Validating USARsim for use in HRI Research

Jijun Wang, Michael Lewis,
and Stephen Hughes
School of Information Sciences
University of Pittsburgh
Pittsburgh, PA 15260 USA

Mary Koes
Carnegie Mellon University
Pittsburgh, PA 15213
Stefano Carpin
International University of Bremen
D-28759 Bremen, Germany

HRI is an excellent candidate for simulator based research because of the relative simplicity of the systems being modeled, the behavioral fidelity possible with current physics engines and the capability of modern graphics cards to approximate camera video. In this paper we briefly introduce the USARsim simulation and discuss efforts to validate its behavior for use in Human Robot Interaction (HRI) research.

INTRODUCTION

Many of the HRI studies reported within the past year have relied on USARsim (Hughes and Lewis 2004a,b, Wang et al. 2004) or other (Chadwick et al. 2004, Nielsen et al. 2004, Ricks et al. 2004, Olsen and Wood 2004) robotic simulations. Although many robotic simulators are available most of them have been built as ancillary tools for developing and testing control programs to be run on research robots. Simulators built before 2000 typically have low fidelity dynamics for approximating the robot's interaction with its environment. More recent simulators including Gazebo (Gerkey et al. 2003), and the commercial Webots (Webots) use the open source Open Dynamics Engine (ODE) physics engine to approximate physics and kinematics more precisely. ODE, however, is not integrated with a graphics library forcing developers to rely on low level libraries such as OpenGL. This limits the complexity of environments that can practically be developed and effectively precludes use of many of the specialized rendering features of modern gpu's. Both high quality graphics and accurate physics are needed for HRI research because the operator's tasks depend strongly on remote perception (Woods et al. 2004) which requires accurate simulation of camera video and interaction with automation which requires accurate simulation of sensors, effectors and control logic.

USARsim

USARSim was developed as a high fidelity simulation of urban search and rescue (USAR) robots and environments intended as a research tool for the study of HRI and multi-robot coordination. It is freely available and can be downloaded from http://usl.sis.pitt.edu/ulab. 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. It's 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* (Jacoff et al. 2001) 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 are under development for the Virtual Robot USAR demonstration at RoboCup 2005.

Robot models

The official release of USARSim currently provides detailed models of six robots: the Pioneer P2AT and P2DX, iRobot ATRV-Jr, the Personal Exploration Rover (PER) (Nourbakhsh et al. 2004), the Corky robot built for this project and a generic four-wheeled car. These models which include commercial robots widely used in USAR competition were constructed using the Karma physics engine (Karma 2002), a rigid body simulation that computes physical interactions in realtime. A hierarchy of sensor classes have 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.

USARsim as a Common Tool

USARsim is a tool for observing and testing alternate designs for automation and the related user interface. The basic concept is that the simulation provides accurate models of robots, environments, and camera video while the experimenter brings his own interface and automation strategies to be tested.

We developed USARsim both to advance our own work in Robot, Agent, Person (RAP) teams and to provide a common tool for use in HRI research and USAR competition. Researchers outside of computer science or mechanical engineering departments specializing in robotics are unlikely to have access to the experimental robots needed to conduct research. Yet researchers from disciplines such as HCI, psychology or human factors often have the greatest interest in issues affecting HRI. Expense, unreliability, and difficulties in running participants in parallel especially in multi-robot experiments make physical robotics inappropriate for the large samples, repeated trials and varied conditions needed for HRI research. Because the object of study in HRI is the behavior of the human with the robot, environment, and task controlled, high fidelity simulation may not only be the most practical, but also the best-suited tool for HRI research.

VALIDATION STUDIES

Validating USARsim for HRI presents a complex problem because the performance of the human-robot system is jointly determined by the robot, the environment, the automation, and the interface. Because only the robot and its environment are officially part of the simulation, validation is necessarily limited to some particular definition of interface and automation. If, for example, sensor-based drift in estimation of yaw were poorly modeled it would not be apparent in validation using teleoperation yet could still produce highly discrepant results for a more automated control regime. Realizing the impossibility of "complete" validation we are relying on a two stage approach. In the first stage we conduct tests comparing the performance of elementary behaviors and sensor readings for real and simulated robots. In the second stage we compare standard HRI tasks for particular interfaces and definitions of automation. Positive results give us some assurance that the simulation is physically accurate (Stage 1) and evidence that it remains consistent for at least some interface and automation definitions (Stage 2)

We are currently conducting tests in replicas of NIST's Orange Arena at Carnegie Mellon University and the International University of Bremen. We hope to link validation studies to the incorporation of new robots and sensors into the simulation. This year are requiring participants in the Virtual Rescue Robot Demo to limit their teams to models of robots they or others have entered in physical league competition. In the future we hope to collect validation data on all models released with the simulation.

Mapping Validation at IUB

Maps to help rescuers find robot-identified victims are a standard requirement for USAR competitions. Research at the International University of Bremen (IUB) has focused on grid based maps and the problem of multi-robot map merging (Carpin and Birk 2005). Recently efforts have turned to other approaches, like simultaneous localization and mapping (SLAM) that require the identification of features. In this context, algorithms were developed to extract natural landmarks in unstructured environments. The algorithms were developed in simulation and then moved to real robots. The IUB rescue robots are equipped with a proximity range finder, odometry, an orientation sensor, and a set of cameras. Victim detection is human supervised, and is assisted by an infrared camera and a CO_2 sensor. Mapping is performed using the robot's pose (provided by odometry and orientation sensor) and the data coming from the range finder.

The proximity range sensor provided by USARsim can be configured for the number of beams used to sample the swept area, the maximum reachable distance, and noise. The real sensor (Hokuyo PB9-11) sweeps an area of 162 degrees with 91 beams. Its detection distance is 3 meters, and we experimentally determined that under the conditions found in the IUB rescue arena the signal to noise ratio was about 30 dB.

These properties were used to configure the simulated range finder. The simulated robot was run in the model of the IUB rescue arena and gathered data from the simulated proximity range finder. Then, the same data was collected for the real robot and arena. Features were extracted using the Hough transform, a widely used tool from image processing. Local maxima of this histogram correspond to lines detected in the image. The following figures show a comparison between the data collected with the simulator and with the real robot (figure 1). The fine tuning of parameters was performed entirely within USARsim. No change was necessary when the code was later used to perform the same processing on real data. You can observe by inspection the close agreement between the real and simulated maps.

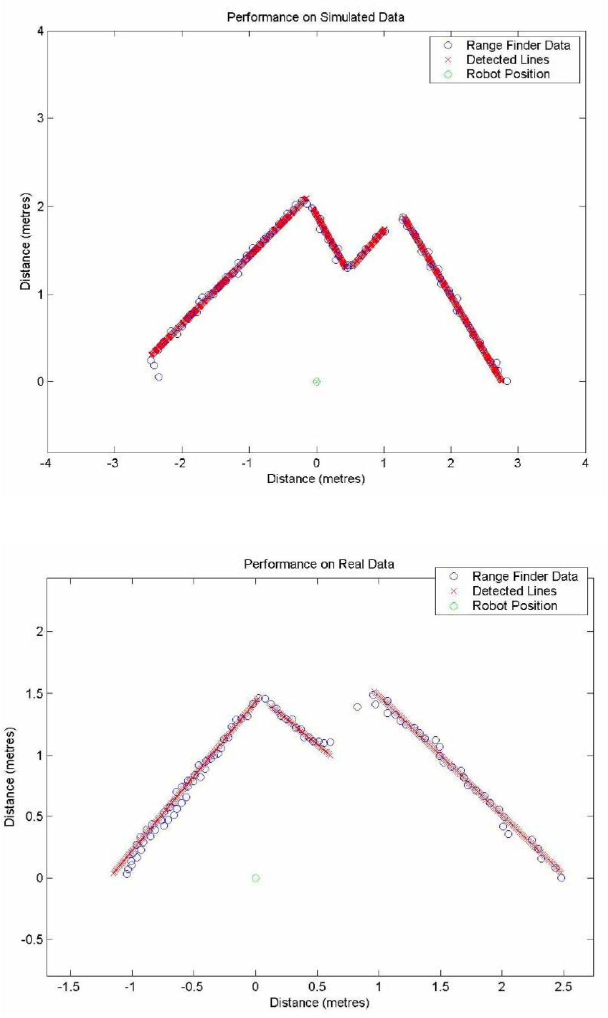

Figure 1. Map of Orange Arena: virtual-top, real-bottom

Robot Control Validation at CMU and Pitt

Informally, we have long observed that tasks that caused difficulties in the real environment also caused problems in the simulation (Wang et al. 2003). It is very difficult, for example, to drive the Corky robot up the ramp of either the real or the simulated Orange Arena. We are currently conducting more rigorous tests to validate the simulation. Participants control remote robots or the simulation using only camera video. Tests with the PER robot are complete and tests of the Pioneer P3AT simulated as the earlier P2AT are underway. In Stage 1 testing of the PER we established times, distances, and errors associated with movements from point to point over a wood floor, paper, and lava rocks. These data were used to adjust the speed of the simulated PER and alter its performance when moving over scattered papers.

Figure 2. Real (top) and Simulated (bottom) Obstructed Paper Surface

In Stage 2 testing, PER robots were repeatedly run along a narrow corridor with varying types of debris (wood floor, scattered papers, lava rocks) while the sequence, timing and magnitude of commands were recorded. Participants were assigned to maneuver the robot with either direct teleoperation or waypoint (specified distance) modes of control. In the initial three exposures to each environment, participants had to drive approximately three-meters, along an unobstructed path to an orange traffic cone. Hypothetically, this could be achieved with a single command, however operators had to compensate

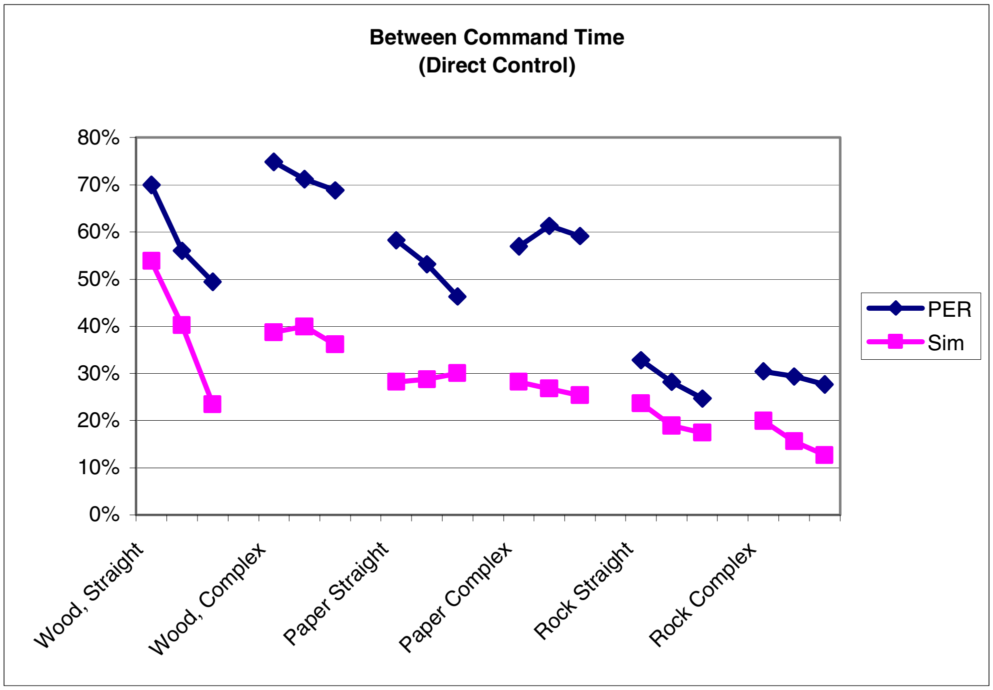

Figure 3. Pauses between commands

for the effects of the debris, which could delay or alter the heading of the robot. In later trials, obstacles were added to the environments, forcing the driver to negotiate at least three turns to reach the objective destination. These same conditions were evaluated with fresh participants using the PER model in USARSim. Although the data collected from five subjects per condition is insufficient for statistical analyses it does provide a qualitative picture of the simulation's performance.

Learning rates-The participants in the study were novice operators, limited to a brief description of the robots capabilities and a quick test drive. Similar learning trends on the repeated tasks were observed between the simulation and the real environments, specifically, the overall execution time and the time between commands trended downwards at comparable rates as the operator became more familiar with the controls.

Terrain effects-The effect of the paper surface on the PER's operation could best be described as marginal. Alternatively, the rocky surface had a considerable impact, including a loss of traction and deflection of the robot. This was reflected by increases in the odometry and number of turn commands issued by the operator. A parallel spike in these metrics is recorded in the simulator data.

Proximity-One metric in which the simulation and the physical robot consistently differed was the proximity to the cone acquired by the operator. Participants were given the instruction to "get as close to the cone as possible without touching it". Operators using the physical robot reliably moved the robot to within 35cm from the cone, while the USARSim operators were usually closer to 80cm from the cone. It is unlikely that the simulation would have elicited more caution from the operators, so this result suggests that there could be a systematic distortion in depth perception, situation awareness, or strategy. In both cases the cone filled the camera's view at the end of the task.

A variety of other data were collected to characterize the effects of the simulation and conditions on operator control behavior. Figure 3 shows idle times between issuing commands for direct (teleoperation) control conditions. Note that there are substantially longer pauses between actions in controlling the real robot. This occurred despite matching frame rates although slight differences in response lag may have played a factor.

Figure 4 shows times to complete tasks. Note that despite the difference in length of pauses completion times remain very close between the robot and the simulation. The average number of commands were also very similar between the simulation and the PER for control mode and environment except for straight travel over rocks in command mode where PER participants issued more than twice as many commands as those in the simulation or direct operation modes. A similar pattern occurs for forward distance traveled with close performance between simulation and PER for all conditions but straight travel over rocks, only now it is the teleoperated simulation that is higher.

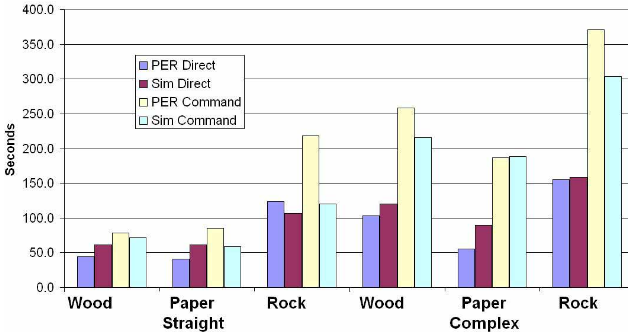

Figure 4. Task Completion Times

These observations illustrate the importance of validation of simulations for HRI. Although the overt performance of the simulation was closely matched to the robot in Stage 1, Stage 2 showed that direct mode pauses were longer for the real robot although most other measures were consistent. To draw valid conclusions from robotic simulations it is important to know the metrics which are consistent with the operation of the actual robot and those which are not. By collecting validation data for all entities within the simulation we hope to create a tool with which researchers can pick and choose manipulations and metrics that are likely to yield useful results. The IUB mapping data, for example, suggests that our sensor model for laser rangefinding is reasonably accurate and could safely be used for generating maps unlikely to differ substantially from those encountered in the field. The PER proximity data, by contrast, should raise a flag for a researcher interested in using the PER model and interface to study control strategies for ordnance removal.

As our library of models and validation data expands we hope to begin incorporating more rugged and realistic robots, tasks and environments. Tame tasks performed by wheeled robots in portable arenas are only a hint at the potential high fidelity HRI simulation has to offer for human control of robots in large scale hazardous environments for which we could not otherwise gain experience.

ACKNOWLEDGEMENTS

This project is supported by NSF grant NSF-ITR-0205526.

REFERENCES

Carpin, S., Birk, A. (2005). Stochastic map merging in rescue environments. In: Robocup 2004: Robot Soccer World Cup VIII. Springer.

Chadwick, R., Gillan, D., Simon, D., and Pazuchanics, S. (2004). Cognitive analysis methods for control of multiple robots: Robotics on $5 a day. Proceedings of the 48th Annual Meeting Human Factors and Ergonomics Society, pp. 688-692.

Gerkey, B., Vaughan, R., Howard, A. (2003). The Player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems Proceedings of the International Conference on Advanced Robotics (ICAR 2003), Coimbra, Portugal, June 30 - July 3, pp. 317-323.

Hughes, S., Lewis, M. (2004) Robotic camera control for remote exploration, Proceedings of the 2004 Conference on Human Factors in Computing Systems (CHI 2004), Vienna, Austria, 511-517.

Hughes, S. and Lewis, M. (2004). Assisted viewpoint control for tele-robotic search, Proceedings of the 48th Annual Meeting of the Human Factors and Ergonomics Society, pp. 2657-2661.

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

Karma (2002) Mathengine Karma User Guide http://udn.epicgames.com/Two/KarmaReference/KarmaUserGuide.pdf Accessed: 2005

Nourbakhsh, I., Hamner, E., Porter, E., Dunlavey, B., Ayoob, E., Hsiu, T., Lotter, M., Shelly, S. (2005) The Design of a Highly Reliable Robot for Unmediated Museum Interaction,. ICRA-2005

Olsen, D. and Wood, S. (2004): Fan-out: measuring human control of multiple robots. Proceedings of ACM CHI 2004 Conference on Human Factors in Computing Systems. p.231-238

Ricks, B. Nielsen, C. and Goodrich, M. (2004). Ecological Displays for Robot Interaction: A New Perspective. Proceedings of IROS 2004. Sept 28-Oct 2, 2004, Sendai, Japan.

UNR, Unreal engine, http://www.epicgames.com, accessed February 8, 2005

Webots, http://www.cyberbotics.com/, accessed February 8, 2005

Wang, J., Lewis, M. and Gennari, J. (2003). USAR: A Game-Based Simulation for Teleoperation. Proceedings of the 47th Annual Meeting of the Human Factors and Ergonomics Society, Denver, CO, Oct. 13-17.

Wang, J., Lewis, M., and Hughes, S. (2004). Gravity-referenced attitude display for teleoperation of mobile robots, Proceedings of the 48th Annual Meeting of the Human Factors and Ergonomics Society, pp. 2662-2666.

Woods, D., Tittle, J., Feil, M., and Roesler, A. (2004). Envisioning human-robot coordination in future operations, IEEE Transactions on Systems, Man, and Cybernetics, Part C, pp. 210-218.