

Trajectory Tracking for Underwater Swimming Manipulators using a Super Twisting Algorithm

Ida Louise G. Borlaug, Jan Tommy Gravdahl, Jørgen Sverdrup-thygeson, Kristin Pettersen, Antonio Loria

▶ **To cite this version:**

Ida Louise G. Borlaug, Jan Tommy Gravdahl, Jørgen Sverdrup-thygeson, Kristin Pettersen, Antonio Loria. Trajectory Tracking for Underwater Swimming Manipulators using a Super Twisting Algorithm. Asian Journal of Control, 2019, SMC based observation, identification, uncertainties compensation and fault detection, 21 (1), pp.208-223. 10.1002/asjc.1840 . hal-02367624

HAL Id: hal-02367624
https://hal.science/hal-02367624

Submitted on 5 Mar 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Asian Journal of Control, Vol. 00, No. 0, pp. 1–17, Month 0000

Published online in Wiley InterScience (www.interscience.wiley.com) DOI: 10.1002/asjc.0000

TRAJECTORY TRACKING FOR UNDERWATER SWIMMING MANIPULATOR USING A SUPER TWISTING ALGORITHM

I.-L. G. Borlaug[1] J. T. Gravdahl[2] J. Sverdrup-Thygeson[1], K.Y. Pettersen[1] and A. Loria[3]

ABSTRACT

The underwater swimming manipulator (USM) is a snake-like, multi-articulated, underwater robot equipped with thrusters. One of the main purposes of the USM is to act like an underwater floating base manipulator. As such, it is essential to achieve good station-keeping and trajectory tracking performance for the USM using the thrusters, while using the joints to attain a desired position and orientation of the head and tail of the USM. In this paper, we propose a sliding mode control (SMC) law, in particular the super-twisting algorithm with adaptive gains, for trajectory tracking of the USMs center of mass. A higher-order sliding mode observer is proposed for state estimation. Furthermore, we show ultimate boundedness of the tracking errors and perform a simulation study to verify the applicability of the proposed control law and show that it has better tracking performance than a linear PD-controller.

Key Words: Underwater Swimming Manipulator, Super-Twisting, Siding Mode Control, Sliding Mode Observer.

I. Introduction

An underwater swimming manipulator (USM) is an underwater snake robot (USR) equipped with

Manuscript received Decemeber 2017

[1] Centre for Autonomous Marine Operations and Systems, Department of Engineering Cybernetics, NTNU, Norwegian University of Science and Technology, Trondheim, Norway (E-mail: {Ida.Louise.Borlaug, Jorgen.Sverdrup-Thygeson, Kristin.Y.Pettersen}@ntnu.no

[2] Department of Engineering Cybernetics, NTNU, Norwegian University of Science and Technology, Trondheim, Norway (E-mail: Tommy.Gravdahl@ntnu.no)

[3] The National Center for Scientific Research (CNRS), France (E-mail: loria@lss.supelec.fr)

This research was funded by the Research Council of Norway through the Centres of Excellence funding scheme, project No. 223254 NTNU AMOS, and by VISTA, a basic research program in collaboration between The Norwegian Academy of Science and Letters, and Statoil.

© 0000 John Wiley and Sons Asia Pte Ltd and Chinese Automatic Control Society

thrusters [1]. The main purposes of the thrusters are to provide forward thrust without requiring the snake robot to follow an undulating gait pattern, which is of particular importance in narrow, confined environments, and to provide sideways thrust for station-keeping and trajectory tracking. The station-keeping and trajectory tracking capabilities enable the USM to act like an underwater floating base manipulator. The slender, multi-articulated body provides the USM with outstanding accessibility and flexibility. As such, the USM is a crossover between a small autonomous underwater vehicle (AUV) and an underwater snake robot. The USM possesses the high kinematic redundancy of the USR, and at the same time it has the advantages of the AUV in terms of full energy-efficient hydrodynamic properties and tether-less operation. Moreover, the USM has the advantages of remotely operated vehicles (ROVs) regarding full actuation and the capability of doing intervention operations. Since the USM can use the thrusters instead of the joints to create forward propulsion, the joints can be used to perform manipulation tasks and, thus, exploit the full potential of the inherent kinematic redundancy. This has been addressed in detail in [2], [3].

As a floating base manipulator, the USM can move itself to an area of interest, position its tail at the initial base location, and then start to operate as a robotic manipulator. When the USM carries out a manipulation task, the overall motion of the USM and the joint angle velocities can be determined by the desired velocities of the end-effector, i.e. the desired motion of the head of the USM. One approach for this is described in [4], where the base motion and the joint angle motion of the USM are assigned using a redundancy resolution technique based on inverse kinematics. The outputs of this procedure are time-varying velocity references for the base and the joints. This inverse kinematics method is only one of many ways to calculate the velocity references.

Controller design for underwater robots (URs) such as the USM and ROVs, is a complex problem [5]. URs are often subject to hydrodynamic and hydrostatic parameter uncertainties, uncertain thruster characteristics, unknown disturbances, and unmodelled dynamic effects, e.g. thruster dynamics and coupling forces caused by joint motion. As the USM has no separate vehicle base and a low mass compared to an ROV, the motion of the joints become more significant for the overall motion of the USM as a rigid body than for the ROV. The coupling forces are therefore more prominent for the USM, and this increases the complexity of motion control of the USM, compared to an ROV.

Sliding mode control (SMC) is a robust and versatile non-linear control approach, and we will in this paper show that it is well suited for control of USMs. For underwater vehicles, in general, some important contributions are given in [6], [7], [8], [9], [10] and [11]. In [6], a singularity-free SMC approach, inspired by [12], is used for set-point regulation of a UR with uncertainties in the hydrodynamic parameters. In [7], [8], SMC is employed to cope with multiplicative

© 0000 John Wiley and Sons Asia Pte Ltd and Chinese Automatic Control Society

uncertainty in the thruster configuration matrix. The combination of sliding mode and adaptive control is studied in [7], [8], [11]. In particular, in [11], sliding mode control is combined with adaptive PID controller gains and an adaptive update of the upper bound on the disturbances and the parameter uncertainties. SMC is also applicable to deal with linearisation errors [9] and the coupling effects between an underwater vehicle and an attached manipulator arm [10]. Sliding mode techniques have been applied to land-based snake robots in [13] to achieve robust tracking of a desired gait pattern and under-actuated straight line path following. However, SMC have to the authors' best knowledge, never been applied to underwater snake robots, and in particular, to USRs with thrusters.

In this paper SMC is applied to the robot model proposed in [14], for which the robot is equipped with thrusters as in [1]. The model in [1] extends the 2D model proposed in [15], by modelling also additional effectors and considering the force allocation among these effectors. In [14] the model from [15] used in [1] has been revised and extended, and we here use the revised model. In [1] a linear PD-controller was used for tracking of the position and heading along the reference path. In this paper, we propose to replace the PD-controller with a super-twisting algorithm (STA) for sliding mode control accompanied by a higher-order sliding mode observer, for the case when only the position measurements are available. We consider the tracking problem for the position of the centre of mass of the USM.

The first-order relay controller [16], has large problems with chattering. To eliminate chattering, we could have used saturation control, but since the sliding mode does not exist inside the boundary layer, the effectiveness of the controller is challenged when parasitic dynamics are considered, [17]. Therefore the super-twisting algorithm will be used. The STA is one of the most powerful second-order continuous sliding mode control algorithms. It was first introduced in [18] and has thereafter been used for multiple applications. The STA attenuates chattering and will, therefore, give a smoother control signal. A challenge with the STA is that it only works with bounded perturbations, and therefore a conservative upper bound has to be used when designing the controller to ensure that sliding is maintained. To avoid this issue, we will use adaptive STA [19]. The gains can then adapt to a level where they are as small as possible but still guarantee that sliding is maintained. Since the STA is only applicable to systems where the control input appears in the equation for the first derivative of the sliding variable, both the position and velocity of the USM need to be available for measurement. For the case when only the position measurements are available, we will use a higher-order sliding mode observer, proposed in [20], to estimate the states. As such, we combine the results from [19] and [20], as done in [21], but we will replace the regular STA with a STA with adaptive gains. We will then apply this control structure to the USM and show ultimate boundedness of the tracking errors. We also present simulations that verify that the proposed

© 0000 John Wiley and Sons Asia Pte Ltd and Chinese Automatic Control Society

approach applies well to USMs, and also compare the results to a standard PD-controller.

The remainder of this paper is organized as follows. In Section II the robot model used will be explained in more detail. The control and observer design is presented in Section III, and in Section IV we prove boundedness of the tracking errors. In Section V the simulation results will be presented. Conclusions and suggestions for future work are given in Section VI.

II. Underwater Swimming Manipulator (USM) Model

In this section, the equations of motion for the USM and the force allocation matrix will be explained. We refer to [1], [14] and [15] for further details.

2.1. Kinematics

The position of the center of mass (CM) of the USM, $p_{CM} \in R^2$, expressed in the global frame is

$$p_{CM} = \begin{bmatrix} p_x \\ p_y \end{bmatrix} = \begin{bmatrix} \frac{1}{m_t}\sum_{i=1}^{n} m_i x_i \\ \frac{1}{m_t}\sum_{i=1}^{n} m_i y_i \end{bmatrix} = \frac{1}{m_t}\begin{bmatrix} e^T M X \\ e^T M Y \end{bmatrix} \tag{1}$$

where (x_i, y_i), $i = 1, \cdots, n$ are the coordinates of the CM of link i in the global frame, m_i is the mass of link i and $m_t = \sum_{i=1}^{n} m_i$ is the total mass of the USM. Eq. (1) is valid because it is assumed that the mass of each link is uniformly distributed. The matrix representation of the force balance for all the links is

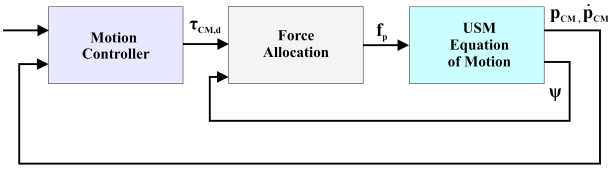


Fig. 1. System overview USM, [1]

$$M\ddot{X} = D^T h_x + f_x + f_{px}, \quad M\ddot{Y} = D^T h_y + f_y + f_{py} \tag{2}$$

where f_{px} and f_{py} are the forces from the additional effectors, h_x and h_y are the joint constraint forces and f_x and f_y are the fluid forces acting on the links. By differentiating Eq. (1) and inserting Eq. (2), the joint constraint forces cancel out, and the translational motion of the CM of the USM can be written as

$$m_t \ddot{p}_x = e^T(f_x + f_{px}), \quad m_t \ddot{p}_y = e^T(f_y + f_{py}). \tag{3}$$

2.2. Force Allocation

The force allocation distribution is given by

$$\begin{aligned} \tau_{CM} &= \begin{bmatrix} F_{CM,x} \\ F_{CM,y} \\ M_{CM,z} \end{bmatrix} \\ &= \begin{bmatrix} e^T & 0^{1\times n} \\ 0^{1\times n} & e^T \\ e^T S_\psi K & -e^T C_\psi K \end{bmatrix} \begin{bmatrix} f_{px} \\ f_{py} \end{bmatrix} = T(\psi) f_p, \end{aligned} \tag{4}$$

where $T(\psi)$ is the allocation matrix and $f_p = [f_{p,k_1}, \ldots, f_{p,k_r}]$ is the vector of scalar effector forces. The allocation matrix represents the mapping between the effector forces and the forces and moments acting on the CM of the USM. It is assumed that the additional effector forces are acting through the CM of each link.

© 0000 John Wiley and Sons Asia Pte Ltd and Chinese Automatic Control Society

The primary objective for the force allocation method is to distribute the efforts among the additional effectors to obtain the desired forces and moments. In the next section, we propose a novel method for calculation of the desired forces and moments, together with a non-linear observer for position and velocity.

III. Control and observer design

Control problem: *Assume that there exist a guidance system which determines a suitable path for the USM to follow. The task at hand is to design a motion controller that calculates the desired forces for the translational motion F_{CM}, and the desired moments for the rotational motion M_{CM}, of the USM.*

We will in the following use a super-twisting algorithm with adaptive gains to calculate the desired forces, F_{CM}. To calculate the desired moments, M_{CM}, we will use a proportional controller. The desired forces and moments are represented by

$$\tau_{\mathrm{CM},d} = \begin{bmatrix} F_{\mathrm{CM},d} \\ M_{\mathrm{CM},d} \end{bmatrix} = \begin{bmatrix} F_{\mathrm{CM},d_x} \\ F_{\mathrm{CM},d_y} \\ M_{\mathrm{CM},d} \end{bmatrix} \tag{5}$$

The control input for the translational motion is therefore $F_{\mathrm{CM,d}}$, which is the desired force we want to impose on the system. This force will be given as input to the force allocation matrix in Eq. (4), which will then distribute the forces on the effectors so that the combined force in x- and y-direction is equal to the desired forces in x- and y-direction, i.e. $F_{\mathrm{CM,d_x}}$ and $F_{\mathrm{CM,d_y}}$, respectively. By assuming that the actuator dynamics is faster than the system dynamics, the following equation is assumed to be satisfied.

$$F_{\mathrm{CM,d}} = \begin{bmatrix} F_{\mathrm{CM,d_x}} \\ F_{\mathrm{CM,d_y}} \end{bmatrix} = F_{CM} = \begin{bmatrix} F_{\mathrm{CM},x} \\ F_{\mathrm{CM},y} \end{bmatrix} = \begin{bmatrix} e^T f_{px} \\ e^T f_{py} \end{bmatrix}. \tag{6}$$

By replacing $e^T f_{px}$ and $e^T f_{py}$ in Eq. (3), with $F_{\mathrm{CM,d_y}}$ and $F_{\mathrm{CM,d_y}}$, the translational motion of the CM of the USM can be rewritten as

$$m_t \ddot{p}_x = e^T f_x + F_{\mathrm{CM,d_x}}, \quad m_t \ddot{p}_y = e^T f_y + F_{\mathrm{CM,d_y}}. \tag{7}$$

3.1. Sliding surface design

First we define the error variable. As the output variable for the translational motion of the USM is p_{CM}, the error variable can be defined as

$$\tilde{p} = \begin{bmatrix} \tilde{p}_x \\ \tilde{p}_y \end{bmatrix} = p_{\mathrm{CM}} - p_{\mathrm{CM,ref}} = \begin{bmatrix} p_x - p_{x,\mathrm{ref}} \\ p_y - p_{y,\mathrm{ref}} \end{bmatrix} \tag{8}$$

where $p_{\mathrm{CM,ref}}$ is the desired position of the CM of the USM in the global frame. The sliding surface should be selected such that the state trajectories of the controlled system are forced onto the sliding surface $\sigma = \dot{\sigma} = 0$, where the system behaviour meets the design specifications. The controller $F_{\mathrm{CM,d}}$ should also appear in the first derivative of σ, so that the relative degree is equal to 1. The sliding surface σ can then be

© 0000 John Wiley and Sons Asia Pte Ltd and Chinese Automatic Control Society

chosen as

$$\begin{aligned}\sigma = \begin{bmatrix}\sigma_x \\ \sigma_y\end{bmatrix} &= \tilde{p} + \lambda\dot{\tilde{p}} = \begin{bmatrix}\tilde{p}_x \\ \tilde{p}_y\end{bmatrix} + \begin{bmatrix}\lambda\dot{\tilde{p}}_x \\ \lambda\dot{\tilde{p}}_y\end{bmatrix} \\ &= \begin{bmatrix}p_x - p_{x,\mathrm{ref}} \\ p_y - p_{y,\mathrm{ref}}\end{bmatrix} + \begin{bmatrix}\lambda(\dot{p}_x - \dot{p}_{x,\mathrm{ref}}) \\ \lambda(\dot{p}_y - \dot{p}_{y,\mathrm{ref}})\end{bmatrix}\end{aligned} \tag{9}$$

Since only the position, p_{CM}, of the centre of mass is available for measurement, an observer for the states is designed. The observer states will be used in the sliding surface, and following the structure of Eq. (9), the revised sliding surface is then

$$\hat{\sigma} = \begin{bmatrix}\hat{\sigma}_x \\ \hat{\sigma}_y\end{bmatrix} = \begin{bmatrix}\hat{p}_x - p_{x,\mathrm{ref}} \\ \hat{p}_y - p_{y,\mathrm{ref}}\end{bmatrix} + \begin{bmatrix}\lambda(\dot{\hat{p}}_x - \dot{p}_{x,\mathrm{ref}}) \\ \lambda(\dot{\hat{p}}_y - \dot{p}_{y,\mathrm{ref}})\end{bmatrix}. \tag{10}$$

3.2. Control input design

In this section the equations describing the STA with adaptive gains and the SMO will be given in detail. These will then be used to find the desired force $F_{\mathrm{CM,d}}$.

3.2.1. The super-twisting algorithm with adaptive gains

The STA with adaptive gains proposed in [19] can be written as

$$\begin{aligned}u_{\mathrm{STA}} = \begin{bmatrix}u_{\mathrm{STA},x} \\ u_{\mathrm{STA},y}\end{bmatrix} &= \begin{bmatrix}-\alpha_x|\sigma_x|^{1/2}\operatorname{sgn}(\sigma_x) + v_x \\ -\alpha_y|\sigma_y|^{1/2}\operatorname{sgn}(\sigma_y) + v_y\end{bmatrix} \\ \dot{v} = \begin{bmatrix}\dot{v}_x \\ \dot{v}_y\end{bmatrix} &= \begin{bmatrix}-\beta_x\operatorname{sgn}(\sigma_x) \\ -\beta_y\operatorname{sgn}(\sigma_y)\end{bmatrix}\end{aligned} \tag{11}$$

where the adaptive gains are defined as

$$\dot{\alpha} = \begin{bmatrix}\dot{\alpha}_x \\ \dot{\alpha}_y\end{bmatrix} = \begin{bmatrix}\begin{cases}\omega_1\sqrt{\frac{\gamma_1}{2}}, & \text{if } \sigma_x \neq 0 \\ 0, & \text{if } \sigma_x = 0\end{cases} \\ \begin{cases}\omega_1\sqrt{\frac{\gamma_1}{2}}, & \text{if } \sigma_y \neq 0 \\ 0, & \text{if } \sigma_y = 0\end{cases}\end{bmatrix} \tag{12}$$

and

$$\beta = \begin{bmatrix}\beta_x \\ \beta_y\end{bmatrix} = \begin{bmatrix}2\varepsilon\alpha_x + \lambda + 4\varepsilon^2 \\ 2\varepsilon\alpha_y + \lambda + 4\varepsilon^2\end{bmatrix}, \tag{13}$$

where $\varepsilon, \lambda, \gamma_1$ and ω_1 are positive constants. For implementation purposes, a small boundary is put on the sliding surface and the adaptive gains can be expressed as

$$\begin{aligned}\dot{\alpha} = \begin{bmatrix}\dot{\alpha}_x \\ \dot{\alpha}_y\end{bmatrix} &= \begin{bmatrix}\begin{cases}\omega_1\sqrt{\frac{\gamma_1}{2}}, & \text{if } |\sigma_x| > \alpha_m \\ 0, & \text{if } |\sigma_x| \leq \alpha_m\end{cases} \\ \begin{cases}\omega_1\sqrt{\frac{\gamma_1}{2}}, & \text{if } |\sigma_y| > \alpha_m \\ 0, & \text{if } |\sigma_y| \leq \alpha_m\end{cases}\end{bmatrix} \\ \beta = \begin{bmatrix}\beta_x \\ \beta_y\end{bmatrix} &= \begin{bmatrix}2\varepsilon\alpha_x + \lambda + 4\varepsilon^2 \\ 2\varepsilon\alpha_y + \lambda + 4\varepsilon^2\end{bmatrix}\end{aligned} \tag{14}$$

where the design parameter α_m is a small positive constant.

© 0000 John Wiley and Sons Asia Pte Ltd and Chinese Automatic Control Society

3.2.2. State observer

By designing the observer structure as in [20], the state observer is chosen as

$$\begin{aligned}\dot{\hat{p}}_1 &= \begin{bmatrix}\dot{\hat{p}}_{1,x}\\ \dot{\hat{p}}_{1,y}\end{bmatrix} = \begin{bmatrix}\hat{p}_{2,x}+z_{1,x}\\ \hat{p}_{2,y}+z_{1,y}\end{bmatrix}\\ \dot{\hat{p}}_2 &= \begin{bmatrix}\dot{\hat{p}}_{2,x}\\ \dot{\hat{p}}_{2,y}\end{bmatrix} = \begin{bmatrix}\hat{p}_{3,x}+z_{2,x}+\frac{1}{m_t}F_{\mathrm{CM,d_x}}\\ \hat{p}_{3,y}+z_{2,y}+\frac{1}{m_t}F_{\mathrm{CM,d_y}}\end{bmatrix}\\ \dot{\hat{p}}_3 &= \begin{bmatrix}\dot{\hat{p}}_{3,x}\\ \dot{\hat{p}}_{3,y}\end{bmatrix} = \begin{bmatrix}z_{3,x}\\ z_{3,y}\end{bmatrix}\end{aligned} \tag{15}$$

where

$$\begin{aligned}z_1 &= \begin{bmatrix}z_{1,x}\\ z_{1,y}\end{bmatrix} = \begin{bmatrix}k_1|e_{1,x}|^{2/3}\operatorname{sgn}(e_{1,x})\\ k_1|e_{1,y}|^{2/3}\operatorname{sgn}(e_{1,y})\end{bmatrix}\\ z_2 &= \begin{bmatrix}z_{2,x}\\ z_{2,y}\end{bmatrix} = \begin{bmatrix}k_2|e_{1,x}|^{1/3}\operatorname{sgn}(e_{1,x})\\ k_2|e_{1,y}|^{1/3}\operatorname{sgn}(e_{1,y})\end{bmatrix}\\ z_3 &= \begin{bmatrix}z_{3,x}\\ z_{3,y}\end{bmatrix} = \begin{bmatrix}k_3\operatorname{sgn}(e_{1,x})\\ k_3\operatorname{sgn}(e_{1,y})\end{bmatrix}\end{aligned} \tag{16}$$

and k_1, k_2 and k_3 are gains to be chosen according to [22] and [23], $e_{1,x} = p_x - \hat{p}_{1,x}$ and $e_{1,y} = p_y - \hat{p}_{1,y}$. One choice of parameters that meets the requirements in [22] and [23], is according to [21], $k_1 = 6L^{1/3}$, $k_2 = 11L^{1/2}$ and $k_3 = 6L$, where L is a sufficiently large constant. By defining $e_2 = \dot{p} - \hat{p}_2$ and $e_3 = -\hat{p}_3 + F(t)$, the error dynamics of the state observer can be written as

$$\begin{aligned}\dot{e}_1 &= -k_1|e_1|^{2/3}\operatorname{sgn}(e_1) + e_2\\ \dot{e}_2 &= -k_2|e_1|^{1/3}\operatorname{sgn}(e_1) + e_3\\ \dot{e}_3 &= -k_3\operatorname{sgn}(e_1) + \dot{F}(t)\end{aligned} \tag{17}$$

3.2.3. Control input

In order for the STA to be applicable, the control input needs to be chosen such that the control appears in the equation of the first derivative of the sliding variable. In particular, we want to have $\dot{\hat{\sigma}} = u_{\mathrm{STA}}$, so that σ reaches zero in finite time. Taking the time derivative of Eq. (10) and substituting $\dot{\hat{p}}_1$ and $\dot{\hat{p}}_2$, defined in Eq. (15), we find that

$$\begin{aligned}\dot{\hat{\sigma}} &= (\dot{\hat{p}}_1 - \dot{p}_{\mathrm{ref}}) + (\dot{\hat{p}}_2 - \ddot{p}_{\mathrm{ref}})\\ &= (\hat{p}_2 + z_1 - \dot{p}_{\mathrm{ref}}) + (\hat{p}_3 + z_2 + \frac{1}{m_t}F_{\mathrm{CM,d}} - \ddot{p}_{\mathrm{ref}})\end{aligned} \tag{18}$$

By choosing $F_{\mathrm{CM,d}}$ to be

$$F_{\mathrm{CM,d}} = m_t(-\hat{p}_2 - z_1 + \dot{p}_{\mathrm{ref}} - \hat{p}_3 - z_2 + \ddot{p}_{\mathrm{ref}} + u_{\mathrm{STA}}) \tag{19}$$

we obtain

$$\dot{\hat{\sigma}} = u_{\mathrm{STA}}. \tag{20}$$

3.2.4. PD-controller

We want to compare the performance of the SMC algorithms to an existing controller for USMs with respect to disturbances and modelling errors. We will to the standard PD-controller that was proposed in [1]. This is implemented by replacing u_{STA} in Eq. (19) with

$$u_{\mathrm{PD}} = k_d^{\mathrm{CM}}\begin{bmatrix}\dot{p}_{x,\mathrm{ref}} - \dot{\hat{p}}_x\\ \dot{p}_{y,\mathrm{ref}} - \dot{\hat{p}}_y\end{bmatrix} + k_p^{CM}\begin{bmatrix}p_{x,\mathrm{ref}} - \hat{p}_x\\ p_{y,\mathrm{ref}} - \hat{p}_y\end{bmatrix} \tag{21}$$

where k_d^{CM} and k_p^{CM} are controller gains.

© 0000 John Wiley and Sons Asia Pte Ltd and Chinese Automatic Control Society

IV. Stability Analysis

In this section we perform a stability analysis of the closed-loop system, and it will be shown that the tracking error converge asymptotically.

4.1. Error dynamics

By defining $p = \begin{bmatrix} p_x & p_y \end{bmatrix}^T$ and dividing Eq. (7) by m_t (the total mass of the USM) the equations of motion can be written as

$$\ddot{p} = \begin{bmatrix} \ddot{p}_x \\ \ddot{p}_y \end{bmatrix} = \begin{bmatrix} \frac{1}{m_t}(e^T f_x + F_{\mathrm{CM},d_x}) \\ \frac{1}{m_t}(e^T f_y + F_{\mathrm{CM},d_y}) \end{bmatrix} \tag{22}$$

where $e^T f_x$ is the sum of all forces acting on the CM in x-direction and $e^T f_y$ is the sum of all forces acting on the CM in y-direction. These forces are hard to model exactly and will therefore be interpreted as a time-varying disturbance called $f(t) = \begin{bmatrix} f_x(t) & f_y(t) \end{bmatrix}^T$, where it is assumed that $\dot{f}(t)$ is bounded. The equation can then be written as

$$\begin{gathered} \ddot{p} = \frac{1}{m_t}(f(t) + F_{\mathrm{CM},d}) = \\ \begin{bmatrix} \ddot{p}_x \\ \ddot{p}_y \end{bmatrix} = \begin{bmatrix} \frac{1}{m_t}(f_x(t) + F_{\mathrm{CM},d_x}) \\ \frac{1}{m_t}(f_y(t) + F_{\mathrm{CM},d_y}) \end{bmatrix}. \end{gathered} \tag{23}$$

The error variable was introduced in Eq. (8). By introducing $\tilde{p}_1 = \tilde{p}$, $\tilde{p}_2 = \dot{\tilde{p}}$ and by differentiating the error variables the error dynamics can be written as

$$\begin{aligned} \dot{\tilde{p}}_1 &= \dot{\tilde{p}} = \tilde{p}_2 \\ \dot{\tilde{p}}_2 &= \ddot{\tilde{p}} = \ddot{p} - \ddot{p}_{\mathrm{ref}}(t) = \frac{1}{m_t}(f(t) + F_{\mathrm{CM},d}) - \ddot{p}_{\mathrm{ref}}(t), \end{aligned} \tag{24}$$

where it is assumed that the reference trajectory and its derivatives is bounded by design. By introducing a new function $F(t) = \frac{1}{m_t} f(t) - \ddot{p}_{\mathrm{ref}}(t)$, the error dynamics can be written as

$$\begin{aligned} \dot{\tilde{p}}_1 &= \tilde{p}_2 \\ \dot{\tilde{p}}_2 &= F(t) + \frac{1}{m_t} F_{\mathrm{CM},d}. \end{aligned} \tag{25}$$

where $\dot{F}(t)$ is bounded since it is a function of two bounded signals.

4.2. Overall closed-loop dynamics

By using the fact that $\hat{p}_1 = p - e_1$ and that $\hat{p}_2 = \dot{p} - e_2$, from Section 3.2.2, Eq. (10) can be written as

$$\hat{\sigma} = p - e_1 - p_{\mathrm{ref}} + \dot{p} - e_2 - \dot{p}_{\mathrm{ref}}. \tag{26}$$

Since $\tilde{p}_1 = p - p_{\mathrm{ref}}$ and $\tilde{p}_2 = \dot{p} - \dot{p}_{\mathrm{ref}}$ then Eq. (26) can be written as

$$\hat{\sigma} = \tilde{p}_1 - e_1 + \tilde{p}_2 - e_2. \tag{27}$$

By using that $\tilde{p}_2 = \dot{\tilde{p}}_1$, from Eq. (25), we get that

$$\hat{\sigma} = \tilde{p}_1 - e_1 + \dot{\tilde{p}}_1 - e_2 \tag{28}$$

and

$$\dot{\tilde{p}}_1 = \hat{\sigma} - \tilde{p}_1 + e_1 + e_2. \tag{29}$$

The overall closed-loop dynamics with $F_{\mathrm{CM,d}}$ given by Eq. (19), $\dot{\hat{\sigma}}$ as in Eq. (20), $\dot{\tilde{p}}$ as in Eq. (29) and the state

© 0000 John Wiley and Sons Asia Pte Ltd and Chinese Automatic Control Society

observer error as in Eq. (17) is then

$$\sum_1 \begin{cases} \dot{\tilde{p}}_1 = \hat{\sigma} - \tilde{p}_1 + e_1 + e_2 \\ \dot{\hat{\sigma}} = -\alpha|\hat{\sigma}|^{1/2}\operatorname{sgn}(\hat{\sigma}) + v \\ \dot{v} = -\beta\operatorname{sgn}(\hat{\sigma}) \end{cases} \qquad \sum_2 \begin{cases} \dot{e}_1 = -k_1|e_1|^{2/3}\operatorname{sgn}(e_1) + e_2 \\ \dot{e}_2 = -k_2|e_1|^{1/3}\operatorname{sgn}(e_1) + e_3 \\ \dot{e}_3 = -k_3\operatorname{sgn}(e_1) + \dot{F}(t) \end{cases} \tag{30}$$

Theorem 1 *Assume that the error dynamics is given by Eq.* (25)*, where* $|\dot{F}(t)| \leq \Delta$*,* m_t *is known and the sliding surface is defined by Eq.* (10)*. Assume that a state observer in Eq.* (15) *is used to estimate* p *and* $\dot{p}$*. Let the control input be given by Eq.* (19)*. Then the origin of the cascaded system in Eq.* (30) *is uniformly globally asymptotically stable (UGAS), which ensures asymptotic convergence of the tracking error.*

Proof. Analysis of subsystem 1, with $e_1 = 0$ *and* $e_2 = 0$*:* With $e_1 = 0$ and $e_2 = 0$, subsystem 1 can be written as

$$\sum_1 \begin{cases} \dot{\tilde{p}}_1 = \hat{\sigma} - \tilde{p}_1 \\ \dot{\hat{\sigma}} = -\alpha|\hat{\sigma}|^{1/2}\operatorname{sgn}(\hat{\sigma}) + v \\ \dot{v} = -\beta\operatorname{sgn}(\hat{\sigma}) \end{cases} \tag{31}$$

This can then be divided in two subsystems:

$$\sum_{11} \begin{cases} \dot{\tilde{p}}_1 = \hat{\sigma} - \tilde{p}_1 \end{cases} \qquad \sum_{12} \begin{cases} \dot{\hat{\sigma}} = -\alpha|\hat{\sigma}|^{1/2}\operatorname{sgn}(\hat{\sigma}) + v \\ \dot{v} = -\beta\operatorname{sgn}(\hat{\sigma}) \end{cases} \tag{32}$$

where [24, Lemma 2.1] can be used. Subsystem $\sum_{11}$ with $\hat{\sigma} = 0$ is analysed first. This is clearly a globally exponentially stable linear system, but since we will need a Lyapunov function for the analysis of this system when $\hat{\sigma} \neq 0$, we will use the Lyapunov function candidate $V_{11}(\tilde{p}) = \frac{1}{2}\tilde{p}_1^2$ for the analysis. The derivative of the this is

$$\begin{aligned} \dot{V}_{11}(\tilde{p}) &= \tilde{p}_1\dot{\tilde{p}}_1 = \tilde{p}_1(-\tilde{p}_1) \\ &= -\tilde{p}_1^2 \leq -||\tilde{p}_1||^2. \end{aligned} \tag{33}$$

This means that the Lyapunov function satisfies:

$$\begin{aligned} &k_1||x||^a \leq V_{11}(x) \leq k_2||x||^a \\ &\frac{\partial V_{11}}{\partial x} f_{11}(t,x) \leq -k_3||x||^a \end{aligned} \tag{34}$$

with $k_1 = k_2 = \frac{1}{2}$, $k_3 = 1$ and $a = 2$. By using Theorem 4.10 in [25] subsystem $\sum_{11}$ is proven globally exponentially stable with $\hat{\sigma} = 0$.

Subsystem $\sum_{12}$ has the structure of the STA algorithm with adaptive gains. In [19] a Lyapunov function is proposed for systems with this structure. Here it is proven that the Lyapunov function proposed is indeed a Lyapunov function for subsystem $\sum_{12}$ and that for any initial conditions, $\sigma, \dot{\sigma} \to 0$ in finite time by using the STA with adaptive gains given by Eq. (12) and Eq. (13) where ε, λ, γ_1 and ω_1 are arbitrary positive constant. Now, since the subsystem is globally finite time stable and autonomous it is also uniformly globally asymptotically stable (UGAS), [26, Proposition 2 and Proposition 3], which also implies that $||\hat{\sigma}(t)|| < \beta_1$.

To check if the solutions of $\sum_1$ are uniformly globally bounded (UGB), subsystem $\sum_{11}$ has to be

© 0000 John Wiley and Sons Asia Pte Ltd and Chinese Automatic Control Society

analysed with $\hat{\sigma} \neq 0$. The derivative of the Lyapunov function V_{11} is then

$$\begin{aligned}\dot{V}_{11}(\tilde{p}) &= -||\tilde{p}_1||^2 + \hat{\sigma}\tilde{p}_1 \\ &\leq -||\tilde{p}_1||^2 + \theta||\tilde{p}_1||^2 - \theta||\tilde{p}_1||^2 + \beta_1||\tilde{p}_1|| \\ &\leq -(1-\theta)||\tilde{p}_1||^2 \quad \forall \quad ||\tilde{p}_1|| \geq \frac{\beta_1}{\theta}\end{aligned} \tag{35}$$

where $0 < \theta < 1$. The solutions are then UGB, since the conditions of [25, Theorem 4.18] are satisfied. Consequently, the conditions of [24, Lemma 2.1] are satisfied, which implies that the origin of subsystem $\sum_1$ is UGAS.

Analysis of subsystem 2: In [27] a Lyapunov function is proposed for a third order observer. It is proven that the Lyapunov function is radially unbounded, positive definite and that it is a Lyapunov function for subsystem $\sum_2$, whose trajectories converge in finite time to the origin $e = 0$ for every value of $|\dot{F}(t)|$ as long as $\dot{F}(t)$ is bounded. Since $\dot{F}(t)$ is bounded by assumptions, the origin is globally finite time stable for every value of $\dot{F}(t)$, which means that the origin is also UGAS [26, Proposition 2 and Proposition 3], which implies $||e(t)|| \leq \beta_2$.

Analysis of the complete system: To analyse the complete system, [24, Lemma 2.1] is used. To check if the solutions of the complete system are UGB, the boundedness of $\tilde{p}_1$ has to be evaluated when $e_1 \neq 0$ and $e_2 \neq 0$, for this the Lyapunov function V_{11} is used. Note that boundedness of $\hat{\sigma}$ follows from $\sum_{12}$ being UGAS, since $\sum_{12}$ is not perturbed by $\sum_2$.

$$\begin{aligned}\dot{V}_{11}(\tilde{p}) &= -||\tilde{p}_1||^2 + (\hat{\sigma} + e_1 + e_2)\tilde{p}_1 \\ &\leq -||\tilde{p}_1||^2 + \theta||\tilde{p}_1||^2 - \theta||\tilde{p}_1||^2 + (\beta_1 + 2\beta_2)||\tilde{p}_1|| \\ &\leq -(1-\theta)||\tilde{p}_1||^2 \quad \forall \quad ||\tilde{p}_1|| \geq \frac{\beta_1 + 2\beta_2}{\theta}\end{aligned} \tag{36}$$

where $0 < \theta < 1$. The solutions are then UGB, since the conditions of [25, Theorem 4.18] are satisfied. Consequently, the conditions of [24, Lemma 2.1] are satisfied, which implies that the complete system is UGAS, and the tracking errors will therefore converge asymptomatically. □

V. Simulation Results

5.1. Implementation

The complete model with the force allocation matrix and controller is implemented in MATLAB Simulink. The USM implemented has $n = 16$ links, each one having length $2l_i = 0.14$ m and mass $m_i = 0.6597$ kg. The thruster configuration used corresponds to configuration 2 in [1]. This has one tail thruster attached to link 1, exerting a force along the x-axis of the link, and four additional thrusters located at link number 3, 6, 11 and 14, exerting forces normal to the links. For more details regarding the parameters used in the model, please see [1]. We have implemented two different case studies, one called torpedo mode, which is described in Section 5.1.1, and one called operation mode, described in Section 5.1.2.

© 0000 John Wiley and Sons Asia Pte Ltd and Chinese Automatic Control Society

5.1.1. Case 1 - Torpedo mode

We want the USM to move as a torpedo-shaped AUV when it is moving from one place to another. To achieve this type of behaviour, the link angles were set to zero, i.e. there was no lateral undulation, and a line-of-sight (LOS) guidance law defined by $\bar{\Psi}_{\text{ref}} = -\arctan(p_y/\Delta)$, where Δ is the look-ahead distance and p_y is the cross-track error from the path, was used for heading control. This was motivated by [28] and [29], but as [1] the heading of the USM was defined as the head link angle $\bar{\Psi} = \Psi_n$. This simulation case is shown in Fig. 2.

5.1.2. Case 2- Operation mode

When the USM is in operation mode, it will use the thrusters to stay in one place or move around, and use the end-effector at the head of the USM to do the operation. The motion of the joints can be seen as a disturbance to the CM position control system, as it will inflict unwanted motion on the CM of the USM. This simulation case investigates how well the proposed STA attenuates the unwanted effects of the joint motion. The simulated operation is an inspection, which entails that the head of the USM first moves in one direction and then the other, while the thrusters should keep the USM on the reference path. This type of simulation is shown in Fig. 3, where the USM head changes direction at 10, 20 and 30 seconds.

5.2. Simulations

As described in Section 3.2.2 the gain parameter L needs to be chosen sufficiently large, and for the simulations L was tuned manually to get good performance. The PD-controller gains were chosen by pole placement, and then slightly tuned to get better performance. The sliding surface parameter λ in Eq. (10) was set to 1. For the simulations, a fixed-step solver, with fixed step size 10^{-5} was used. In Table 1

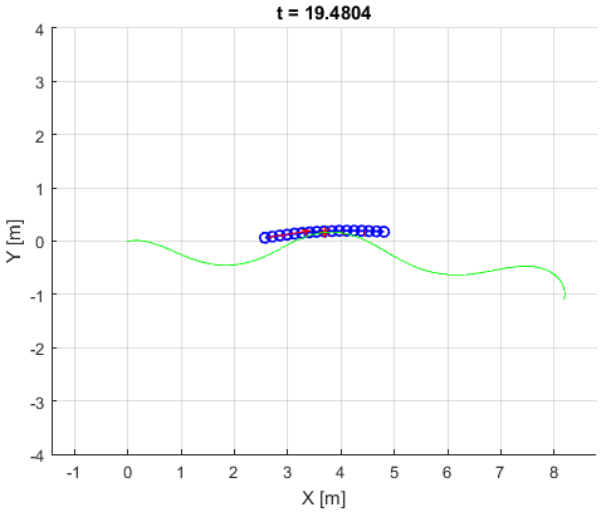


Fig. 2. Torpedo mode USM simulation

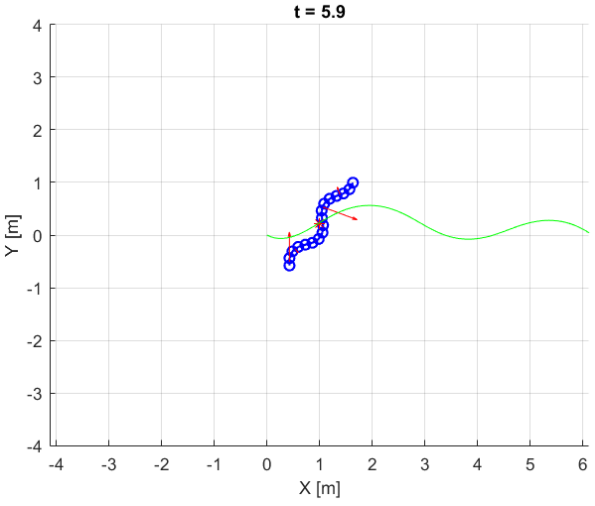


Fig. 3. Operation mode USM simulation

© 0000 John Wiley and Sons Asia Pte Ltd and Chinese Automatic Control Society

the maximum position error after settling is presented for both the PD and SMC controllers.

5.2.1. The super-twisting algorithm with adaptive gains:

The gains in the super-twisting algorithm with adaptive gains were set to: $\varepsilon = 1, \lambda = 1, \gamma_1 = 1, \omega_1 = 8, \alpha_m = 0.05$, and the observer gain was set to: $L = 55$. The simulations for torpedo mode can be seen in Fig. 4, and the simulations for operation mode can be seen in Fig. 5. The position error for case 2, operation mode can be seen better in Fig. 6.

5.2.2. The PD-controller

The gains for the PD-controller were set to: $k_d^{CM} = 6$ and $k_p^{CM} = 200$. The torpedo mode simulation can be seen in Fig. 7 and the operation mode simulation can be seen in Fig. 8. The position error for case 2, operation mode can be seen better in Fig. 9.

Table 1. Absolute maximum value for position error

Algorithm	**Error**			
	Torpedo		Operation	
	x	y	x	y
The STA with adaptive gains	$3.6134 \cdot 10^{-4}$	$2.8766 \cdot 10^{-4}$	0.0126	0.0264
PD-controller	0.0018	0.0095	0.0195	0.0227

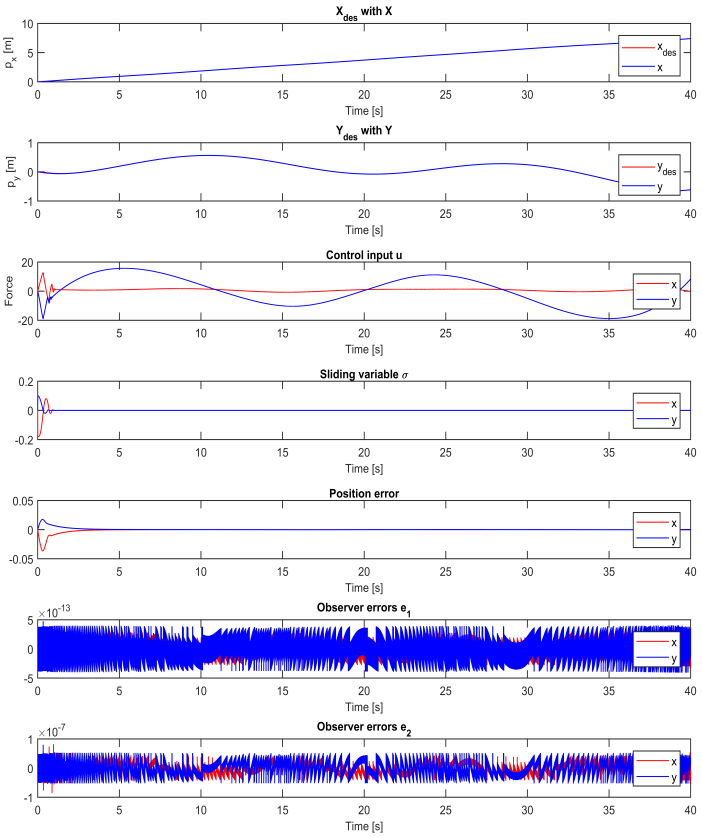


Fig. 4. Torpedo mode:Simulation of STA with state observer

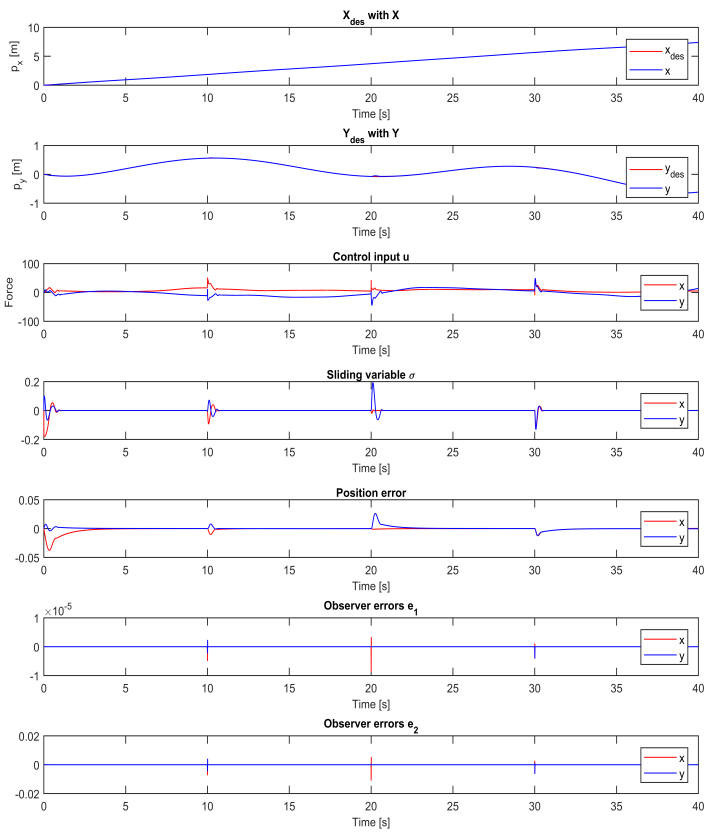


Fig. 5. Operation mode: Simulation of STA with state observer

© 0000 John Wiley and Sons Asia Pte Ltd and Chinese Automatic Control Society

5.3. Discussion

From Figs. 4 and 5 we can see that the proposed control law is indeed applicable since the position error converges to zero. From Figs. 4 to 8 and Table 1, we can also see that the STA algorithm with adaptive gains is superior to the PD-controller because it has smaller position errors in both simulation cases. The improved tracking performance is important to be able to control the tail or head of the USM better, to perform high-precision work, and to be able to move around in confined spaces.

It is worth noticing that for case 2, operation mode, the difference in position error is not very large. From Fig. 6 and Fig. 9, it can be seen that for the PD-controller the absolute position error is more varying than for the STA with adaptive gains. The reason for the larger absolute position error for the STA is the peaks that can be seen in Fig. 6. These peaks are from when the USM shifts position, and the error is therefore only larger in some small time period while the USM shifts position. The absolute position error when it has settled is for the STA in x-direction equal to $3.6088 \cdot 10^{-4}$ and in y-direction equal to $2.8847 \cdot 10^{-4}$, while for the PD-controller it is in x-direction equal to 0.0032 and in y-direction equal to 0.0073. This means that the error most of the time is less for the STA. From Fig. 9 it can also be seen that the error is not constant and adding integral effect would therefore not improve the results noticeably.

From the control input in the plots it is possible to see that the increase in performance does not change

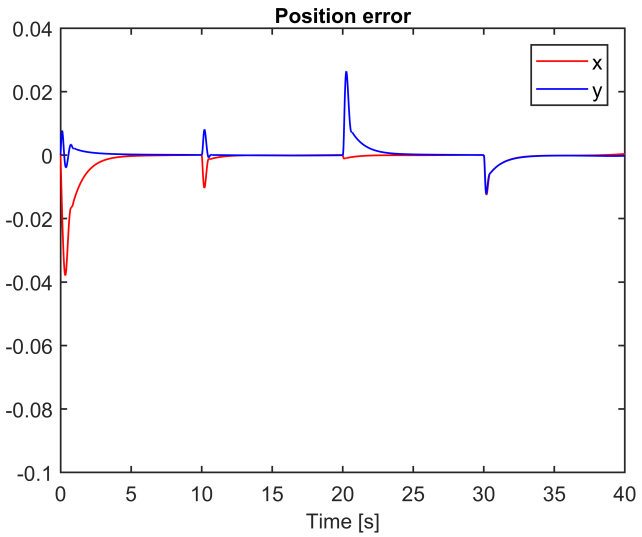


Fig. 6. Operation mode: Position error for the STA

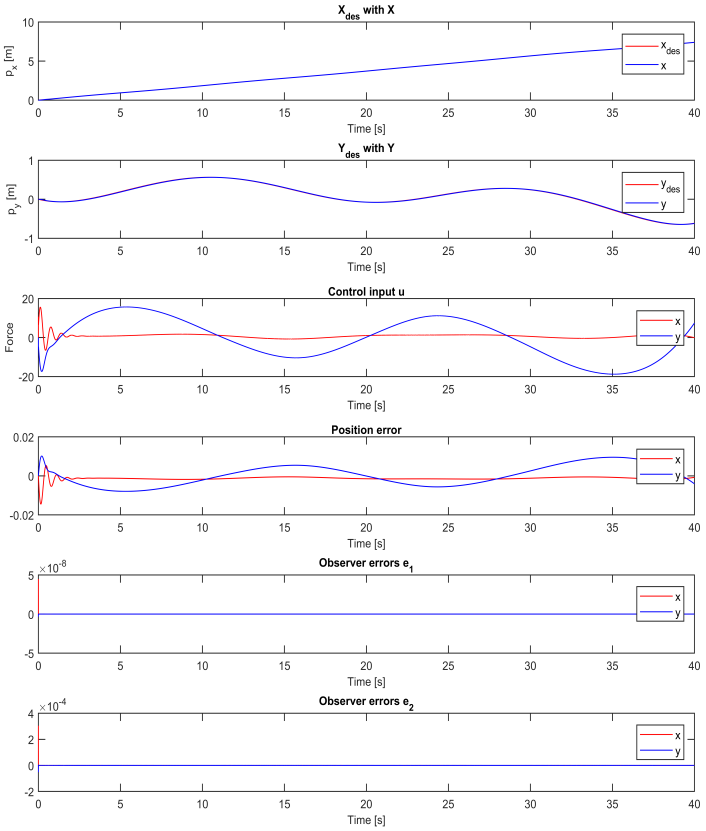


Fig. 7. Torpedo mode: Simulation of PD-controller

© 0000 John Wiley and Sons Asia Pte Ltd and Chinese Automatic Control Society

the magnitude of the force needed to control the USM noticeably. In operation mode, when using the STA the control input does have some peaks when the USM shifts position, but that is to be expected as the position error is nearly not affected at all by the shift. It is important that the increase in force needed is not too large, as that will affect the power usage of the USM.

From Figs. 4 and 5 we can see that the sliding surface does indeed converge to zero, and so do the observer errors. From Fig. 7 and Fig. 8 we can see that the observer errors also converge to zero when the PD-controller is used.

The PD gains for the linear controller might not be completely optimal, since finding the optimal gains is a difficult task. This gives the STA with adaptive gains one more advantage as finding the optimal gains is no longer a problem.

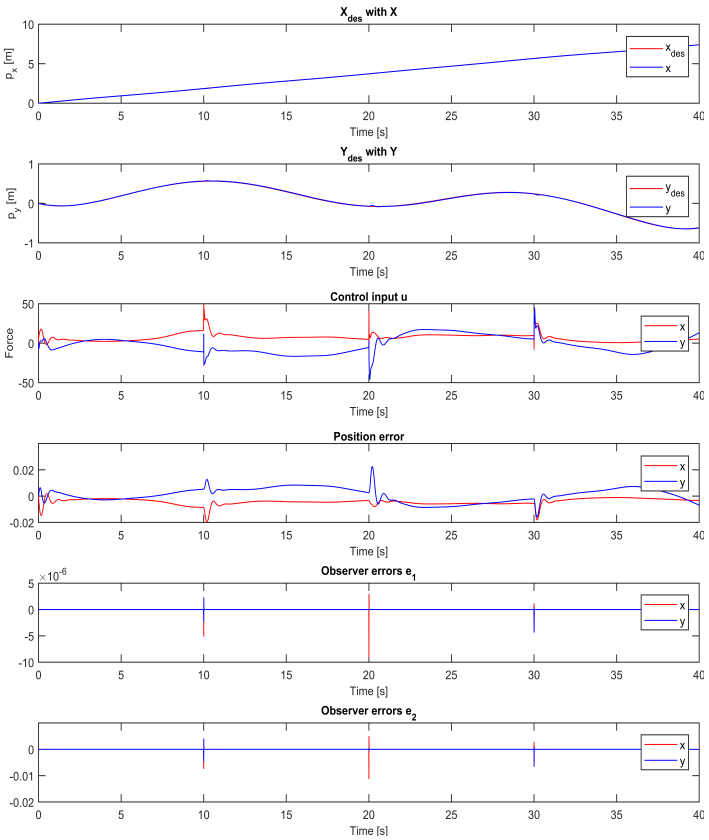


Fig. 8. Operation mode: Simulation of PD-controller

VI. Conclusions and Future Research

In this paper, we have discussed the use of the USM as a floating base manipulator, for which the trajectory tracking performance is important, and how the complexity of motion control is larger for USMs than for ROVs. We have proposed a second-order sliding mode control law for trajectory tracking, with the use of a sliding mode observer for the case when velocity measurements are not available. Furthermore, we have proved asymptotic convergence of the tracking error and performed a simulation study to verify the applicability of the proposed control law and shown that

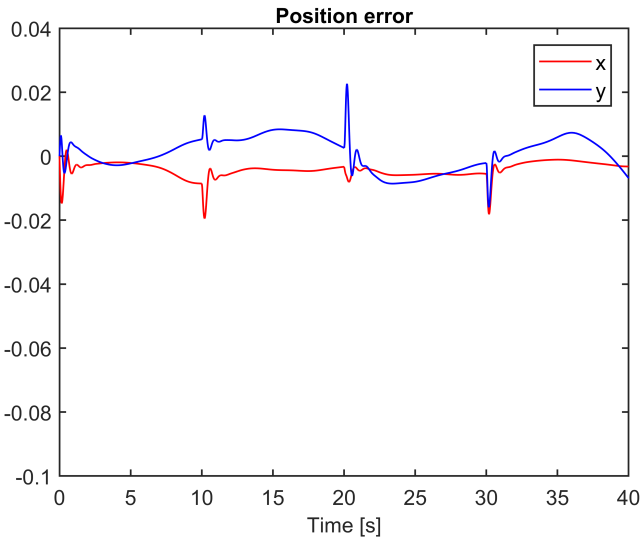


Fig. 9. Operation mode: Position error for the PD-controller

© 0000 John Wiley and Sons Asia Pte Ltd and Chinese Automatic Control Society

it gives better tracking performance than a linear PD-controller.

Future work includes investigating the best choice of control parameters and extending the results to 3D.

REFERENCES

1. Sverdrup-Thygeson J, Kelasidi E, Pettersen KY, Gravdahl JT. Modeling of underwater swimming manipulators. In: Proc. 10th IFAC Conference on Control Applications in Marine Systems. vol. 49. Trondheim, Norway; 2016. p. 81–88.
2. Sverdrup-Thygeson J, Kelasidi E, Pettersen KY, Gravdahl JT. A control framework for biologically inspired underwater swimming manipulators equipped with thrusters. In: Proc. 10th IFAC Conference on Control Applications in Marine Systems. vol. 49. Trondheim, Norway; 2016. p. 89–96.
3. Sverdrup-Thygeson J, Kelasidi E, Pettersen KY, Gravdahl JT. The Underwater Swimming Manipulator - A Bio-Inspired AUV. In: Proc. 2016 IEEE OES Autonomous Underwater Vehicles. Tokyo, Japan; 2016. .
4. Sverdrup-Thygeson J, Moe S, Pettersen KY, Gravdahl JT. Kinematic singularity avoidance for robot manipulators using set-based manipulability tasks. In: Proc. 1st IEEE Conference on Control Technology and Applications. Kohala Coast, Hawaii; 2017. .
5. Antonelli G. Underwater Robots. vol. 96 of Springer Tracts in Advanced Robotics. Springer International Publishing; 2014.
6. Antonelli G, Chiaverini S. Singularity-free regulation of underwater vehicle-manipulator systems. In: Proc. American Control Conference. Philadelphia, Pennsylvania; 1998. p. 399–403.
7. Fossen TI. Adaptive macro-micro control of nonlinear underwater robotic systems. In: Proc. 5th International Conference on Advanced Robotics. Pisa, Italy; 1991. p. 1569–1572.
8. Fossen T, Sagatun S. Adaptive control of nonlinear underwater robotic systems. Modeling, Identification and Control. 1991;12(2):95–105.
9. Cristi R, Papoulias FA, Healey AJ. Adaptive Sliding Mode Control of Autonomous Underwater Vehicles in the Dive Plane. IEEE Journal of Oceanic Engineering. 1990;15(3):152–160.
10. Dannigan MW, Russell GT. Evaluation and reduction of the dynamic coupling between a manipulator and an underwater vehicle. IEEE Journal of Oceanic Engineering. 1998;23(3):260–273.
11. Soylu S, Buckham BJ, Podhorodeski RP. A chattering-free sliding-mode controller for underwater vehicles with fault-tolerant infinity-norm thrust allocation. Ocean Engineering. 2008;35(16):1647–1659.
12. Fjellstad OE, Fossen TI. Singularity-free tracking of unmanned underwater vehicles in 6 DOF. In: Proc. 33rd IEEE Conference on Decision and Control. Lake Buena Vista, Florida; 1994. p. 1128–1133.

© 0000 John Wiley and Sons Asia Pte Ltd and Chinese Automatic Control Society

13. Rezapour E, Pettersen KY, Liljebäck P, Gravdahl JT. Differential geometric modelling and robust path following control of snake robots using sliding mode techniques. In: Proc. 2014 IEEE International Conference on Robotics and Automation. Hong Kong, China; 2014. p. 4532–4539.
14. Kelasidi E, Pettersen KY, Gravdahl JT, Strømsøyen S, Sørensen AJ. Modeling and Propulsion Methods of Underwater Snake Robots. In: Proc. 1st IEEE Conference on Control Technology and Applications. Kohala Coast, Hawaii; 2017. .
15. Kelasidi E, Pettersen KY, Gravdahl JT, Liljebäck P. Modeling of underwater snake robots. In: Proc. 2014 IEEE International Conference on Robotics and Automation. Hong Kong, China; 2014. p. 4540–4547.
16. Hung JY, Gao W, Hung JC. Variable Structure Control: A Survey. IEEE Transactions on Industrial Electronics. 1993;40(1):2–22.
17. Young KD, Utkin VI, Özgüner Ü. A control engineer's guide to sliding mode control. IEEE Transactions on Control Systems Technology. 1999;7(3):328–342.
18. Levant A. Sliding order and sliding accuracy in sliding mode control. International Journal of Control. 1993;58(6):1247–1263.
19. Shtessel YB, Moreno JA, Plestan F, Fridman LM, Poznyak AS. Super-twisting adaptive sliding mode control: A Lyapunov design. In: Proc. 49th IEEE Conference on Decision and Control. Atlanta, GA, USA; 2010. p. 5109–5113.
20. Kumari K, Chalanga A, Bandyopadhyay B. Implementation of Super-Twisting Control on Higher Order Perturbed Integrator System using Higher Order Sliding Mode Observer. In: Proc. 10th IFAC Symposium on Nonlinear Control Systems.. vol. 49. California, USA; 2016. p. 873–878.
21. Chalanga A, Kamal S, Fridman LM, Bandyopadhyay B, Moreno JA. Implementation of Super-Twisting Control: Super-Twisting and Higher Order Sliding-Mode Observer-Based Approaches. IEEE Transactions on Industrial Electronics. 2016;63(6):3677–3685.
22. Levant A. Robust exact differentiation via sliding mode technique [Journal Article]. Automatica. 1998;34(3):379–384.
23. Levant A. Higher-order sliding modes, differentiation and output-feedback control. International Journal of Control. 2003;76(9-10):924–941.
24. Loría A, Panteley E. Cascaded nonlinear time-varying systems: Analysis and design. Lecture Notes in Control and Information Sciences. 2005;311:23–64.
25. Khalil HK. Nonlinear systems. 3rd ed. Upper Saddle River, N.J: Prentice Hall; 2002.
26. Polyakov A, Fridman L. Stability notions and Lyapunov functions for sliding mode control systems. Journal of the Franklin Institute. 2014;351(4):1831–1865.

© 0000 John Wiley and Sons Asia Pte Ltd and Chinese Automatic Control Society

27. Moreno JA. Lyapunov function for Levant's Second Order Differentiator. In: Proc. IEEE 51st IEEE Conference on Decision and Control. Maui, Hawaii, USA; 2012. p. 6448–6453.
28. Kelasidi E, Pettersen KY, Gravdahl JT. A waypoint guidance strategy for underwater snake robots. In: Proc. IEEE 22nd Mediterranean Conference on Control and Automation. Palermo,Italy.; 2014. p. 1512–1519.
29. Liljebäck P, Pettersen KY, Stavdahl Ø, Gravdahl JT. Snake Robots : Modelling, MMechatronics, and Control. Snake Robots : Modelling, Mechatronics, and Control. Dordrecht: Springer; 2012.

© 0000 John Wiley and Sons Asia Pte Ltd and Chinese Automatic Control Society