Available online at www.sciencedirect.com

ScienceDirect

IFAC Papers Online CONFERENCE PAPER ARCHIVE

IFAC-PapersOnLine 48-16 (2015) 043–048

A Pitch-Depth Bottom Following Controller for AUVs using Eigenstructure Assignment ⋆

José Melo * **Aníbal Matos** *,**

* *FEUP - Faculty of Engineering, University of Porto, Porto, Portugal (e-mail: jose.melo@fe.up.pt)*
** *INESC TEC - INESC Technology and Science, Porto, Portugal (e-mail: anibal@fe.up.pt)*

Abstract: This paper addresses the problem of bottom following by the MARES Autonomous Underwater Vehicle, and presents the derivation of a controller able to cope with the peculiarities of such problem. In specific, the main requirement for the controller is the existence of no overshoot both in the depth and pitch outputs of the system. The existence of such time-domain requirements motivates the use of Eigenstructure Assignment techniques in the formulation of the controller. Simulation results obtained with a dynamic model of the MARES AUV are presented and discussed, indicating the validity of the proposed approach.

Keywords: Control of underwater vehicles; Navigation, guidance, and control of autonomous vehicles; Eigenstructure assignment, Multivariable control systems; Marine systems

1. INTRODUCTION

Traditional applications for the use of Autonomous Underwater Vehicles (AUVs) are mostly related with bathymetric tasks, where the objective of mapping the bottom of the river or sea is achieved by using advanced sonar sensors. However, other applications for these vehicles have been envisioned, particularly in open waters environments where the benefits of using them are more dramatic. Nowadays AUVs are already being used for a variety of missions, including the inspection of the sea bottom or underwater structures, or even remote environmental sensing within oceanographic expeditions.

Performing visual inspection of the bottom with an AUV obviously requires the vehicles to navigate closely to the bottom. With poor lighting conditions and turbid water, the bottom of the sea is usually a quite adverse system for acquiring images. Whenever this is necessary, the vehicle needs to navigate as close to the bottom as possible in order to obtain satisfactory results. Such inspection tasks would also greatly benefit if the trajectories of the vehicle closely resemble the profile of the bottom. In this way, bottom features would be depicted according to their natural size and orientation ratio, decreasing distortion and other disturbances that otherwise may affect the final images.

In known environments navigating close to the bottom doesn't represent a challenging navigation problem, as it is easy to plan ahead a given trajectory. In most cases, however, it is not possible to known in advance the profile of the bottom and the problem of having an autonomous vehicle navigating close to the bottom becomes a non-trivial task that could even put in danger the safety of the vehicle.

1.1 Related work

In one of the initial works on the topic, Bottom Following has been described by Bennett et al. (1995) as "maintaining a fixed altitude above an arbitrary surface whose characteristics may or may not be known". Following to that, the problem of bottom following, sometimes also known as seabed tracking, has been addressed in the literature on several occasions, with diverse approaches being proposed.

A study of the bottom following problem for Remotely Operated underwaver Vehicle (ROV) has been detailed in a series of articles by Caccia (see Caccia et al. (1999) and Caccia et al. (2003)). There, high precision bottom following algorithms using high-frequency pencil beam profiling sonars were derived; the approach is based on a multi-hypothesis Extended Kalman Filter for motion and environment estimation techniques, and a Lyapunov-based guidance system. Yoerger et al. (2000) provided a description of a seafloor over rugged volcanic terrain at depths of 2600 meters, where an AUV was used to obtain high-precision detailed data from the seabed. By using an acoustic rangefinder, a bottom following behaviour was developed, but little details were provided about the implementation of it.

More recently different vehicle controllers have been derived to address the problem of bottom following. Gao et al. (2008) proposed the use of the potential field method to derive a controller that addresses both the problems of bottom avoidance and bottom following with a two-level hierarchical motion control approach. The first step includes the motion planning phase, on which the com-

⋆ This work is financed by the ERDF – European Regional Development Fund through the COMPETE Programme (operational programme for competitiveness) and by National Funds through the FCT – Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) within project «FCOMP-01-0124-FEDER-037281».

Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2015.10.256

manded pitch angle is generated based on measurements from altimeter and depth sensor; the subsequent execution control phase, on the other hand, is responsible for the regulation of the stern rudder to track pitch references using for that a linear sliding mode controller. Adhami-Mirhosseini et al. (2011) adopted a different strategy, by converting the bottom-following problem into a trajectory tracking problem. First a Fourier series expansion of the seabed profile is obtained, which is then used with the non-linear output regulation framework to address the seabed tracking problem. Silvestre et al. (2009) proposed a different approach on which the bathymetric characteristics ahead of the vehicle are measured by two echo sounders and taken into account and preview control theory is used to develop a suitable bottom following controller.

1.2 Contribution

In this article we present an extension of previous work on which a Bottom Estimation and Bottom Following guidance-based approach was developed (see Melo and Matos (2012)). There, the focus was on developing a reactive bottom following behaviour based on environment sensing, which was able to provide depth and pitch references to the existing control layer of the MARES AUV. These references were derived by continuous real time estimation of the slope of the seabed using measurements provided by an altimeter. Experimental validation of the aforementioned work demonstrated the success of the approach, but lacked any guarantee in terms of overall stability of the system. Occasionally some oscillations were also observed that could degrade the performance of the approach.

On what follows we will further extend those results, by deriving a dedicated depth-pitch controller using Eigenstructure Assignment (EA) techniques. Such controller should accept pitch and depth references as inputs, and be able to drive the AUV to perform trajectories that closely resemble the profile of the bottom. Furthermore, we are interested that the performance of the controller to be subject to strict time-domain requirements. For obvious reasons, it is desired that no overshoot or oscillations are observed on the pitch and depth transient response of the controller.

This paper is organized as follows. On Section 2 we introduce the general Vehicle Dynamics for AUVs, and derive the linearized diving equations for the MARES AUV. On Section 3 we present the Eigenstructure Assignment algorithm, and in Section 4 we derive a suitable control law using full state feedback, using eigenstructure assignment. Section 5 presents some simulation results and finally, Section 6 provides some conclusion remarks and comments on future work.

2. VEHICLE DYNAMICS

In this section we describe a general dynamical model for AUVs and, in this specific case, for the MARES AUV, which will be the object of the present article. The MARES AUV is a 1.5 meter long modular AUV, with a slender body form and with a weight of around 32kg. The vehicle is equipped with four thrusters, two vertical and two horizontal, which, due to their specific spatial configuration, provide 4 degrees of freedom and allow the vehicle to be to fully controlled in both the horizontal and vertical movements. Even though different configurations are possible, by adding or removing additional payload, the focus will be on the standard configuration of the vehicle as modelled by Ferreira et al. (2010).

Fig. 1. MARES AUV

On what follows, we will use the notation and structure of the vehicle model according to Fossen (1994). Therefore, the state vector $\mu = [u, v, w, p, q, r]^T$ refers to the body-fixed vector of linear and angular velocities while $\eta = [x, y, z, \phi, \theta, \psi]^T$ is the absolute position vector. With that in mind, the general 6 DOF non-linear vectorial equation of motion of an underwater vehicle in a body-fixed reference frame can be written as:

$$M\dot{\nu} + C(\nu)\nu + D(\nu)\nu + g(\eta) = \tau \tag{1}$$

In (1), the matrix M refers to the inertia, $C(\nu)$ is the matrix with the Coriollis and centripetal terms, $D(\mu)$ contains the Hydrodynamical damping terms and $g(\tau)$ refers to the vector of restoring forces and moments.

$$M = M_{RB} + M_A \quad C(\nu) = C_{RB}(\nu) + C_A(\nu) \tag{2}$$

The matrix M commonly encompasses both the rigid body dynamics and added mass terms, the same happening with $C(\nu)$, as indicated by (2). Finally, $g(\eta)$ refers to the restoring forces and moments acting on the vehicle, while τ describes the external forces and moments applied to the vehicle. Following the standard approach, no hydrodynamic interactions between the seabed and the vehicle are considered. Further details on the derivation of this 6 DOF equation of motion can be found in Fossen (1994).

Even though the model derivation for generic AUVs can be somehow intricate, some geometrical properties of MARES, depicted on Figure 1, alleviate that effort. Examples of this are the existence of different planes of symmetry and an appropriate choice of origin of the body vector frame. Another particularity of MARES is the use of 4 thrusters, whose forces and moments produced will compose τ. This thruster configuration provides peculiar ability to move with arbitrary low velocities, and even stop and hover in the water column, a characteristic that is of enormous interest when performing inspections operations. The complete analysis and modelling of the MARES AUV was covered in Ferreira et al. (2010) and, for the sake of brevity, wont be further pursued.

2.1 Linearized Model

The general equations of motion for underwater vehicles, described in this section, are a set of differential non-linear equations. From those (1), a set of linear equations can be obtained by applying a Taylor series approximation around the equilibrium point ($\nu_0(t)$, $\eta_0(t)$), and neglecting the terms with order two or higher. The following notation can then be introduced:

$$\Delta\nu(t) = \nu(t) - \nu_0(t) \quad \Delta\eta(t) = \eta(t) - \eta_0(t) \tag{3}$$

The linearized model obtained by applying (3) to (1) assumes that the vehicle is moving on the longitudinal plane with non-zero surge and heave velocities, u_0 and w_0 respectively. Also, if the equilibrium point is characterized by roll and pitch angles equal to zero, and that the steady-state linear and angular velocities v_0, p_0, q_0 and r_0 are also equal to zero, it can be shown (see Fossen (1994)) that a linear time-invariant equations of motion for an underwater vehicle can be described as:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -M^{-1}[C+D] & -M^{-1}G \\ J & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} M^{-1} \\ 0 \end{bmatrix} \tau \tag{4}$$

where $x_1 = \Delta\nu$, $x_2 = \Delta\eta$ and the matrices C, D, and G arise from the linearization of $C(\nu)\nu$, $D(\nu)\nu$ and $G(\eta)$, respectively, around the equilibrium point.

Further detailing (4), the vector of forces applied by the thrusters if given by

$$\tau = Pf_p = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ z_{p1} & z_{p2} & -x_{p3} & -xp4 \\ -y_{p1} & -y_{p2} & 0 & 0 \end{bmatrix} \begin{bmatrix} F_{p1} \\ F_{p2} \\ F_{p3} \\ F_{p4} \end{bmatrix}, \tag{5}$$

where $p1$ and $p2$ refer to the horizontal left and right thrusters, respectively, while $p3$ and $p4$ refer to the vertical back and front thrusters. In (5) the matrix P is the propulsion matrix, assigning the contribution of each of the thrusters to the different state variables, and x_{p1}, x_{p2}, y_{p1}, y_{p2}, z_{p1} and z_{p2} refer to geometric properties of the location of the thrusters.

Note that (4) can now be expressed in the standard state-space form, which will be of particular usefulness for developments presented ahead in the paper.

$$\dot{x} = Ax + Bu, \tag{6}$$

2.2 Decoupled Pitch and Depth Control

The nature of the combined pitch and depth control we are considering, together with the particular characteristics of the MARES AUV, allow for additional simplifications of the vehicle model to be used, namely by considering only state variables relevant for the motion on the longitudinal plane. That means that we will only consider the state variables corresponding to both surge velocity u, heave velocity w, pitch rate q and pitch angle θ, and also the depth z. Thus, the following model can be used, which only the diving equations of motion:

$$\begin{bmatrix} \Delta\dot{u} \\ \Delta\dot{w} \\ \Delta\dot{q} \\ \Delta\dot{\theta} \\ \Delta\dot{z} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & 0 & 0 \\ a_{21} & a_{22} & a_{23} & a_{24} & 0 \\ a_{31} & a_{32} & a_{33} & a_{34} & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \Delta u \\ \Delta w \\ \Delta q \\ \Delta\theta \\ \Delta z \end{bmatrix} + \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ 0 \\ 0 \end{bmatrix} u \tag{7}$$

The diving motion model of (7), introduced by Fossen (1994), was obtained by linearization around the point $[u_0, w_0, q_0, \theta_0, z_0]^T = [1, 0, 0, 0, 0]^T$. The term u naturally referes to the forces produced by the thrusters. The terms of the matrix A, a_{ij}, are obtained from the corresponding terms of (4). For the sake of simplicity of notation, in what follows the Δ will be dropped when referring to any of the linearized state vectors.

3. EIGENSTRUCTURE ASSIGNMENT

Among the main advantages of EA techniques are the ability to deal with MIMO systems in a natural fashion, while at the same time using the available extra degrees of freedom to assign the eigenvectors in a way that mode decoupling can be achieved. Successful applications of EA have been mostly used to the control of aerial vehicles like (see Andry et al. (1983) for example), but little work as been done concerning autonomous underwater vehicles. In this section we describe the general EA algorithm for output feedback control.

Consider the following multivariable linear system

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx \end{cases} \tag{8}$$

where $x \in \mathcal{R}^n$, is the state vector, $u \in \mathcal{R}^r$ is the input vector, $y \in \mathcal{R}^m$ output vector and A, B and C are matrices of appropriate dimensions. It is possible to demonstrate that the system time response to an initial condition x_0 of the state vector, is given by

$$x(t) = Me^{\Lambda_i t}M^{-1}x_0 \tag{9}$$

where M is the matrix of eigenvectors of the system, $\Lambda = diag(e^{\lambda_1 t}, e^{\lambda_2 t}, \dots e^{\lambda_m})$ with λ_i the eigenvalues of the system, and x_0 the initial conditions (for more details see Andry et al. (1983)). Equation 9 demonstrates that the transient response of a system is dependent, apart from the initial conditions, on its eigenvalues but also on its eigenvectors. From (9) it is also possible to see that while the eigenvalues λ_i are mostly responsible for determining the decay rate of the response, the eigenvectors are responsible for the shape of the solution. It is now obvious that the transient response of a system depends most critically on its eigenstructure - set of eigenvalues and eigenvectors of the system. Thus, any results concerning changing the shape of the transient response of a system must change its eigenstructure.

Consider now applying a output feedback control law to the system(8) such that $u = Ky$. Under the influence of such control, the closed loop system becomes

$$\dot{x(t)} = (A + BKC)x(t). \tag{10}$$

For such system, EA is then reduced to the problem of finding the matrix K such that the eigenstructure of $A + BKC$ is as desired. Note that (10) doesn't make any restrictions on the number of inputs or outputs of the systems, thus making EA a technique suitable for the control of Multi-Input Multi-Output (MIMO) systems. The eigenvalues of such system represent its poles and, therefore, are important for the stability and speed of the system. The eigenvectors, on the other hand, are linked with the shape of the system and are used to induce decoupling among the different modes.

Recalling the definition of eigenvalues and eigenvectors, from (10) it is possible to write

$$(A + BKC)v_i = \lambda_i v_i. \tag{11}$$

or equivalently,

$$v_i = (\lambda_i I - A)^{-1} BKCv_i. \tag{12}$$

Equation (12) provides the only restriction as to the nature of the choice of eigenvectors. That means that they should be chosen such that they lie in subspace spanned by the columns of $L_i = (\lambda_i I - A)^{-1}B$

Fully specifying all the elements of all the eigenvectors is usually not possible due to the mathematical properties of the problem. According to Faleiro (1998), the number of specified elements in each eigenvector that can be decoupled is equal to $m - 1$, where m is the number of inputs of the systems. Because of that, the usual common is to use only specify the elements that are likely to induce the decoupling of the modes, thus leading to the construction of the vector v_i^d of the desired eigenvectors.

$$v_i^d = \begin{bmatrix} v_{i1} & X & X & \dots & v_{ij} & v_{in} \end{bmatrix}^T \tag{13}$$

In (13), v_{ij} are the specified components of the desired eigenvector, while X are the components of no interest. A common approach to induce decoupling is to set to zero the respective elements of v_i^d. In general a desired eigenvector, v_i^d doesn't lie on the subspace L_i. Therefore, an achievable eigenvector is obtained by projecting the specified elements of v_i^d onto an appropriate achievable subspace. The usual way to do this, as described by Andry et al. (1983), is by first applying a reorder operator $\{.\}^{R_i}$ such that

$$\{v_i^d\}^{R_i} = \begin{bmatrix} l_i \\ d_i \end{bmatrix} \tag{14}$$

where l_i are the specified elements of v_i^d and d_i the unspecified ones. The same operator should be applied to the subspace L_i so that

$$\{L_i\}^{R_i} = \begin{bmatrix} \tilde{L_i} \\ D_i \end{bmatrix}. \tag{15}$$

An achievable eigenvector, which on the general case differs from the desired one, can then be obtained as:

$$v_i^a = L_i z_i = L_i \tilde{L_i}^{\dagger} l_i \tag{16}$$

Obtaining all the desired achievable eigenvectors involves using (16) several times, on which the notation $(.)^{\dagger}$ refers to the Moore–Penrose pseudoinverse. Due to the involved matrix inversion operation, the numerical integrity of such operations can be at risk. Because of that, an alternative to (14 - 16) can be used using a constrained least square formulation.

By considering the case of a full output feedback where $Kv_i = u_i$, it is possible to rewrite (11) in a matricial form as

$$\begin{bmatrix} (\lambda_i I - A) | B \end{bmatrix} \begin{bmatrix} v_i \\ u_i \end{bmatrix} = 0 \tag{17}$$

What we are now seeking is to find an achievable vector v_i^a such that (17) is respected, while at the same time being as close as possible to the desired eigenvector v_i^d. In specific, the corresponding elements of v_i^a should be as close as possible to the specified elements of v_i^d. Introducing the selection matrix S, this can be translated into the following:

$$\underset{x}{\text{minimize}} \ \|Sv_i^a - l_i\|_2 \tag{18}$$

From (17 - 18), it becomes obvious that the EA problem can be transformed into a standard least square problem subject to equality constrains, that can be solved using any of the widely available optimization engines.

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \|A\mathbf{x} - \mathbf{b}\|_2 \\ \text{subject to} \quad & B\mathbf{x} = \mathbf{d} \end{aligned} \tag{19}$$

After solving (19), the gain matrix K can then be calculated straightforwardly as

$$K = UV^{-1}, \tag{20}$$

where $U = [u_1 \ \dots \ u_n]$ and $V = [v_1 \ \dots \ v_n]$.

4. CONTROLLER DESIGN

In this section we will present the design of the desired depth-pitch controller for the MARES AUV. We wish to derive a control that is able to follow depth and pitch references that are fed to the system.

Recall from Section 2 the expressions for the simplified 5 DOF diving equations of the MARES AUV. Gathering all the appropriate terms, the matrix A in (7) is:

$$A = \begin{bmatrix} -0.12 & 0 & 0.0006 & 0 & 0 \\ 0 & -1.5483 & 0 & -0.9782 & 0 \\ 0.0032 & 0 & -1.5858 & 0 & 0 \\ 0 & 0 & 1.0000 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix} \tag{21}$$

By analysing (21), it is possible to verify that it presents five different modes, whose values are listed in Table 1.

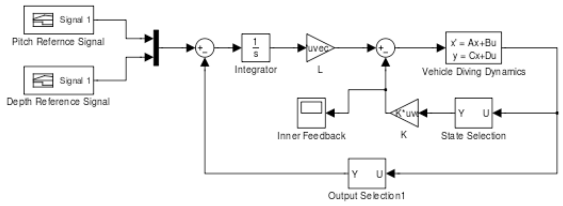

Fig. 2. Controller structure

From there it can be seen that all the open-loop eigenvalues are real, but tow of them are zero, and one is very close to zero. In order to improve overall stability of the system we wish to bring those eigenvalues further inside the left s-plane, but without demanding excessive control power. In what follows, we also assume all state variables to be directly are measured, which are relatively mild assumptions for current day AUVs. This means that the output matrix C of the linearised system is simply an identity matrix.

Table 1. Open loop eigenvalues

	λ_1	λ_2	λ_3	λ_4	λ_5
Value	-1.54	-1.59	-0.12	0	0

In order to implement a tracking system, we use the standard approach which is to introduce additional states in the state vector, one for each signal to be tracked. We augment the system with $z = [z_1 \ z_2]^T$ such that

$$\dot{z} = r - w = r - Ex, \tag{22}$$

where $r = [r_1 \ r_2]$ is the references vector and the matrix E assigns the outputs which are required to follow the input vector r. The augmented system is now as in (23).

$$\begin{bmatrix} \dot{x} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} A & 0 \\ -E & 0 \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u \tag{23}$$

Similarly to (10), the closed-loop system that results after applying a output feedback law $u = Ky$ is

$$\begin{bmatrix} \dot{x} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} A + BK_1 & BK_2 \\ -E & 0 \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} r. \tag{24}$$

A schematics depicting the structure of the controller derived can be seen in Figure 2. It is interesting to note that by using EA, all the gains involved are calculated in a single step.

Using empirical knowledge, we specify the eigenvalues of the tracking system as in Table 2. There the desired and obtained eigenvalues can be compared. As for the eigenvectors used for decoupling they were specified as indicated in Table 3.

Table 2. Closed-loop eigenvalues

	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	λ_7
Desired	-2	-5	-3	-2.5	-3.5	-1.5	-1.7
Obtained	-2.00	-4.58	-3.00	-2.50	-3.50	-1.50	-1.71

Table 3. Desired eigenvectors

	v_1	v_2	v_3	v_4	v_5	v_6	v_7
u	X	1	0	X	1	0	0
w	X	X	X	0	X	X	X
q	X	X	X	X	X	X	X
θ	0	X	X	1	X	0	X
z	X	0	X	X	X	X	0
r_θ	1	X	1	X	X	1	X
r_z	X	1	1	X	X	0	1

5. SIMULATION RESULTS

The overall performance of the controller derived in the previous sections was assessed in a MATLAB/Simulink simulation environment.

We started by testing the system performance when subject to a step reference input on depth only. The purpose of it is to assess the coupling between states. The result of this can be seen in Fig. 3. Naturally, the reference in depth induces a variation not only in z but also in w, as expected. As for the other states, there are minimal transient variations, suggesting that the decoupling was successful.

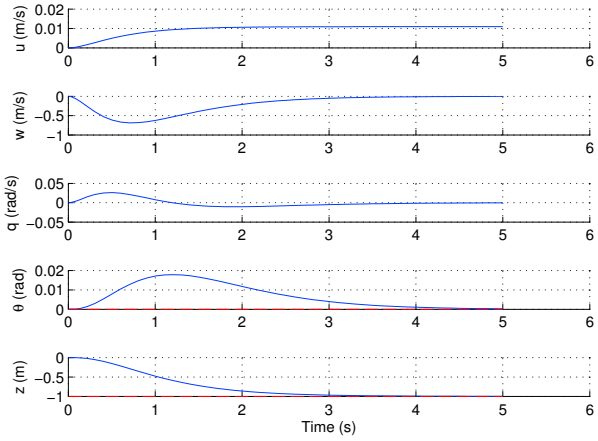

Fig. 3. Evolution of the different state variables when the system is following a step depth reference signal

Next, we simulate a constant slope bottom with an inclination of about 15°. This was simulated with a negative ramp reference signal for depth, and a step reference signal for pitch. The results of it are depicted on Fig. 4 and Fig. 5. On the first one we see the evolution of all the relevant states along time, and it is possible to see that there is a change in w compatible with the variations in depth z, while at the same time variations in q match the changes in pitch, θ. Therefore, the results are very satisfactory. Most importantly, there is no overshoot neither in depth or pitch.

Figure 5 presents a detailed view on the error signal, that is the difference between reference and output of the system along the time. It is possible to see that while pitch is correctly tracked along time with zero error, there is a small steady state error in depth, of less than 0.25 meters. This is because the reference signal from time step $t = 5s$ till $t = 20$ is a ramp, and only one integrator is included in the controller. Should this error be required to be zero, another integrator is needed.

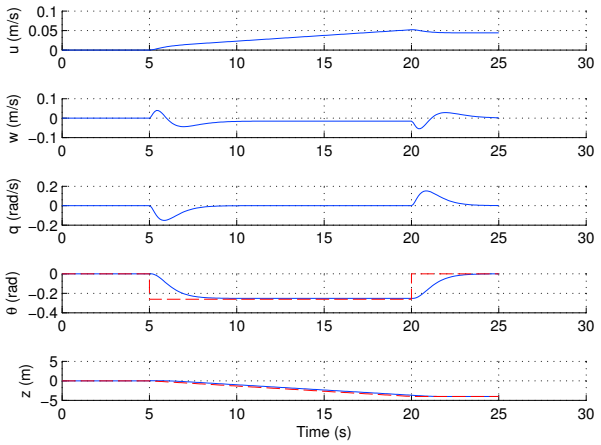

Fig. 4. Evolution of the different state variables to dept-pitch reference. Solid line correspond to the output of the system, dashed lines correspond to the reference signals.

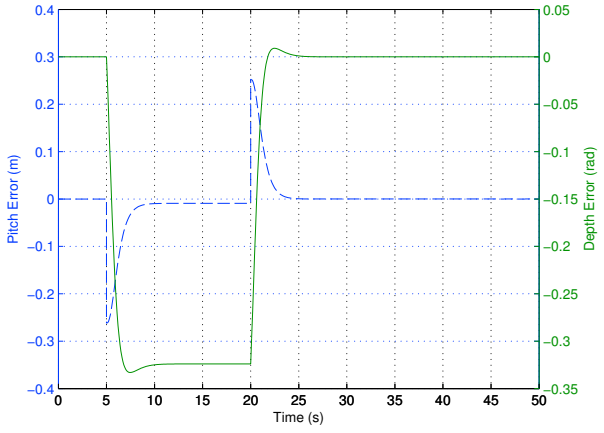

Fig. 5. Transient error response of both depth and pitch following a reference signal

6. CONCLUSION AND FUTURE WORK

This paper addressed the problem of designing a bottom following depth-pitch controller using Eigenstructure Assignment. Among the main advantages of this technique are the ability to deal with MIMO systems in a natural fashion, and the calculation of all the gains involved in a single step, without worrying on the effects cascading implementations may suffer. Moreover, EA allows to take into account restrictions related with the transient response of the system and, in specific, with the overshoot in the response to a desired reference signal. This of particular importance for the problem we are addressing.

The simulation results clearly indicate that the controller derived has a decent performance for bottom following scenarios. Stability and zero tracking error of the closed-loop system were achieved in response to step inputs in both depth and pitch, and in both cases there is no overshoot observed, which was the main objective. Naturally, there exists a small steady-state error in response to ramps, but this can be tackled by introducing an additional integrator in the system, though care must be taken to ensure the overall stability of the system. Future research will address such issue, as well as implementing the controller on the on-board computer of the MARES AUV in order to assess the performance real-world scenarios.

REFERENCES

Adhami-Mirhosseini, A., Aguiar, A., and Yazdanpanah, M. (2011). Seabed tracking of an autonomous underwater vehicle with nonlinear output regulation. In *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, 3928 –3933. doi: 10.1109/CDC.2011.6161411.

Andry, A., Shapiro, E., and Chung, J. (1983). Eigenstructure assignment for linear systems. *Aerospace and Electronic Systems, IEEE Transactions on*, AES-19(5), 711–729. doi:10.1109/TAES.1983.309373.

Bennett, A., Leonard, J., and Bellingham, J. (1995). Bottom following for survey-class autonomous underwater vehicles. *International Symposium on Unmanned Untethered Submersible Technology*, 1, 327–336.

Caccia, M., Bruzzone, G., and Veruggio, G. (1999). Active sonar-based bottom-following for unmanned underwater vehicles. *Control Engineering Practice*, 7(4), 459 – 468. doi:10.1016/S0967-0661(98)00168-3.

Caccia, M., Bruzzone, G., and Veruggio, G. (2003). Bottom-following for remotely operated vehicles: Algorithms and experiments. *Autonomous Robots*, 14, 17–32. doi:10.1023/A:1020923402707.

Faleiro, L.F. (1998). *The Application of Eigenstructure Assignment to the Design of Flight Control Systems*. Ph.D. thesis, Loughborough University.

Ferreira, B., Matos, A., Cruz, N., and Pinto, M. (2010). Modeling and control of the mares autonomous underwater vehicle. *Marine Technology Society Journal*, 44(2), 19–36.

Fossen, T.I. (1994). *Guidance and control of ocean vehicles*. John Wiley & Sons, Ltd.

Gao, J., Xu, D., Zhao, N., and Yan, W. (2008). A potential field method for bottom navigation of autonomous underwater vehicles. In *Intelligent Control and Automation, 2008. WCICA 2008. 7th World Congress on*, 7466 –7470. doi:10.1109/WCICA.2008.4594083.

Melo, J. and Matos, A. (2012). Bottom estimation and following with the mares auv. In *Oceans, 2012*, 1–8. doi:10.1109/OCEANS.2012.6404917.

Silvestre, C., Cunha, R., Paulino, N., and Pascoal, A. (2009). A bottom-following preview controller for autonomous underwater vehicles. *Control Systems Technology, IEEE Transactions on*, 17(2), 257–266. doi: 10.1109/TCST.2008.922560.

Yoerger, D., Bradley, A., Walden, B., Cormier, M.H., and Ryan, W. (2000). Fine-scale seafloor survey in rugged deep-ocean terrain with an autonomous robot. In *Robotics and Automation, 2000. Proceedings. ICRA '00. IEEE International Conference on*, volume 2, 1787 –1792 vol.2. doi:10.1109/ROBOT.2000.844854.