A Review of Virtual Simulators for Autonomous Underwater Vehicles (AUVs)

O. Matsebe*. C.M. Kumile**
N.S. Tlale***

*Department of Material Science and Manufacturing, Centre for Scientific and Industrial Research, Pretoria, South Africa (Tel: 0027 73833 7987; e-mail: OMatsebe@ csir.co.za).
**Department of Mechanical Engineering, Tshwane University of Technology, Pretoria, South Africa (email:kumilecm@tut.ac.za
*** Department of Material Science and Manufacturing, Centre for Scientific and Industrial Research, Pretoria, South Africa, (e-mail: ntlale@csir.co.za)}

Abstract: This paper overviews graphical simulators used for Autonomous Underwater Vehicles (AUVs) development**.** Graphical simulators allow researchers to develop autonomous software without the need for the actual vehicle. There are many graphical simulators developed to date, for somebody who is involved in AUV research it becomes difficult to decide which simulator to use without going through extensive literature review, this then necessitate a summary and classification of available simulators. Instead of a review of all underwater virtual environments developed to date, the paper describes a selection which only reflects the authors' preference and proposes a classification.

Keywords: Autonomous Underwater Vehicles, graphical simulators, online simulation, offline simulation.

1. INTRODUCTION

An AUV is an unmanned, untethered, sub-sea vehicle with sensors, actuators and on-board intelligence to carry out a mission without human operator interference or support from a ship (Groaning *et al*., 1998). Tasks that can be carried out by these vehicles include pipe assessment, cable inspection, seabed exploration, surveying, chemical and biological oceanography but to mention a few. As the complexity of AUV missions increases, it becomes necessary to design and develop new control architectures and algorithms. Validation of these control algorithms and architectures presents a challenge to researchers as it requires real – life testing and monitoring of the vehicles. However, real – life testing presents a high risk of damage or even loss of AUVs if the algorithm fails. It is then necessary to substitute real – life testing with simulation which is safer and cost effective (Gracanin *et al*., 1998). The use of a simulator avoids on- board data collection, scenario construction and underwater recovery of AUVs. Controller tuning, mission planning, fault tolerance, monitoring and analysis can all be performed in a 3-Dimensional (3D) virtual environment prior to the actual sea – trials (Choi *et al*., 1993). For a simulator to model the underwater environment and the vehicle accurately all features of the environment must be represented and it must also provide a user interface (Gracanin *et al*., 1998).

During the early stages of AUV development, offline simulators like MATLAB/SIMULINK play a very important role in modelling the vehicle and control systems. A bottleneck of this kind of simulation is that it does not represent the external environment and hence the external sensors such as sonar cannot be modelled. Again in offline simulation one second of simulation can be less or more than one second of reality as a result the time properties of the developed algorithms are not taken into consideration. On the other hand, the online simulators guarantee consistency between simulation and reality. It is also possible for a team of engineers to work concurrently on different features of the control system of a virtual robot (Ridao *et al*., 2004).

Another type of simulation is Hardware in the loop (HIL), it permits the developed algorithm to be executed in the real vehicle hardware while the vehicle's actions are directed towards the simulator. For an enhanced performance of the simulator, real and virtual systems can function together in an improved environment. This then makes the simulation of a vehicle navigating in a test tank while using virtual sensors possible. This is a HIL simulator configuration is known as Hybrid simulation (HS). The reliability of offline, online and HIL simulation depends on the accuracy of the vehicle, world, environment and sensor models being used (Ridao *et al*., 2004).

The world model is made up of data representing underwater agents like fish, vehicles and underwater objects like wrecks, Remotely Operated Vehicle (ROV) panels and the sub-sea structure. It permits the simulation of the vehicle's actions while underwater using its sensors to acquire information about its immediate environment. The environment model allows the simulation of environmental effects like tides, currents, waves and winds, though waves and winds affect the vehicle while it is on the surface. These affect the performance of the vehicle as it moves underwater (Ridao *et al*., 2004).

Similar work to review virtual simulators has been carried out by (Ridao *et al*., 2004 *and* Gracanin *et al*., 1998*)*.
(Gracanin *et al*., 1998) gives a brief description of several simulators but does not propose any classification whereas (Ridao *et al*., 2004) proposes a classification without any

description of those testbeds. This paper overviews, summarises, gives a description and proposes a classification of different virtual simulators.

The rest of the paper is structured as follows. Sections 2, 3, 4, 5, 6, 7, and 8 give descriptions and features of MVS, CADCON, IGW, NEPTUNE, DVECS, SUBSIM, and DEEPWORKS simulation testbeds respectively. Section 9 presents the discussions and the conclusions are found in Section 10.

2. MULTI-VEHICLE SIMULATOR (MVS)

The MVS was developed at the Underwater Robotics and Application Laboratory of the University of Tokyo, Japan for the Twin – Burger I AUV (Gracanin *et al*., 1998). This virtual underwater environment was designed as an incorporated developing environment to handle the various development stages, i.e. from consideration of mission strategies of a vehicle up to the last stage just before launching the vehicle into the ocean. The MVS was developed especially for handling multiple vehicle simulation and on-line works with actual vehicles. The synthetic world is produced by substituting some part of the real data with the virtual data hence facilitating the creation of virtual agents like vehicles and obstacles. This makes the actual AUV not to be able to discriminate between the real and virtual agents, it then becomes possible to create an underwater terrain in a testing tank. Since the actual AUV in the testing tank supplies the essential data, AUV related simulation (dynamics) is discarded (Kuroda *et al*., 1996). Detailed information on the MVS can be found in (Kuroda *et al*., 1996).

2.1 Features of MVS

- Multi – agent simulation.

The MVS can manage many agents (vehicles, obstacles, divers, seabed etc) concurrently. It has a distributed architecture where agents are linked by manager processes which provides the agents with virtual data such as disturbance, collision, and ranging sonar.

- Multi-user accessibility.

Work-group users can connect to the MVS through network and work together in the shared workspace.

- Real-time performance & connectivity to the actual systems.

The real-time simulation is important for investigating the availability of the developed software and hardware in the actual scene. The connectivity to the actual / virtual systems provides a large-scale simulation under worldwide network. MVS can also be used for tele-operation since researchers can get access to any networked agent.

- Multi-world & multi-CPU availability.

The distributed architecture of the MVS provides multi-cpu arrangement as a result the heavy processes can be carried out on other platforms linked through the network. A distinctive feature of MVS is the multi-world creativity. This allows construction of independent virtual worlds to other users hence reducing heavy network resource consumption.

- Real-time rendered 3D visualization

The MVS permits connecting the graphic subsystems to the actual vehicles that are equipped with vision sensors thereby providing them with visual images of the virtual environment.

- Portable system

The MVS usually works on any type of processing units connected through any type of networks. The core programs of the MVS are currently running under SUN, SGI, SVR4 and Linux on PCs, and INMOS's Transputers. They are connected to each other through TCP/IP or Transputer links (Kuroda *et al*., 1996).

3. COOPERATIVE AUV DEVELOPMENT CONCEPT (CADCON)

CADCON was developed at the Autonomous Undersea Systems Institute (AUSI) to support research in the area of multiple cooperative vehicles. It incorporates a multi – agent simulation, visualisation system and a control harness designed to simulate a fairly accurate underwater environment that can be shared by simulated or actual vehicles connected through the Internet (Chappell *et al*., 1999). It provides an open and flexible simulation environment for use by as many researchers as possible. CADCON tries to address issues in complex systems such as Autonomous Oceanographic Sampling Network (AOSN), i.e. problems related to the interactions amongst multiple heterogeneous agents; be those agents' simulations, real vehicles, or human users. CADCON has been used to test and evaluate solar powered AUV (SAUV) systems components and multiple cooperating vehicle mission profiles.

3.1 Features of CADCON

- All CADCON components have been developed to run on the IBM compatible computer.
- The client/server model is used for CADCON environment components and communication between them is through the Internet: TCP/IP.
- CADCON permits joint simulation scenarios by distant institutions since components are linked via TCP/IP sockets, therefore they may be distributed across intranets and Internet.

- The client/server model allows modular development. TCP/IP protocol permit.
- CADCON permits for real hardware in the loop simulation (HIL).
- The client/server model allows modular development. TCP/IP protocol permit.

4. INTEGRATED GRAPHICS WORKSTATION (IGW)

IGW was developed at the Autonomous Systems Laboratory of the University of Hawaii, USA for the Omni – Directional Intelligent Navigator (ODIN) AUV. It is made up of the 3-Dimensional background graphics (the data visualizer module), the vehicle graphics and interface (the advanced visualizer) module, the numerical (AUV dynamics) module, the control module, the communication module and the gage panel module (Choi *et al*., 1993).

The IGW is used for both the vehicle control station and graphic simulator. It works in two modes: remote control and autonomous modes. As a vehicle graphic simulator, the dynamic vehicle model in IGW with other actual data is used to build advanced vehicle technologies. IGW is used in the planning stage to simulate and practice the vehicle motion prior to the mission. It is also used to supervise vehicle motion (Choi *et al*., 1993).

4.1 Features of IGW

- Modular approach

As a result of the modular approach, development, trouble shooting, testing and incorporation of new modules are all independent.

- Remote control mode

The global coordinate data from the vehicle is forwarded straight into the workstation, this information is then utilized by IGW to continuously update graphic views of the vehicle on the screen. The operator directly controls the vehicle with a mouse, or joystick.

- Autonomous mode

In the autonomous mode, IGW works as a monitor and the onboard computer takes control of the vehicle.

5.NEPTUNE

NEPTUNE is a real-time graphical simulator developed at the Institute of Informatics and Applications, University of Girona for the URIS Unmanned Underwater Vehicle (UUV). It has the potential for running on-line, hardware in the loop (HIL) and hybrid simulation. It models UUV using their hydrodynamic equations whereas thrusters are modelled through the affine model. The world is modelled using VRML together with bathymetry model and sonar is modelled using a geometric method. NEPTUNE can simulate the following sensors: range detection sonar, vision and internal sensors (position, attitude, speed and depth) (Ridao *et al*., 2004).

5.1 Features of NEPTUNE

- Flexibility

New virtual worlds and UUV models can be easily incorporated.

- Multi – Vehicle Simulator

More than one robot can be simulated at the same time.

- Distributed application

To permit real time performance, the application entails a number of processes i.e. NEPTUNE main program, one robot dynamics process for each simulated robot and a name server.

- All the programs communicate via TCP/IP network

6. DISTRIBUTED VIRTUAL ENVIRONMENT COLLABORATIVE SIMULATOR (DVECS)

DVECS is a 3-dimensional, virtual, hybrid, synthetic, collaborative world simulator developed at the Autonomous Systems Laboratory of the University of Hawaii for the Semi - Autonomous Underwater Vehicle for Intervention Missions (SAUVIM). It is used for assessment of unmanned underwater vehicles (UUV) of both actual and simulated worlds where collaboration of other real and simulated vehicles, situations, obstacles, conditions and disturbances in a virtual environment can be observed without physical interference (Choi *et al*.).

6.1 Features of DVECS

- It can be used to establish the best performance and criteria for the cooperating vehicles and its relative application.
- It can be used to determine the advantages and disadvantages of cooperative application tasks between multiple UUVs.
- To find optimal communication links between the cooperating vehicles and its remote control stations.
- It is used as monitoring system.
- Distributed approach

DVECS's architecture has been developed to work in a networked environment where each component of the simulation can be processed on a separate system, hence distributing the computation load.

- Users can develop and assess their AUV simulations across the Internet using common servers.

Multiple simulated or actual agents can interact over a networked environment without sharing code. Protected algorithms can be assessed in a shared environment without making them public.

- Links via the Internet also permits collaboration by researchers from many different research institutions

7. SUBSIM

SUBSIM is a 3 – dimensional dynamic simulator for AUVs and the external environment developed at the University of Western Australia. It permits assessment of application design, mission simulation, controller tuning and fault – tolerance, therefore there is no need for onboard data collection, scenario construction and underwater recovery of AUVs. SUBSIM accurately models the motion of objects under the influence of forces and collisions using a 3rd party engine called Newton. Sensors, motor, and liquid effects were modelled using the Physics Abstraction Layer (PAL) library. It uses OpenGL along with a custom high-level interface for graphical models. SubSim supports Milkshape 3D models. It has been tested on the Windows XP and Windows 98 Second Edition operating systems, on both Intel Pentium and AMD Athlon processors. Detailed information on SUBSIM can be found in the University of Western Australia Web page.

7.1 Features of SUBSIM

- Simulation start/pause/stop and simulation time display.

- Application programming interface that compatible with C and C++.

- Customization of simulation parameters, AUV and environment through XML files.

- Extensibility through C++ plugin model.

8. DEEPWORKS

Deep Works was developed for the 3 dimensional simulations of all aspects of underwater systems by GRL in the United Kingdom. Its base product Deep View has the capacity to create complex sub-sea environments. Deep Works has three other add-on modules, namely Deep Live, DeepSim, and DeepTouch which enhances its real-time dynamic simulation capabilities. DeepView has play and replay functions of any sub-sea scenario. These permit operator training and mission planning. DeepLive allows real-time monitoring and visualisation of offshore assets for improved safety and performance. DeepSim models the true hydrodynamic response of components when exposed to water effects therefore it adds dynamic, real-time simulation to the platform. DeepTouch permits interactions between any objects in the world model so that the simulation is as close to reality as possible. Detailed information can be found in the GRL website.

8.1 Features of DeepWorks

- Mission planning in a virtual environment

- Operator training

- Visualisation of complete environments with real-time vehicle tracking

9. DISCUSSIONS

Table 1 in Appendix A summarises the main features of the simulators described above. From Table 1 it can be noticed that different simulators possess different characteristics depending on the applications they were designed for. Table 2 gives a classification of functions that can be performed by simulators according to the features they possess. The classification provided below is only for the simulators discussed in this paper.

10. CONCLUSIONS

Graphical simulators play a key role in AUV development. Depending on the application, they can perform in different ways that is; offline, online, hardware in the loop (HIL) and hybrid simulation (HS). Graphical simulators can also be used as monitoring systems and/or mission playback or for operator training. They also vary in the type of AUV, world, environment and sensor models they use.

REFERENCES

Albert, E., J. Bilodeau and R.M. Turner (2003). Interfacing the CODA and CADCON simulators: A multi–fidelity simulation testbed for autonomous oceanographic sampling networks. *Proceedings of the 13th international symposium on unmanned untethered submersible technology (UUST).* August 2003, Durham,NH. http://mainesail.umcs.maine.edu/Papers/2003/UUST03-coda/AlbertBilodeauTurnerUUST03.pdf accessed on 18/12/2007.

Chappell, S.G., R.J. Komerska, L. Peng and Y. Lu (1999). Cooperative AUV development concept (CADCON):

An environment for high-level multiple AUV simulation. *In proceedings of the eleventh international symposium on unmanned submersible technology.* August 23–25 1999, Durham NH. Web site http://www.ausi.org/publications/cadcon.pdf accessed 17/12/2007.

Choi, S.K. and J. Yuh (1993). Design of advanced underwater robotic vehicle and graphic workstation. IEEE web page http://ieeexplore.ieee.org/iel2/904/7227/00292131.pdf?tp=&isnumber=&arnumber=292131 accessed on 18/12/2007.

Choi, S.K. and O.T. Easterday. An underwater vehicle monitoring system and its sensors. Web site http://books.google.com/books?id=6ic4L4NtvBAC&pg=PA551&lpg=PA551&dq=choi+sk+and+ot+easterda+an+underwater+vehicle+monitoring+system+and+its+sensors+autonomous+systems+laboratory+department+of+mechanical+engineering+university+of+hawaii+at+manoa&source=web&ots=s4n8GJcpvK&sig=qWJ1oUkFXzAZ4RKeER4fYjHyDlY#PPA554,M1 accessed on the 19/12/2007.

Gracanin, D., K.P. Valavanis and M. Matijasevic (1998). Virtual environment testbed for autonomous underwater vehicles. *Control Engineering Practice*, **6,** 653-660**.** IEEE xplore web site http://ieeexplore.ieee.org/iel3/4880/13463/00620272.pdf?isNumber= accessed on 19/12/2007.

Kuroda, Y., K. Aramaki and T. Ura (1996). AUV test using real/virtual synthetic world. *Proceedings of IEEE Conference on autonomous underwater vehicle technology.* June 3-6, Monterey, CA. IEEE xplore web site http://ieeexplore.ieee.org/iel3/3780/11039/00532436.pdf?isNumber= accessed on 17/12/2007.

Ridao, P., E. Batlle, D. Ribas and M. Carreras (2004). NEPTUNE: A HIL simulator for multiple UUVs. Web site http://vicorob.udg.es/fitxers/articles/doc/137.pdf accessed on the 19/12/2007.

University of Western Australia Web page, http://robotics.ee.uwa.edu.au/auv/subsim.html, accessed on 18/12/2007.

GRL Web page http://www.generalrobotics.co.uk/html/downloadpage_47.html accessed 04/01/2008.

Appendix A

Table 1. Summary of simulation testbeds (features and functions)

	SUBSIM	CADCON	NEPTUNE	MVS	DVECS	IGW	DeepWorks
Graphics	3D (using OpenGL, support milkshape models)	3D	3D (using OpenGL)	3D	3D (using OpenGL)	3D	3D
Type of Simulation	Offline	Online, HIL	Online, HIL, HS	Online, HIL, HS, (Performs Mission Playback)	Online, HS, HIL (Performs Monitoring)	Online, HIL, HS, (performs Monitoring, Operator Training)	Offline, HIL, (Performs Monitoring, Mission Playback, Operator Training)
Real Time	NO	YES	YES	YES	YES	YES	YES
World modelling	YES (Using Newton physics engine)	Bathymetry model	VRML model, Bathymetry model	YES	YES	YES	YES
Environment modelling	YES (Using Newton physics engine)	YES (SMTIC model)	Not yet supported	YES	YES (Models Currents)	YES (using local underwater mapping data)	YES (models currents, tides, clouding, particles)
Internal (I) & External (E) sensors	YES (using Physics Abstraction Layer)	YES	Sonar (E), Vision (E) Position, speed, depth & Attitude sensor(I)	Speed sensor (I), Depth sensor (I) Sonar (E), vision (E)	Sonar (E), vision (E)	YES	Vision (E), Sonar (E)
Multiple-vehicle	NO	YES	YES	YES	YES	NO	NO
Vehicle (V) & Thruster (T) modelling	Hydrodynamics (V), Dynamic Model (T)	YES	Hydrodynamics (V) Affine model (T)	YES	Hydrodynamics (V) Affine model (T)	Hydrodynamics (V)	Hydrodynamics (V)
Distributed Environment	NO	YES (TCP/IP network)	YES (TCP/IP network)	YES (TCP/IP network)	YES (TCP/IP network)	YES	NO
Institution / Organisation	Robotics & Automation Lab, The University of Western Australia	Autonomous Undersea Systems Institute, USA	Computer Vision and Robotics, University of Girona, Spain	Underwater Robotics and Application Laboratory, the University of Tokyo,	Autonomous Systems Laboratory, University of Hawaii, USA.	Autonomous Systems Laboratory, University of Hawaii, USA.	GRL

Application (Field robots)	Developed for the 1st International Autonomous Underwater Vehicle Competition in Australia	SAUV – AUSI, EAVE, EST	URIS, GARBI	The Twin-Burger I, PTEROA, MANTA	SAUVIM, ODIN	ODIN AUV	Developed for the simulation of all aspects of underwater systems
Multi-user accessibility	NO	YES	YES	YES	YES	YES	NO
Computer System / language compatability	Windows XP, Windows 98/ C, C++	IBM compatible computers		Runs on any type of processing unit, on any type of network. SUN, SGI,SVR4,LINUX		UNIX / C	Microsoft Windows
Type	Open source	Open source	Open source	Open source	Open source	Open source	Commercial

Table 2. Classification of functions according to features possessed by simulators.

Functions

	Monitoring	Mission Playback	Operator Training	Hardware In the Loop simulation	Hybrid Simulation	Online Simulation	Offline Simulation
Features							
Distribution	√	√		√	√	√	
TCP/IP	√	√		√	√	√	
Vehicle model	√	√	√	√	√	√	√
Thruster model	√			√	√	√	√
Environment model	√	√	√	√	√	√	√
World model	√	√	√	√	√	√	√
External sensors	√	√	√	√	√	√	√
Internal sensors	√	√		√	√	√	√
3D graphics	√	√	√	√	√	√	√
Real Time	√	√	√	√	√	√	√