Modeling and Simulation of the LAUV Autonomous Underwater Vehicle

Jorge Estrela da Silva, Bruno Terra, Ricardo Martins and João Borges de Sousa

***Abstract*— We derive the numerical coefficients for the nonlinear mathematical model of a new AUV designed at Oporto University. This is made using theoretical and empirical methods as also by adapting the known results from similar AUVs. We use the derived model on MVS, a simulator which can be embedded in the loop of the control software, by replacing the interface with the sensors and actuators.**

***Index Terms*— Underwater Vehicles, Modeling, Simulation**

I. Introduction

On most methodologies, the design and tuning of controllers requires a mathematical model of the system. We present a mathematical model for the new generation of autonomous underwater vehicles (AUV) designed and built at the Underwater Systems and Technology Laboratory (USTL) from Oporto University. These AUVs have a torpedo shape and they are optimized for small size and low cost mechanical structure. The first vehicle of this generation is called LAUV. The mathematical model will be employed on early testing of vehicle control software.

The computation of the coefficients of the nonlinear model is done by resorting to theoretical and empirical formulas and also by establishing analogies with known models from similar vehicles, namely the Isurus AUV, a REMUS class vehicle created at the Woods Hole Oceanographic Institution and customized at the USTL. Figure I shows both Isurus and LAUV. The derived model will allow the tuning of controllers which, on their turn, will enable the execution of some in-water tests – such as operation at constant depth, far from the influence of the surface – that will provide data for the refinement of the model.

We describe how the derived model is incorporated on MVS, a multiple vehicle simulation system being developed at USTL using the Open Dynamics Engine (ODE) library. The ODE is an open source library for simulating rigid body dynamics. It was designed to be used in interactive or real-time simulation. It is particularly suited for simulating several moving objects in changeable virtual reality environments. It has already been used in many applications (see [1], for instance) including games. However, we are not aware of any published work concerning submarine simulation.

Finally we describe the incorporation of the MVS on the vehicle's on-board software. Basically, the MVS must replace the interface with the physical sensors and actuators.

J. Silva is with the Department of Electrical Engineering, Instituto Superior de Engenharia do Porto, R. Dr. António Bernardino Almeida 431, 4200-072 Porto, Portugal. Phone: +351 22 8340500. jes@isep.ipp.pt

B. Terra, R. Martins and J. Sousa are with the Department of Electrical Engineering, Faculdade de Engenharia da Univesidade do Porto, Rua Dr. Roberto Frias, 4200-465 Porto, Portugal. {bterra,rasm,jtasso}@fe.up.pt

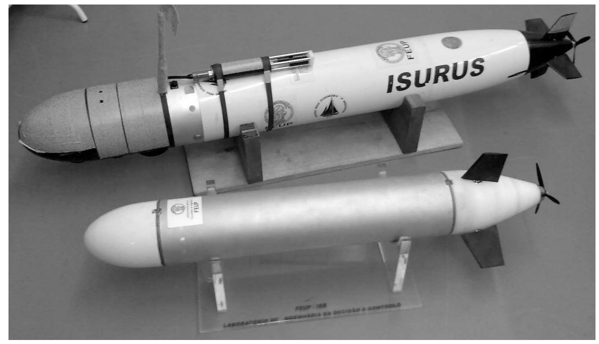

Fig. 1. Isurus (top) and LAUV (bottom) side by side, at USTL.

The objective of this approach is to allow a single implementation of the control software to function unmodified in both real-life and simulated environments. The typical design cycle involves the test of different control laws or navigation schemes. In most cases the control system must be replicated in a simulation environment, usually on a different language. Even when that is done correctly, it is difficult to keep consistency between that implementation and the final control system, which may be subject to updates from other sources. Instead of writing separate code for a prototyping environment and then for the final version, our approach allows the employment of the stable/final software in the overall design cycle.

The paper is organized as follows. In section II we review the nonlinear model structure usually employed for AUVs, with some remarks for the particular configuration of the torpedo shaped vehicles. In section III we derive the actual coefficient values for the LAUV model. We discuss steady state operation and the sensitivity of the model to certain parameters. In section IV we describe the implementation of the simulator. Finally, on section V we present the conclusions.

II. Vehicle's model

Autonomous underwater vehicles (AUV's) are best described as nonlinear systems (see [2] for details). In order to define the model, two coordinate frames are considered: body-fixed and earth-fixed. In what follows, the notation from the Society of Naval Architects and Marine Engineers (SNAME) [3] is used. The motions in the body-fixed frame are described by 6 velocity components $\nu = [\nu_1^T, \nu_2^T]^T = [u, v, w, p, q, r]^T$ respectively, surge, sway, heave, roll, pitch, and yaw, relative to a constant velocity coordinate frame moving with the ocean current. The six components of position and attitude in the earth-fixed frame are $\eta =$

$[\eta_1^T, \eta_2^T]^T = [x,y,z,\phi,\theta,\psi]^T$. The earth-fixed reference frame can be considered inertial for the AUV.

The velocities in both reference frames are related through the Euler angle transformation

$$\dot{\eta} = J(\eta_2)\nu \qquad (1)$$

with

$$J(\eta_2) = \begin{bmatrix} J_1(\eta_2) & \mathbf{0} \\ \mathbf{0} & J_2(\eta_2) \end{bmatrix}$$

$$J_1(\eta_2) = \begin{bmatrix} \mathrm{c}\psi\mathrm{c}\theta & (\mathrm{c}\psi\mathrm{s}\theta\mathrm{s}\phi - \mathrm{s}\psi\mathrm{c}\phi) & (\mathrm{s}\psi\mathrm{s}\phi + \mathrm{c}\psi\mathrm{c}\phi\mathrm{s}\theta) \\ \mathrm{s}\psi\mathrm{c}\theta & (\mathrm{c}\psi\mathrm{c}\phi + \mathrm{s}\phi\mathrm{s}\theta\mathrm{s}\psi) & (\mathrm{s}\theta\mathrm{s}\psi\mathrm{c}\phi - \mathrm{c}\psi\mathrm{s}\phi) \\ -\mathrm{s}\theta & \mathrm{c}\theta\mathrm{s}\phi & \mathrm{c}\theta\mathrm{c}\phi \end{bmatrix}$$

$$J_2(\eta_2) = \begin{bmatrix} 1 & \mathrm{s}\phi\tan\theta & \mathrm{c}\phi\tan\theta \\ 0 & \mathrm{c}\phi & -\mathrm{s}\phi \\ 0 & \frac{\mathrm{s}\phi}{\mathrm{c}\theta} & \frac{\mathrm{c}\phi}{\mathrm{c}\theta} \end{bmatrix}$$

The equations of motion are composed of the standard terms for the motion of an ideal rigid body and, additionally, the terms due to hydrodynamic forces and moments. The usual approach to model the hydrodynamic terms is to consider three main effects: restoring forces, the simplest one, which depends only on the vehicle weight, buoyancy and relative positions of the centers of gravity and buoyancy; added mass, which describes pressure induced forces/moments due to forced harmonic motion of the body; and damping, caused by skin friction (laminar and turbulent) and vortex shedding. Usually the elements of the damping matrix are defined so that linear and quadratic components arise (e.g., $X_u + X_{u|u|}|u|$ for D_{11}).

The hydrodynamic damping and added mass are very hard to describe accurately. They can be estimated by usually expensive hydrodynamic tests but a frequent alternative is the employment of heuristical formulas, an approach which trades-off accuracy by simplicity.

In the body-fixed frame the nonlinear equations of motion are:

$$M\dot{\nu} + C(\nu)\nu + D(\nu)\nu + L(\nu)\nu + g(\eta) = \tau \qquad (2)$$

where M is the constant inertia and added mass matrix of the vehicle, $C(\nu)$ is the Coriolis and centripetal matrix, $D(\nu)$ is the damping matrix, $L(\nu)$ is the lift matrix (some authors include these terms on the damping matrix), $g(\eta_2)$ is the vector of restoring forces and moments and τ is the vector of body-fixed forces from the actuators. If the vehicle's weight equals its buoyancy and the center of gravity is coincident with the center of buoyancy, $g(\eta_2)$ is null. Additionally, for an AUV with port/starboard, top/bottom and fore/aft symmetries, M and $D(\nu)$ are diagonal.

The considered AUVs are not fully actuated. There is a propeller for actuation in the longitudinal direction and fins for lateral and vertical actuation. This mechanical configuration leads to a simpler dynamic model. τ depends only on 3 parameters: propeller velocity n ($0 < n \leq n_{max}$), horizontal fin inclination δ_s ($-\delta_{smax} \leq \delta_s \leq \delta_{smax}$) and vertical fin inclination δ_r ($-\delta_{rmax} \leq \delta_r \leq \delta_{rmax}$). The dynamics of the thruster motor and fin servos are generally much faster than the remaining dynamics therefore, for the purposes of this work, they can be excluded from the model. We also consider that the vehicle is port/starboard and bottom/top symmetric in shape. For safety reasons, the vehicles usually are slightly buoyant. The center of gravity is slightly below the center of buoyancy, providing a restoring moment in pitch and roll which is useful for these underactuated vehicles.

III. LAUV MODEL

The Light Autonomous Underwater Vehicle (LAUV) is a low-cost submarine for oceanographic and environmental surveys designed and built at USTL. It is a torpedo shaped vehicle, with a length of 108cm, a diameter d of 15cm and a mass of approximately 18kg. The actuator system is composed of one propeller and 3 or 4 control fins (depending on the vehicle version), all electrically driven. It has a miniaturized computer system running the control system software. It uses an IMU unit, a depth sensor and LBL system for navigation. The maximum expected velocity is 2m/s.

Since our modeling methodology will be based on the results gathered for another AUV, Isurus, we will make a brief review of those results. The Isurus AUV is a REMUS class vehicle customized at the USTL. It is a torpedo-shaped AUV weighting approximately 50kg vehicles and with a length of 1.4 meters. For the Isurus AUV, the values of the coefficients were derived using results from the literature and from our field experiments. The added mass terms were computed using heuristic formulas for an ellipsoidal body, as described in [2]. This is an acceptable approximation for this kind of vehicle's shape. The values did not differ significantly from those derived with strip theory in [4], where a similar AUV is analyzed. For the quadratic cross-flow drag coefficients we used the values derived in [4]. For the linear drag coefficients we used the results of our field experiments, namely the circle test using the procedure described in [2]. Notice that due to the symmetries of the vehicle, some of the coefficients affecting the motion on the vertical plane are the same as those affecting the motion on the horizontal plane.

For the inertia and added mass matrix of the LAUV, a ellipsoidal form is assumed, the same way as was done for Isurus. Like Isurus, in normal operation, this AUV does not have any form of direct actuation over the roll dynamics. Therefore, roll stabilization is performed in a passive fashion, by lowering the center of gravity relatively to the center of buoyancy, in order to create a restoring moment. The origin of the body fixed referential is the center of buoyancy and $z_G = 0.01m$ is the distance from the origin to the center of gravity. For simplicity, we assume that the mass is distributed in such a way that the inertia tensor of the vehicle can be approximated by that of an prolate ellipsoid. Therefore:

$$M = \begin{bmatrix} m-X_{\dot{u}} & 0 & 0 & 0 & mz_G & 0 \\ 0 & m-Y_{\dot{v}} & 0 & -mz_G & 0 & 0 \\ 0 & 0 & m-Z_{\dot{w}} & 0 & 0 & 0 \\ 0 & -mz_G & 0 & I_x-K_{\dot{p}} & 0 & 0 \\ mz_G & 0 & 0 & 0 & I_y-M_{\dot{q}} & 0 \\ 0 & 0 & 0 & 0 & 0 & I_z-N_{\dot{r}} \end{bmatrix}$$

$$M = \begin{bmatrix} 19 & 0 & 0 & 0 & 0.18 & 0 \\ 0 & 34 & 0 & -0.18 & 0 & 0 \\ 0 & 0 & 34 & 0 & 0 & 0 \\ 0 & -0.18 & 0 & 0.04 & 0 & 0 \\ 0.18 & 0 & 0 & 0 & 2.1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2.1 \end{bmatrix}$$

The inertia moment of a prolate ellipsoid with uniformly distributed mass m along the body fixed y and z axis, is given by $m(L^2/20+r^2/5)$, where L is the length of the ellipsoid and r is its largest sectional radius. Just for comparison, the inertia moment of a cylinder enclosing the described ellipsoid is given by $m(L^2/12+r^2/4)$. For the actual vehicle dimensions, the values for the ellipsoid are 60% of those obtained for the cylinder. For the Isurus vehicle, the true value is somewhere between the two cases but closer to that of the ellipsoid, therefore we rounded up the obtained value. The added mass terms were calculated using the ellipsoid formulas in[2]. We neglect the added mass terms that would arise due to the asymmetry between the nose and tail ($Y_{\dot{r}} = -Z_{\dot{q}}$,$N_{\dot{v}} = -M_{\dot{w}}$)). Our experience with the simulation model of the Isurus shows that the impact is minimal.

In what concerns the restoring terms, the vehicle will be slightly buoyant, with $W-B = -1$N, where $W = 176$N is the vehicle's weight and B is the vehicle's buoyancy force. Therefore:

$$g(\eta_2) = \begin{bmatrix} (W-B)\sin\theta \\ -(W-B)\cos\theta\sin\phi \\ -(W-B)\cos\theta\cos\phi \\ z_G W\cos\theta\sin\phi \\ z_G W\sin\theta \\ 0 \end{bmatrix}$$

The damping matrix has the following expression:

$$D(v) = -\begin{bmatrix} X_u & 0 & 0 & 0 & 0 & 0 \\ 0 & Y_v & 0 & 0 & 0 & Y_r \\ 0 & 0 & Z_w & 0 & Z_q & 0 \\ 0 & 0 & 0 & K_p & 0 & 0 \\ 0 & 0 & M_w & 0 & M_q & 0 \\ 0 & N_v & 0 & 0 & 0 & N_r \end{bmatrix} -$$

$$\begin{bmatrix} X_{u|u|}|u| & 0 & 0 & 0 & 0 & 0 \\ 0 & Y_{v|v|}|v| & 0 & 0 & 0 & Y_{r|r|}|r| \\ 0 & 0 & Z_{w|w|}|w| & 0 & Z_{q|q|}|q| & 0 \\ 0 & 0 & 0 & K_{p|p|}|p| & 0 & 0 \\ 0 & 0 & M_{w|w|}|w| & 0 & M_{q|q|}|q| & 0 \\ 0 & N_{v|v|}|v| & 0 & 0 & 0 & N_{r|r|}|r| \end{bmatrix}$$

In this case, the considered symmetries are top/bottom and port/starboard. The asymmetry between the tail, which contain the fins, and the nose make the damping matrix non-diagonal. Even so the vehicle's symmetries allow us the following simplifications: $Y_{v|v|} = Z_{w|w|}$, $N_{v|v|} = -M_{w|w|}$, $Y_{r|r|} = -Z_{q|q|}$, $N_{r|r|} = M_{q|q|}$. The same relations apply to the linear damping terms:$Y_v = Z_w$, $N_v = -M_w$, $Y_r = -Z_q$, $N_r = M_q$. Concerning the actual coefficient values, we will use the normalized hydrodynamic derivatives from the Isurus model, due to the strong similarity between the form of the two vehicles. For the coefficients related to forces due to linear velocities or to moments due to angular velocities we will have relations such as $X_{u|u|} = \frac{\rho}{2}U_0L^2X'_{u|u|}$, where $X'_{u|u|}$ is the normalized coefficient; for forces due to angular velocities or moments due to linear velocities the relations will be like $Y_{r|r|} = \frac{\rho}{2}U_0L^3Y'_{r|r|}$. The typical velocity for this vehicle will be $U_0 \simeq 1.5$m/s. Thus, the damping matrix has the following numerical expression:

$$D(v) = \begin{bmatrix} 2.4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 23 & 0 & 0 & 0 & -11.5 \\ 0 & 0 & 23 & 0 & 11.5 & 0 \\ 0 & 0 & 0 & 0.3 & 0 & 0 \\ 0 & 0 & -3.1 & 0 & 9.7 & 0 \\ 0 & 3.1 & 0 & 0 & 0 & 9.7 \end{bmatrix} +$$

$$\begin{bmatrix} 2.4|u| & 0 & 0 & 0 & 0 & 0 \\ 0 & 80|v| & 0 & 0 & 0 & -0.3|r| \\ 0 & 0 & 80|w| & 0 & 0.3|q| & 0 \\ 0 & 0 & 0 & 6\times10^{-4}|p| & 0 & 0 \\ 0 & 0 & -1.5|w| & 0 & 9.1|q| & 0 \\ 0 & 1.5|v| & 0 & 0 & 0 & 9.1|r| \end{bmatrix}$$

Notice that, for low velocities, the quadratic terms, e.g. $Y_{v|v|}|v|$, may be considered negligible.

We consider lift forces and moments due to the fin surfaces and also due to the body surface. For a in-depth description of this terms see, for instance, [5].

The values for the body lift force and moment coefficients ($Y_{uv_b} = Z_{uw_b}$ and $N_{uv_b} = -M_{uw_b}$) were obtained using the following formulas, where $C_{LB} = 1.24$ is an empirical coefficient which depends on body length and diameter:

$$Z_b = -\frac{1}{2}\rho\pi\left(\frac{d}{2}\right)^2 C_{LB}uw \quad (3)$$

$$M_b = -(-0.65L - x_B)Z_b \quad (4)$$

The term $0.65L$ is an empirical formula for the center of pressure, the point where the body lift forces are applied[5]; $x_B = -0.4$m is the position of the center of buoyancy relatively to the nose of the vehicle.

The LAUV will have two versions: one with a four fin tail (two vertical and two horizontal), the one considered here, and another version with a three fin tail (one vertical and the other two at ±120 degree from the vertical one). The empirical formulas for the pitch fins' lift force and moment are presented below ($S_{fin} = 64\text{cm}^2$ is the fin's face area, $x_{fin} = -40$cm is the position of the fin relatively to the center of buoyancy and $C_{LF} = 3$ depends essentially on the geometrical aspect of the fin):

$$Z_f = -\rho C_{LF} S_{fin}(u^2\delta_s + uw - x_{fin}uq) \quad (5)$$

$$M_f = -x_{fin}Z_f \quad (6)$$

The formulas for the rudder fins' force and moment are analogous:

$$Y_f = \rho C_{LF} S_{fin}(u^2\delta_r - uv - x_{fin}ur) \quad (7)$$

$$N_f = x_{fin}Y_f \quad (8)$$

Therefore the lift matrix comes as follows:

$$L(\nu) = -\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -30 & 0 & 0 & 0 & 7.7 \\ 0 & 0 & -30 & 0 & -7.7 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -9.9 & 0 & -3.1 & 0 \\ 0 & 9.9 & 0 & 0 & 0 & -3.1 \end{bmatrix} u$$

Taking in account all above mentioned assumptions, we define the matrix $C(\nu)$ with the Coriolis and centripetal terms (including the effect of the added mass):

$$C(\nu) = \begin{bmatrix} \mathbf{0} & C_{12}(\nu) \\ C_{21}(\nu) & C_{22}(\nu) \end{bmatrix}$$

with

$$C_{12}(\nu) = \begin{bmatrix} mz_G r & (m - Z_{\dot{w}})w & -(m - Y_{\dot{v}})v \\ -(m - Z_{\dot{w}})w & mz_G r & (m - X_{\dot{u}})u \\ -mZ_G p + (m - Y_{\dot{v}})v & c_{12_{32}} & 0 \end{bmatrix}$$

$$c_{12_{32}} = -mZ_G q - (m - X_{\dot{u}})u$$

$$C_{21}(\nu) = \begin{bmatrix} -mz_G r & (m - Z_{\dot{w}})w & mZ_G p - (m - Y_{\dot{v}})v \\ -(m - Z_{\dot{w}})w & -mz_G r & mZ_G q + (m - X_{\dot{u}})u \\ (m - Y_{\dot{v}})v & -(m - X_{\dot{u}})u & 0 \end{bmatrix}$$

$$C_{22}(\nu) = \begin{bmatrix} 0 & (I_z - N_{\dot{r}})r & -(I_y - M_{\dot{q}})q \\ -(I_z - N_{\dot{r}})r & 0 & (I_x - K_{\dot{p}})p \\ (I_y - M_{\dot{q}})q & -(I_x - K_{\dot{p}})p & 0 \end{bmatrix}$$

The actuator system is modelled in the following way: we assume that the propeller creates a constant thrust force X_{prop}, in order to keep the desired steady state surge. The induced roll moment, due to the thrust force, is given by $-0.06X_{prop}$. The force and moments created by the fins are calculated using Equations 7 and 8. The respective coefficients's values are $Y_{uu\delta_r} = -Z_{uu\delta_s} = 19.2$ and $N_{uu\delta_r} = M_{uu\delta_s} = -7.7$.

A. Linear damping

Some of the models found in the literature, e.g. [6], [7], [8], [9], do not consider the linear damping terms X_u, Z_w, etc. This terms are mainly due to laminar skin friction [2] and may play an important role in the design of the control system, namely in the local stability analysis. For low velocities scenarios, such as when regulating to constant depth, the quadratic damping terms become very small. If the linear damping is ignored, the linearization of the system model around the equilibrium point may falsely reveal a locally unstable system. This leads the control system designer to counteract, generally by adding a derivative action, i.e., linear damping in the form of velocity feedback. In practice, this will lead to a conservative design, since the overlooked damping terms contribute to system stability. In fact, it is possible to find examples in the literature where the authors perform a worst case analysis, by totally disregarding the damping matrix [10], [11]. While our field experience reveals that it is possible to perform depth regulation of the Isurus vehicle using a cascade of two proportional-integrative controllers, the analysis of the linearized models with null linear damping pointed out the mandatory use of derivative action (in the present case, the feedback of the state variable q).

When designing low cost vehicles, it is of interest to use the smallest and cheapest possible set of sensors. Thus, assuming no direct velocity measurement is made, even a velocity estimate may become problematic if some of the sensors present appreciable measurement errors or noise. This illustrates the importance of a correct estimation of the linear damping terms.

The analysis was made by applying the Routh-Hurwitz method to the characteristic polynomial of the linearized system and taking in account the Lyapunov's linearization method (LLM). The LLM is based on the following theorem (see, e.g., [12]):

- If the linearized system is strictly stable, then the equilibrium point is asymptotic stable (for the actual nonlinear system).
- If the linearized system is unstable, then the equilibrium point is unstable (for the nonlinear system).
- if the linearized system is marginally stable, then one cannot conclude anything from the linear approximation.

We considered the following linearized model of the AUV's pitch motion, already including a state feedback scheme similar to a proportional-integrative-derivative controller, where $k_{\theta p}$ is the proportional action's gain, $k_{\theta i}$ is the integrative action's gain and $k_{\theta q}$ is the derivative action's gain:

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ a_{21} - b_2k_{\theta p} & a_{22} & a_{23} - b_2k_{\theta q} & b_2k_{\theta i} \\ a_{31} - b_3k_{\theta p} & a_{32} & a_{33} - b_3k_{\theta q} & b_3k_{\theta i} \\ -1 & 0 & 0 & 0 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 & b_2k_{\theta p} & b_3k_{\theta p} & 1 \end{bmatrix}^{\mathrm{T}} \theta_{ref} \quad (9)$$

with $\dot{\mathbf{x}} = \begin{bmatrix} \theta_l & w_l & q & e_{\theta i} \end{bmatrix}$

Through the analysis of the characteristic polynomial associated to Equation 9, using the respective numerical coefficients, we conclude that it is not possible to stabilize the system either with proportional control or proportional-integrative control when no linear damping is considered.

B. Model analysis

The main focus of our analysis will be on depth operation thus, in what follows we consider operation on a vertical plane, with negligible roll. Therefore, we will have $y = \phi = \psi = v = p = r = 0$. Thus, based on [2], the dynamic model of the AUV becomes:

$$
\begin{aligned}
(m - X_{\dot{u}})\dot{u} + mz_g\dot{q} =& -(W-B)\sin\theta + X_u u \\
&+ X_{u|u|}u|u| + (X_{wq}-m)wq \\
&+ X_{qq}q^2 + X_{\text{prop}} && (10)\\
(m - Z_{\dot{w}})\dot{w} - Z_{\dot{q}}\dot{q} =& (W-B)\cos\theta + Z_{uw}uw \\
&+ (Z_{uq}+m)uq + Z_w w \\
&+ Z_{w|w|}w|w| + Z_q q + Z_{q|q|}q|q| \\
&+ mz_g q^2 + Z_{uu\delta_s}u^2\delta_s && (11)\\
mz_g\dot{u} - M_{\dot{w}}\dot{w} + (I_{yy} - M_{\dot{q}})\dot{q} =& -z_g W\sin\theta + M_{uw}uw + M_{uq}uq \\
&+ M_w w + M_{w|w|}w|w| - mz_g wq \\
&+ Mqq + M_{q|q|}q|q| \\
&+ M_{uu\delta_s}u^2\delta_s && (12)
\end{aligned}
$$

The choice of nonzero coefficients reflects the symmetries considered for the AUVs.

In order to obtain the earth-fixed coordinates, the following kinematic relation is employed:

$$
\begin{aligned}
\dot{z} =& -\sin\theta u + \cos\theta w && (13)\\
\dot{\theta} =& q && (14)
\end{aligned}
$$

where $z(t)$ is the depth of the vehicle (positive downwards) and $\theta(t)$ is the vehicle's pitch angle (positive "upwards").

In what follows, we calculate the steady state values of u, w and θ for different values of δ_s. At any equilibrium point we have the following equations:

$$
\begin{aligned}
0 =& -(W-B)\sin\theta + X_u u + X_{u|u|}u|u| + X_{\text{prop}} \\
0 =& (W-B)\cos\theta + Z_w w + Z_{w|w|}w|w| \\
&+ Z_{uw}uw + Z_{uu\delta_s}u^2\delta_s \\
0 =& -z_g W\sin\theta + M_w w + M_{w|w|}w|w| \\
&+ M_{uw}uw + M_{uu\delta_s}u^2\delta_s
\end{aligned}
$$

Remember that by the definition of equilibrium point ($\dot{\mathbf{x}} = \mathbf{0}$) and since $\dot{\theta} = q$, one can conclude that the value of q at any equilibrium point is zero.

We solve the system of three equations using numerical methods. We remark that the steady state value of δ_s is part of the solution of the system of equations.

The solution of these equations is useful for an accurate model linearization but they can also be employed to perform useful steady state analysis. For instance, at this stage we are not sure of the exact final location of the center of gravity, since different hardware arrangements will be tested. In section III we considered $z_G = 1$cm. Table I shows the steady state values of the state variables, as also the linearized system's poles, for different values of δ_s considering the assumed value. If we assume that the center of gravity is lowered to $z_G = 3$cm, the behavior of the vehicle is slightly different, as shown on Table II. As expected, since the opposing restoring moment is higher, higher actuation values are needed in order to achieve a certain pitch angle. However, the numerical results of both tables give us a good estimative of the quantitative impact of the variation of the center of gravity.

δ_s	u	w	$\theta()$	$\dot{z}$	poles
-0.070	1.56	0.043	85.2	-1.55	-0.00, -2.08, -7.09
-0.030	1.55	0.006	20.8	-0.55	-0.12, -1.89, -6.97
-0.020	1.55	-0.001	11.8	-0.32	-0.13, -1.86, -6.96
-0.010	1.55	-0.008	3.3	-0.10	-0.13, -1.89, -6.96
-0.0056	1.55	-0.010	-0.4	0.00	-0.13, -1.90, -6.96
0.000	1.55	-0.014	-5.1	0.12	-0.13, -1.91, -6.96
0.010	1.55	-0.020	-13.3	0.34	-0.13, -1.94, -6.96
0.020	1.55	-0.026	-21.8	0.55	-0.13, -1.96, -6.97
0.069	1.55	-0.044	-85.5	1.54	-0.02, -2.05, -7.07

TABLE I

EQUILIBRIUM POINT FOR DIFFERENT VALUES OF δ_s, WITH $z_G = 1$cm

δ_s	u	w	$\theta()$	$\dot{z}$	poles
-0.210	1.56	0.122	86.1	-1.54	-0.02, -2.43, -7.12
-0.100	1.55	0.051	27.2	-0.67	-0.36, -2.05, -6.79
-0.029	1.55	0.005	6.4	-0.17	-0.42, -1.83, -6.72
-0.020	1.55	-0.001	3.9	-0.11	-0.43, -1.81, -6.72
-0.005	1.55	-0.011	-0.4	0.00	-0.43, -1.85, -6.72
0.000	1.55	-0.014	-1.7	0.03	-0.43, -1.86, -6.72
0.020	1.55	-0.027	-7.2	0.17	-0.42, -1.92, -6.72
0.100	1.55	-0.072	-30.0	0.71	-0.37, -2.13, -6.76
0.210	1.55	-0.123	-81.6	1.51	-0.07, -2.39, -7.02

TABLE II

EQUILIBRIUM POINT FOR DIFFERENT VALUES OF δ_s, WITH $z_G = 3$cm

IV. SIMULATION

On this section we describe the implementation of the LAUV model on the simulator (MVS) and the incorporation of the MVS on the LAUV's on-board software. It must be remarked that, thanks to the software architecture, the software can be executed on different kinds of computers and operating systems, as described later.

As stated in the introduction, the MVS is based on the ODE. The ODE will solve the following equation.

$$
M_{RB}\dot{\nu} = -M_a\dot{\nu} - C(\nu)\nu - D(\nu)\nu - L(\nu)\nu - g(\eta) + \tau
$$

The force (and moments, if convenient) terms of the second member must be input to the library as described below.

The ODE engine allows different types of numerical solvers. However, at this stage, the developers recommend a fixed step solver. We use a first order algorithm (Euler) with a step size of 0.01 seconds to perform the integration of the equations of motion.

For this simulation we choose not to use any collision detection, as it is a one vehicle simulation. This improves the overall simulation performance, reducing the calculations required at each time step.

The algorithm starts with creation of a dynamic world where the global properties (like gravity and correction factors) are defined, and the world building takes place (obstacles and world boundaries creation). This world's structure can be changed while the simulation is running. The vehicle's initial state is defined and then the simulation loops until termination, performing the following steps: a) apply forces to the vehicle (thrust, fins); b) take simulation step; c) read vehicle's position, orientation and velocity;

In what concerns AUV modeling, the hydrodynamic forces and moments must be provided to the library at each step since these effects are not calculated by the ODE engine. Additionally, the actuation must also be applied. The ODE library provides primitives that allow the application of forces at any specified point of the body fixed referential. This way, if the point of application of the force is specified (the default is the center of gravity) the ODE calculates the respective moment. For instance, in the case of the fins, we only provide the produced force and the position of the fin relatively to the center of gravity. The same applies to the restoring forces. However, in other cases, as for the damping matrix, the forces and moments are provided assuming the center of gravity as the origin of the body fixed referential.

In order to mimic overall system operation, the LAUV simulator is embedded in the AUV control software. When the simulator component is enabled, apart from simulating world and body physics, it takes the place of the sensors and actuators. In what follows we describe the implementation of the simulator in the LAUV's on-board software. For that end, we make a brief description of the software architecture.

Running on top of the computational system is the operating system Linux with real-time preemptive scheduling and the LAUV's on-board software DUNE (DUNE: uniform navigational environment), which is also used in ROV and ASV class vehicles [13]. At the core of DUNE sits a platform abstraction layer, written in C++, enhancing portability among different computer architectures and operating systems.

DUNE attains loose coupling between components by partitioning related logical operations into isolated sets (or Tasks in DUNE's nomenclature). Tasks are executed in a concurrent or serialized fashion and may also be grouped into single concurrent or serialized execution entities. Usually several concurrent and serialized execution tasks will coexist within DUNE. Tasks can be started or terminated at any time during DUNE's execution period.

Communication and synchronization between tasks is achieved by the exchange of messages using a lock-free/wait-free message bus. The internal message format is also used for logging purposes and communicating with external software modules over network links. As no restrictions are imposed by DUNE's core on the source and destination of a message, tasks may be scattered among different networked computers working together to achieve the same goal.

For the first in-water tests, proportional-integrative controllers for heading and depth were implemented. These controllers were packaged as simple periodic tasks with the ability of receiving and updating their parameters at run-time.

Finally, the LAUV simulator is implemented by wrapping MVS in a DUNE's non-intrusive periodic task. To keep the real-time synchronization and the simulation's behaviour stable and predictable, we use a time accumulator. Therefore the effect of cumulative time errors is eliminated. Because of DUNE's distributed capabilities the simulator may run on the LAUV's computational system, sharing CPU with other controllers, or offloaded to a separate computer (visible via network) relieving LAUV's computational system. Either way simulation results are identical. The LAUV's computational system consists of a Intel XScale PXA255 processor at 400MHz, mounted on a dedicated SBC (Single Board Computer), and additional modules to interface with the vehicle's sensors and actuators. When running the simulation application, the software uses 11% of the CPU's processing power. This shows that the current model can be simulated faster than real-time even on relatively modest CPU.

V. Conclusions

The simulation of the derived model has shown results that are consistent with the expected behavior of the vehicle. However, due to the uncertainty on some of the coefficients, such as those associated with the drag effects, the model can only be validated with the underwater tests that will be performed in a very short term. On the other hand, most of the remaining coefficients be estimated accurately by the described methods. In a parallel work (not reported here), we already used this model on the design of a basic control scheme which will allow us to perform closed loop depth and heading regulation. The embedding of the simulator in the vehicle's control system provides a very practical way of testing the developed software.

References

[1] O. Michel, "Webots: Professional mobile robot simulation," *International Journal of Advanced Robotic Systems*, vol. 1, no. 1, pp. 39–42, 2004.

[2] T. I. Fossen, *Guidance and Control of Ocean Vehicles*. John Wiley and Sons Ltd., 1994.

[3] E. Lewis, Ed., *Principles of Naval Architecture*. Society of Naval Architects and Marine Engineers, 1989, 2nd revision.

[4] T. J. Prestero, "Verification of a six-degree of freedom simulation model for the remus auv," Master's thesis, Massachusetts Institute of Technology / Woods Hole Oceanographic Institution, Departments of Ocean and Mechanical Engineering, 2001.

[5] S. F. Hoerner and H. V. Borst, *Fluid Dynamic Lift*. author.

[6] T. Prestero, "Development of a six-degree of freedom simulation model for the remus autonomous underwater vehicle," in *OCEANS, 2001. MTS/IEEE Conference and Exhibition*, vol. 1, 5-8 Nov. 2001, pp. 450–455vol.1.

[7] N. E. Leonard and J. G. Graver, "Model-based feedback control of autonomous underwater gliders," *IEEE Journal of Oceanic Engineering, Special Issue on Autonomous Ocean-Sampling Networks*, vol. 26, no. 4, pp. 633–645, October 2001.

[8] G. Conte and A. Serrani, "Modelling and simulation of underwater vehicles," in *Computer-Aided Control System Design, 1996., Proceedings of the 1996 IEEE International Symposium on*, 15-18 Sept. 1996, pp. 62–67.

[9] P. Ridley, J. Fontan, and P. Corke, "Submarine dynamic modelling," in *Proc. Australian Conf. Robotics and Automation*, Brisbane, December 2003.

[10] N. Leonard, "Stabilization of steady motions of an underwater vehicle," in *Proceedings of the 1996 IEEE Conference on Decision and Control*, December 1996, pp. 961–966.

[11] M. Chyba, N. Leonard, and E. Sontag, "Singular trajectories in multi-input time-optimal problems: Application to controlled mechanical systems," *Journal of Dynamical and Control Systems*, vol. 9, no. 1, pp. 73–88, 2003.

[12] J. Slotine and W. Li, *Applied Nonlinear Control*. Prentice-Hall, 1991.

[13] H. Ferreira, R. Martins, E. Marques, J. Pinto, A. Martins, J. Almeida, J. Sousa, and E. Silva, "Swordfish: An autonomous surface vehicle for network centric operations," in *Proceedings of the Oceans 2007 Conference (to appear)*, Aberdeen, Scotland, June 2007.