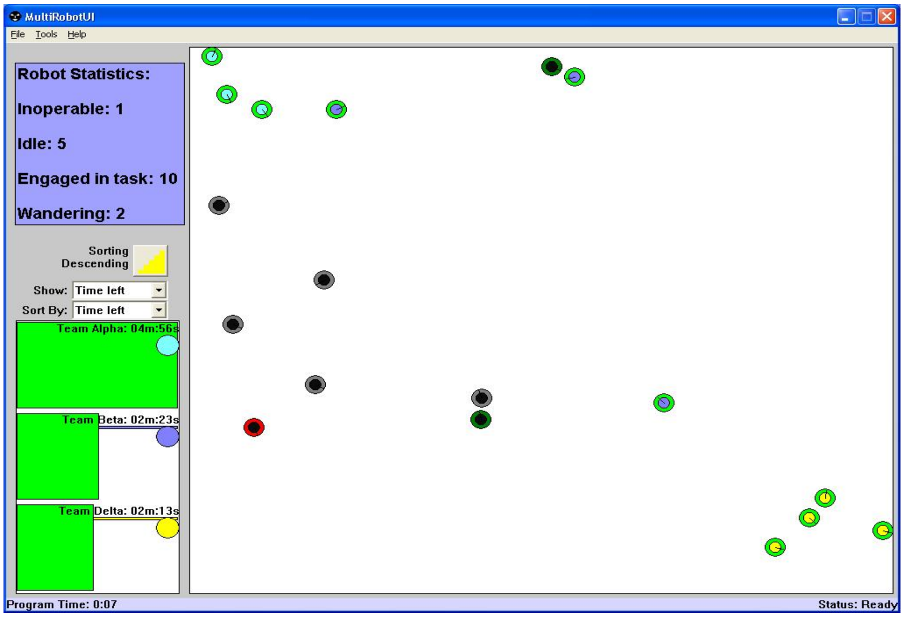
One important aspect is raising the operator's situational awareness. Defining Situational Awareness is difficult. Nofi [4] has shown that there are many possible definitions. Situational awareness (SA) has been described in three levels [5]. SA level 1 involves the perception of simple information regarding the environment. SA level 2 involves combining information from the interface to perform user tasks, creating a level of comprehension. SA level 3 involves making predictions about the future, based on current information.

Scholtz et al. [6] have been developing HRI specific situational awareness measurement tools based upon the broadly accepted situation awareness measurement tools. Daily et al. [7] have developed an augmented reality technique for an improved interface, in which they represent robots as single pixels. Baker et al. [8] have focused on a specific application: Urban search and rescue situations. They have decided to center activity and attention around a video window. Lastly, Yanco et al. [9] have compared different interfaces and teams in a single study.

Larger systems of robots will likely result in subteams being formed to complete various sub-tasks. There will be levels of importance to such sub-tasks that will need to be understood by an operator. For instance, in a search-and-rescue environment, a team that is working on saving a particular victim should have higher priority than a team that is building a map of the environment. If a problem arises in both groups simultaneously, the supervisor should solve the problem for the rescue team first. It may be obvious to the supervisor in this case, but not all situations will be as clear. In certain situations, supervisors may not know or be aware of the task priorities. If a factor of the system utility is task priorities then operators may fail to maximize the system utility even when their performance level is high.

---

* This work was supported by the Vanderbilt University Summer Undergraduate Research Program.



Robot Statistics:

Inoperable: 1

Idle: 5

Engaged in task: 10

Wandering: 2

Sorting Descending

Show: Time left

Sort By: Time left

Team Alpha: 04m:56s

Team Beta: 02m:23s

Team Delta: 02m:13s

Program Time: 0:07

Status: Ready

Fig. 1 The main HRI window.

Section II of this paper presents the preliminary human-robot interface. The user evaluation experimental design is provided in Section III while Section IV presents the evaluation results. Section V provides a discussion and Section VI outlines future work and the conclusions.

II. THE HUMAN-MULTIPLE ROBOT INTERFACE

The human-multiple robot interface (HRI) in this work focuses on the user's supervisor role based on the five roles Scholtz [10] defined. The robots are simulated and completely autonomous. Currently supervisors have complete control over robot assignments to tasks; however this will be handled by a coalition formation algorithm [11]. The supervisor will retain ultimate control over the assignments, but the intent is to simplify the supervisor's task.

The HRI consists of a main window, a task management window, a group information window, and a user task window. The main interface window is depicted in Fig. 1.

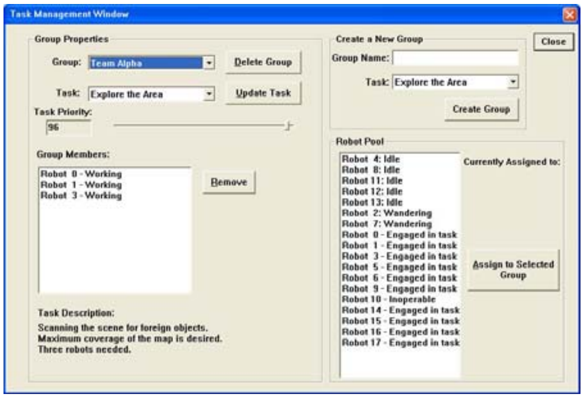
The environment consists of eighteen robots and three robot teams, which are assigned tasks. Each task requires a predefined number of robots; no progress is made unless the team has the appropriate number of robots. The main map conveys the location and the status of each robot in the environment. A circle with a center and an outer ring represents each robot. The outer ring color indicates the robot's status (Gray: 'idle', dark gray: 'wandering', red: 'inoperable', and bright green: 'engaged in task'). The color of the robot's center identifies the robot's team. A robot with a black center is currently unassigned. This structure allows a supervisor to quickly locate the team members.

The bar chart (bottom left corner of Fig. 1) provides a means for accessing team level information. Its structure allows comparison of teams. Team-related information is displayed and sorted by task time remaining, task priority, percentage of completion and team ID, using the dropdown lists. Right-clicking a bar provides that team's popup menu. The color of each bar indicates the team status. Green indicates regular progress towards task completion, orange indicates that the team has encountered a problem and is attempting to resolve the issue without human intervention, and red indicates that supervisor intervention is required.

The top left corner of Fig. 1 shows information regarding the robot population. This information includes the number of idle, wandering, working, and inoperable robots.

The task management window is provided in Fig. 2. It is used for creating and destroying teams, assigning new tasks to teams, inspecting the robot pool, adding or removing robots to and from teams, and adjusting teams' task priorities. Supervisors have full access to managing tasks from this window.

Fig. 3 provides the team information window that conveys information regarding a specific team. Task priority and team members can be modified. Information regarding percentage of task completion, expected time remaining, time already committed, task type, and task priority are accessible. Extra information about each member robot (Sensory capabilities) is also available. The supervisor may also modify the robot team composition by adding or removing members.



Group: Team Alpha
Task: Explore the Area
Task Priority: 96

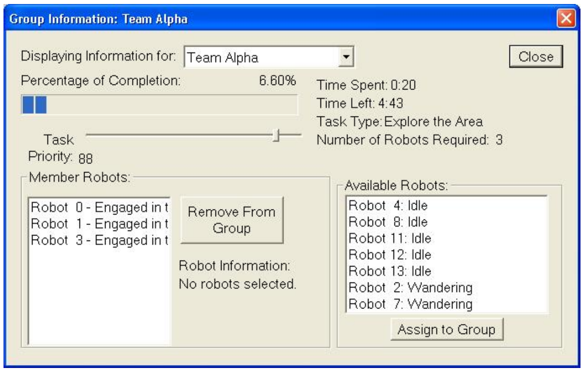
Group Members:
Robot 0 - Working
Robot 1 - Working
Robot 3 - Working

Task Description:
Scanning the scene for foreign objects.
Maximum coverage of the map is desired.
Three robots needed.

Group Name:
Task: Explore the Area

Robot 4: Idle
Robot 8: Idle
Robot 11: Idle
Robot 12: Idle
Robot 13: Idle
Robot 2: Wandering
Robot 7: Wandering
Robot 0 - Engaged in task
Robot 1 - Engaged in task
Robot 3 - Engaged in task
Robot 5 - Engaged in task
Robot 6 - Engaged in task
Robot 9 - Engaged in task
Robot 10 - Inoperable
Robot 14 - Engaged in task
Robot 15 - Engaged in task
Robot 16 - Engaged in task
Robot 17 - Engaged in task

Fig. 2 Task Management Window.



Displaying Information for: Team Alpha
Percentage of Completion: 6.60%
Time Spent: 0:20
Time Left: 4:43
Task Type: Explore the Area
Number of Robots Required: 3
Task Priority: 88

Robot 0 - Engaged in t
Robot 1 - Engaged in t
Robot 3 - Engaged in t

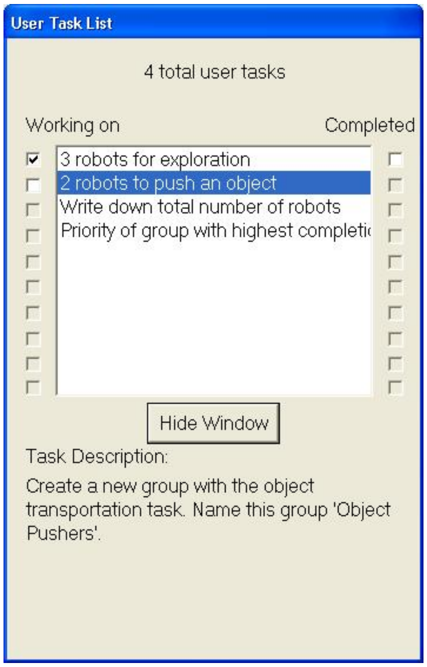
Robot Information:
No robots selected.

Robot 4: Idle
Robot 8: Idle
Robot 11: Idle
Robot 12: Idle
Robot 13: Idle
Robot 2: Wandering
Robot 7: Wandering

Fig. 3 Team Information Window.

A primary feature introduced in this work is a user task list. The user task list (shown in Fig. 4) permits a supervisor to track relevant events in the environment. Events that appear on this list are malfunctioning robots/robot teams, new tasks requiring new teams, task completions, and any other task-pertinent information to be conveyed to the supervisor. Items in the user task list are sorted by priority so that the most important user tasks are at the top. The supervisor is guided towards working on the most critical items first. This tool may enhance SA level 1 by providing a second path of information flow in the case of failures. If a team member fails and the supervisor does not notice the red outer robot ring or the red bar in the chart, a notification on the task window appears. The *task description* attempts to tap deeper levels of situational awareness by providing suggestions on how to handle a situation.

The task list has certain restrictions. The supervisor must first select the *working on* box for a task and then when it is completed, check the *completed* box to delete the task. The system forces the user to choose the task with the highest priority. However, working on multiple items is allowed since some tasks cannot be completed until certain resources become available. The only restriction is that users cannot skip a task and start working on lower priority tasks. In addition, to prevent disorientation, if a task is checked (*working on*), the insertion of a new task does not replace it, even if the new task has higher priority.



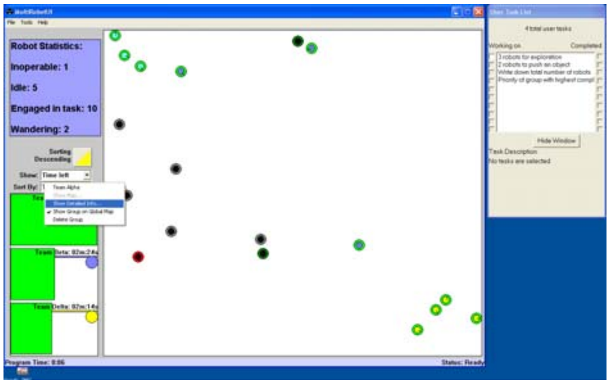
4 total user tasks

Working on		Completed
☑	3 robots for exploration	☐
☐	2 robots to push an object	☐
☐	Write down total number of robots	☐
☐	Priority of group with highest completi	☐

Task Description:
Create a new group with the object transportation task. Name this group 'Object Pushers'.

Fig. 4 User Task List Window.

Fig. 5 provides a screenshot of the interface in use. The user-task list window is docked to the right of the main window. This prevents the user-task list window from being hidden by other windows and provides the supervisor with continuous access to the task list.



Robot Statistics:
Inoperable: 1
Idle: 5
Engaged in task: 10
Wandering: 2

Fig. 5 Screenshot of system in action.

III. USER EVALUATION

A. Apparatus

The interface was evaluated using a predetermined scenario that simulates a real multi–robot environment. The interface did not interact with any robotic systems and was purely simulation. Eighteen robots were used in the scenario. The simulation began with three teams that were already assigned tasks. Robot failures were simulated at predefined times. Participants were responsible for resolving these situations as they arose. In addition, participants were given tasks to create new teams and assign certain tasks to each team. There were also some tasks that involved obtaining information about the environment. These tasks varied in complexity and were geared towards providing an understanding of the user's situational awareness. Some of the identification tasks were SA Level 1 tasks and required recording the number of robots in the environment. Other tasks were geared at the operator's SA Level 2 and required recoding the task completion rate of the team with the highest task priority. The participants' workload was varied as well. There were periods when the participants had no tasks or only one task, and there were periods when numerous tasks were assigned within a short time span. The scenario included a total of 16 tasks. The recorded objective data included:

1. The time required to finish a task, measured from the time that the task was assigned.
2. Accuracy of the user's responses.

The HRI evaluation focus was evaluating the user task list. There were two trials: One that included the user task list (Task list condition) and one without (Index card condition). The index card condition required the examiner to give an index card to the participant when a task was required. If the task was to remedy a situation with a problem robot, an index card was not provided, the participant was expected to note the problem based upon the interface display. The exact same scenario was used in both conditions. Half of the participants completed the task list condition first and the other half completed the index card condition first. The trial order was randomly assigned to participants by alternating the order for each participant. The participants' workload was measured with the NASA TLX questionnaire. The participants completed the questionnaire at the end of each trial.

The task list condition required participants to attend to the task list window and clear the listed tasks. Team failures were included as items on the user task list. The index card condition provided the user tasks with printed 3 x 5 index cards. The name and description of the task were printed on the cards in exactly the same manner as the user task list. The cards were handed to the participant by the experimenter at the predetermined times. The participant was responsible for detecting and attending to team failures. Participants were given response sheets for recording written response (i.e., "Find out how many robots are inoperable. Write this information down.") for both conditions.

B. Participants

There were twenty participants, all of whom were undergraduate students at Vanderbilt University. All participants were Microsoft Windows users, but none had prior training in Robotics or Artificial Intelligence. They varied in age from 18 to 22. Participants were randomly assigned the order of trials by alternating the order of trials with each participant. They were acquaintances of the experimenter and received no compensation for participation.

Participants received training before engaging in the timed trials. They read through a manual and at the completion of each section, the experimenter demonstrated that portion of the reading. In this manner, participants were shown all interface aspects. At the end of this walkthrough, they were given a few moments to interact with the program, during which they had a chance to interact with the user task list as well. Finally, they worked on four tasks that resembled the tasks they were to receive in the trials, to complete their training.

Each full session consisted of the informed consent, training, completing the first trial, completing the NASA TLX form, completing the second trial, completing the NASA TLX form, and finally a debriefing. Each session lasted approximately 40-minutes.

IV. RESULTS

The recorded completion time data for each task was analyzed. The task assignment times were predetermined, and the task finishing times were recorded either by the interface (Task list condition) or by the experimenter (Index card condition). Tasks that involved creating a new team or fixing a malfunctioning team were considered complete once the team resumed progress and was automatically recorded. When the task involved writing information down, the finish time was based on the checking the 'completed' box for the task-list condition (automatically recorded by the system), and the handing back of a task card for the non-list condition (recorded manually by experimenter). Turnaround time was calculated as the time when the task was assigned (handing an index card, insertion of a task in the user task list, or the simulation of a team failure) subtracted from the task completion time. These times were added for each trial to obtain total turnaround times for each participant. Each user task had a different priority, and another total turnaround was calculated factoring in the task priorities. Tasks were ranked by priority, and the turnaround time for each task was divided by the task's priority rank. This helped maximize the penalty for the most important tasks, and minimize the penalty for the least important tasks.

Practice effect had a significant impact on performance. Participants had higher performance on the second trial independent of the condition completed first.

A. *Task Completion Times*

Table 1 provides the results for the mean total turnaround times by trial. The mean total turnaround time was 19 minutes 14 seconds (s) for the task list trials, and 20 minutes 1 second for the index card condition.

TABLE I
Turnaround Times

	Mean	Standard Dev.
User Task List	1154.80 s	272.96s
Index Cards	1201.10 s	416.17s

A within-subjects t-test revealed that $t(19) = 0.41$, $p = 0.85$. Thus no statistically significant difference existed between the two conditions. This result may be due to very high variability caused by practice effects. Participants consistently performed better on the second trial, independent of which condition they completed in the first trial. Therefore, turnaround times were also computed for each trial by condition as shown in Table 2. It shows the means for four different task-trial combinations (Ten people in two groups): Performance on the user task list condition for people that used it in the first trial, performance on the same condition for people that used it in the second trial, performance on the index card condition for people that completed the task on the first trial, and performance on the same condition for people that did it on the second trial.

TABLE II
Turnaround Times Based on Order of Trials

	1st Trial		2nd Trial	
	Mean	St. Dev	Mean	St. Dev
User task list	1314.30s	194.87s	995.30s	250.56s
Index cards	1449.30s	418.32s	952.90s	231.84s

The second trial results are more relevant, as they reflect performance after complete familiarization with the system. Surprisingly, the mean was lower for the index card condition. Participants who used index cards in the second trial averaged a total turnaround time of 952.90s, whereas participants using the task list in the same trial averaged 995.30s. However, a between-subjects t-test analysis revealed $t(18) = 0.39$, $p = 0.86$, showing the difference to be highly insignificant.

B. *Task Priorities*

The task priorities were factored into the computations. For each task, the turnaround time was divided by the task's priority rank, decreasing the penalty for the less important tasks. The results were summed to obtain new total turnaround times. Table 3 provides the mean results for these calculations.

TABLE III
Turnaround Times with Factored-in Task Priorities

	Mean	Standard Dev.
User Task List	262.80s	57.07s
Index Cards	359.25s	139.01s

The user task list clearly outperforms the index card condition. The score for the user task list was 262.80s versus 359.25s for the index cards. A within-subjects t-test ($t(19) = 2.73$, $p < 0.01$) revealed a statistically significance difference. Results were also computed for the different trial orders, depicted in Table 4.

TABLE IV
Turnaround Times with Factored-in Task Priorities and trial Order

	1st Trial		2nd Trial	
	Mean	St. Dev	Mean	St. Dev
User task list	298.20s	43.04s	227.40s	47.30s
Index cards	438.60s	134.86s	279.90s	92.81s

The mean score for the user task list condition when completed as the second trial is smaller (227.40s) than the index card condition (279.90s). A between-subjects t-test analysis for the 2nd trial revealed $t(18) = 1.59$, $p = 0.06$, barely failing to pass the 0.05 significance level. The t-test for the first trial ($t(18) = 3.136$, $p < 0.01$) revealed a clear statistical difference.

C. *Assessment of Workload*

At the end of each trial, participants completed the NASA Task Load Index questionnaire that measures perceived workload. Results are depicted in Table 5. The mean score for the second trial of the task list condition was 28.07 and 45.50 for the index card condition. A between-subjects t-test ($t(18) = 2.66$, $p < 0.01$) has shown this difference to be significant.

TABLE V
Assessment of Workload by Participants Based on the NASA TLX Survey

	1st Trial		2nd Trial	
	Mean	St. Dev	Mean	St. Dev
User task list	43.97	14.08	28.07	11.22
Index cards	38.10	13.98	45.50	17.41

Accuracy was also monitored for participants' responses. Extensive training prior to the tasks aimed to minimize incorrect responses to questions that required a written response. In all cases where participants made mistakes in creating teams, they were able to correct their mistakes. The types of mistakes the participants made included creating a team with the wrong task, creating a team with the wrong name, and trying to assign robots to a team before creating the team. In such cases, a participant's lack in accuracy was indirectly penalized by the increase in turnaround time.

V. DISCUSSION

The results have shown that the task list produced improved performance when the users had little familiarity with the system (1st trial), both with and without taking the task priorities into account. When task priorities were taken into account for the second trial, the task list produced better performance, however the difference barely failed the significance test. These findings suggest that the task list is an effective guide for users that find themselves in an ambiguous situation to improve performance, and that the list is a potentially good method to prioritize tasks.

The standard deviation was much higher for the index card condition during all trials and all measurements. This finding makes that the user task list causes the user's actions and performance more predictable. This can be attributed to the predetermination of the order that participants attended to tasks. An important implication is that by designing a task list that maximizes utility, we can boost a supervisor's performance.

The results of the NASA TLX questionnaire have shown that participants perceived the task list to decrease their overall workload. The difference was larger and more significant for the second trials, implying that after becoming familiar with the system, the participants perceived the task list to require less workload than the index cards. The difference is more significant for the perception of workload than for their performance. There are two possible interpretations: 1) The participants overestimated the change in workload, or 2) The smaller difference in performance is due to the overhead devoted to the extra work (checking and unchecking items) on the task list.

VI. CONCLUSION AND FUTURE WORK

We have determined that using a user task list can be helpful in prioritizing tasks. The results indicate that this concept can be useful in applications where supervisors are not well informed of the tasks and the environment or in applications where task priorities are of great importance.

Certain weaknesses were identified regarding the task list. Participant feedback showed that creating an item for a task's completion did more harm than good. It ended up wasting time that was not worth the change in awareness. Our next design will immediately disband a team once it has completed the assigned task. We also intend to investigate whether the practice effect is enhanced by the Task List condition that demonstrates prioritizing tasks, which may be carried on by participants to the Index Card condition.

Another improvement will be sorting the user tasks. Currently, if a task is checked, nothing can replace it and force it further down in the list. This prevents a less important task from being replaced by more important ones. The new design will include moving a higher priority task up the list once the user checks it. This way, distracting the user's orientation will still be avoided, but devotion to task priorities should be better achieved by alerting the supervisor that the task they just started to work is more important.

Last, our goal is to develop other methods to increase situational awareness. We plan to integrate the user interface with the Player/Stage environment, actual robots and the existing coalition formation algorithm [11].

REFERENCES

[1] Murphy, R. R. (2004). "Human-Robot Interaction in Rescue Robotics," *IEEE Systems, Man and Cybernetics Part C: Applications and Reviews, special issue on Human-Robot Interaction*, Vol. 34, No. 2.

[2] Bruemmer, D. J., Few, D. A., Boring, R. L., Marble, J. L., Walton, M. C. and Nielsen, C. (2005). "Shared Understanding for Collaborative Control," to appear *IEEE Transactions on Systems, Man and Cybernetics – Part A,, S*pecial Issue on Human-Robot Interaction, to appear.

[3] Parasuraman, R., Galster, S., Squire, P., Furukawa, H., and Miller, C. (2005). "A Flexible Delegation-Type Interface Enhances System Performance in Human Supervision of Multiple Robots: Empirical Studies with RoboFlag." To appear: *IEEE Transactions on Systems, Man and Cybernetics – Part A,* Special Issue on Human-Robotic Interaction, to appear.

[4] Nofi, A.A. (2000). "Defining and Measuring Shared Situational Awareness." *Center for Naval Analyses,* November 2000.

[5] Endsley, M. R. (1988). "Design and evaluation for situation awareness enhancement." *Proc. of the Human Factors Society 32nd Annual Meeting,* pp. 97 – 101.

[6] Scholtz, J., Antonishek, B., and Young, J. (2005). "Implementation of a Situation Awareness Assessment Tool for Evaluation of Human-Robot Interfaces." *IEEE Transactions on Systems, Man and Cybernetics – Part A,* Special Issue on Human-Robotic Interaction, to appear.

[7] Daily, M., Cho, Y., Martin, K., and Payton, D. (2003). "World Embedded Interfaces for Human-Robot Interaction." *36th Annual Hawaii International Conference on System Sciences*.

[8] Baker, M., Casey, R., Keyes, B., and Yanco, H.A. (2004). "Improved Interfaces for Human-Robot Interaction in Urban search and Rescue." *IEEE Conference on Systems, Man and Cybernetics*.

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

[10] Scholtz, J. (2003). "Human-robot Inter-actions: Creating Synergistic Cyberforces." *36th Annual Hawaii International Conference on System Science*.

[11] Vig, L. and Adams, J.A. (2005). "Issues in Multi-Robot Coalition Formation." *Multi-Robot Systems: From Swarms to Intelligent Automata Vol. III*, L.E. Parker, F.E. Schneider & A.C. Schultz (Eds.), Springer: Germany, pp. 15-26.