A Robot Simulator Classification System for HRI

Jeff Craighead, Robin Murphy, Jenny Burke
Department of Computer Science
University of South Florida
Tampa, FL 33620
{*craighea,murphy,jlburke4*}@*cse.usf.edu*

Brian Goldiez
Institute for Simulation and Training
University of Central Florida
Orlando, FL 32826
bgoldiez@ist.ucf.edu

ABSTRACT

This paper presents a classification system for computer-based robot simulators that is based on the FAA guidelines for aircraft simulators. Low fidelity computer simulation has been used extensively for testing artificial intelligence and control algorithms for robotic systems. Until recently operator training using simulators has been impractical due to the cost of the computer systems necessary to simulate robot operation with high fidelity. The rapid increase in the power of desktop computers over the last decade has led to cheap, high fidelity vehicle simulation. A review of the literature shows that there are many robot simulators in use with a variety of features and fidelity levels. There has been no prior work attempting to classify the functionality of these robot simulators.

KEYWORDS: Serious Games, Simulation, Training, Simulator Rating

1. INTRODUCTION

This paper presents a new simulator classification system specific to mobile robots and autonomous vehicles. The proposed system is based on the Federal Aviation Administration's (FAA) requirements for aircraft simulators[17] with some modification for use with remotely operated vehicles. A classification system for robot simulators will allow researchers to identify existing simulators which may be useful in conducting a wide variety of robotics research from testing low level or autonomous control to human robot interaction. Craighead[14] recently reviewed 14 open source and commercial robot simulators based on Alexander's[11] definitions of fidelity and found that there are many simulators that have similar or overlapping target vehicles, features, and users. For example, Player/Stage[3] can simulate many ground vehicles as a generic robot with sensors such as sonar or laser scanner out of the box. SimRobot[6], Webots[10], and Simbad[4] also have similar target vehicles and can all do out of the box simulation of a basic robot. This leads to several questions regarding the uniqueness of a simulator for researchers as well as the developers of these products.

The classification system we present should help robotics researchers answer these questions quickly when examining a simulator. The FAA simulator requirements chart classifies simulators, based on features that affect fidelity, into four classes A through D. Class A simulators have fewer features and so can be used for fewer training exercises. Class D simulators should be an exact replica of a particular aircraft's cockpit on a motion platform that meets the specified timing requirements. The robot simulator classification system classifies simulators into five classes A-E. Class A is a catch all class for most simulators that do not meet class B requirements. Class B has few requirements, but specifies a minimal level of fidelity. Class E simulators require a full hardware mockup of the specific robot operator control unit (OCU) in addition to the high fidelity physical simulation and rendering required by class D simulators.

The remainder of this article is presented as follows. Section 2 discusses the FAA requirements chart as well as past work on classification of robot simulators. Section 3 explains the details of the proposed robot simulator classification chart shown in Figure 1. Section 4 demonstrates how the classification system can be used to assist HRI researchers when selecting a simulator. Finally Section 5 summarizes the classification method and discusses how it may improved in the future.

2. RELATED WORK

To date there has been no work towards the development of requirements for computer based robot simulators. It is important to establish a minimum set of requirements for the simulator so that researchers can easily identify systems that are adequate for studying human robot interaction. The FAA Advisory Circular AC120-C provides the model for the development of our robot simulator classification system. AC120-C was published in 1995 identifies the important aspects of flight training which must be included in an aircraft simulator.

Another FAA document, Qualification Guidelines for Personal Computer-Based Aviation Training Devices: Private Pilot Certificate[19], published in 2001 presents a task analysis for the private pilot certificate requirements as part of the development of guidelines for PC based simulators. The paper explains that it is important to establish simulator guidelines so the FAA can verify that maneuvers performed in simulation based training device will provide an adequate emulation of the maneuvers performed in a real aircraft. Clearly the FAA is convinced that simulator training is valuable and desired for use as a training aid, but realizes that for effective training to take place a simulator must provide a minimum set of features.

A quick look at the use of simulation in other fields shows that is becoming quite popular for both research and training. Some specialties in the medical industry have approved the use of computer based (Virtual Reality)[1] simulations for procedures such as laproscopy[1]. Commercial simulators such as those developed by Immersion[2] are used at major universities as part of the medical curriculum. While there are no clear guidelines published, the devices used by medical schools attempt to provide a high fidelity simulation of the device being simulated. SimPort[18, 5] is a management game specifically developed to train "Port professionals" during simulated expansion of the Port of Rotterdam. Hazmat:Hotzone[16] was developed to train fire fighters in hazmat response procedures.

Within the field of human robot interaction, Richer and Drury[15] presented a rough computer game based framework for studying HRI. This paper did not detail simulator features, but explained what types of input and output devices are available. Additionally discussed are several example display types and what can data be easily displayed to the user. In contrast the classification system we present focuses specifically on lower level fidelity requirements for the simulation engine.

3. SIMULATOR CLASSIFICATION

This section discusses the classes and requirements for each simulator class and how these relate to the FAA simulator requirements. A large portion of the FAA requirements are specific to full cockpit mock-ups which are mounted on motion platforms. Our classification system ignores these specific motion requirements, however this could be added to a future version as class F. We will focus on the requirements for simulating physics, rendering, and methods of interaction. The remainder of this section is organized into subsections describing the requirements of each class and how these map over to the FAA requirements.

The requirements for each simulator class are summarized in Figure 1. Note that the FAA classification has only four classes, A-D, while the classification system presented in this paper has five. An extra catch-all class, class A, was created to classify most simulators that do not meet the requirements of the higher levels. There are three areas of interest in the classification system, Physics, OCU, and Environment in which the requirements are divided. We will begin with the most relaxed class, class B. As stated above, any simulator that does not meet the requirements of one of the following classes will be considered a class A simulator.

3.1. Class B

Class B simulators have the most relaxed requirements and are the minimum level for a simulator suitable for HRI work. Class B simulators approximate the motion of a robot and effectors, simulation of physical forces and control surfaces is not necessary. Class B simulators run in a basic 3D environment. Any sensors present on the real robot should be present in the simulation and provide some information to the user through the on screen display. Sensor simulations must be rough approximations of the real output, high fidelity is not required. Additionally class B simulators must support all relevant features of class A simulators. *Environment:* The most important feature that separates class B simulators from class A simulators is that the environment is rendered in 3D. There are no additional requirements on the resolution, lighting, or texture detail of the rendering; primitive shapes and colors will suffice. *OCU:* The simulated vehicle must be operable by a user through some onscreen interface or external control device such as a joystick. *Physics:* Class B simulators have very relaxed physics simulation requirements. A class B simulator must only approximate vehicle motion, simulation of physical forces on the vehicle is not necessary.

3.2. Class C

Class C simulators simulate some forces, control surfaces, and effectors of a robot so that operation of the simulated device approximates the real device. Sensor simulations must be equivalent to the output of the real sensor. Visual sensor simulation must be higher fidelity if the goal is human operation and training. Additionally class C simulators

[1]The VR simulators for medical procedures usually have a control unit similar to the real device, such as an endoscope. A computer generated 3D image replaces the video that would be displayed on a real device.

must support all relevant features of class B simulators. *Environment:* The simulator must present environmental and vehicle sounds at appropriate volumes. This includes the sound of the robot tracks or wheels on various surfaces, robot effectors, precipitation, flowing water, fire, nearby vehicles, etc. The simulator must also provide an interface for the user or instructor to configure environmental conditions such as time of day and weather if the simulator supports multiple environmental conditions. *Physics:* Effector failure should be simulated appropriately (ie. a motor that fails to neutral on the real robot should do so in the simulation). Physical damage such as broken or missing parts (ie. flat tire or missing track) should affect the operation of the simulated vehicle in a manner consistent with the real vehicle. Simulation of dynamic forces must account for friction and center of gravity.

3.3. Class D

Class D simulators are the most complex, requiring many features that enhance the overall look and feel of the simulated vehicle and environment as well as requiring more complex physics simulation. Class D simulators provide simulation of all forces, control surfaces, and effectors of a robot so that operation of the simulated device is equivalent to the real device. Sensor simulations must approximate real output to the fullest ability of the simulation platform. Additionally class D simulators must support all relevant features of class C simulators. *Environment:* The simulator must support operation in all environmental conditions targeted by the real vehicle. Systems must operate in a manner consistent with the real vehicle in each environment. Environmental conditions, such as time of day, and weather must be user or instructor configurable. Environmental sounds must correspond to real sounds at appropriate volumes. Wheel and track sounds should change with the contact surface (sand, gravel, grass, etc). *OCU:* All relevant instrument indications quickly and automatically respond to control movement or external disturbances to the simulated robot. Communications equipment such a headset and microphone should be present, operable, and interact with the simulated environment if the equipment is present on the real robot. Any force feedback present on the real robot should respond in the same manner for the simulated vehicle. Significant OCU sounds which result from operator actions should be present. *Physics:* Frictional forces on surfaces with various frictional coefficients such as icy pavement vs wet pavement vs dry pavement should be simulated for all intended operating environments. Effects of dynamic forces for various combinations of drag, thrust, gravity, and friction encountered in typical use conditions must be simulated. Changes in pose, center of gravity, gross weight, and configuration as well as altitude and temperature if these have a significant impact on performance must be handled accordingly. Class D also include special requirements for UAV simulators: ground effect, ground reaction, ground handling, and windshear should be simulated in each operating environment.

3.4. Class E

Class E simulator requirements are the most rigorous and closest to the FAA simulator requirements. The difference between a class E simulator and a class D simulator is that class E simulators require a full hardware mock-up of the robot OCU be used to interact with the simulator. The use of a real OCU to control the simulated vehicle will provide the best training experience for the operator. This class maps over to the FAA class D simulator, which require the simulator hardware to provide realistic instrumentation and force feedback to the pilot in a cockpit identical to that which would be seen in a real aircraft of the model being simulated. A simulator on a motion platform that is not tied to the simulation, which would simulate driving a remote vehicle from within moving vehicle, could possibly be considered a class E simulator with the motion being part of the full scale model of the control station. This is unlike the FAA class D requirements which specify the time in which the motion platform must respond to control inputs.

4. DISCUSSION AND EXAMPLE

In this section we give an example of how the classification system should be used to compare simulators and select one that best meets a researcher's requirements. The following is a comparison of two open source simulators, USARSim[9] and YARS (Yet Another Robot Simulator)[13]. USARSim was first released 2003 as a simulator for the US National Institute of Standards and Technologies (NIST) USAR test arenas. It is the currently simulator used by the RoboCup simulation league. YARS is currently in development at the University of South Florida in parallel with SARGE (Search And Rescue Game Environment)[12] was initially released in 2007.

The first step in choosing a simulator using the classification system is to identify a set of requirements for the experiment you will be conducting. This will determine the target class from which a simulator should be selected. The second step is to identify which simulators are in the target class. Ideally simulator developers would publish the rating for their simulator to make the selection process simple. In the case where no rating has been published, a simulator can be rated using the classification table. The final step, once a set of suitable simulators from the target class have been identified, is to choose one that best fits your budget and target platform.

The remainder of this section uses the classification table to

Requirement	Class A	Class B	Class C	Class D	Class E
• Class Descriptions					
• Class E simulators provide full scale mock up of the control unit of the simulated robot. Additionally Class E simulators must support all relevant features of Class D simulators.					
• Class D simulators provide simulation of all forces, control surfaces, and effectors of a robot so that operation of the simulated device is equivalent to the real device. Sensor simulations must approximate real output to the fullest ability of the simulation platform. Additionally Class D simulators must support all relevant features of Class C simulators.					
• Class C simulators simulate some forces, control surfaces, and effectors of a robot so that operation of the simulated device approximates the real device. Sensor simulations must approximate real output in a manner consistent with the goal of the simulator. Visual sensor simulation must be higher fidelity if the goal is human operation/training. Additionally Class C simulators must support all relevant features of Class B simulators.					
• Class B simulators approximate the motion of a robot and effectors, simulation of physical forces and control surfaces is not necessary. Class B simulators run in a basic 3D environment. Sensor simulations must be rough approximations of the real output, high fidelity is not required. Additionally Class B simulators must support all relevant features of Class A simulators.					
• Class A encompasses all simulators that do not meet the requirements for Class B or higher. Class A simulators approximate the motion of a robot, operation of effectors and sensors can be assumed. Class A simulators are not required to provide any visual output.					
• Physics					
• Surfaces with various frictional coefficients (dry cement vs. wet cement vs. icy cement) should be simulated if the robot will operate in these conditions.			X	X	X
• Effect of dynamic forces for various combinations of drag, thrust, and friction encountered in typical use conditions including the effect of change in robot pose, center of gravity, gross weight, and configuration; additionally altitude and temperature if these have significant impact on performance.			X	X	X
• Effector failure should be simulated appropriately. Failed motors should fail in a manner consistent with the robot (i.e. fail to brake vs fail to neutral). Vehicles with physical damage or lost parts should respond appropriately.			X	X	X
• Vehicle motion must be approximated, but simulation of forces is not required.	X	X			
• Vehicle/obstacle collisions should accounted for in the simulation.		X	X	X	X
• Sensors should be represented if present on the real vehicle.		X	X	X	X
• Aerial Vehicles Only				X	X
• Ground Effect - for example: roundout, flare, and touchdown. This requires data on lift, drag, pitching moment, trim, and power in ground effect.				X	X
• Ground Reaction - reaction of the aerial vehicle upon contact with the runway or landing surface during landing. Must include strut deflections, tire friction, side forces, and other appropriate data, such as weight and speed, necessary to identify the flight condition and configuration.				X	X
• Ground Handling characteristics - steering inputs to include crosswind, braking, thrust reversing, deceleration, and turing radius.				X	X
• Windshear models should be present				X	X
• OCU					
• A full-scale replica of the OCU for the robot simulated. Movement of the controls and switches is identical to that of the real OCU.					X
• All relevant instrument indications involved in the simulation of the robot quickly and automatically respond to control movement or external disturbances to the simulated robot.				X	X
• Communications equipment (microphone, headset) should be present and operate in the simulation as expected.				X	X
• Any control forces (force feedback) should respond in the same manner as the real robot under the same simulated operating conditions.				X	X
• Significant OCU sounds which result from operator actions should be present at similar volumes.				X	X
• Simulated vehicle must be operable by user through onscreen display or input device if vehicle is teleoperated.		X	X	X	X
• Environment					
• Simulator systems must simulate the robot in all intended environments. Systems must be operative to the extend that normal, abnormal, and emergency operating procedures can be accomplished.				X	X
• Environmental sounds correspond to actual sound at similar volumes. Wheel and track sounds should change with the operating surface (sand, gravel, carpet, etc.)				X	X
• Environmental conditions such as weather and time of day must be present if the robot will operate in such conditions and must be configurable by the user or instructor in the case of a training simulator.				X	X
• Environmental conditions such as weather and time of day should be configurable by the user or instructor in the case of a training simulator if the simulator supports multiple environmental conditions.			X	X	X
• Environmental sounds such as precipitation, fire, wind, etc. and vehicle sounds such as motor and wheel/track sounds should be present.			X	X	X
• Vehicles & environment must be represented in 3D		X	X	X	X

Figure 1. The Robot Simulator Classification Chart. Categories A-E Are Listed By Column With The Rows Describing Simulator Features.

rate USARSim and YARS. We will assume the desired simulation task involves a remotely operated unmanned ground vehicle (UGV) and a human operator that will drive the robot manually. The minimum level for a driving task should be class B, however higher classes increase the realism of the simulation. This choice is up to the researcher and does not affect how a simulator is examined for classification.

Start by examining the simulator with class A requirements and work up the scale until a requirement is not met. Both USARSim and YARS approximate vehicle motion using some force based calculations, so both can be considered usable for any class A task.

Both USARSim and YARS detect collisions between a robot and objects in the environment; both simulate robot sensors including audio, range, and video; both support joystick control of vehicles and effectors as well as control via commands sent over a network; and both render the environment in 3D. USARSim and YARS are usable for any class B task.

USARSim and YARS support surface friction variation for simulating different operating surfaces such as ice, and pavement. USARSim does not support dynamic force effects on the vehicle while YARS does support this feature. Effector failures cannot be simulated in USARSim correctly (parts cannot be removed on the fly to simulate vehicle damage), however YARS does support this feature. Both USARSim and YARS are capable of providing varying environmental conditions over time, however neither do as of February 2007, thus there is no need for a configuration option. USARSim and YARS simulate environmental and vehicle sounds. YARS meets all class C requirements, however USARSim does not currently meet class C requirements because dynamic force effects are not completely simulated.

At this point the evaluation for USARSim can stop as it will obviously not meet any of the higher classification levels. YARS can be evaluated against class D requirements, however few are supported at this time. Using the classification method presented, USARSim receives a class B rating and YARS receives a class C rating. At this point in the evaluation it is clear that YARS is a higher fidelity simulator. However there are other subjective factors that must be considered that are not part of the objective rating provided by the classification system. The cost of the development platform, cost of distribution, and ease of use must be considered when choosing a simulator.

For example USARSim runs on top of the UnrealTournament 2004 (UT2004)[8] video game, which must be purchased by both developers and users. The development environment only runs on Windows, however the add on packages can be used with the Linux and Mac versions of the game. UT2004 game is available for around $10 as of this writing. YARS is developed using the Unity[7] game engine, which must be purchased by the developer, however compiled games can be distributed freely by the developer. The Unity development environment only runs on the Macintosh OS but games created with Unity Pro can be run on Mac or Windows[2]. Unity is available as of this writing for $250 and the academic price for Unity Pro is $750. In addition to initial cost, development time and ease of use must also be considered. Development using the Unreal engine is much more complicated and time consuming than development using Unity.

5. CONCLUSION

This paper has proposed a classification system for mobile robot simulators in an attempt to aid robotics researchers who are in need of a robot simulator. The robot classification system is based on the FAA pilot training simulator classification system which has been in use and evolving for over two decades. Simulators are classified into five groups (A-E) depending on many characteristics derived from the FAA system. The system will allow HRI researchers to easily choose a simulator to meet their needs based on published specifications. By comparing the published information with the requirements chart, the proper classification can be determined. Ideally simulator developers will adopt the classification system and publish the classification along with the specification.

Just as the FAA system has been evolving since its adoption, the robot classification system must be updated as new simulator features are developed. Additionally the current system does not take into account subjective features such as ease of use and development costs. While the class rating focuses on simulation fidelity features, the subjective elements are important and must be considered when examining a simulator. A future version of the classification system should account for this.

This work was sponsored, in part, by the US Army Research Laboratory under Cooperative Agreement W911NF-06-2-0041. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the ARL or the US Government. The US Govern-

[2]Unity Pro is required to compile games to run on Windows

ment is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

REFERENCES

[1] Framework for post-residency surgical education and training. Available from: http://www.sages.org/sagespublication.php?doc=17.

[2] Immersion medical. Available from: http://www.immersion.com/medical/.

[3] Player/stage/gazebo. Available from: http://sourceforge.net/projects/playerstage

[4] Simbad 3d robot simulator. Available from: http://simbad.sourceforge.net/

[5] Simport. Available from: http://www.simport.eu/

[6] Simrobot. Available from: http://www.informatik.uni-bremen.de/simrobot/.

[7] Unity. Available from: http://www.unity3d.com.

[8] Unreal. Available from: http://www.epicgames.com/

[9] Usarsim. Available from: http://usarsim.sourceforge.net/.

[10] Webots. Available from: http://www.cyberbotics.com/products/webots/

[11] Alexander, A.L., T. Brunyé, J. Sidman, and S.A. Weil. "From gaming to training: A review of studies on fidelity, immersion, presence, and buy-in and their effects on transfer in pc-based simulations and games", November 2005. Available from: http://www.darwars.bbn.com/downloads/DARWARS Paper 12205.pdf

[12] Craighead, J., Sarge - search and rescue game environment. Available from: http://sarge.sourceforge.net.

[13] Craighead, J., Yars - yet another robot simulator. Available from: http://sarge.sourceforge.net.

[14] Craighead, J., R. Murphy, J. Burke, and B. Goldiez, "A survey of commercial and open source unmanned vehicle simulators," In *Proceedings of the 2007 International Conference on Robotics and Automation*, April 2007.

[15] Richer, J. and J.L Drury, "A video game-based framework for analyzing human-robot interaction: Characterizing interface design in real-time interactive multimedia applications," In *Proceeding of the 1st ACM SIGCHI/SIGART Conference on Human-Robot Interaction*, 2006, pp. 266 – 273.

[16] Schell, J., Hazmat: Hotzone. Available from: http://www.etc.cmu.edu/projects/hazmat_2005/.

[17] U.S. Department of Transportation Federal Aviation Admisistration, *Advisory Circular 120-40C*, July 1995. Available from: http://www.faa.gov/.

[18] Warmerdam, J., M. Knepfle, R. Bidarra, G. Bekebrede, and I. Mayer, "Simport: a multiplayer management game framework," In *Proceedings of the 9th International Conference on Computer Games*, 2006.

[19] Williams,K.W., "Qualification guidelines for personal computer-based aviation training devices: Private pilot certificate," Technical report, Federal Aviation Administration, 2001.