Path Following of Underwater Robots using Lagrange Multipliers

Ehsan Peymani, Thor I. Fossen

Center for Ships and Ocean Structures
Department of Engineering Cybernetics, Norwegian University of Science and Technology
Trondheim, Norway

Abstract

This article proposes a framework for motion control of mechanical systems with application to underwater marine craft. Motivated by analytical mechanics, control problems are reformulated as modeling of constrained multi-body systems. Then, techniques for forward dynamics simulations are employed to obtain valid models under any circumstances. The suggested method is then used to develop a novel path-following controller for UUVs. Moreover, a strategy is proposed to handle underactuation. Computer simulations are also conducted to verify the validity of the theoretical developments.

Keywords: path following, virtually constrained motion, constraint force, underactuation.

1. Introduction

Interacting with and exploring the subsea world are inconceivable without the help of robotic systems. However, the unstructured and hazardous ocean environment encourages the use of semi- and fully autonomous unmanned underwater vehicles (UUVs) to take the danger away from personnel. Generally speaking, UUVs consist of remotely operated vehicles (ROVs) and autonomous underwater vehicles (AUVs) [1].

Inaccurate path following may cause uncertainties in the characteristics of recorded information in seabed mapping or may lead to losing the UUV in under-ice operations. To provide an enhanced solution, the article is dedicated to path following of UUVs. In contrast to a considerable number of articles, the present study calls attention to the subtle but powerful capabilities of analytical mechanics for handling constrained multi-body systems.

In fact, the gist of the proposed method is to view motion control problems as modeling of mechanical systems, consisting of several bodies, subject to various constraints. Then, with the help of Lagrangian mechanics [2], a mathematical representation for motion of each body is individually derived as if the body is allowed to move freely in the selected configuration space. Imposing constraints, the variational principle of analytical mechanics [3] and the Lagrange multiplier method are taken into account in order to find the forces that have to be exerted on each body so that it meets all corresponding constraints. Hence, the system is artificially constrained by virtue of control objectives and the forces due to virtual constraints are applied to the system as the control forces.

This bright idea has been less used in maritime applications. References [4, 5] are of the few articles in this field. They encompass trajectory tracking and formation control of fully actuated marine surface craft. However, this has been of limited use in practice since marine craft are mainly underactuated. On the other hand, most applications are concerned with path following and speed control, usually called path maneuvering [6, 7]. In the path maneuvering scenario, steering the object along the path takes precedence over the speed assignment. Therefore, by relaxing temporal constraints, compared to trajectory tracking, path maneuvering becomes less restrictive, resulting in better performance [8] and more robust behavior in the case of thruster saturations [7].

In the present article, we take this technique one stage further to be able to apply it to path maneuvering. The requirement is to place velocity constraints on the system since [4, 5] only consider positional restrictions. Then, path maneuvering of UUVs is taken into consideration to illustrate the proposed technique. It is demonstrated that the suggested methodology results in a novel controller which gives the path following task priority over the speed assignment task, facilitates the convergence proof, and simplifies dealing with external disturbances. Moreover, to handle underactuation, a method is suggested.

Previous Work

The concept of virtual constraints were first introduced in [9] and developed within the task frame formalism [10]. It is very popular in the context of force control to handle robot manipulators in contact with the environment [11]. Another thread of thought of virtually constrained systems is presented in [12] in which orbital stabilization and periodic motion planning are considered while the concept

Email addresses: ehsan@ntnu.no (Ehsan Peymani), fossen@ieee.org (Thor I. Fossen)

of constraint forces are not used. Pertinent to [4, 5], a formation controller for spacecraft is designed in [13] where virtual holonomic constraints are placed on the system.

In [14], a fundamental equation of motion for constrained systems was developed based on Gauss's principle. Without utilizing the Lagrange multiplier method, it determines the force of constraints. Subsequently, in [15], the explicit equation of constrained motion was used for tracking control of nonlinear systems. Afterwards, the idea was expanded to trajectory tracking of robot manipulators [16], synchronization of gyroscopes [17], and formation keeping of satellites [18]. They all consider full state trajectory tracking which leads to placing holonomic constraints and do not discuss disturbances.

2. Equations of Motion of Mechanical Systems

Mathematical representation of an unconstrained mechanical system takes the following form [2]:

$$\mathcal{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{n}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{\tau} \tag{1}$$

where $\boldsymbol{q} \in \mathbb{R}^n$ is a vector of generalized coordinates that uniquely characterizes the position and orientation of the system in an n-dimensional configuration space. n is termed as the number of degrees of freedom (DOF). Obviously, $\dot{\boldsymbol{q}}$ is the time-derivative of $\boldsymbol{q}$ and is called the generalized velocity vector. $\mathcal{M}(\boldsymbol{q}) = \mathcal{M}(\boldsymbol{q})^T \in \mathbb{R}^{n\times n}, \mathcal{M}(\boldsymbol{q}) > 0$ is the inertia matrix. The n-vector $\boldsymbol{n}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ includes Coriolis-centripetal, potential and viscous forces. $\boldsymbol{\tau} \in \mathbb{R}^n$ is a vector of generalized forces, arising from actuators, external disturbances, or reactions among various bodies in a complex system.

2.1. 6 DOF Model of UUVs

A robot-like model for marine craft is introduced by [19]. Consider $\boldsymbol{q} = [\boldsymbol{p}^T, \boldsymbol{\Theta}^T]^T \in \mathbb{R}^3 \times \mathcal{S}^3$ as the vector of the position and orientation expressed in the inertial frame $\{i\}$, and $\boldsymbol{\nu} = [\boldsymbol{\upsilon}^T, \boldsymbol{\omega}^T]^T \in \mathbb{R}^6$ as the linear and angular velocity vector expressed in the body-fixed reference frame $\{b\}$ (see Fig. 1). The kinetics is described by

$$\mathcal{M}_b\dot{\boldsymbol{\nu}} + \mathcal{C}_b(\boldsymbol{\nu})\boldsymbol{\nu} + \mathcal{D}_b(\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{\tau}_p^b + \boldsymbol{\tau}_{dist}^b \tag{2}$$

in which $\mathcal{M}_b = \mathcal{M}_b^T > 0, \dot{\mathcal{M}}_b = 0, \mathcal{C}_b = -\mathcal{C}_b^T$, and $\mathcal{D}_b > 0$. In (2), $\boldsymbol{\tau}_p^b$ represents the vector of actuator forces and moments, and $\boldsymbol{\tau}_{dist}^b$ captures forces due to external disturbances, where the superscript "b" stands for the body representation. The transformation between $\dot{\boldsymbol{q}}$ and $\boldsymbol{\nu}$ is described by the kinematic equation

$$\dot{\boldsymbol{q}} = \mathcal{J}(\boldsymbol{\Theta})\boldsymbol{\nu} \tag{3}$$

where the transformation matrix $\mathcal{J}(\boldsymbol{\Theta})$ can be found in [20]. Using (3), (2) is translated to the form (1) with

$$\mathcal{M}(\boldsymbol{q}) = \mathcal{J}^{-T}(\boldsymbol{\Theta})\mathcal{M}_b\mathcal{J}^{-1}(\boldsymbol{\Theta}),\ \boldsymbol{\tau} = \mathcal{J}^{-T}(\boldsymbol{\Theta})(\boldsymbol{\tau}_p^b + \boldsymbol{\tau}_{dist}^b)$$
$$\boldsymbol{n}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \mathcal{C}_n(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \mathcal{D}_n(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \mathcal{J}^{-T}(\boldsymbol{\Theta})\boldsymbol{g}(\boldsymbol{q})$$

in which, $\mathcal{C}_n$ and $\mathcal{D}_n$ can be found by inspection.

2.2. Constrained Motion

The motion of a system might be restricted to assorted constraints classified under various criteria. *Holonomic constraints* are expressed in terms of generalized coordinates; thus, a set of k holonomic constraints can be put in the vector form:

$$\boldsymbol{G}(\boldsymbol{q}, t) = 0_k \qquad k < n \tag{4}$$

where t is the time variable and 0_k is the k-vector of zeros. In case constraints involve generalized velocities, *kinematic constraints* arise [21]. A set of m first-order kinematic constraints can be generally described by:

$$\boldsymbol{K}(\boldsymbol{q}, \dot{\boldsymbol{q}}, t) = 0_m \qquad m \leq n \tag{5}$$

If kinematic constraints are integrable, they do not depend on generalized velocities and they are inherently holonomic even if the integrated form is not known.

Otherwise, it is said to be *nonholonomic*. A well-trodden form of (5) is linear in the generalized velocities:

$$\boldsymbol{K}(\boldsymbol{q}, \dot{\boldsymbol{q}}, t) \triangleq \mathcal{A}(\boldsymbol{q}, t)\dot{\boldsymbol{q}} + \boldsymbol{\mathcal{B}}(\boldsymbol{q}, t) \tag{6}$$

in which $\mathcal{A} \in \mathbb{R}^{m\times n}, \boldsymbol{\mathcal{B}} \in \mathbb{R}^m$. The stationary, driftless form of (6) is called *Pfaffian* constraints. Details of terminology and types of kinematic constraints can be found in [2]. This article focuses on equality constraints.

2.3. Handling Constrained Motion

Contrary to the methods involved with elimination of dependency between position variables [2] or velocities [22], there exists another method that keeps constraints and considers the effect of constraints as the forces acting on the system so that the constraints hold.

For holonomic constraints, the reaction forces are obtained by generalization of D'Alembert's principle and utilizing the Lagrange multiplier method [3]. Considering the multiplier vector $\boldsymbol{\lambda}_G \in \mathbb{R}^k$, the additional force

$$\boldsymbol{\tau}_G = -\mathcal{W}_G^T(\boldsymbol{q}, t)\boldsymbol{\lambda}_G, \qquad \mathcal{W}_G = \partial\boldsymbol{G}/\partial\boldsymbol{q} \in \mathbb{R}^{k\times n} \tag{7}$$

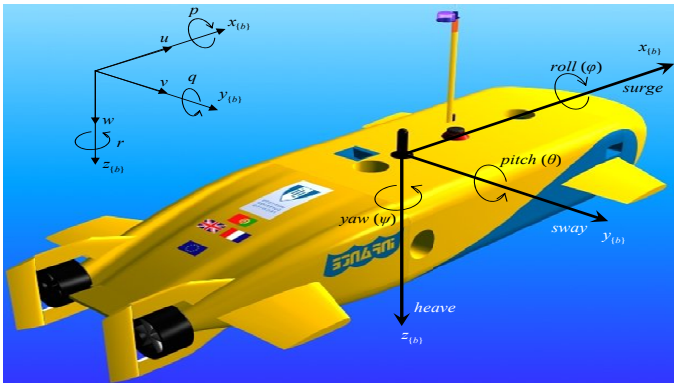

Figure 1: INFANTE AUV with the body-fixed reference frame. $\boldsymbol{\upsilon} = [u, v, w]^T$ is the vector of linear velocities along the axes of $\{b\}$. $\boldsymbol{\omega} = [p, q, r]^T$ is the vector of angular velocities about the axes of $\{b\}$. The vector $\boldsymbol{p} = [x, y, z]^T$ denotes the position in $\{i\}$, and the Euler angle vector $\boldsymbol{\Theta} = [\phi, \theta, \psi]^T \in \mathcal{S}^3$ represents the orientation of the craft.

is applied to the system to satisfy holonomic constraints. The result also follows from the principle of virtual work stating that the force of constraints do no work under any virtual displacements.

However, general nonholonomic constraints (5) are outside the scope of any principle that consists in the principle of D'Alembert [23]. Nonetheless, the basic form of the principle of D'Alemebert can be utilized to derive the force of linear-in-velocity nonholonomic constraints (6). It gives rise to

$$\boldsymbol{\tau}_K = -\mathcal{A}^T(\boldsymbol{q},t)\boldsymbol{\lambda}_K, \quad \boldsymbol{\lambda}_K \in \mathbb{R}^m \tag{8}$$

The force of holonomic and nonholonomic constraints are superimposed onto the model of the system as if the system is not constrained but under the influence of additional forces; thus, the overall constraint force vector is

$$\boldsymbol{\tau}_c = \boldsymbol{\tau}_G + \boldsymbol{\tau}_K \tag{9}$$

Defining the *multiplier* vector $\boldsymbol{\lambda} \triangleq [\boldsymbol{\lambda}_K^T, \boldsymbol{\lambda}_G^T]^T \in \mathbb{R}^{k+m}$ and the *constraint Jacobian* matrix $\mathcal{W} \triangleq [\mathcal{A}^T, \mathcal{W}_G^T]^T \in \mathbb{R}^{(m+k)\times n}$, in view of (7) and (8), the constraint force vector (9) is recast as

$$\boldsymbol{\tau}_c = -\mathcal{W}^T\boldsymbol{\lambda} \tag{10}$$

Remark 1. The forces of constraints, $\boldsymbol{\tau}_c$, are generalized forces. Hence, they are superimposed to the right-hand side of (1) to correct the accelerations of the system.

3. "Virtually Constrained Motion" Controller

In the previous section, naturally constrained motion and the concept of reaction forces were explained for mechanical systems. However, a system can be artificially restricted to desired functions by means of control strategies. The resulted reaction forces are then conceived as the control inputs required to control the system as specified. In this section, we go through the procedure of designing controllers based on the concept of virtual constraints.

3.1. Overview of the control methodology

Fig. 2 illustrates the overall perspective of the method. For this control strategy, the control objectives are formulated in the forms (4) and (6), and regarded as virtual constraints imposed on the system. The system along with the virtual confinements comprises the "constrained system" block, explained in Section 2. The other block, namely "constraint stabilization", is to circumvent inconsistency to the initial conditions, as well as solving the drift problem. This block is inspired by the stabilization method of Baumgarte [24] for numerical resolutions of constrained systems. From the control viewpoint, it makes the control system robust to external disturbances, measurement noise, and modeling uncertainties by virtue of a feedback loop. In this section, the concept of constraint stabilization is enunciated. At the end of the section, we can grasp the physical interpretation of the control methodology.

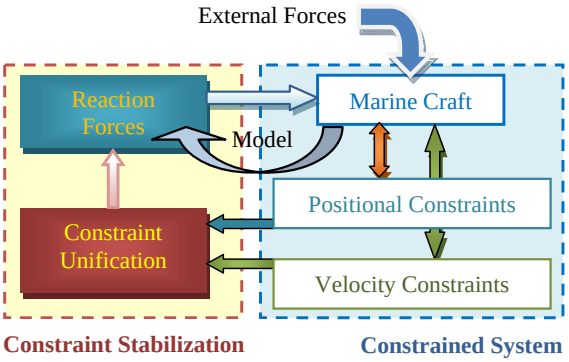

Figure 2: Flow chart showing the proposed method for control of virtually constrained mechanical systems.

3.2. Constraint Stabilization

The goal of the constraint stabilization task is to make the constraint manifold invariant and attractive so that if the constraints are violated for any reason, whether due to inconsistent initial conditions or due to unknown perturbations, they are properly enforced. To this end, holonomic constraints are stabilized at the velocity level in order to be merged with nonholonomic confinements; after that, the unified function is stabilized to find the control law. The unification method is now clarified.

3.2.1. Constraint Unification

As k holonomic constraints (4) must hold always, the corresponding manifold is made invariant using $\dot{\boldsymbol{G}} = \boldsymbol{u}_{2G}$. Picking $\boldsymbol{u}_{2G} = -\mathcal{P}_G\boldsymbol{G}$ with positive definite $\mathcal{P}_G \in \mathbb{R}^{k\times k}$, $\boldsymbol{G} = 0_k$ is globally exponentially stable (GES) and

$$\dot{\boldsymbol{G}} + \mathcal{P}_G\boldsymbol{G} = 0_k \Rightarrow \mathcal{W}_G\dot{\boldsymbol{q}} + \boldsymbol{G}^t + \mathcal{P}_G\boldsymbol{G} = 0_k \tag{11}$$

in which $\mathcal{W}_G = \partial\boldsymbol{G}/\partial\boldsymbol{q} \in \mathbb{R}^{k\times n}$, and $\boldsymbol{G}^t = \partial\boldsymbol{G}/\partial t \in \mathbb{R}^k$. Eq. (11) is identical to the nonholonomic linear-in-velocity constraints (6) in structure. Putting (11) and (6) into a unified representation, it follows that

$$\boldsymbol{\Phi}(\boldsymbol{q},\dot{\boldsymbol{q}},t) \triangleq \mathcal{W}(\boldsymbol{q},t)\dot{\boldsymbol{q}} + \boldsymbol{a}(\boldsymbol{q},t) \tag{12}$$

$$\boldsymbol{a}(\boldsymbol{q},t) = \begin{bmatrix} \boldsymbol{\mathcal{B}}(\boldsymbol{q},t) \\ \boldsymbol{G}^t(\boldsymbol{q},t) + \mathcal{P}_G\boldsymbol{G}(\boldsymbol{q},t) \end{bmatrix} \tag{13}$$

$\boldsymbol{\Phi} \in \mathbb{R}^{m+k}$ is called the *unified* constraint function, and it must be always zero. With this aim in view, the system $\dot{\boldsymbol{\Phi}} = \boldsymbol{u}_\Phi$ is made uniformly globally exponentially stable (UGES) by a proper selection for $\boldsymbol{u}_\Phi$. Considering the positive definite matrix $\mathcal{P}_\Phi \in \mathbb{R}^{(m+k)\times(m+k)}$, $\boldsymbol{u}_\Phi$ may be chosen as $-\mathcal{P}_\Phi\boldsymbol{\Phi}$; it then gives rise to

$$\dot{\boldsymbol{\Phi}} + \mathcal{P}_\Phi\boldsymbol{\Phi} = 0_{m+k} \Rightarrow \mathcal{W}\ddot{\boldsymbol{q}} + \dot{\mathcal{W}}\dot{\boldsymbol{q}} + \dot{\boldsymbol{a}} + \mathcal{P}_\Phi\boldsymbol{\Phi} = 0_{m+k} \tag{14}$$

where $\mathcal{W}$ is the constraint Jacobian matrix defined in (10).

3.2.2. Reaction Forces

To derive constraint forces, one should solve the model (1) for acceleration; i.e.

$$\ddot{\boldsymbol{q}} = \mathcal{M}(\boldsymbol{q})^{-1}\left(\boldsymbol{\tau} + \boldsymbol{\tau}_c - \boldsymbol{n}(\boldsymbol{q},\dot{\boldsymbol{q}})\right) \tag{15}$$

where $\boldsymbol{\tau}$ is the impressed force vector and $\boldsymbol{\tau}_c$ is given by (10). Substituting (15) into (14) yields

$$\mathcal{W}\mathcal{M}^{-1}\mathcal{W}^T\boldsymbol{\lambda} = \mathcal{W}\mathcal{M}^{-1}\left(\boldsymbol{\tau} - \boldsymbol{n}\right) + \dot{\mathcal{W}}\dot{\boldsymbol{q}} + \dot{\boldsymbol{a}} + \mathcal{P}_\Phi\boldsymbol{\Phi} \quad (16)$$

In (16), if $\mathcal{W} \in \mathbb{R}^{(m+k)\times n}$ is a full-row-rank matrix (i.e. $\text{rank}(\mathcal{W}) = m + k \leq n$), the product $\mathcal{W}\mathcal{M}^{-1}\mathcal{W}^T$ is a non-singular square matrix of order $(m + k)$. Then, following computation of the vector of multipliers $\boldsymbol{\lambda}$, the reaction forces are found using (10) at each instant of time. In case $\mathcal{W}$ has n linearly independent columns (i.e. $\text{rank}(\mathcal{W}) = n \leq m + k$), the product $\mathcal{W}\mathcal{M}^{-1}\mathcal{W}^T$ is not invertible. To solve the problem, the constraint force vector has to be directly computed from (16) by utilizing Moore-Penrose (MP) pseudo-inverse [25] for $\mathcal{W}\mathcal{M}^{-1}$. Since $\mathcal{W}\mathcal{M}^{-1}$ is full column rank, its MP pseudo-inverse, denoted by $(\mathcal{W}\mathcal{M}^{-1})^\dagger$, is equal to $\mathcal{M}\mathcal{W}^\dagger$ in which $\mathcal{W}^\dagger = (\mathcal{W}^T\mathcal{W})^{-1}\mathcal{W}^T$. Consequently, the vector of constraint forces emerges as

$$\boldsymbol{\tau}_c = \boldsymbol{n}(\boldsymbol{q}, \dot{\boldsymbol{q}}) - \boldsymbol{\tau} - \mathcal{M}\mathcal{W}^\dagger\left(\dot{\mathcal{W}}\dot{\boldsymbol{q}} + \dot{\boldsymbol{a}} + \mathcal{P}_\Phi\boldsymbol{\Phi}\right) \quad (17)$$

Theorem 1 states that applying the constraint force vector to (1) ensures achievement of the control objectives.

Theorem 1 ("Virtually Constrained Motion" Controller)**.** *Consider the n-DOF mechanical system (1) subject to k holonomic (4) and m nonholonomic constraints (6). The reaction force vector (16) ensures that all trajectories of the system move such that the manifold*

$$\mathbb{E} = \{(\boldsymbol{q}, \dot{\boldsymbol{q}}) : \boldsymbol{G} = 0_k, \dot{\boldsymbol{G}} = 0_k, \boldsymbol{K} = 0_m\} \quad (18)$$

is rendered uniformly globally exponentially stable (UGES), provided $\mathcal{W}$ is full (row/column) rank.

Proof. Applying (16) to (1) as the constraint force vector results in (14), which implies (11). Considering $V = \frac{1}{2}(\boldsymbol{G}^T\boldsymbol{G} + \boldsymbol{\Phi}^T\boldsymbol{\Phi}) > 0$, for $\boldsymbol{G}, \boldsymbol{\Phi} \neq 0$ as the Lyapunov function candidate and differentiating along the solutions of (11) and (14), it follows that $\dot{V} < 0$, for $\boldsymbol{G}, \boldsymbol{\Phi} \neq 0$. According to Lyapunov stability [26], $(\boldsymbol{G}, \boldsymbol{\Phi})$ is GES at the origin. Since $\boldsymbol{\Phi} = [\boldsymbol{K}^T, (\dot{\boldsymbol{G}} + \mathcal{P}_G\boldsymbol{G})^T]^T$, it results in GES of $(\boldsymbol{G}, \dot{\boldsymbol{G}}, \boldsymbol{K}) = 0$. Therefore, the relation (16), coming out of the stabilization process, makes $\mathbb{E}$ GES. The full rank property of $\mathcal{W}$ insures existence and uniqueness of the reaction forces $\boldsymbol{\tau}_c$, as explained. □

The control laws derived from the proposed control methodology can be physically interpreted. Considering control objectives as virtual constraints together with stabilization process resembles placing generalized dampers and springs between bodies; these imaginary objects create forces when they leave their equilibrium. Put alternatively, once the bodies of the system violate the constraints, the imaginary dampers and springs turn active to preserve the constraints. One can recognize the description of the generalized dampers and springs in (16). On the other hand, although the control laws are derived from physical first principles, the procedure bears superficial resemblance to the recursive method of backstepping [27] and sliding-mode control [28].

Remark 2. From the control point of view, positional constraints are stabilized using a PD-type controller and velocity constraints are stabilized using a P-type controller.

Remark 3. The reader might wonder how to ensure full rank property of $\mathcal{W}$. It is observed that if the constraints are independent, the Jacobian matrix $\mathcal{W}$ possesses linearly independent rows. By independent, we mean that constraints are not redundant or unnecessary. Moreover, inconsistent constraints must be avoided. Those constraints that contradict one another are called *conflicting* or *inconsistent* [29].

4. Case Study: Path Maneuvering of UUVs

Path maneuvering is considered to illustrate the proposed method.

System Description. A realistic assumption for UUVs is to use an independent depth controller; thus, the craft is thought to move in the horizontal plane. Assuming that the dynamics of heave, roll, and pitch is negligible, the model is reduced to 3 DOFs [20]. Therefore, the vector of generalized coordinates is $\boldsymbol{q} = [x, y, \psi]^T$, see Fig. 1. The generalized velocities are related to the body-frame velocities $\boldsymbol{\nu} = [u, v, r]^T$ through the kinematic relation (3) in which the transformation matrix is $\mathcal{J}(\psi) = \text{diag}(\mathcal{R}(\psi), 1) \in \mathcal{SO}(3)$ where $\forall\alpha \in [-2\pi, 2\pi]$

$$\mathcal{R}(\alpha) \triangleq \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix} \mathcal{S}(\alpha) \triangleq \begin{bmatrix} 0 & -\dot{\alpha} \\ \dot{\alpha} & 0 \end{bmatrix} \quad (19)$$

The kinetic equation (2) is valid but $\boldsymbol{g}(\boldsymbol{q})$ is zero due to horizontal motion and neutral bouyancy. Presuming homogeneous mass distribution and xz-plane symmetry, the system matrices are in the forms

$$\mathcal{M}_b = \begin{bmatrix} m_{11} & 0 & 0 \\ 0 & m_{22} & m_{23} \\ 0 & m_{23} & m_{33} \end{bmatrix}, \mathcal{D}_b = \begin{bmatrix} d_{11} & 0 & 0 \\ 0 & d_{22} & d_{23} \\ 0 & d_{32} & d_{33} \end{bmatrix} \quad (20a)$$

$$\mathcal{C}_b = \begin{bmatrix} 0 & 0 & -(m_{22}v + m_{23}r) \\ 0 & 0 & m_{11}u \\ m_{22}v + m_{23}r & -m_{11}u & 0 \end{bmatrix} \quad (20b)$$

$\boldsymbol{\tau}_p^b \triangleq [\tau_u, \tau_v, \tau_r]^T$. If the vehicle is not capable of moving sideways independently by an immediate force, $\tau_v = 0$ and the vehicle is termed as underactuated.

Path Parametrization. A path parameterized by the variable $\varrho \in \mathbb{R}$ is considered. The position of every point on the path is denoted by $\boldsymbol{p}_p(\varrho) = [x_p(\varrho), y_p(\varrho), z_p(\varrho)]^T \in \mathbb{R}^3$. Presume the path is smooth and physically feasible to be followed by the craft. Consider a particle, called the *path particle*, whose motion is restricted to the path; therefore, its position takes that of the path at the current instant of time; thus, one can parameterize the position of the path

particle with the parameter $\varpi(t) \in \mathbb{R}$ and denote it by $\boldsymbol{p}_p(\varpi)$. A local reference frame, which is named the *path frame* and denoted by $\{p\}$, is placed on the path particle such that its x-axis is tangent to the path in the direction of the particle's motion. Let the speed vector of the path particle, expressed in $\{i\}$, be $\boldsymbol{V}_p = \dot{\boldsymbol{p}}_p$; since the speed is always tangent to the path, the velocity vector, expressed in $\{p\}$, is $\boldsymbol{\nu}_p = [u_p(t), 0, 0]^T$.

The projection of the path on the xy plane creates a 2-D path. Similarly, a particle moving on the projected path is considered and a frame, denoted by $\{p'\}$, is located on that. These notations are also defined: $\bar{\boldsymbol{p}}_p(\varpi) = [x_p, y_p]^T$, $\bar{\boldsymbol{V}}_p = \dot{\bar{\boldsymbol{p}}}_p$, $\bar{\boldsymbol{\nu}}_p = [\bar{u}_p, 0]^T$. If $x^{'}(\varrho) \triangleq \frac{dx}{d\varrho}$, the azimuth angle, ψ_p, and the elevation angle, θ_p, are given by [30]

$$\psi_p = \tan^{-1}\left(\frac{y_p^{'}}{x_p^{'}}\right), \theta_p = \tan^{-1}\left(\frac{-z_p^{'}}{\sqrt{(x_p^{'})^2 + (y_p^{'})^2}}\right) \tag{21}$$

Notice $\bar{u}_p = u_p \cos(\theta_p)$. The kinematic relation $\bar{\boldsymbol{V}}_p = \mathcal{R}(\psi_p)\bar{\boldsymbol{\nu}}_p$ holds. Since $\left\|\bar{\boldsymbol{V}}_p\right\| = \|\bar{\boldsymbol{\nu}}_p\| = |\bar{u}_p|$, one can obtain

$$\dot{\varpi} = \frac{\bar{u}_p}{\sqrt{(x_p^{'})^2 + (y_p^{'})^2}} \tag{22}$$

Problem Formulation. The path maneuvering problem is to make the craft converge to and follow the path with the commanded forward speed profile, $u_d(t) > 0, \forall t$. It can be defined as a dynamic task and a geometric task, as classified in [31]. From the constrained multi-body systems standpoint, it is tantamount to restricting the motion of the vehicle to that of the path particle so that the vehicle travels on the path in the desired direction.

We set out to design a 2-D path-following controller based on the 3-DOF horizontal model of UUVs. To that end, the error vector between the vehicle and the projected path particle, $\boldsymbol{\varepsilon} \triangleq [s, e]^T$, is given by

$$\boldsymbol{\varepsilon}(t) = \mathcal{R}(\psi_p)^T \left(\bar{\boldsymbol{p}} - \bar{\boldsymbol{p}}_p(\varpi)\right) \tag{23}$$

in which $\bar{\boldsymbol{p}} = [x, y]^T$. $\boldsymbol{\varepsilon}$ is the error vector expressed in $\{p'\}$; see Fig. 3. To put it clearly, the along-track error, s, and the cross-track error, e, represent the distance between the vehicle and the particle along the x-axis and y-axis of $\{p'\}$, respectively. Therefore, one objective is to make $\boldsymbol{\varepsilon}(t) \to 0$ as time tends to infinity.

Inspired by the line-of-sight (LOS) guidance method, the following approach angle is proposed to guide the vehicle to reach $\bar{\boldsymbol{p}}_p$ [30]

$$\psi_r = \tan^{-1}\left(-e/\Delta_e\right), \quad \Delta_e > 0 \tag{24}$$

Consequently, the desired angle is found using

$$\psi_d = \psi_p + \psi_r \tag{25}$$

Thus, another constraint is placed on the heading. For the path maneuvering scenario, the path particle can be simply asked to move with $\bar{u}_p = u_d$. However, it is justifiable

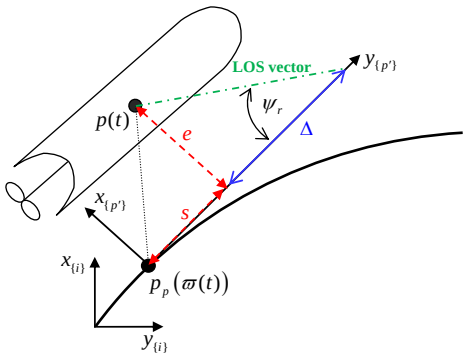

Figure 3: Geometric representation of the path-following problem

to use $\bar{u}_p = u_d + \sigma s$, $\sigma > 0$. This choice makes the path particle be at the shortest distance relative to the UUV since if $s > 0$ the particle speeds up and if $s < 0$ it slows down; see Fig. 3. This choice also helps to achieve better performance when UUV's thrusters saturate.

Eventually, the constraint set is written as

$$\boldsymbol{K} \triangleq \mathcal{R}^T(\psi)\dot{\bar{\boldsymbol{p}}} - [u_d, v_d]^T \tag{26}$$

$$\boldsymbol{G} \triangleq \left[\psi - \psi_d, \boldsymbol{\varepsilon}^T\right]^T \tag{27}$$

We gloss over the reasoning behind placing constraints on the sway velocity. It suffices to mention that it is neither redundant nor inconsistent.

Control Law. Considering (26) and (27), it is straightforward to utilize the aforementioned stabilization procedure and the control law (16) in order to make the set $\mathbb{E}$ invariant and force all the states of the vehicle to approach it. The requirement is to find the Jacobian matrix $\mathcal{W}$. With this object in view, the time-derivative of $\boldsymbol{\varepsilon}$ is found. It is

$$\dot{\boldsymbol{\varepsilon}} = \mathcal{R}(\psi_p)^T\dot{\bar{\boldsymbol{p}}} + \mathcal{S}(\psi_p)^T\boldsymbol{\varepsilon} - \bar{\boldsymbol{\nu}}_p \tag{28}$$

Accordingly, the constraint Jacobian matrix is written as

$$\mathcal{W} = \begin{bmatrix} \mathcal{R}(\psi)^T & 0_2 \\ 0 & 1 \\ \mathcal{R}(\psi_p)^T & 0_2 \end{bmatrix} \in \mathbb{R}^{5\times 3} \tag{29}$$

Obviously, $\text{rank}(\mathcal{W}) = 3$; thus, (17) is directly used. However, to have the constraint forces expressed in $\{b\}$, using (3), (14) is converted to

$$\mathcal{W}_b\dot{\boldsymbol{\nu}} + \dot{\mathcal{W}}_b\boldsymbol{\nu} + \dot{\boldsymbol{a}} + \mathcal{P}_\Phi\boldsymbol{\Phi} = 0_5 \tag{30}$$

where $\mathcal{W}_b \triangleq \mathcal{W}\mathcal{J}(\psi) \in \mathbb{R}^{5\times 3}$. Using the kinetic relation (2), the vector of constraint forces emerges as

$$\boldsymbol{\tau}_p^b = \boldsymbol{n}_b(\boldsymbol{\nu}) - \mathcal{M}_b\mathcal{W}_b^\dagger\left(\dot{\mathcal{W}}_b\boldsymbol{\nu} + \dot{\boldsymbol{a}} + \mathcal{P}_\Phi\boldsymbol{\Phi}\right) \tag{31}$$

in which $\boldsymbol{n}_b(\boldsymbol{\nu}) = \mathcal{C}_b(\boldsymbol{\nu})\boldsymbol{\nu} + \mathcal{D}_b(\boldsymbol{\nu})\boldsymbol{\nu}$. Corrolary 2 states the result formally.

Corollary 2 (2-D Path Maneuvering for fully actuated UUVs)**.** *Assume a fully actuated UUV, described by model*

(1), *structurally stable in roll and pitch. The control law* (31) *makes the UUV converge to the path with the commanded forward velocity* $u_d(t) > 0 \ \forall t > t_0$ *if: 1)* $v_d = 0$ *to have no sideways motion; 2) the desired heading is given by* (25)*; 3)* ϖ *is updated using* (22).

Proof. It follows from Theorem 1 that $\tilde{u} \triangleq u - u_d$, $\tilde{v} \triangleq v - v_d$, $\tilde{\psi} \triangleq \psi - \psi_d$, and $\boldsymbol{\varepsilon}$ are GES at the origin. Therefore, the UUV converges to the path particle exponentially fast while $u(t)$ converges to u_d with an exponential rate. □

To deepen our understanding of the control law, let us look into it by finding the MP inverse $\mathcal{W}_b^\dagger$:

$$\mathcal{W}_b^\dagger = \begin{bmatrix} 0.5\mathcal{I}_2 & 0_2 & 0.5\mathcal{R}(\gamma) \\ 0_2^T & 1 & 0_2^T \end{bmatrix} \in \mathbb{R}^{3\times 5} \tag{32}$$

where $\gamma \triangleq \psi - \psi_p = \psi_r + \tilde{\psi}$. It is recast as $\mathcal{W}_b^\dagger = \mathcal{Y} + \mathcal{Z}(\gamma)$ where $\mathcal{Y} = [\mathcal{I}_3, 0_{3\times 2}]$ and $\mathcal{Z}$ is found by inspection. The decomposition suggests splitting the control law up into two parts as

$$\boldsymbol{\tau}_p^b = \boldsymbol{\tau}_n + \boldsymbol{\tau}_\varepsilon \tag{33}$$

$\boldsymbol{\tau}_n$ eliminates nonlinearities of the system and acts like a feedback linearizing PD controller so that $\tilde{u}$, $\tilde{v}$, and $\tilde{\psi}$ become zero. Defining $\boldsymbol{h} \triangleq \dot{\mathcal{W}}_b \boldsymbol{\nu} + \dot{\boldsymbol{a}} + \mathcal{P}_\Phi \boldsymbol{\Phi}$, it is given by

$$\boldsymbol{\tau}_n = \boldsymbol{n}_b(\boldsymbol{\nu}) - \mathcal{M}_b \mathcal{Y} \boldsymbol{h} \tag{34}$$

$\boldsymbol{\tau}_\varepsilon = -\mathcal{M}_b \mathcal{Z}(\gamma) \boldsymbol{h}$ arises from the inclusion of the geometric error $\boldsymbol{\varepsilon}$ in the design procedure and holds

$$\ddot{e} + p_{e2}\dot{e} + p_{e1} e = 0 \tag{35}$$

to ensure that the craft exponentially converges to the path. Therefore, e, $\dot{e}$, and $\ddot{e}$ are globally bounded. Thus, global boundedness of $\dot{\psi}_r$ and $\ddot{\psi}_r$ is established because

$$\dot{\psi}_r = -\kappa \dot{e}, \qquad \ddot{\psi}_r = -\kappa \ddot{e} + 2\kappa^2 \dot{e}^2 e / \Delta \tag{36}$$

where $\kappa \triangleq \Delta/(\Delta^2 + e^2)$. An inspection of $\mathcal{Z}(\gamma)$ and $\boldsymbol{\tau}_\varepsilon$ reveals that: 1) as long as $\gamma \neq 0$, the first component of $\boldsymbol{\tau}_\varepsilon$ affects the surge dynamics. Thus, if the craft is off the path and/or $\tilde{\psi} \neq 0$, the controller alters the forward speed to reach the path faster. A similar effect is seen in the second component. The contribution of $\boldsymbol{\tau}_\varepsilon$ to the rudder command is multiplied by m_{23} and is not as major as the others; 2) at steady state when all conditions are satisfied, $\boldsymbol{\tau}_\varepsilon$ is null; 3) due to the special shape of $\mathcal{Z}(\gamma)$, the error system of $\tilde{\psi}$ is not affected by inclusion of $\boldsymbol{\varepsilon}$; i.e.

$$\dddot{\tilde{\psi}} + p_{\psi 2}\ddot{\tilde{\psi}} + p_{\psi 1}\dot{\tilde{\psi}} = 0 \tag{37}$$

while the error systems of $\tilde{u}$ and $\tilde{v}$ get affected with that. To put is clearly, inclusion of $\boldsymbol{\varepsilon}$ in the design mainly results in manipulation of the forward (and sway) velocity in order for fast convergence to the path when the craft does not move on the desired path. On the negative side, it might cause large commands which exceed saturation levels if $\boldsymbol{\varepsilon}$ is large, or it may lead to reverse thrust.

4.1. Extension to Underactuated UUVs

Assuming fully actuated marine craft is too idealistic, as a great number of UUVs are unactuated in sway. Moreover, at high speeds, sway thrusters become inefficient even if they are mounted on the craft. Hence, for the planar motion of UUVs, we will rely only on the forward force (τ_u) and the turning moment (τ_r), and it is assumed that $\tau_v = 0$. In this regard, underactuation places a constraint on the resulted control law. In fact, since there is no instantaneous control action in sway, it is impossible to assign arbitrary values to $v_d(t)$ and it must be adapted in conformity with the system dynamics and the designed controller. Thus, setting $\tau_v = 0$, a differential equation for v_d comes out. It is given by

$$\tfrac{1}{2} m_{22} \dot{v}_d = -d_{22} v_d + g(\boldsymbol{\zeta}, t) + u_{v_d}(t) \tag{38}$$

in which $\boldsymbol{\zeta} \triangleq [\tilde{u}, \tilde{v}, \dot{\tilde{\psi}}, \tilde{\psi}, \dot{\boldsymbol{\varepsilon}}^T, \boldsymbol{\varepsilon}^T]^T$, $g(\boldsymbol{\zeta}, t)$ and u_{v_d} are nonlinear terms which are given in Appendix A. The result for underactuated UUVs is formalized in Corollary 3.

Corollary 3 (2-D Path Maneuvering for 3-DOF Underactuated UUVs)**.** *Consider a sway-unactuated UUV, described by model* (1) *with the system matrices* (20)*, structurally stable in roll and pitch. The first and the third elements of* (31) *make the UUV converge to the path with the desired velocity* $u_d(t) > 0 \ \forall t > t_0$ *if: 1)* v_d *is found by integration of* (38)*; 2) the desired heading is given by* (25)*; 3)* ϖ *is updated using* (22)*. It is ensured that* $v_d \in \mathcal{L}_\infty$ *and the control signals are well-defined.*

Proof. It follows from Corollary 2 that $\boldsymbol{\zeta}$ is UGAS/ULES at the origin; so the path maneuvering objectives are accomplished as specified. The concern is about the unactuated dynamics, which prescribes a strict constraint on the derived control law so that $\tau_v = 0$. Taking v_d as a DOF to cope with this restriction, (38) is obtained to adjust v_d properly to satisfy the condition. Since $\lim_{t\to 0} \tilde{v} = 0$, and τ_u and τ_r depend on v_d, the characteristics of v_d is vital for boundedness of the unactuated dynamics and internal stability of the closed-loop system. Because the UUV is open-loop stable in sway ($d_{22} > 0$), the system $\frac{1}{2} m_{22} \dot{v}_d = -d_{22} v$ is UGES. It follows from Lemma 4.6 in [26] that the system (38) is input-to-state stable from the input $r_{v_d} = g(\boldsymbol{\zeta}, t) + u_{v_d}(t)$ to the state $v_d(t)$ since it is globally Lipschitz in (v_d, r_{v_d}). The input $r_{v_d} \in \mathcal{L}_\infty$ since $\boldsymbol{\zeta}, u_d \in \mathcal{L}_\infty$, and the path is smooth. It is inferred that if the path is straight, $v(t)$ converges to zero as $t \to \infty$. □

Now, it is conspicuous why $v - v_d$ is included in the constraint set (26). Indeed, the approach can employ the idea proposed in [32] to have a smooth transition from fully actuation at low speeds to underactuation at high speeds, and the other way around.

Remark 4. For marine craft unactuated in sway, τ_v is not identically zero since rudder deflection δ produces a sway force, denoted by $\tau_{0v}(\delta)$, besides a yaw moment. In [33–35], a coordinate transformation is employed for removing

the contribution of the rudder deflection from the sway dynamics. For the proposed method, there is no need for such transformations, and it suffices to modify the differential equation for v_d to include the contribution of the rudder. In fact, the constraint arising from underactuation bcomes $\tau_{0v}(\delta) + \tau^b_{p,v} = 0$, in which $\tau^b_{p,v}$ is the second component of the control law in (31).

4.2. Dealing with Ocean Currents

It is assumed that the underactuated UUV is subject to irrotational constant ocean currents. An inherent drawback of LOS guidance-based path-following controllers is that they are fragile in the presence of disturbances and steady-state errors are inevitable, as the LOS guidance system provides a wrong heading angle. The explanation can be given as follows: in order to counteract the component of currents perpendicular to the path, UUVs rotate to create a sideslip angle (β) so as to provide a component of the forward velocity against currents. It implies that the heading angle of the AUV cannot be equal to the slope of the path tangential; i.e. the relative angle ψ_r can never be zero. Equivalently, e cannot be zero.

One remedy is to align the total speed, V, along the line of sight between the UUV and the desired point on the path tangential. However, for underactuated UUVs, the total speed is an uncontrollable value. In [19], the following control law is proposed

$$\tau_r = \tau_{n,r} - k_d\dot{e} - k_p e - k_i \int_0^t e(\varrho)d\varrho \tag{39}$$

where $\tau_{n,r}$ is appropriately designed. Since the cross-track error linearly affects the yaw moment, saturation may easily come up. Alternatively, the integral LOS guidance strategy is put forward in [34, 36] as

$$\psi_r = \tan^{-1}\left(\frac{e+\sigma\int_0^t e_{int}(\varrho)d\varrho}{\Delta}\right) \tag{40}$$

in which the relative angle ψ_r is modified to allow the craft to have a sideslip angle while $e = 0$. In fact, the method aims to correct the desired heading angle ψ_d. Akin to the previous method, this method employs the yaw moment to reject disturbances. As the integral of e influences through $\tan^{-1}$ function, the amplitude of e does not have that much effect on the control signal; however, large e or improper choice of σ may easily lead to poor performance.

For the proposed method, augmentation of integral action can robustify the controller to constant or slowly varying ocean currents. A PID-type controller is only needed for the geometric error $\boldsymbol{\varepsilon}$, which converts (35) to

$$\ddot{e} + p_{e2}\dot{e} + p_{e1}e + p_{e0}\int_0^t e(\varrho)d\varrho = 0 \tag{41}$$

It does not change the results of Corollaries 2 and 3. Adding integral action to the $\boldsymbol{\varepsilon}$-subsystem adjusts the forward velocity so that the UUV catches the path. If the exact dynamic task is demanded, integral action can be augmented

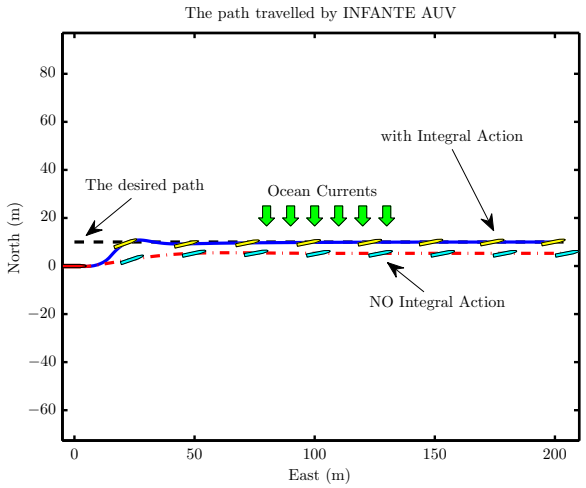

Figure 4: The paths travelled by the AUV. Performance in the presence of ocean currents. $\Delta = 5m$ is equal to the length of the AUV.

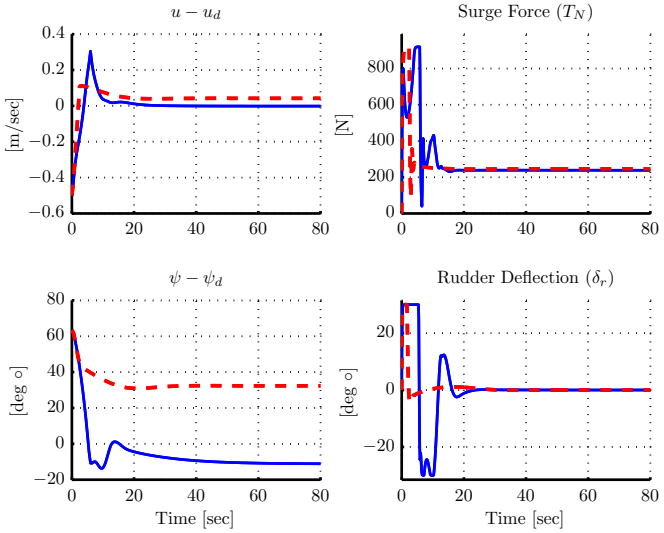

Figure 5: u and $\tilde{\psi}$ on the left, and the control signals on the right. The solid blue lines belong to the controller with integral action and the dash red lines are for the controller without integral action.

to the $\tilde{u}$-dynamics but care must be taken to avoid windup. Note that there is no need for a PID-type controller for the heading loop because underactuated UUVs have to rotate and have a sideslip angle; thus, $\tilde{\psi}$ cannot be zero if $e = 0$. We would like to draw the reader's attention to the point that decreasing Δ might be one solution to *reduce* the steady-state error as increasing the gains of the controller associated with $\boldsymbol{\varepsilon}$ dynamics can be an alternative. The former manipulates τ_r to redirect the vehicle and may lead to aggressive behavior. The latter exploits τ_u and modifies the velocity to attenuate disturbance effects.

4.3. Simulation Results

Simulation 1. The nonlinear model of INFANTE AUV (Fig. 1) described in [37] is used for simulating a realistic environment while a simplified model according to (20) is considered for control design. The INFANTE AUV is unactuated in sway. The maximum propeller thrust is 920N which makes the AUV move at a speed of 5m/s in calm

water. The rudder deflection saturates at 30°. Actuator dynamics is approximated by a first-order transfer function. The path is described by $\boldsymbol{p}_p(\varrho) = [10, \varrho]^T$. The initial conditions are $\boldsymbol{q}_0 = [0, 0, \pi/2]^T$ and $\boldsymbol{\nu}_0 = [2\text{m/s}, 0, 0]^T$. Ocean currents with a speed vector $\dot{\boldsymbol{q}}_c = [0, 0.5\text{m/s}, 0]^T$ are considered. The results are shown in Figs. 4 and 5.

A comparison is drawn between the case integral action is included and the case it is not. The parameters of the two controllers are kept identical to have comparable results. It is inferred from the results that the path-maneuvering objectives, $\tilde{u} = 0$ and $\boldsymbol{\varepsilon} = 0$, are achieved when integral action is augmented to the controller.

Simulation 2. For this experiment, the ODIN AUV [38] is used to extend the controller to 3-D case. The AUV is spherical and equipped with four horizontal thrusters and four vertical ones; the AUV is underactuated and neutrally buoyant. The model parameters and the thruster configuration matrix are taken from [39]. Each thruster, whose dynamics is approximated by a first-order model, can provide a maximum thrust of 27N. The AUV is controlled separately in heave with a decoupled depth controller.

The ODIN AUV is asked to set out on a voyage from rest when it is located at $\boldsymbol{p}_0 = [0, 0, -102]^T$ while $\boldsymbol{\tau}_{dist} = [0, 5, 0]^T$. The given path is a helix with variable curvature, defined by $\boldsymbol{p}_p(\varrho) = [10\sin(\varrho/5), 8\cos(\varrho/5), 0.1\varrho - 100]^T$. The results are displayed in Figs. 6 and 7.

Fig. 6 shows the desired path and the traveled path in a 3-D plot, indicating that the controller performance is satisfactory. Path-following trajectory in the horizontal plane and the distance error components are plotted in Fig. 7, providing a better insight into the way the controller performs.

The constraint $\psi - \psi_d$ with ψ_d given by (25) means that the x-axis of $\{b\}$ must be ideally tangent to the path. However, when the path's curvature is nonzero and the UUV is underactuated, the sway velocity differs from zero even at the absence of external disturbances. Thus, the total speed is not aligned with the x-axis of $\{b\}$, and a sideslip angle exists. Because the path that an UUV makes is constructed by the total speed, geometric errors (see Fig. 7) will be nonzero. On the other hand, when the LOS guidance method is used to find the desired heading, the UUV is always guided towards a point on the path tangential that is perhaps not on the path when the curvature is different from zero (see Fig. 3). In other words, in curved path following with the LOS guidance system, the path is approximated to straight segments at each instant; this can be seen as an inherent deficiency of the method. Decreasing Δ would be suggested to reduce errors; however, it might cause aggressive behavior. Another remedy is to modify the desired heading and compensate for the sideslip angle as $\bar{\psi}_d(t) = \psi_p(t) + \psi_r(t) - \beta$. Assuming β is known, $\bar{\psi}_d$ forces the total speed to be aligned with the LOS. Nonetheless, the controller requires $\dot{\beta}$ and $\ddot{\beta}$, which makes it difficult to implement. Alternatively, the proposed method where $\boldsymbol{\varepsilon}$ is explicitly included in the

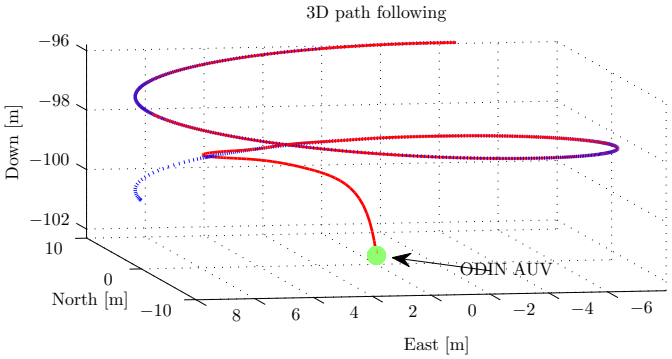

Figure 6: 3D representation of the reference path (blue dash-dot line) and the travelled path by ODIN (red solid).

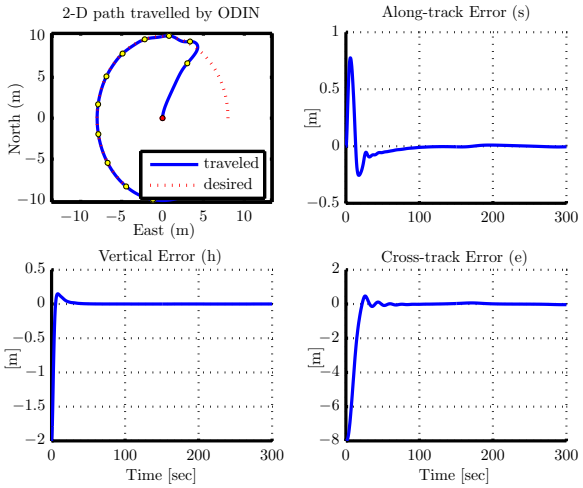

Figure 7: 2-D representation of the path for the first 200s is depicted. Along-track, cross-track, and vertical errors are also shown.

controller parallel with integral action can lower errors significantly.

5. Concluding Remarks

A methodology for motion control of marine craft is proposed. Inspired by analytical mechanics, we establish a link between modeling and simulation of constrained multi-body systems in one hand and motion control of underwater vehicles, on the other hand. The beauty of the presented approach consists in simplicity and physical interpretability, where imaginary dampers and springs introduce a deformable body and take care of motion of the vehicle. Moreover, a novel path-maneuvering controller, which gives precedence to the geometric task, is designed. Consequently, the vehicle converges to the path faster, and the drawback of the traditional LOS guidance strategy in the presence of ocean currents is resolved.

Appendix A.

The time-varying function υ_{v_d} is

$$\upsilon_{v_d}(t) = (m_{11}u_d + d_{23})\,\dot{\psi}_p + m_{23}\ddot{\psi}_p$$

which is $\mathcal{L}_\infty$ by assumption. $\mathcal{A}^{\{i,j\}}$ denotes (i,j) entry of the matrix $\mathcal{A}$. Thus, $p_{ei} \triangleq \mathcal{P}_{\varepsilon i}^{\{2,2\}}$ for $i = 1, 2$. The following notations are used:

$$\mathcal{P}_G = \text{diag}\{p_{G1}, \mathcal{P}_{G\varepsilon}\} \qquad \mathcal{P}_{G\varepsilon} = \text{diag}\{p_{G2}, p_{G3}\}$$
$$\mathcal{P}_\Phi = \text{diag}\{p_{\Phi 1}, p_{\Phi 2}, p_{\Phi 3}, \mathcal{P}_{\Phi\varepsilon}\} \qquad \mathcal{P}_{\Phi\varepsilon} = \text{diag}\{p_{\Phi 4}, p_{\Phi 5}\}$$
$$\mathcal{P}_{\varepsilon 2} = \mathcal{P}_{G\varepsilon} + \mathcal{P}_{\Phi\varepsilon} \qquad \mathcal{P}_{\varepsilon 1} = \mathcal{P}_{G\varepsilon}\mathcal{P}_{\Phi\varepsilon}$$
$$p_{\psi 2} = p_{G1} + p_{\Phi 3} \qquad p_{\psi 1} = p_{G1}p_{\Phi 3}$$
$$\mathcal{H}_{\varepsilon 2} \triangleq \tfrac{1}{2}\mathcal{R}(\gamma)\left(\mathcal{S}(\gamma)^T + \mathcal{P}_{\varepsilon 2}\right), \mathcal{H}_{\varepsilon 1} \triangleq \tfrac{1}{2}\mathcal{R}(\gamma)\left(\mathcal{S}(\dot\gamma)^T + \mathcal{P}_{\varepsilon 1}\right)$$

The coupling term is formulated as

$$g(\boldsymbol{\zeta}, t) = \boldsymbol{W}(\boldsymbol{\zeta}, t)^T \boldsymbol{\zeta} + 0.5 m_{22} \sin(\gamma)\dot u_d$$

where $\boldsymbol{W}$ is equal to

$$\begin{bmatrix} m_{11}(\dot{\tilde\psi} + \dot\psi_p - \kappa\dot e) \\ d_{22} - 0.5 m_{22} p_{\Phi 2} \\ d_{23} + m_{11}u_d - m_{23}p_{\psi 2} + 0.5 m_{22}(\tilde u + u_d) \\ -m_{23}p_{\psi 1} \\ -\kappa m_{11}u_d + m_{23}\kappa p_{e2}\dot e - m_{22}(\mathcal{H}_{\varepsilon 1}^{\{2,2\}} + 0.5\kappa u) - \kappa d_{23} \\ 2m_{23}\kappa^2\dot e^2/\Delta + \kappa m_{23}p_{e1} - m_{22}\mathcal{H}_{\varepsilon 1}^{\{2,2\}} \\ -m_{22}(\mathcal{H}_{\varepsilon 2}^{\{2,1\}} - 0.5\sin(\gamma)\sigma) \\ -m_{22}\mathcal{H}_{\varepsilon 1}^{\{2,1\}} \end{bmatrix}$$

References

[1] G. Antonelli, Underwater Robots. Motion and Force Control of Vehicle-Manipulator Systems, Springer, 2nd edition, 2006.

[2] L. A. Pars, A Treatise on Analytical Dynamics, Heinemann, London, 1965.

[3] C. Lanczos, The variational principles of mechanics, Dover Publications, New York, fourth edition, 1986.

[4] I. A. F. Ihle, J. Jouffroy, T. I. Fossen, Formation control of marine surface craft: A lagrangian approach, Oceanic Engineering, IEEE J. 31 (2006) 922–934.

[5] I. A. F. Ihle, J. Jouffroy, T. I. Fossen, Robust formation control of marine craft using lagrange multipliers, in: K. Pettersen, J. Gravdahl, H. Nijmeijer (Eds.), Group Coordination and Cooperative Control, Springer, 2006, pp. 113–129.

[6] J. Hauser, R. Hindman, Aggressive flight maneuvers, in: CDC, 1997, Proc. of IEEE Conf., volume 5, pp. 4186–4191 vol.5.

[7] R. Skjetne, T. I. Fossen, P. V. Kokotovic, Robust output maneuvering for a class of nonlinear systems, Automatica 40 (2004) 373–383.

[8] A. P. Aguiar, J. P. Hespanha, P. V. Kokotovic, Performance limitations in reference tracking and path following for nonlinear systems, Automatica 44 (2008) 598–610.

[9] M. T. Mason, Compliance and force control for computer controlled manipulators, Systems, Man and Cybernetics, IEEE Trans. on 11 (1981) 418–432.

[10] H. Bruyninckx, J. De Schutter, Specification of force-controlled actions in the task frame formalism-a synthesis, Robo. & Auto., IEEE Trans. on 12 (1996) 581–589.

[11] B. Siciliano, L. Villani, Robot Force Control, volume 540 of *The Springer Inter. Series in Eng. & Comp. Sci.*, Springer, 2000.

[12] A. S. Shiriaev, L. B. Freidovich, S. V. Gusev, Transverse linearization for controlled mechanical systems with several passive dof, Auto. Cont., IEEE Trans. on 55 (2010) 893–906.

[13] J. Xing, Y. Lei, W. Cheng, G. Tang, Nonlinear control of multiple spacecraft formation flying using the constraint forces in lagrangian systems, Sci. in China 52 (2009) 2930–2936.

[14] F. E. Udwadia, R. E. Kalaba, A new perspective on constrained motion, Proc.: Math. & Phys. Sci. 439 (1992) pp. 407–410.

[15] F. E. Udwadia, A new perspective on the tracking control of nonlinear structural and mechanical systems, Royal Society of London. Math., Phys. & Eng. Sci. 459 (2003) 1783–1800.

[16] J. Peters, M. Mistry, F. Udwadia, J. Nakanishi, S. Schaal, A unifying framework for robot control with redundant dofs, Autonomous Robots 24 (2008) 1–12.

[17] F. E. Udwadia, B. Han, Synchronization of multiple chaotic gyroscopes using the fundamental equation of mechanics, Journal of Applied Mechanics 75 (2008) 021011.

[18] H. Cho, F. E. Udwadia, Explicit solution to the full nonlinear problem for satellite formation-keeping, Acta Astronautica 67 (2010) 369–387.

[19] T. I. Fossen, Guidance and control of ocean vehicles, Wiley, 1994.

[20] T. I. Fossen, Handbook of marine craft, hydrodynamics, and motion control, John Wiley & Sons Ltd., 2011.

[21] B. Siciliano, L. Sciavicco, L. Villani, G. Oriolo, Robotics, Springer, 2009.

[22] A. M. Bloch, M. Reyhanoglu, N. H. McClamroch, Control and stabilization of nonholonomic dynamic systems, Automatic Control, IEEE Trans. on 37 (1992) 1746–1757.

[23] M. R. Flannery, The enigma of nonholonomic constraints, American J. of Physics 73 (2005) 265–272.

[24] J. W. Baumgarte, A new method of stabilization for holonomic constraints, J. of Applied Mechanics 50 (1983) 869–870.

[25] R. A. Horn, C. R. Johnson, Matrix analysis, Cambridge University Press, 1985.

[26] H. K. Khalil, Nonlinear systems, Prentice Hall, 3rd ed., 2002.

[27] M. Krstic, P. Kokotovic, I. Kanellakopoulos, Nonlinear and adaptive control design, John Wiley & Sons, 1995.

[28] J. Slotine, W. Li, Applied nonlinear control, Prentice Hall, 1991.

[29] F. E. Udwadia, R. E. Kalaba, Analytical Dynamics: A New Approach, Cambridge University Press, 1996.

[30] M. Breivik, T. I. Fossen, Guidance laws for autonomous underwater vehicles, in: A. V. Inzartsev (Ed.), Underwater Vehicles, InTech, 2009.

[31] R. Skjetne, T. I. Fossen, P. V. Kokotovic, Adaptive maneuvering, with experiments, for a model ship in a marine control laboratory, Automatica 41 (2005) 289–298.

[32] M. Breivik, T. I. Fossen, A unified control concept for autonomous underwater vehicles, in: ACC, 2006, p. 7 pp.

[33] E. Fredriksen, K. Pettersen, Global [kappa]-exponential waypoint maneuvering of ships: Theory and experiments, Automatica 42 (2006) 677 – 687.

[34] E. Børhaug, Nonlinear Control and Synchronization of Mechanical Systems, Phd thesis, Dep. of Eng. Cybernetics, Norwegian University of Sci. & Tech., Trondhiem, Norway, 2008.

[35] K. D. Do, J. Pan, Global tracking control of underactuated ships with nonzero off-diagonal terms in their system matrices, Automatica 41 (2005) 87–95.

[36] E. Børhaug, A. Pavlov, K. Y. Pettersen, Integral los control for path following of underactuated marine surface vessels in the presence of constant ocean currents, in: CDC 2008, pp. 4984–4991.

[37] C. J. F. Silvestre, Multi-objective optimization theory with applications to the integrated design of controllers/plants for autonomous vehicles, Ph.D. thesis, Inst. Superior Tecnico, 2000.

[38] S. K. Choi, J. Yuh, G. Y. Takashige, Development of the omni directional intelligent navigator, Robo. & Auto. Mag., IEEE 2 (1995) 44–53.

[39] G. Antonelli, T. I. Fossen, D. R. Yoerger, Underwater robotics, in: B. Siciliano, O. Khatib (Eds.), Springer Handbook of Robotics, Springer, 2008, pp. 987–1008.