

A REAL-TIME SIMULATOR FOR AUV DEVELOPMENT

João Lucas Dozzi Dantas, joao.dantas@poli.usp.br

Ettore Apolonio de Barros, eabarros@usp.br

Escola Politécnica da Universidade de São Paulo

Abstract.

This work presents a real-time software for simulating the motion and operation of the main systems of an AUV. The real-time characteristics were defined so that the AUV control system can be tested taking into account the real characteristics of the embedded hardware, such as sensors and actuators. The simulation software includes a number of modules that represent the vehicle dynamics, environmental perturbations, actuators and sensors dynamics. In order to provide an easy integration of the simulation into the HIL system, the "Real-Time Workshop" together with the xPC Target were chosen as the development environment. They provide the easy and simple execution on real-time of models generated by Matlab/Simulink tools.

Keywords: *Hardware-in-the-Loop, Simulator, AUV, Real-Time, Vehicle Dynamics*

1 INTRODUCTION

The presence of simulators in the academy or industrial environment is quite usual since they can provide fast testing of devices and more complex systems at lower cost when compared to experiments on prototypes.

However, in mechatronic systems, modeling components such as sensors and actuators may provide unreliable results. In order to improve the realism of simulators, the hardware in the loop, HIL, approach includes components such as sensors or actuators that are embedded into a virtual environment. In field robotics, HIL has been used for testing the performance control and navigation software of autonomous aerial (Shixianjun et al., 2006, Imado et al., 2006 and Göktogan et al., 2003) and underwater (Lane et al., 2001, Song et al., 2003 and Devie and Lemaire, 1998) vehicles, for instance.

This work presents the real-time simulation software that is going to be integrated into a HIL system of an autonomous underwater vehicle, AUV. The vehicle equations of motion are implemented into the simulation software, which generates the motion variables to be used by the real-time control system. In the future, such variables will be used as reference values for moving a mechanical platform which supports the real sensors included into the AUV. The actual signals from the sensors are going to be used for feeding the control and navigation software instead of the simulated ones.

The final objective of this simulator is to test the control systems in the AUV embedded computer, both in software and hardware levels. The embedded computer test is going to be conducted using the simulator operating in real time, i.e. the simulation loop of the HIL system only considers the hardware and software of the AUVs embedded computer, and a computer running the simulation software in real time. In such way, the HIL simulator calculates the dynamics of the simulated vehicle, and it generates the sensors signals in a manner close to the actual operation of the AUV.

The technique of simulation in a HIL system was studied and implemented in the works Shixianjun et al. (2006) and Imado et al. (2006). In the first work, the implementation occurred using a mechanical platform with one degree of freedom to perform the simulation of the attitude of an unmanned aerial vehicle (UAV). The second paper presents a robot manipulator of six degrees of freedom to simulate the position and orientation of the simulated vehicle.

As the HIL simulator must operate online with actual devices, one of its main requirements is to run in real time, providing good synchronization among all its critical systems Lane et al. (2001). As in the system implemented by Shixianjun et al. (2006), in order to make the development of real-time simulator easy and practical, the following off the shelf, software tools were adopted in the HIL system. The Matlab (MathWorks, 2008a) was selected for high-level programming; while the Simulink(MathWorks, 2008c) was chosen for modeling and simulation, The Real-Time Workshop (MathWorks, 2008b) was selected as a conversion tool of the code for real-time operation, and the xPC Target (MathWorks, 2008d), is the real-time environment of simulation. All those systems provide a number of tools for simulation and communication, that generates very complex, optimized and real-time software in a quickly and practical way.

A second important requirement for this software concerns the AUV dynamics simulation, which needs to be quite realistic. This work borrows from Song et al. (2003) which includes a HIL simulator containing the dynamics of the simulated vehicle, sensors and the environment around the AUV. Moreover, the actuators dynamics were modeled as well.

The mathematical models of dynamic equations governing the movement of the vehicle, used in this simulator, were based on the methodology presented by Ishidera et al. (1986). However, in order to model the hydrodynamic coefficients of AUVs, semi-empirical and analytical formula based on the vehicle geometry have been adopted, which were thoroughly studied and verified in the works de Barros et al. (2009), de Barros et al. (2004), de Barros et al. (2006) and Dantas (2008).

Section 2 introduces the AUV Pirajuba, which is the AUV to be simulated in the HIL system.

The software structure is described in Section 3, together with the main characteristics of its modules.

In Section 4 are present the equations of motion for representing the AUV dynamics. The hydrodynamic and hydrostatic efforts acting upon the vehicle are described. This module is also supposed to include representations of the environmental effects (like current forces, for example), models of sensors and actuators.

Section 5 shows some results generated by the simulator when representing typical maneuvers, the visual animation resource, and the records of real-time operation.

Finally, Section 6 presents final considerations and points to future developments of the HIL simulator.

2 THE AUV PIRAJUBA

Pirajuba (Fig. 1) is a AUV with 1.78m in length, 0.234m in maximum diameter that has the "Myring" geometry (Myring, 1976), which has already been applied in the investigation of aircraft's fuselage and other AUVs such as Maya and Remus (Prestero, 2001).

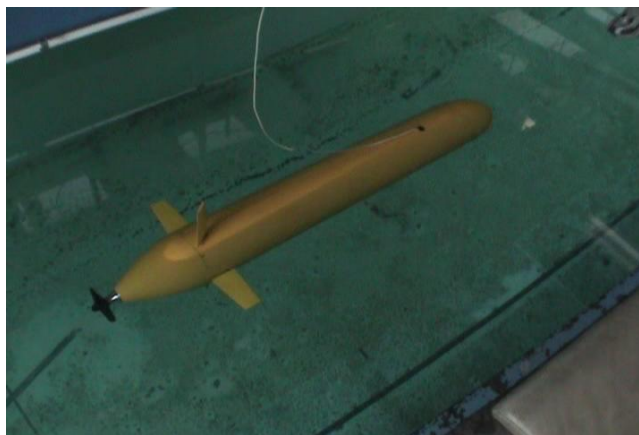


Figure 1. AUV Pirajuba under a self-propelled tank test in the IPT.

The vehicle has autonomy of four hours at cruising speed of 1m/s.

The embedded systems are grouped in three pressure vessels, consisting by cylinders of aluminum: the main, which includes the control, sensing and energy systems; the maneuver vessel, which includes the servo-actuators of rudders and corresponding drive electronics; and the propulsion vessel, with include the motor reducer system of the propeller actuation. The three vessels are linked by sea connectors and grouped in a glass-fiber hull flooded ("free-flooding").

The embedded system in the main pressure vessel is arranged in a rack structure of nylon, with longitudinal metallic elements of bulkhead union (which also are of nylon), as shown at Fig. 2.

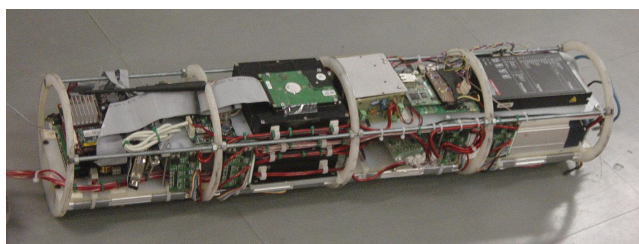


Figure 2. Embedded electronics loaded in the principal pressure vessel.

The hardware architecture includes a central unit based on a PC/104, and a number of distributed units based on a microprocessor boards. The main unit performs the principal tasks during the mission as the acquisition of data from sensors, navigation, guidance, control and management of the mission.

The embedded sensors are an Inertial Measurement Unit (IMU), a GPS receiver, a module of electronic compass TCM2, a depth sensor, current sensors in the actuators and a liquid level sensor. The IMU, compass and GPS have RS232 standard output and are read directly by the central unit.

The actuation system consists of four "high torque" futaba servos and a servo motor of 150W. All actuators are controlled from an Ethernet network link. And to make the interface between the actuators input (PWM and +10V) and the Ethernet network were used RCM4200 modules, which have capacity of A/D conversion and PWM generation. The

RCM4200 modules also make the interface with analog sensors in general, as the depth sensor, the current sensors, and the liquid presence sensors.

The protocol used in the Ethernet network is the UDP-IP, exchanging messages in the ASCII format standard among the main unit and the other nodes. On the same network is found the data link, which is a 9Xtend-pkg-E Ethernet RF modem, that has a bandwidth of 115Kbps and connects the embedded network and the base station when the vehicle is near the surface.

The base station consists of a PC with a Linux based operating system that execute a program to capture the commands on a joystick, display on screen the states of AUV and send commands to the embedded system.

A schematic representation of the main components of the embedded electronic, except for propulsion and the base station, is shown in the Fig. 2.

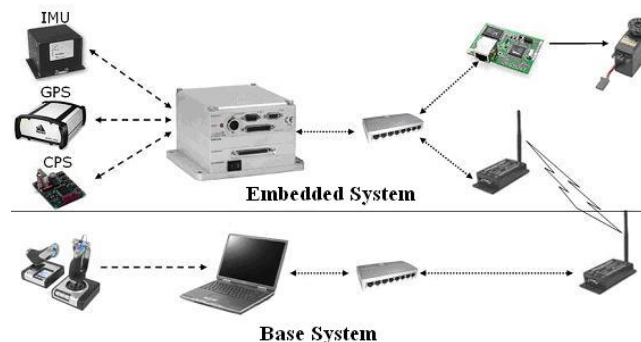


Figure 3. Schematic representation of the main embedded system and the base system.

3 SOFTWARE STRUCTURE

In order to provide flexibility to the simulator software architecture, one of the main requirements imposed during its development was the modular design. Using a modular system implies that one can modify or replace a subsystem without imposing changes to other parts of the software.

Other important requirements for the software architecture are related to the simulation inclusion into the HIL system. These requirements are: operations should be synchronized in real-time and the software should have the ability to communicate with other systems such as hardware devices. The adopted solutions for implementing these features are based on functions from the Mathworks toolboxes that allow the user to define and expose very specific configurations of the software.

Therefore, the simulation software presented in this work was divided into the following modules, with each of these composed of other sub-modules:

- **Communication Controller:** This module has the implementation of the protocol for communication between the simulator computer and the embedded computer (to be used inside the AUV). In general, the hardware and software tested in the HIL system are not modified, and in order to test them the simulator must have a very customizable and robust interface of communication. Considering this, it was implemented communication systems of Ethernet type using the UDP protocol, with systems of generating and reading messages in different formats (such as double, string and binary) that are not native to the Mathworks Simulink software. The functions of communication in this module were based on the devices existing in the communication's software of the embedded system in the AUV of study, presented on Amianti (2008).
- **Communication to the Platform:** This module is responsible for the communication between the simulator and the mechanical platform which where the real sensors are located. Except for the messages, that are encoded differently, this module uses the same type of communication described in the Communication Controller module.
- **Dynamics:** In this module, the AUV motion dynamics in the marine environment is simulated, including also the actuators and sensors representation. A more detailed explanation of this module will be given in the Section 4.
- **Results Interface:** This module implements the interface between the motion simulation and the user. Graphics showing the time history of specific variables are included as well as animation resources for analysis of the AUV performance.

The HIL system should be a flexible and modular architecture, so that the system is able to operate replacing a module by the equivalent simulated one. This HIL system will consist of five modules: a controller, executed on a embedded

hardware (PC/104); a mechanical platform, which held the actual movement of the real sensors; the platform's computer, which will do the reading of the platform's sensors and the execution of actuators; a computer interface, that uses a graphical interface to show the dynamics of the AUV simulated in real time; and a base computer, which will perform the calculations of dynamics, simulation of some sensors (not present in mechanical platform) and communication between the modules of the system.

4 DYNAMIC MODELING

The dynamics module has the main function of modeling all systems that interact with the dynamics of an AUV, in other words, it makes the calculations of the vehicle trajectory, the environment efforts acting upon it, and the response of sensors and actuators.

Since these systems present very distinct functions they should be modeled in smaller parts in order to ensure the high flexibility required for the implementation of this software.

Therefore, the dynamics simulation software of the entire AUV will be divided into four different sub-modules, which will be presented with more detail in the next sections.

4.1 Equations of Motion Sub-Module

The AUV main dynamics is represented by this part of the software. It means that, in this sub-module the differential equations governing the motion of the AUV are solved, producing the efforts acting upon the vehicle during its trajectory. The methodology presented in this section is mainly based on Ishidera et al. (1986), but the work Nelson (1989) had a significant contribution to understanding the physical and mathematical modeling of submerged bodies.

This module should also perform the transformation of coordinates between the inertial and non-inertial (fixed to the vehicle) reference frames, so that external disturbances can be represented into the vehicle reference frame.

The equations of motion governing the dynamics of a rigid body immersed in a fluid, in this case an AUV, can be expressed by Eq. (1) when represented in the coordinate system fixed in the vehicle. This vector equation has six lines, each line represents a linear or angular velocity in an axis of the vehicle.

$$[m_{ij} + A_{ij}] \cdot [\dot{V} \cdot \dot{\omega}] + F_I = F_H + F_G + F_B + F_T \quad (1)$$

The first term on the left side of the Eq. (1) includes the generalized mass matrix of the vehicle, presented in Ishidera et al. (1986). It is composed by the inertial mass terms added to the added mass terms Newman (1977). This matrix is multiplied by the vector of linear and angular acceleration. The definition of these matrices can be seen in Eq. (2).

$$[m_{ij} + A_{ij}] \cdot [\dot{V}, \dot{\omega}] = \begin{bmatrix} m & 0 & 0 & 0 & m \cdot z_G & -m \cdot y_G \\ 0 & m & 0 & -m \cdot z_G & 0 & m \cdot x_G \\ 0 & 0 & m & m \cdot y_G & -m \cdot x_G & 0 \\ 0 & -m \cdot z_G & m \cdot y_G & I_X & -I_{XY} & -I_{XZ} \\ m \cdot z_G & 0 & -m \cdot x_G & -I_{XY} & I_Y & -I_{YZ} \\ -m \cdot y_G & m \cdot x_G & 0 & -I_{XZ} & -I_{YZ} & I_Y \end{bmatrix} + \begin{bmatrix} A_{11} & A_{12} & A_{13} & A_{14} & A_{15} & A_{16} \\ A_{21} & A_{22} & A_{23} & A_{24} & A_{25} & A_{26} \\ A_{31} & A_{32} & A_{33} & A_{34} & A_{35} & A_{36} \\ A_{41} & A_{42} & A_{43} & A_{44} & A_{45} & A_{46} \\ A_{51} & A_{52} & A_{53} & A_{54} & A_{55} & A_{56} \\ A_{61} & A_{62} & A_{63} & A_{64} & A_{65} & A_{66} \end{bmatrix} \cdot \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \\ \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} \quad (2)$$

The second term on the left side of Eq. (1), " F_I ", is called inertial effort, which represents the efforts generated in the vehicle due to its linear and angular velocities. Its formulation is defined by Eq. (3).

$$F_I = \begin{bmatrix} m \cdot [-v \cdot r + w \cdot q - x_G \cdot (q^2 + r^2) + y_G \cdot p \cdot q + z_G \cdot p \cdot r] \\ m \cdot [-w \cdot p + u \cdot r - y_G \cdot (r^2 + p^2) + z_G \cdot q \cdot r + x_G \cdot p \cdot q] \\ m \cdot [-u \cdot q + v \cdot p - z_G \cdot (p^2 + q^2) + x_G \cdot p \cdot r + y_G \cdot q \cdot r] \\ m \cdot [y_G \cdot (-u \cdot q + v \cdot p) - z_G \cdot (-w \cdot p + u \cdot r)] \\ m \cdot [z_G \cdot (-v \cdot r + w \cdot q) - x_G \cdot (-u \cdot q + v \cdot p)] \\ m \cdot [x_G \cdot (-w \cdot p + u \cdot r) - y_G \cdot (-v \cdot r + w \cdot q)] \end{bmatrix}$$

$$+ \begin{bmatrix} 0 \\ 0 \\ 0 \\ (I_Z - I_Y) \cdot q \cdot r - I_{XZ} \cdot p \cdot q + I_{YZ} \cdot (r^2 - q^2) + I_{XY} \cdot p \cdot r \\ (I_X - I_Z) \cdot p \cdot r - I_{XY} \cdot q \cdot r + I_{XZ} \cdot (p^2 - r^2) + I_{YZ} \cdot p \cdot q \\ (I_Y - I_X) \cdot p \cdot q - I_{YZ} \cdot p \cdot r + I_{XY} \cdot (q^2 - p^2) + I_{XZ} \cdot q \cdot r \end{bmatrix} \quad (3)$$

The first term on the right side of the Eq. (1), " F_H ", represents the hydrodynamic efforts generated by vehicle, due to orientation (angle of attack, " α ", and angle of sideslip, " β "), angular velocities (" p, q, r ") and the deflection of the control surfaces (of rudder, " δ_r ", and elevator, " δ_e ") of the vehicle in relation with the water velocity.

This term may be represented by the Eq. (4), where " ρ " is the water density, " V " is the relative velocity between vehicle and water, " S_{REF} " and " L_{REF} " are the area and length of reference of the hydrodynamic coefficients and terms " C_X ", " C_Y ", " C_Z ", " C_L ", " C_M " e " C_N " are respectively the hydrodynamic coefficients of axial, lateral and vertical forces, and of roll, pitch and yaw moments.

$$F_H = 1/2 \cdot \rho \cdot V^2 \cdot S_{REF} \cdot \begin{bmatrix} C_X(\alpha, \beta, p, q, r, \delta_r, \delta_e) \\ C_Y(\beta, r, \delta_r) \\ C_Z(\alpha, q, \delta_e) \\ C_L(p) \cdot L_{REF} \\ C_M(\alpha, q, \delta_e) \cdot L_{REF} \\ C_N(\beta, r, \delta_r) \cdot L_{REF} \end{bmatrix} \quad (4)$$

Since the AUV Pirajuba has a lateral and horizontal plane of symmetry the coefficients of vertical and lateral forces and moments have equal magnitude. The signals of those coefficients are defined according to the body fixed reference frame:

$$C_Y = C_Z = -C_{FN} \quad (5)$$

$$C_M = -C_N \quad (6)$$

The estimation of these hydrodynamic coefficients can be based on analytical and semi-empirical, ASE, approaches such as those presented by Hoak and Finck (1978), Jorgensen (1977), and Roskam (1985). Expressions for the vertical force and moment coefficients are:

$$C_{FN} = C_{FN_{BODY}}(\alpha, q) + K \cdot C_{FN_{WING}}(\alpha, q, \delta_e) \quad (7)$$

$$C_M = C_{M_{BODY}}(\alpha, q) + K \cdot C_{FN_{WING}}(\alpha, q, \delta_e) \cdot x_{HC_{WING}} \quad (8)$$

For the lateral motion case, the arguments change to " β ", " r " and " δ_r ", in the functions of force and moment.

The variable " K " is the coefficient of influence between body and the wings of the vehicle, derived from the slender body theory, Pitts et al. (1957), and the variable " $x_{HC_{WING}}$ " is the longitudinal distance between the hydrodynamic center of wings and the vehicle coordinate origin.

For the bare hull of Pirajuba, it was possible to improve the estimation of both coefficients (" $C_{FN_{BODY}}$ " and " $C_{M_{BODY}}$ "), combining ASE and CFD approaches de Barros et al. (2009), de Barros et al. (2004), de Barros et al. (2006) and Dantas (2008).

On the right side of the Eq. (1), it is also considered the efforts of the vehicle propeller, " F_T ". In the case of an AUV with the propeller positioned on its longitudinal axis such efforts have only one force component, due the force of thrust, and one of moment, due to the residual moment generated by the propeller. Both efforts are given in terms of the vehicle speed and the rotation of the propeller using the " $K_T - K_Q$ " curves, in Lewis (1967) is found a formulation for this type of propulsion.

Finally, efforts of the gravity, " F_G ", and buoyancy, " F_B ", are accounted for. The first depends on the mass of the vehicle (" m "), the acceleration of gravity (" g "), center of mass of the vehicle (" x_G, y_G, z_G ") and the orientation of the vehicle in inertial reference given by the Euler angles (" ϕ, θ, ψ ").

The buoyancy force depends on the volume of the vehicle (" ∇ "), the density of water (" ρ "), the acceleration of gravity, the center of buoyancy of the vehicle (" x_B, y_B, z_B ") and the Euler angles, in a similar manner to the gravitational efforts. The Eq. (9) defines the sum of these two terms, also known as hydrostatic forces.

$$F_G + F_B = g \cdot \begin{bmatrix} -(m - \rho \cdot \nabla) \cdot \sin(\Theta) \\ (m - \rho \cdot \nabla) \cdot \sin(\Phi) \cdot \cos(\Theta) \\ (m - \rho \cdot \nabla) \cdot \cos(\Phi) \cdot \cos(\Theta) \\ (m \cdot y_G - \rho \cdot \nabla \cdot y_B) \cdot \cos(\Phi) \cdot \cos(\Theta) - (m \cdot z_G - \rho \cdot \nabla \cdot z_B) \cdot \sin(\Phi) \cdot \cos(\Theta) \\ -(m \cdot x_G - \rho \cdot \nabla \cdot x_B) \cdot \cos(\Phi) \cdot \cos(\Theta) - (m \cdot z_G - \rho \cdot \nabla \cdot z_B) \cdot \sin(\Theta) \\ (m \cdot x_G - \rho \cdot \nabla \cdot x_B) \cdot \sin(\Phi) \cdot \cos(\Theta) + (m \cdot y_G - \rho \cdot \nabla \cdot y_B) \cdot \sin(\Theta) \end{bmatrix} \quad (9)$$

Table 1. Turning radius in function of the rudder angle.

Rudder Angle ("δ _r ")	Turning Radius
5°	10.0 m
10°	4.2 m
15°	2.8 m
20°	2.8 m

4.2 Environment Sub-Module

This module is responsible for calculating both the vehicle's relative velocity in the water, considering flow models, and the hydrodynamic properties of water, like pressure and density as function of depth and temperature. The flow model considers that water has a constant speed with a specific direction in relation to the sea bottom fixed coordinate system.

In the future, in order to carry out a more efficient test of the AUV's control system, this sub-module will implement a generator of disturbance in the velocity of water through models of waves, so that water has velocity components in three directions in as function of time and space. The implementation of this model will be performed using the same methodology presented in Song et al. (2003), using models of waves based on well know sea spectra, Newman (1977), for fully developed waves generated by wind in the sea.

4.3 Virtual Sensors Sub-Module

This unit is in charge of modeling the dynamics and inaccuracies of the AUV sensors. The sensor response is modeled as function of the vehicle dynamics and it includes the effect of noise, bias, quantization, frequency and delays of the signal.

In this version of the simulator, the represented sensors are: Compass, Global Position System (GPS), Inertial Measurement Unit (IMU) and a Doppler Velocity Log (DVL).

4.4 Virtual Actuators Sub-Module

In this unit, the AUV propulsion and maneuver system are represented. The maneuver system includes rudders and elevators that are deflected by servomechanisms based on electrical actuators. These servomechanisms are modeled by second order transfer functions together with saturation for limiting maximum/minimum angle and his rate.

5 RESULTS

Some typical maneuvers of the AUV Pirajuba were simulated. At first, the time history of main variables is presented graphically. Then, the animation interface was used for visual analysis of the AUV performance. Finally, a methodology is presented for verifying the simulation for real time performance.

5.1 Maneuvering Results

The first maneuver tested was the Turning Maneuver, where the controller provides a step reference signal to the vertical surface, making the vehicle perform a circular maneuver. In that way it is possible to measure the diameter of the trajectory performed by the vehicle when it is on steady state, and compare this result with a result obtained by analytical methods.

In Fig. 4, it can be observed the trajectories related to four different rudder angles (5, 10, 15 and 20 degrees). The Tab. 5.1 shows the approximated values of the diameters of these trajectories for each of these deflections.

The AUV responses during Zig-Zag maneuver are represented in Fig. 5. From those is represented the transitory and nonlinear behavior of the AUV dynamics can be analyzed.

5.2 Results Interface

As mentioned, to have a better understanding of the simulated AUV behavior it necessary implement an interface for displaying three-dimensional animation, with both the vehicle and the simulated environment.

The animation is implemented using two different softwares. The Ligos V-Realm Builder, with uses the Virtual Reality Modeling Language (VRML) to realize the three-dimensional modeling, and the Matlab and Simulink to perform the real-time motion of the animation. Figure 6 shows an image containing an example of animation created by this interface.

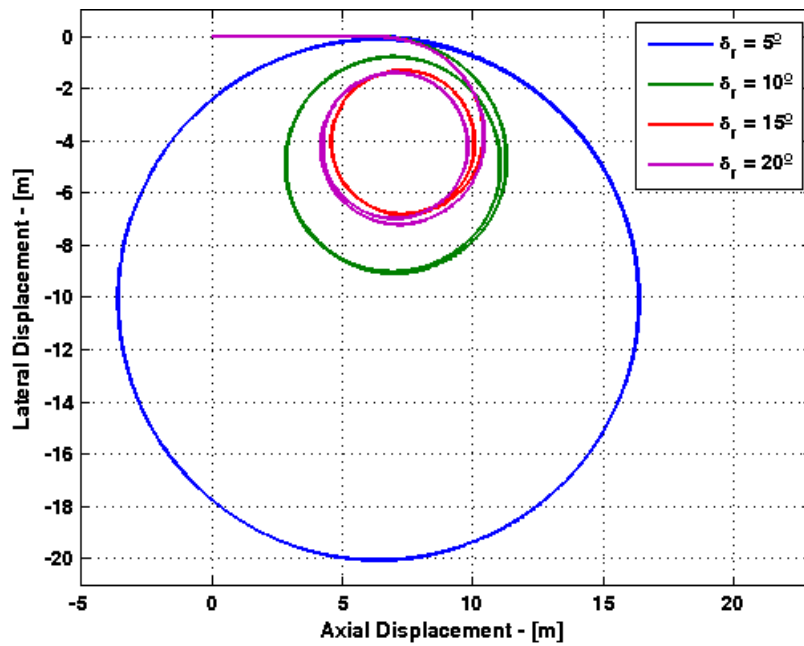


Figure 4. Trajectory performed by the simulated AUV, in a turning manoeuvre, for 4 deflection's angles of the vertical control surface.

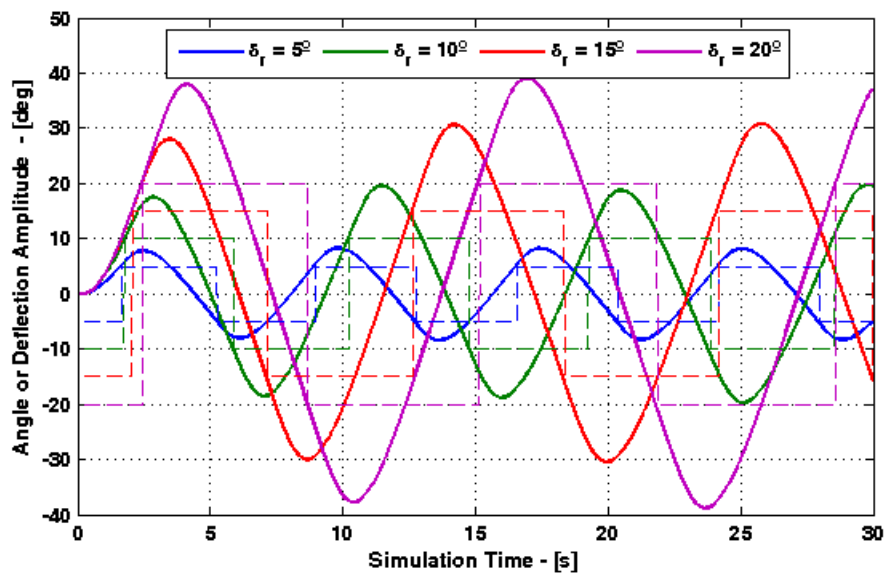
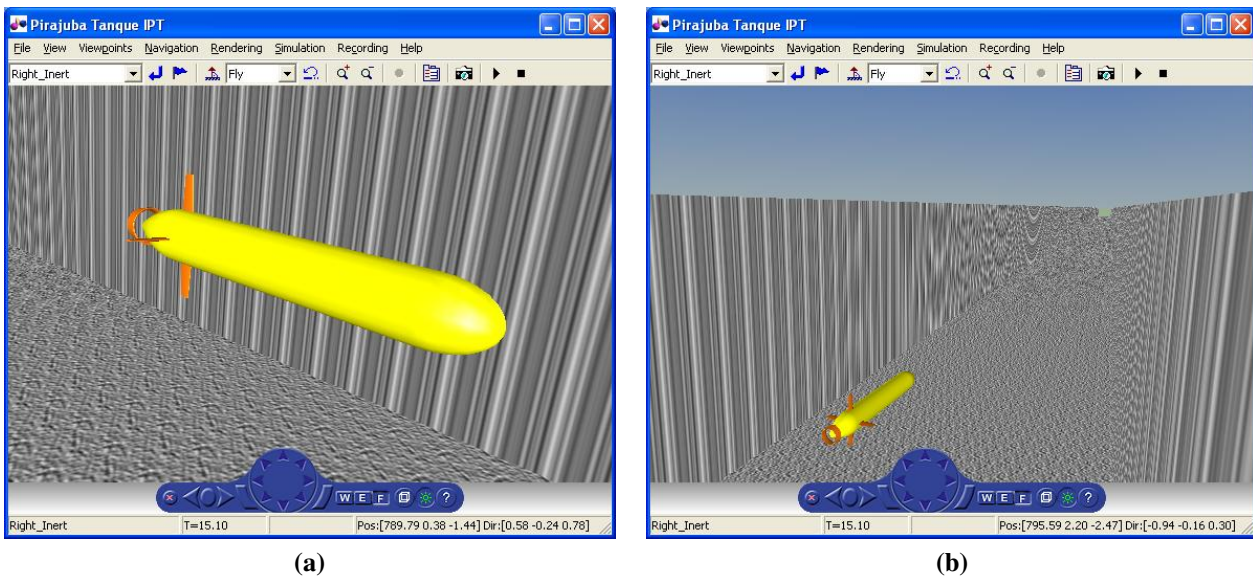


Figure 5. Heading angle (full line) and the control signal (dashed line) of the simulated AUV, in a Zig-Zag manoeuvre, for 4 angles of the rudder.



(a)

(b)

Figure 6. Virtual models (VRML) used in the results interface, being (a) the image of the AUV and (b) the image of the virtual ambient in which this vehicle is simulated.

5.3 Real-Time Verification

There are several methodologies to determine whether or not a software is operating in real-time. A very common and simple method is to check if the execution time of the software "loops", defined as Task Execution Time, or TET, in the xPC Target nomenclature. This should be no greater than the software's operation period, which is called the sampling time.

As the xPC Target has a function that returns the TET automatically as a function of the sampling period. The TET values will be used to calculate the percentage of use of the CPU's processors, which is given by the Eq. (10).

$$CPU(t) = 1 - (T_{TS} - T_{TET}(t)/T_{TS}) \quad (10)$$

Where " T_{TS} " is the sampling period of the simulation software, and " $T_{TET}(t)$ " is the TET obtained in a determined time " t ".

By means of such definition, it was possible to obtain the graph shown in Fig. 7, showing the percentage of processing usage of the computer that runs the simulator software. Thus, it can be observed that the processing is well below 100%, thereby ensuring the operation of the software in real-time, and the non occurrence of processing overflow, or delay in processing due to a need of processing of more than 100%.

6 CONCLUSION

This work presented a methodology for developing simulation software of the AUVs dynamics, with requirements for operation in HIL systems, as the realization of operations in real time.

From the tests observed in the Section 5 was possible to verify that the simulated dynamics of the AUV could generate results, that demonstrate the features of this software, although practical tests have not yet been conducted to validate the AUV predicted behavior. To improve reliability and generality of this simulator, in the future, new dynamic models will be implemented, as the wave induced motions, and improved models for sensors and actuators dynamics.

Using xPC Target, it was possible to verify that the real-time performance has been achieved. Future tests will be aimed at the verification of the real time performance for the whole HIL system.

7 ACKNOWLEDGEMENTS

Authors would like to acknowledge the Fapesp for the scholarship support of the first author (Proc. n° 2006/5504-0), and for the financial support (Proc. n° 550934/2005-7).

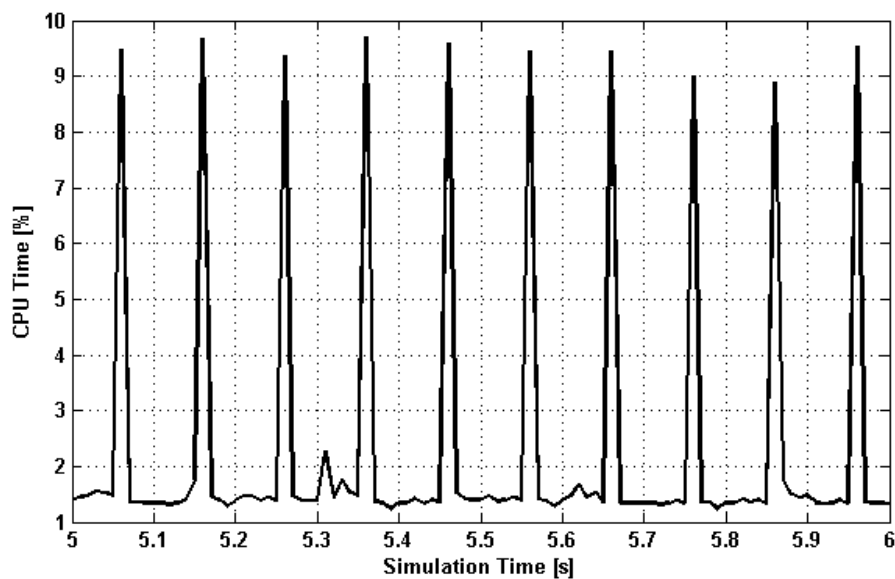


Figure 7. Simulator computer percentage of use (process time).

8 REFERENCES

- Amianti, G., 2008, "Arquitetura de Software Aviônico de um VANT com Requisitos de Homologação", Master's thesis, Escola Politécnica da Universidade de São Paulo.
- Dantas, J. L. D., 2008, "Estudo sobre a Identificação Dinâmica de Veículos Autônomos Submarinos", Scientific report, Escola Politécnica da Universidade de São Paulo.
- de Barros, E. A., Dantas, J. L. D., Pascoal, A. M. and de Sá, E., 2009, "Investigation of Normal Force and Moment Coefficients for an AUV at Nonlinear Angle of Attack and Sideslip Range", Vol. 33, pp. 538–549.
- de Barros, E. A., Pascoal, A. M. and de Sá, E., 2004, "AUV dynamics: modeling and parameter estimation using analytical, semi-empirical, and CFD methods".
- de Barros, E. A., Pascoal, A. M. and de Sá, E., 2006, "Progress Towards a Method For Predicting AUV Maneuvers".
- Devie, F. and Lemaire, J., 1998, "A Flexible Hardware in the Loop Simulator for a Long Range Autonomous Underwater Vehicle", Vol. 3, pp. 1359–1363.
- Göktogan, A. H., Nettleton, B., Ridley, M. and Sukkarieh, S., 2003, "Real Time Multi-UAV Simulator", pp. 2720–2726.
- Hoak, D. B. and Finck, R. D., 1978, "USAF Stability and Control Datcom", Wright-Paterson Air Force Base.
- Imado, F., Abe, S. and Kinoshita, T., 2006, "The Development of Three-Dimensional Scale Model UAV Simulator", pp. 4921–4925.
- Ishidera, H., Tsusaka, Y., Ito, Y., Oishi, T., Chiba, S. and Maki, T., 1986, "Simulation and Experiment of Automatic Controlled ROV".
- Jorgensen, L. H., 1977, "Prediction of static aerodynamic characteristics for slender bodies alone and with lifting surfaces to very high angles of attack".
- Lane, D. M., Falconer, G. J. and Randall, G., 2001, "Interoperability and Synchronisation of Distributed Hardware-in-the-Loop Simulation for Underwater Robot Development: Issues and Experiments", Vol. 1, pp. 909–914.
- Lewis, E. V., 1967, "Principles of Naval Architecture", SNAME - The Society of Naval Architects and Marine Engineers.
- MathWorks, 2008a, "MATLAB Getting Started Guide", MathWorks Inc.
- MathWorks, 2008b, "Real-Time Workshop User's Guide", MathWorks Inc.

MathWorks, 2008c, “Using Simulink”, MathWorks Inc.

MathWorks, 2008d, “xPC Target User’s Guide”, MathWorks Inc.

Myring, D. F., 1976, “A Theoretical Study of Body Drag in Subcritical Axisymmetric Flow”, Vol. 27, pp. 186–194.

Nelson, R. C., 1989, “Flight Stability and Automatic Control”, Aerospace and Mechanical Engineering Department, University of Notre Dame.

Newman, J. N., 1977, “Marine Hydrodynamics”, The MIT Press.

Pitts, W. C., Nielsen, J. N. and Kaattari, G. E., 1957, “Lift and Center of Pressure of Wing-Body-Tail Combinations at Subsonic, Trans-Sonic, and Supersonic Speeds”.

Prestero, T., 2001, “Verification of a Six-Degree of Freedom Simulation Model for the Remus Autonomous Underwater Vehicle”, PhD thesis, MIT- Massachusetts Institute of Technology.

Roskam, J., 1985, “Airplane Design - Part VI Preliminary Calculation of Aerodynamic, Thrust and Power Characteristic”, Roskam Aviation and Engineering Corporation.

Shixianjun, Jiakun, S. and Hongxing, L., 2006, “Hardware-in-the-Loop Simulation Framework Design For a UAV Embedded Control System”, pp. 1890–1894.

Song, F., An, P. E. and Folleco, A., 2003, “Modeling and Simulation of Autonomous Underwater Vehicles: Design and Implementation”, Vol. 28, pp. 283–296.

9 Responsibility notice

The author(s) is (are) the only responsible for the printed material included in this paper