Assessing Coordination Overhead in Control of Robot Teams

Jijun Wang and Michael Lewis, *Member, IEEE*

***Abstract*—Conventional models of multirobot control assume independent robots and tasks. This allows an additive model in which the operator controls robots sequentially neglecting each until its performance deteriorates sufficiently to require new operator input. This paper presents a model and experiment intended to extend the neglect tolerance model to situations in which robots must cooperate to perform dependent tasks. In the experiment operators controlled 2 robot teams to perform a box pushing task under high cooperation demand (teleoperation), moderate demand (waypoint control/heterogeneous robots), and low demand (waypoint control/homogeneous robots) conditions. Measured demand and performance were consistent with the model's predictions.**

I. Introduction

APPLICATIONS for multirobot systems (MRS) such as interplanetary construction or cooperating uninhabited aerial vehicles will require close coordination and control between human operator(s) and teams of robots in uncertain environments. Human supervision will be needed because humans must supply the perhaps changing, goals that direct MRS activity. Robot autonomy will be needed because the aggregate decision making demands of a MRS are likely to exceed the cognitive capabilities of a human operator. Autonomous cooperation among robots, in particular, will likely be needed because it is these activities [1] that theoretically impose the greatest decision making load.

Controlling multiple robots substantially increases the complexity of the operator's task because attention must constantly be shifted among robots in order to maintain situation awareness and exert control. In the simplest case an operator controls multiple independent robots interacting with each as needed. Control performance at this task can be characterized by the average demand of each robot on human attention [2] or the distribution of demands coming from multiple robots [4]. Increasing robot autonomy allows robots to be neglected for longer periods of time making it possible for a single operator to control more robots.

The study of autonomy modes for MRS has been more restrictive. Because of the need to share attention between robots, teloperation has only been used for one robot out of a team [5] or as a selectable mode [6]. Some variant of waypoint control has been used in most of the MRS studies we have reviewed [2,5,6,7] with differences arising primarily in behavior upon reaching a waypoint. A more fully autonomous mode has typically been included involving things such as search of a designated area [5], travel to a distant waypoint [7], or executing prescribed behaviors [6]. In studies in which robots did not cooperate and had varying levels of individual autonomy [2,5,7] (team size 2-4) performance and workload were both higher at lower autonomy levels and lower at higher ones. So although increasing autonomy in these experiments reduced the cognitive load on the operator, the automation could not perform the replaced tasks as well.

For cooperative tasks and larger teams individual autonomy is unlikely to suffice. The round-robin control strategy used for controlling individual robots would force an operator to plan and predict actions needed for multiple joint activities and be highly susceptible to errors in prediction, synchronization or execution. Estimating the cost of this coordination, however, proves a difficult problem.

Established methods of estimating MRS control difficulty, neglect tolerance and fan-out [2] are predicated on the independence of robots and tasks. In neglect tolerance the period following human intervention but preceding a decline in performance is considered time during which the operator is free to perform other tasks. If the operator services other robots over this period the measure provides an estimate of the number of robots that might be controlled. Fan-out works from the opposite direction, adding robots and measuring performance until a plateau without further improvement is reached. Both approaches presume that operating an additional robot imposes an additive demand on cognitive resources. If robots must cooperate to perform a task such as searching a building without redundant coverage or acting together to push a block, independence no longer holds. Where coordination demands are weak, as in the search task, the round robin strategy implicit in the additive approach may still hold, although the operator must now consciously deconflict search patterns to avoid redundancy. For box pushing, coordination demands are simply too strong forcing the operator to either control the robots simultaneously or alternate rapidly to keep them synchronized in their joint activity. In this case it seems reasonable to assume that the instantaneous workload required to coordinate two robots simultaneously will be greater than that required to operate them sequentially in independent tasks. Crandall and Cummings [3] recently proposed extensions to the neglect tolerance metrics to incorporate attention allocation among robots but do not explicitly address coordination.

Estimating the cost of coordination requires designing experiments that allow comparisons between "equivalent" conditions with and without coordination demands. Finding such equivalences is much easier for weak coordination. Wang and Lewis [8], for example, compared search performance between a team of autonomously coordinating robots, manually (waypoint) controlled robots, and mixed

Manuscript received March 16, 2007. This work was supported in part by the Air Force Office of Scientific Research under Grant BS123456.

Jijun Wang is with the School of Information Sciences, University of Pittsburgh, Pittsburgh, PA 15260, USA.

Michael Lewis is with the School of Information Sciences, University of Pittsburgh, Pittsburgh, PA 15260, USA (phone: 412-624-9426, fax: 412-624-2788, email: ml@sis.pitt.edu)

initiative teams with autonomously coordinated robots that accepted operator inputs. The impact of coordination demand was observable through the difference in performance between the manually controlled teams and the mixed initiative ones. The fully automated teams provided a control ensuring that the benefits in the mixed initiative condition were not due solely to the superior performance of the automation.

While [2] examined coordination demand indirectly by comparing performance between conditions in which it was filled either manually or through automation, the present experiment attempts to manipulate coordination difficulty directly. In the control condition participants controlled two robots through teleoperation requiring continuous coordination. In the medium difficulty condition they controlled heterogeneous robots making it difficult to synchronize actions due to differences in kinematics. In the low effort condition they controlled homogeneous robots easing the task of synchronizing distances and times. Because identical robots were used in the first two conditions and the heterogeneous robots had equivalent capabilities differences in performance could be attributed to variation in coordination demand between the conditions. To separate coordination demand (CD) from the demands of interacting with independent robots we have extended Crandall's [2] neglect tolerance model by introducing the notion of occupied time (OT) as illustrated in Figure 1.

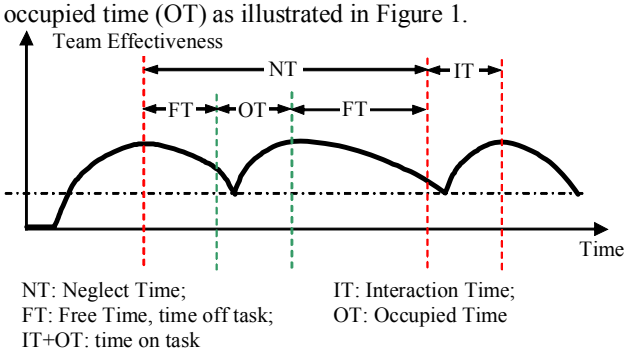

Figure 1. Extended neglect tolerance model for cooperative robot control

The neglect tolerance model describes an operator's interaction with multiple robots as a sequence of control episodes in which an operator interacts with a robot for period IT raising its performance above some upper threshold after which the robot is neglected for the period NT until its performance deteriorates below a lower threshold when the operator must again interact with it. To accommodate dependent tasks we introduce OT to describe the time spent controlling other robots in order to synchronize their actions with those of the target robot. The episode depicted in Figure 1 starts just after the first robot is serviced. The ensuing FT preceding the interaction with a second dependent robot, the OT for robot-1 (that would contribute to IT for robot-2), and the FT following interaction with robot-2 but preceding the next interaction with robot-1 together constitute the neglect time for robot-1. Cooperation demand, CD, is then defined as:

$$\text{CD}=1-\frac{\sum FT}{NT}=\frac{\sum OT}{NT} \qquad (1)$$

Where, CD for a robot is the ratio between the time required to control cooperating robots and the time still available after controlling the target robot, i.e.; the portion of a robot's free time that must be devoted to controlling cooperating robots. Note that OT_n associated with $robot_n$ is less than or equal to IT_n because OT_n covers only that portion of NT_n needed for synchronization. By choosing the strictly cooperative task of box-pushing we can assume that $OT_n \equiv IT_n$, e.g.; that most of the operator's interactions with each robot will involve coordinating its actions with the other. Under such conditions of high CD the operator must repeatedly interact with a robot to synchronize its actions with others as well as to perform the basic task. These increased interactions will reduce Fan-out (number of robots that can be controlled) and add to the cognitive load of the operator. A related measure, team task demand (TAD), adds IT's to both numerator and denominator to provide a measure of the proportion of time devoted to the cooperative task; either performing the task or coordinating robots.

II. USARSIM

These experiments were conducted in the high fidelity USARSim robotic simulation environment we developed as a simulation of urban search and rescue (USAR) robots and environments intended as a research tool for the study of human-robot interaction (HRI) and multi-robot coordination. It is freely available and can be downloaded from www.sourceforge.net/projects/usarsim. USARSim uses Epic Games' UnrealEngine2 to provide a high fidelity simulator at low cost. USARSim supports HRI by accurately rendering user interface elements (particularly camera video), accurately representing robot automation and behavior, and accurately representing the remote environment that links the operator's awareness with the robot's behaviors. The current version of USARSim consists of models of standardized disaster environments, models of commercial and experimental robots, and sensor models. USARSim also provides users with the capability of building their own environments and robots. Its socket-based control API was designed to allow users to test their own control algorithms and user interfaces without additional programming.

USARSim includes detailed models of the NIST Reference Test Arenas for Autonomous Mobile Robots [9] including a replica of the fixed Nike site. The portable arenas model buildings in various stages of collapse and are intended to provide objective performance evaluation for robots as they perform a variety of urban search and rescue tasks. The arenas are used for USAR competitions at RoboCup and other meetings. USARSim offers the possibility of providing more realistic challenges and significantly larger disaster environments have been developed for Virtual Robot USAR competitions at RoboCup 2006 and 2007. The official release of USARSim currently provides detailed models of thirteen robots including the Pioneer P2AT and P2DX robots used in this experiment.

These models which include commercial robots widely used in USAR competition were constructed using the Karma physics engine, a rigid body simulation that computes physical interactions in realtime. A hierarchy of sensor classes has been defined to simulate sensor data. Sensors are defined by a set of attributes stored in a configuration file, for example, perception sensors are commonly specified by range, resolution, and field-of- view. The scenes viewed from the simulated camera are acquired by attaching a spectator, a special kind of disembodied player, to the robot. USARSim provides two ways to simulate camera feedback: direct display and image server. Direct display uses the Unreal Client, itself, for video feedback, either as a separate sensor panel or embedded into the user interface. While this approach is the simplest, the Unreal Client provides a higher frame rate than is likely to be achieved in a real robotic system and is not accessible to the image processing routines often used in robotics. The image server intermittently captures scenes in raw or jpeg format from the Unreal Client and sends them over the network to the user interface. Using the image server, researchers can tune the properties of the camera, specifying the desired frame rate, image format, noise, and/or post processing needed to match the camera being simulated.

III. Experiment

A. Experimental Design

Finding a metric for cooperation demand (CD) is difficult because there is no widely accepted standard. In this experiment, we investigated CD by comparing performance across three conditions selected to differ substantially in their coordination demands. We selected box pushing, a typical cooperative task that requires the robots to coordinate, as our task. We define CD as the ratio between occupied time (OT), the period over which the operator is actively controlling a robot to synchronize with others, and FT+OT, the time during which he is not actively controlling the robot to perform the primary task. This measure varies between 0 for no demand to 1 for maximum demand. When an operator teleoperates the robots one by one to push the box forward, he must continuously interact with one of the robots because neglecting both would immediately stop the box. Because the task allows no free time (FT) we expect CD to be 1. However, when the user is able to issue waypoints to both robots, the operator may have FT before she must coordinate these robots again because the robots can be instructed to move simultaneously. In this case CD should be less than 1. Intermediate levels of CD should be found in comparing control of homogeneous robots with heterogeneous robots. Higher CD should be found in the heterogeneous group since the unbalanced pushes from the robots would require more frequent coordination. In the present experiment, we measured CDs under these three conditions.

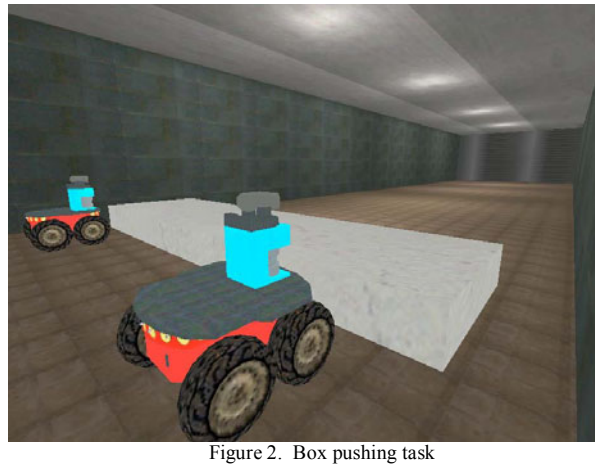

Figure 2. Box pushing task

Figure 2 shows our experiment setting simulated in USARSim [10,11]. The controlled robots were either two Pioneer P2AT robots or one Pioneer P2AT and one less capable three wheeled Pioneer P2DX robot. Each robot was equipped with a GPS, a laser scanner, and a RFID reader. On the box, we mounted two RFID tags to enable the robots to sense the box's position and orientation. When a robot pushes the box, both the box and robot's orientation and speed will change. Furthermore, because of irregularities in initial conditions and accuracy of the physical simulation the robot and box are unlikely to move precisely as the operator expected. In addition, delays in receiving sensor data and executing commands were modeled presenting participants with a problem very similar to coordinating physical robots.

We introduced a simple matching task as a secondary task to allow us to estimate the FT available to the operator. Participants were asked to perform this secondary task as possible as possible when they were not occupied controlling a robot. Every operator action and periodic timestamped samples the box's moving speed were recorded for computing CD .

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

The operator controlled the robots using a distributed multi-robot control system (MrCS) shown in Figure 3. On the left and right side are the teleoperation widgets that control the left and right robots separately. The bottom center is a map based control panel that allows the user to monitor the robots and issue waypoint commands on the map. On the bottom right corner is the secondary task window where the participants were asked to perform the matching task when possible.

B. Participants and Procedure

14 paid participants, 18-57 years old were recruited from the University of Pittsburgh community. None had prior experience with robot control although most were frequent computer users.

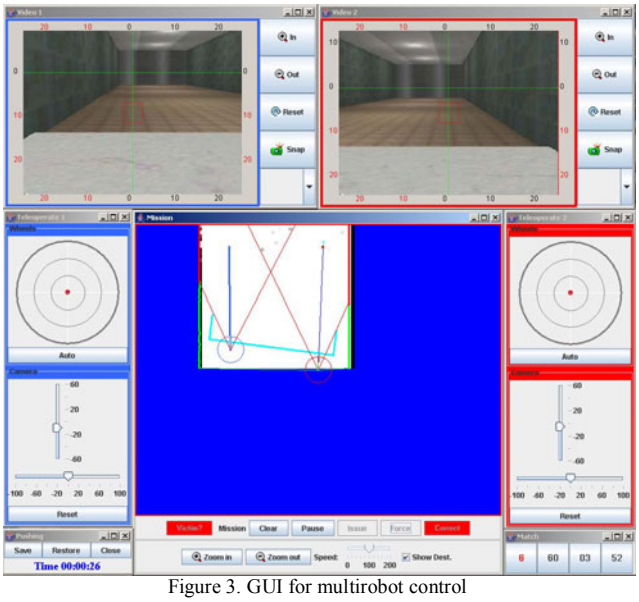

Figure 3. GUI for multirobot control

The experiment started with collection of the participant's demographic data and computer experience. The participant then read standard instructions on how to control robots using the MrCS. In the following 8 minutes training session, the participant practiced each control operation and tried to push the box forward under the guidance of the experimenter. Participants then performed three testing sessions in counterbalanced order. In two of the sessions, the participants controlled two P2AT robots using teleoperation alone or a mixture of teleoperation and waypoint control. In the third session, the participants were asked to control heterogeneous robots (one P2AT and one P2DX) using a mixture of teleoperation and waypoint control. The participants were allowed eight minutes to push the box to the destination in each session. At the conclusion of the experiment participants completed a questionnaire about their experience.

C. Results

Figure 4 shows a time distribution of robot control commands recorded in the experiment. As we expected no free time was recorded for robots in the teleoperation condition and the longest free times were found in controlling homogeneous robots with waypoints. The box speed shown on Figure 4 is the moving speed along the hallway that reflects the interaction effectiveness (IE) of the control mode. The IE curves in this picture show the delay effect and the frequent bumping that occurred in controlling heterogeneous robots revealing the poorest cooperation performance.

None of the 14 participants were able to perform the secondary task while teleoperating the robots. Hence, we uniformly find TAD=1 and CD=1 for both robots under this condition. Within participants comparison found that under waypoint control the team attention demand in heterogeneous robots is significantly higher than the demand in controlling homogeneous robots, $t(13)=2.213$, $p=0.045$ (Figure 5). No significant differences were found between the homogeneous

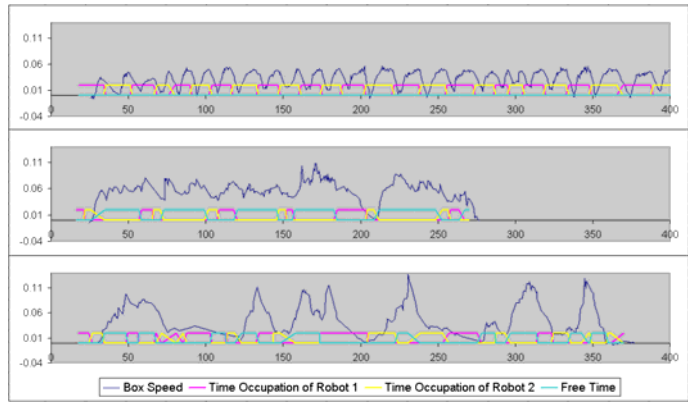

Figure 4 The time distribution curves for teleoperation (upper) and waypoint control (middle) for homogeneous robots, and waypoint control (bottom) for heterogeneous robots

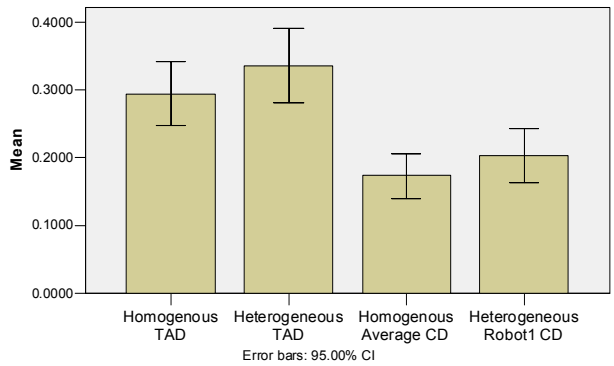

Figure 5. Team task demand (TAD) and Cooperation demand (CD)

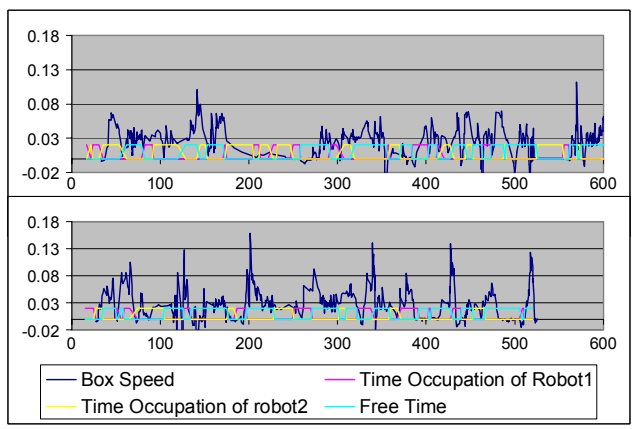

Figure 6. Exception I: Homogeneous (upper) and heterogeneous (bottom) robots control with unfamiliar UI

P2AT robots in terms of the individual cooperation demand ($P=0.2$). Since the robots are identical, we compared the average CD of the left and right robots[1] with the CDs measured under heterogeneous condition. Two-tailed t-test shows that when a participant controlled a P2AT robot, lower CD was required in homogeneous condition than in the heterogeneous condition, $t(13)=-2.365$. $p=0.034$. The CD

[1] In the following, without specific claim, the CD of homogeneous robots means the average individual CD of the robot group.

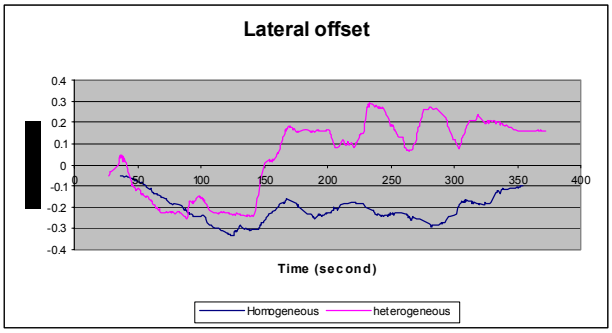

Figure 7. Exception III: Different satisficing levels

required in controlling the P2DX under heterogeneous condition is marginally higher than the CD required in controlling homogenous P2ATs, $t(13)=-1.868$, $p=0.084$ (Figure 5). Surprisingly, no significant difference was found in CDs between controlling P2AT and P2DX under heterogeneous condition ($p=0.79$). This can be explained by the three observed robot control strategies: 1) the participant always issued new waypoints to both robots when adjusting the box's movement, therefore similar CDs were found between the robots; 2) the participant tried to give short paths to the faster robot (P2DX) to balance the different speeds of the two robots, thus we found higher CD in P2AT; 3) the participant gave the same length paths to both robots and the slower robot needed more interactions because it trended to lag behind the faster robot, so lower CD for the P2AT was found for the participant. Among the 14 participants, 5 of them (36%) showed higher CD for the P2DX contrary to our expectations.

IV. Discussion

Although we expected uniformly higher CD for the P2AT robot under heterogeneous condition, 3 exceptions were found in the experiment. The first exception occurred for a participant who commented on having problems in using the control interface. This was confirmed by the recorded irregular time distribution (Figure 6). The close CDs (0.23 and 0.22 for P2AT robot under homogeneous and heterogeneous conditions) demonstrate that lack of operational skill overwhelmed the impact of task and robotic system. In the second exception, we observed that an abnormally long time (41.25 sec) in controlling homogeneous P2ATs was spent in figuring out and recovering from a mistake. Because of the short task completion time (380 sec), this mistake led relatively high CD. The last exception occurred when the participant changed her control strategy, specifically the satisficing level of performance, between the homogeneous and heterogeneous conditions. While controlling the homogeneous robots, she paid more attention to keeping the box in the center of the hallway and made many more adjustments to the robots leading to a total lateral offset of 0.28 meters. However, in the following heterogeneous robots trial, she lowered her criteria for accuracy and finished this session with a total lateral offset of 0.54 meter (Figure 7). The higher CD for homogeneous robots (0.17 and 0.11 under homo and heterogeneous conditions respectively) reflects the impact of this change in criteria.

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

References

[1] B. Gerkey and M. Mataric. A formal framework for the study of task allocation in multi-robot systems. International Journal of Robotics Research, 23(9):939–954, 2004.

[2] J. W. Crandall, M. A. Goodrich, D. R. Olsen, and C. W. Nielsen. Validating human-robot interaction schemes in multitasking environments. IEEE Transactions on Systems, Man, and Cybernetics, Part A, 35(4):438–449, 2005.

[3] J. Crandall and M. Cummings. Developing performance metrics for the supervisory control of multiple robots, Proceedings of the 2007 ACM/IEEE Conference on Human-Robot Interaction (HRI'07), 33-40, 2007.

[4] J. Nickerson and S. Steven. Attention and communication: Decision scenarios for teleoperating robots. In Proceedings of the 38th Annual Hawaii International Conference on System Sciences, January 2005.

[5] C. Nielsen, M. Goodrich, and J. Crandall. Experiments in human-robot teams. In *Proceedings of the 2002 NRL Workshop on Multi-Robot Systems*, October 2003

[6] R. Parasuraman, S. Galster, P. Squire, H. Furukawa, and C. Miller. A flexible delegation-type interface enhances system performance in human supervision of multiple robots: Empirical studies with roboflag. IEEE Systems, Man and Cybernetics-Part A, Special Issue on Human-Robot Interactions, 35(4):481–493, July 2005

[7] B. Trouvain and H. L. Wolf. Evaluation of multi-robot control and monitoring performance. In Proceedings of the 2002 IEEE Int. Workshop on Robot and Human Interactive Communication, pages 111–116, September 2002.

[8] J. Wang and M. Lewis (2007) Human control of cooperating robot teams, 2007 Human-Robot Interaction Conference, ACM.

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

[10] J. Wang, M. Lewis, S. Hughes, M. Koes and S. Carpin, (2005). Validating USARsim for use in HRI Research, Proceedings of the Human Factors and Ergonomics Society 49th Annual Meeting, 457-461.

[11] S. Carpin, T. Stoyanov, Y. Nevatia, M. Lewis, J. Wang, (2006). Quantitative assessments of USARSim accuracy". *Proceedings of PerMIS 2006*