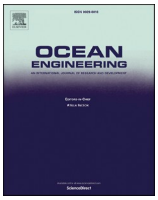
Contents lists available at ScienceDirect

Ocean Engineering

journal homepage: www.elsevier.com/locate/oceaneng



Comparison of two second-order sliding mode control algorithms for an articulated intervention AUV: Theory and experimental results

Ida-Louise G. Borlaug *, Kristin Y. Pettersen, Jan Tommy Gravdahl

Centre for Autonomous Marine Operations and Systems, Department of Engineering Cybernetics, Norwegian University of Science and Technology (NTNU), Trondheim, Norway

ARTICLE INFO

Keywords:
Experimental results
Sliding mode control
Autonomous underwater vehicle
Robotics
Control application

ABSTRACT

An articulated intervention autonomous underwater vehicle (AIAUV) is a slender, multi-articulated underwater robot. Accurate trajectory tracking is essential for AIAUV operations. Furthermore, due to hydrodynamic and hydrostatic parameter uncertainties, uncertain thruster characteristics, unknown disturbances, and unmodelled dynamic effects, robustness is crucial. In this paper, we present a super-twisting algorithm (STA) with adaptive gains and a generalized super-twisting algorithm (GSTA) for trajectory tracking of the position and orientation of AIAUVs. A higher-order sliding mode observer (HOSMO) for estimating the linear and angular velocities when velocity measurements are unavailable is also presented. The tracking errors for the resulting system are proven to converge asymptotically to zero. Finally, we demonstrate the applicability of the presented control laws with comprehensive simulation and experimental results and perform a comparison study, with two tests (C-shape and C-shape with a moving head), between the two algorithms and also a benchmark PID controller. The STA with adaptive gains exhibits the best overall tracking performance, with average position root mean square error (RMSE) 0.0121 m and average orientation RMSE 0.0335 rad. The GSTA also presented good results with average position RMSE 0.0267 m and average orientation RMSE 0.0292 rad. The PID controller gave average position RMSE 0.0371 m and average orientation RMSE 0.0491 rad.

1. Introduction

An articulated intervention autonomous underwater vehicle (AIAUV) is an underwater swimming manipulator. It has the slender, multi-articulated body of an underwater snake robot (USR), which imparts the AIAUV with accessibility and flexibility, but it also has multiple thrusters along its body. The thrusters enable the AIAUV to move forward without using an undulating gait pattern and provides it with the ability to hover. This property is especially important for station-keeping and trajectory tracking in narrow and confined spaces. These capabilities enable the AIAUV to operate as a floating base manipulator. Moreover, the AIAUV has adopted the flexibility and high-kinematic redundancy of the USR and the energy-efficient shape and tether-less operation of autonomous underwater vehicles (AUVs). Compared to standard survey AUVs, the AIAUV has the advantage that it has full actuation, and it can perform intervention tasks. Since it can use its slender body to access narrow spaces, use its thrusters to keep itself stationary and then use its joints to perform intervention tasks, the AIAUV can exploit the full potential of the inherent kinematic redundancy, which has been addressed in detail in (Sverdrup-Thygeson et al., 2016), (Sverdrup-Thygeson et al., 2018).

The controller design for underwater vehicles (UVs) such as AIAUVs, AUVs, USRs and remotely operated vehicles (ROVs) is a complex problem (Antonelli, 2014). Underwater vehicles are often subject to hydrodynamic and hydrostatic parameter uncertainties, uncertain thruster characteristics, unknown disturbances, and unmodelled dynamic effects. However, the control design for the AIAUV is more complex than for the other mentioned UVs, and control methods used for these type of vehicles can therefore not be directly applied. Specifically, the AIAUV has the same mechanical structure as a USR while also possessing hovering and intervention capabilities because of the thrusters. Path following and manoeuvring control of USRs are discussed in (Kelasidi et al., 2014, 2017a; Kohl et al., 2016a, 2016b, 2017a, 2017b); however, these method are not suitable for utilizing the AIAUV's hovering and intervention capabilities. The AUV is a single-body UV, and some methods for control design have been proposed in (Fossen, 2011), while the AIAUV is a floating-base manipulator arm. We therefore face the additional challenge of handling the

* Corresponding author.
E-mail addresses: Ida.Louise.Borlaug@ntnu.no (I.-L.G. Borlaug), Kristin.Y.Pettersen@ntnu.no (K.Y. Pettersen), Tommy.Gravdahl@ntnu.no (J.T. Gravdahl).

https://doi.org/10.1016/j.oceaneng.2020.108480
Received 18 May 2020; Received in revised form 1 December 2020; Accepted 6 December 2020
Available online 14 January 2021

interaction forces, also referred to as coupling forces, between the arm and the vehicle. Several different control approaches that address these challenges are summarized and presented in (Antonelli, 2014). However (Antonelli, 2014), considers mostly ROVs. ROVs also face these challenges, but for the AIAUV, the motion of the joints has a more significant effect on the overall motion of the AIAUV than for ROVs. This is because the AIAUV has no separate vehicle base and a low mass compared to an ROV, while ROVs have a separate base with a larger inertia than that of the manipulator arm. Furthermore, inertia, drag forces, and restoring forces and moments caused by the joint motion have a significant effect on the overall motion of the whole mechanism, for which any joint motion leads to a corresponding change in the direction of the thrusters that are distributed along the robot. All this increases the complexity of the motion for the AIAUV, thereby rendering the control problem even more complex than for ROVs. For the AIAUV in particular, some methods for propulsion have been proposed in (Kelasidi et al., 2016, Kelasidi et al., 2017b), and energy-efficient methods for long-distance travel have been proposed in (Wrzos-Kaminska et al., 2019). In (Sverdrup-Thygeson et al., 2016, 2018), a control framework for AIAUVs that uses inverse kinematic (IK) control together with a dynamic controller is proposed. However, in both papers, IK is the main focus. The dynamic controller proposed in (Sverdrup-Thygeson et al., 2016) is only a simple P-controller, and the controller proposed in (Sverdrup-Thygeson et al., 2018) is a feedback-linearisation controller combined with a velocity-based proportional-integral-derivative (PID)-controller. Moreover, none of these methods focus on robust control, which is essential for the AIAUV due to large coupling forces, unknown disturbances and other unmodelled dynamic effects that affect the AIAUV. Sliding mode control (SMC) is a robust and versatile non-linear control approach that is particularly well suited for situations where unknown non-linearities affect the system, as in the case of AIAUVs. In this paper we therefore investigate using SMC to obtain a robust trajectory tracking controller.

In recent years, several results have been reported regarding the use of SMC for many complex dynamic systems. For UVs, in general, some relevant contributions are the following: in (Antonelli and Chiaverini, 1998), an SMC approach for the regulation problem of an underwater vehicle-manipulator system is developed. This control law is inspired by (Fjellstad and Fossen, 1994) and avoids the inversion of the system Jacobian and is therefore singularity-free. In (Fossen, 1991) and (Fossen and Sagatun, 1991), an SMC is used to address input uncertainty due to partly known non-linear thruster characteristics. In (Soylu et al., 2008), a chattering-free SMC is proposed for trajectory control. The chattering-free approach is developed by combining the SMC with adaptive proportional-integral-derivative controller gains and having an adaptive update of the upper bound on the disturbance and parameter uncertainties. In (Dannigan and Russell, 1998), an SMC is used to address the coupling effects between a manipulator and an UV. By combining a virtual velocity control and SMC, a hybrid control strategy is developed in (Zhu and Sun, 2013) for the trajectory tracking of an unmanned UV. In (Xu et al., 2015), the trajectory tracking problem of an under-actuated unmanned UV is studied by combining backstepping and SMC. An attitude controller for an AUV is designed in (Cui et al., 2016) by using a sliding-mode-based adaptive controller. In (Liu et al., 2017), the trajectory tracking problem for an UV subject to unknown system uncertainties and time-varying external disturbances is considered and is solved by using a non-linear disturbance observer-based backstepping finite-time SMC scheme. In (von Ellenrieder and Henninger, 2019) a third-order sliding mode disturbance observer, coupled with a modified super twisting controller, is proposed for the trajectory tracking of fully-actuated marine vehicles in the presence of unknown, time-varying disturbances. However, as mentioned in the previous section the control design for the AIAUV is more complex than for the other mentioned UVs, and control methods used for these type of vehicles can therefore not be directly applied.

In recent years, SMC has been further developed into higher-order SMC schemes, which removes the chattering problem. The super-twisting algorithm (STA) with adaptive gains, proposed by (Shtessel et al., 2010), is the most powerful second-order continuous SMC algorithm as it attenuates chattering and does not require the consideration of any conservative upper bound on the disturbance to maintain sliding because of the adaptive gains. Motivated by this (Borlaug et al., 2019a), applied the STA together with a higher-order sliding mode observer (HOSMO) (Kumari et al., 2016) to an AIAUV. The tracking problem of the centre of mass of the AIAUV in 2DOF was considered. The tracking errors were proven to converge asymptomatically to zero, and the simulation results demonstrated that the proposed control method provided excellent tracking capabilities. However, the tracking control problem was restricted to considering the position of the AIAUV in 2DOF, and the movements of the joints were considered to be a disturbance. We therefore extended these results in (Borlaug et al., 2018) by considering the AIAUV in 6DOF and examining the tracking problem for the position, orientation and joint angles. Equally good results were obtained in 6DOF as in 2DOF both in theory and in simulations. In (Borlaug et al., 2019a) and (Borlaug et al., 2018), an HOSMO had to be used because velocity measurements were unavailable. In (Borlaug et al., 2019b), we proposed using the generalized super-twisting algorithm (GSTA) proposed in (Castillo et al., 2018) for trajectory tracking of an AIAUV in 6DOF. In (Borlaug et al., 2020), we extended the results obtained in (Borlaug et al., 2019b) to include results with and without an HOSMO for trajectory tracking of an AIAUV in 6DOF. The GSTA is an extension of the STA that provides finite-time convergence in the presence of time- and state-dependent perturbations, which is essential for robust control of the AIAUV. These properties provide the GSTA with additional theoretical advantages over the STA with adaptive gains. These theoretical advantages enable verification that the tracking errors converge asymptotically to zero when the GSTA is used without an HOSMO, which is not possible for the STA with adaptive gains.

In this paper, we investigate the use of the STA with adaptive gains compared to the use of the GSTA. Specifically, we use both the STA with adaptive gains and the GSTA for the trajectory tracking of the position and orientation of the base of an AIAUV in 6DOF. We also solve the tracking problem using a PID controller to evaluate how the SMC algorithms perform compared to a linear controller. An HOSMO for estimating the linear and angular velocities when velocity measurements are unavailable is also presented. Furthermore, the tracking errors for the control law using the STA with adaptive gains in combination with the HOSMO are proven to converge asymptotically to zero. Finally, we demonstrate the applicability of the proposed control laws with comprehensive simulation and experimental results, and we compare the results. The main objective of this paper is a comparison between the STA with adaptive gains, the GSTA and a PID controller. In addition, the paper extends the results obtained in (Borlaug et al., 2019a) from 2DOF to 6DOF. In (Borlaug et al., 2019a), the tracking problem of the centre of mass of the AIAUV in 2DOF was considered and solved using the STA with adaptive gains in combination with an HOSMO, and the tracking errors were proven to converge asymptotically to zero. Preliminary results were presented in (Borlaug et al., 2018). In this paper, these results are extended by also solving the tracking control problem using the STA with adaptive gains without an HOSMO. We also include extensive simulation and experimental results not presented in (Borlaug et al., 2018) using the STA with adaptive gains. In (Borlaug et al., 2019b), the tracking control problem is considered and solved using the GSTA without an HOSMO. In (Borlaug et al., 2020), the tracking control problem is considered and solved using the GSTA with and without an HOSMO, and simulation and experimental results are given. The simulation and experimental results given in (Borlaug et al., 2020) are included in this paper to perform a comparison study between the STA with adaptive gains proposed here with the GSTA proposed in (Borlaug et al., 2020). The two algorithms are further compared with a benchmark PID controller.

The contributions of this paper can be summarized as follows. The trajectory tracking control problem of an AIAUV in 6DOF is solved using the STA with adaptive gains. The trajectory tracking control problem is also solved by using the algorithm in combination with an HOSMO. Additionally, the tracking errors for the control law using the STA with adaptive gains in combination with the HOSMO are proven to converge asymptotically to zero. We demonstrate the applicability of the presented control laws with comprehensive simulation and experimental results, and we compare these results with the GSTA proposed in (Borlaug et al., 2020) as well as with a PID controller.

The remainder of this paper is organized as follows. In Sec. 2, the model and the tracking control problem for the AIAUV are defined mathematically. The control law and observer design for tracking the desired trajectory are presented and analysed in Sec. 3 for all three algorithms. A description of the implemented simulation model and the simulation results are provided in Sec. 4. In Sec. 5, the experimental setup and the experimental results are given. The conclusions and suggestions for future work are presented in Sec. 6.

2. AIAUV model and problem statement

In this section, we provide the model and the mathematical definition of the tracking control problem for the AIAUV, Fig. 1. The AIAUV consists of n links connected by $n-1$ motorized joints. Each joint is treated as a one-dimensional Euclidean joint. To follow the convention used in the robotics community, the first link is referred to as the base of the manipulator. The base link is link 1, and the front link, where the end-effector is positioned, is link n. A visual representation of the relevant frames is given in Fig. 2. In the right corner of Fig. 2 the link frames (blue coordinate-systems) have been visualized for a completely outstretched robot. Note that link frame 1 corresponds to the base frame. The joints are numbered from $i=1$ to $n-1$ such that link i and link $i+1$ are connected by joint i.

Furthermore, the AIAUV is equipped with m thrusters, including one or more thrusters acting along the body of the AIAUV to provide forward thrust and tunnel thrusters acting through the links to provide station-keeping capability. The AIAUV is considered to be a floating base manipulator operating in an underwater environment subject to added mass forces, dissipative drag forces, and gravity and buoyancy forces, which allows us to model the AIAUV as an underwater vehicle-manipulator system (UVMS), with dynamic equations given in matrix form by (Antonelli, 2014), (From et al., 2014)

$$M(q)\dot{\zeta}+C(q,\zeta)\zeta+D(q,\zeta)\zeta+g\left(q,R_B^I\right)=\tau(q) \tag{1}$$

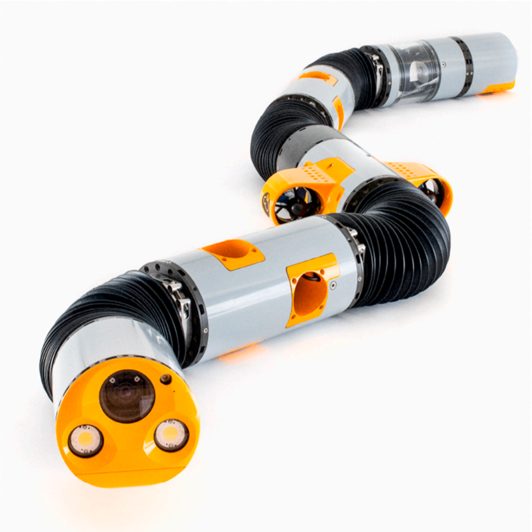


Fig. 1. The Eelume AIAUV (courtesy of Eelume).

where $q\in\mathbb{R}^{(n-1)}$ is the vector representing the joint angles, $\zeta\in\mathbb{R}^{6+(n-1)}$ is the vector of body-fixed velocities, $M(q)$ is the inertia matrix including added mass terms, $C(q,\zeta)$ is the Coriolis-centripetal matrix, $D(q,\zeta)$ is the damping matrix, and $g(q,R_B^I)$ is the matrix of gravitational and buoyancy forces. The control input is given by the generalized forces $\tau(q)$:

$$\tau(q)=\begin{bmatrix} T(q) & 0_{6\times(n-1)} \\ 0_{(n-1)\times m} & I_{(n-1)\times(n-1)} \end{bmatrix}\begin{bmatrix} \tau_{thr} \\ \tau_q \end{bmatrix} \tag{2}$$

where $T(q)\in\mathbb{R}^{6\times m}$ is the thruster configuration matrix, $\tau_{thr}\in\mathbb{R}^m$ is the vector of thruster forces, and $\tau_q\in\mathbb{R}^{(n-1)}$ represents the joint torques. To implement the control input $\tau(q)$, a thruster allocation scheme as proposed in (Sverdrup-Thygeson et al., 2018) needs to be implemented to distribute the desired control inputs onto the m thrusters. The vector of body-fixed velocities, ζ, is defined as

$$\zeta=\begin{bmatrix} v^T & \omega^T & \dot{q}^T \end{bmatrix}^T \tag{3}$$

where v and ω are the body-fixed linear and angular velocities of the base of the AIAUV, respectively, and $\dot{q}$ is the vector of joint angle velocities. The desired velocities are denoted as

$$\zeta_d=\begin{bmatrix} v_d^T & \omega_d^T & \dot{q}_d^T \end{bmatrix}^T \tag{4}$$

in the body-fixed frame. The velocity errors can then be defined as

$$\tilde{\zeta}=\zeta-\zeta_d \tag{5}$$

where ζ and ζ_d are defined as in (3) and (4), respectively. To represent the orientation of the AIAUV, both Euler angles and quaternions can be used. In this paper, we will describe both methods, as we use quaternions to describe the dynamics when we create the control laws without the HOSMO and Euler angles when we create the control laws that use the estimated velocities from the HOSMO. The reason is that the HOSMO does not work with quaternions as a different number of states in position versus velocity then occurs.

2.1. Model representation using euler angles

In this section, we will describe the model using Euler angles. A well-known problem when we use Euler angles is that the Jacobian can become singular at $\theta=\pm\pi/2$ (xyz-convention); however, if we stay away from that angle, the Jacobian is well defined and the inverse of the Jacobian matrix can still be used. When we use Euler angles to describe the AIAUV, we can use the HOSMO from (Kumari et al., 2016) to estimate the velocity such that we do not require velocity measurements. The complete state vector specifying the position, orientation, and shape of the AIAUV when Euler angles are used is defined as

$$\xi_e=\begin{bmatrix} \eta_1^T & \eta_2^T & q^T \end{bmatrix}^T \quad \in\mathbb{R}^{6+(n-1)} \tag{6}$$

where $\eta_1=\begin{bmatrix} x & y & z \end{bmatrix}^T\in\mathbb{R}^3$ is the position of the base, and $\eta_2=\begin{bmatrix} \varphi & \theta & \psi \end{bmatrix}^T\in\mathbb{R}^3$ are the Euler angles describing the orientation of the base in the inertial frame. The relationship between the body-fixed velocities and the complete state vector is given by the differential equation

$$\dot{\xi}_e=J_e(\eta_2)\zeta=\begin{bmatrix} R_B^I & 0_{3\times3} & 0_{3\times(n-1)} \\ 0_{3\times3} & J_{k,o}^{-1} & 0_{3\times(n-1)} \\ 0_{(n-1)\times3} & 0_{(n-1)\times3} & I_{(n-1)\times(n-1)} \end{bmatrix}\zeta \tag{7}$$

where

$$R_I^B(\eta_2)\ =\begin{bmatrix} c_\psi c_\theta & s_\psi c_\theta & -s_\theta \\ -s_\psi c_\varphi+c_\psi s_\theta s_\varphi & c_\psi c_\varphi+s_\psi s_\theta s_\varphi & s_\varphi c_\theta \\ s_\psi s_\varphi+c_\psi s_\theta c_\varphi & -c_\psi s_\varphi+s_\psi s_\theta c_\varphi & c_\varphi c_\theta \end{bmatrix} \tag{8}$$

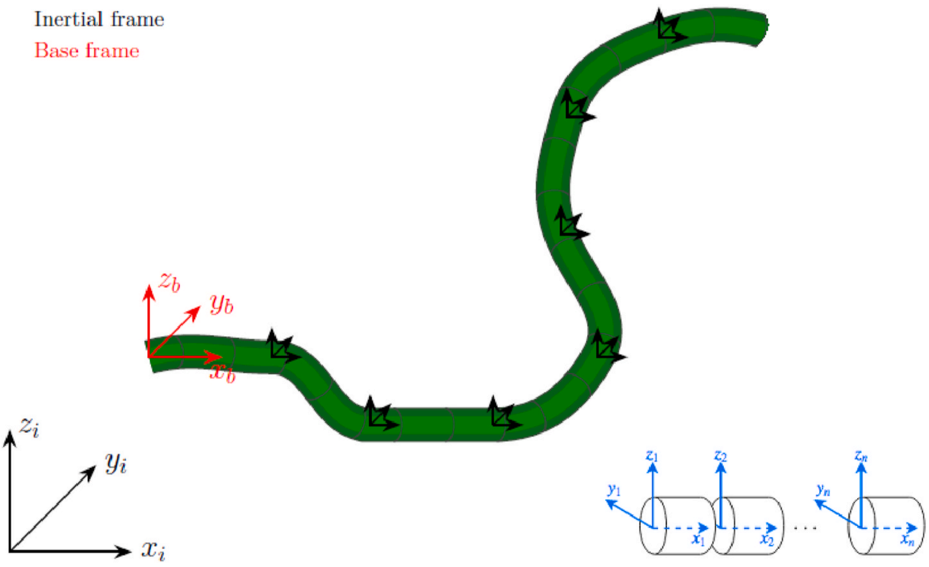


Fig. 2. Coordinate frames for the AIAUV (Credit: Arnt Erik Stene and Marianna Wrzos-Kaminska).

with $c_x = \cos(x)$ and $s_x = \sin(x)$, is the rotation matrix expressing the transformation from the inertial frame to the body-fixed frame and

$$J_{k,o}(\eta_2) = \begin{bmatrix} 1 & 0 & -s_\theta \\ 0 & c_\varphi & c_\theta s_\varphi \\ 0 & -s_\varphi & c_\theta c_\varphi \end{bmatrix} \tag{9}$$

is the Jacobian matrix. By defining the desired trajectory as $\xi_{e,d} = \begin{bmatrix} \eta_{1,d}^T & \eta_{2,d}^T & q_d^T \end{bmatrix}^T$, the tracking error vector can be defined as

$$\tilde{\xi}_e = \begin{bmatrix} \tilde{\eta}_1 \\ \tilde{\eta}_2 \\ \tilde{q} \end{bmatrix} = \begin{bmatrix} \eta_1 - \eta_{1,d} \\ \eta_2 - \eta_{2,d} \\ q - q_d \end{bmatrix} \tag{10}$$

The goal of the tracking problem is to make the error vector, $\tilde{\xi}_e$, converge to zero. Therefore, the tracking control objective is to make $(\tilde{\xi}_e, \tilde{\zeta}) = (0, 0)$ an asymptotically stable equilibrium point of (1) and (7), which will ensure that the tracking errors will converge to zero.

2.2. Model representation using quaternions

In this section, we will describe the model using quaternions. When we use quaternions to represent the model, we avoid the singularities in the Jacobian matrix that arise from the Euler angles. We therefore obtain the advantage of a well-defined Jacobian, which means that we can use the inverse of the Jacobian matrix. The complete state vector specifying the position, orientation, and shape of the AIAUV when quaternions are used is defined as

$$\xi_q = \begin{bmatrix} \eta_1^T & p^T & q^T \end{bmatrix}^T \quad \in \mathbb{R}^{7+(n-1)} \tag{11}$$

where $\eta_1 = [x \quad y \quad z]^T \in \mathbb{R}^3$ is the position of the base, and $p = \begin{bmatrix} \varepsilon^T & \eta \end{bmatrix}^T = [\varepsilon_1 \quad \varepsilon_2 \quad \varepsilon_3 \quad \eta]^T \in \mathbb{R}^4$ is the unit quaternion describing the orientation of the base in the inertial frame. The relationship between the body-fixed velocities and the complete state vector specifying the position, orientation, and shape of the AIAUV is given by the differential equation

$$\dot{\xi}_q = J_q(p)\zeta = \begin{bmatrix} R_B^I(p) & 0_{3\times3} & 0_{3\times(n-1)} \\ 0_{4\times3} & J_{k,oq}(p) & 0_{4\times(n-1)} \\ 0_{(n-1)\times3} & 0_{(n-1)\times3} & I_{(n-1)\times(n-1)} \end{bmatrix} \zeta \tag{12}$$

where R_I^B is the rotation matrix expressing the transformation from the inertial frame to the body-fixed frame, and $J_{k,oq}(p) = \frac{1}{2}\begin{bmatrix} \eta I_3 + S(\varepsilon) \\ -\varepsilon^T \end{bmatrix}$, where I_3 is the (3×3) identity matrix, and $S(\cdot)$ is the cross-product operator defined as in [10, Definition2.2]. By defining the desired trajectories as

$$\xi_{q,d} = \begin{bmatrix} \eta_{1,d}^T & p_d^T & q_d^T \end{bmatrix}^T \tag{13}$$

where $p_d = \begin{bmatrix} \varepsilon_d^T & \eta_d \end{bmatrix}^T$, the orientation error, which is computed via the composition (quaternion product) (Antonelli, 2014), is then given by

$$\tilde{p} = \begin{bmatrix} \tilde{\varepsilon} \\ \tilde{\eta} \end{bmatrix} = \begin{bmatrix} \eta\varepsilon_d - \eta_d\varepsilon + S(\varepsilon_d)\varepsilon \\ \eta\eta_d + \varepsilon^T\varepsilon_d \end{bmatrix} \tag{14}$$

The complete tracking error can then be defined as

$$\tilde{\xi}_q = \begin{bmatrix} \tilde{\eta}_1 \\ \tilde{\varepsilon} \\ \tilde{q} \end{bmatrix} = \begin{bmatrix} \eta_1 - \eta_{1,d} \\ \eta\varepsilon_d - \eta_d\varepsilon + S(\varepsilon_d)\varepsilon \\ q - q_d \end{bmatrix} \tag{15}$$

Remark 1. Note that for the orientation, the goal is to ensure that $p = \pm p_d$, which corresponds to $\tilde{p} = [0_{1\times3} \quad \pm1]^T$. Now, since the Euler parameters satisfy $\eta^2 + \varepsilon^T\varepsilon = 1$, it is sufficient to make $\tilde{\varepsilon} \to 0$ because then $\tilde{p} = [0_{1\times3} \quad \pm1]^T$. Therefore, $\tilde{\eta}$ is not included as an independent state in (15).

The goal of the tracking problem is to make the error vector (15) converge to zero. Therefore, the tracking control objective is to make $(\tilde{\xi}_q, \tilde{\zeta}) = (0, 0)$ an asymptotically stable equilibrium point of (1) and (12), which will ensure that the tracking error will converge to zero.

3. Tracking control laws

In this section, we will present the tracking control laws for the AIAUV using three different algorithms: the STA with adaptive gains, the GSTA and a PID controller. The control laws based on the GSTA obtained in (Borlaug et al., 2020) are briefly presented here for completeness. The control structure used when velocity measurements are available is presented in Fig. 3, and the control structure used when a HOSMO is used for state estimation is presented in Fig. 4.

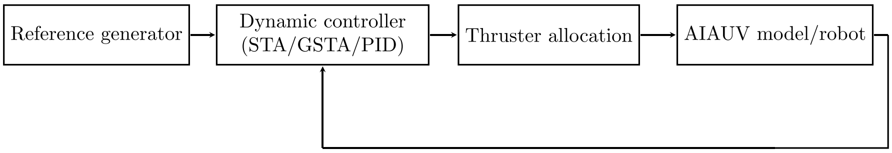


Fig. 3. Illustration of the control structure when velocity measurements are available.

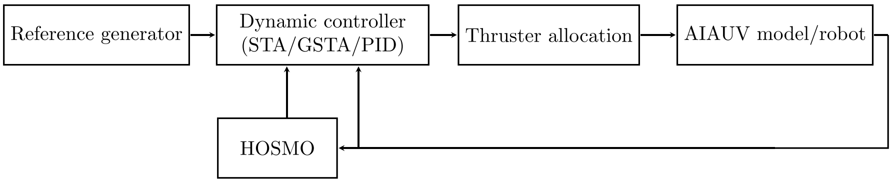


Fig. 4. Illustration of the control structure when a HOSMO is used for state estimation.

3.1. *Super-twisting algorithm with adaptive gains*

In this section, we will present two tracking control laws for the AIAUV based on the STA with adaptive gains: one where only the STA with adaptive gains is used, and one where the STA with adaptive gains is combined with an HOSMO.

3.1.1. *Control law based on the STA with adaptive gains*

In this section, a control law based on the STA with adaptive gains is presented.

Sliding surface: For SMC to be used, a sliding surface needs to be designed. The surface should be designed such that when the sliding variable σ goes to zero, the state variables asymptotically converge to zero, and such that the control input $\tau(q)$ appears in the first derivative of σ. Define $\widetilde{x}_{1,q} = \widetilde{\xi}_q$, where $\widetilde{\xi}_q$ is given by (15) and

$$\begin{aligned}\widetilde{x}_{2,q} &= \begin{bmatrix} R_B^I(\widetilde{p}) & 0_{3\times 3} & 0_{3\times(n-1)} \\ 0_{3\times 3} & \frac{1}{2}(\widetilde{\eta} I_3 + S(\widetilde{\varepsilon})) & 0_{3\times(n-1)} \\ 0_{(n-1)\times 3} & 0_{(n-1)\times 3} & I_{(n-1)\times(n-1)} \end{bmatrix} (\zeta - \zeta_d) \\ &= T_q(\widetilde{p})\widetilde{\zeta}\end{aligned} \tag{16}$$

where ζ and ζ_d are defined in (3) and (4), respectively. The sliding surface can then be chosen as

$$\sigma_q = \widetilde{x}_{1,q} + \widetilde{x}_{2,q} \quad \in \mathbb{R}^{6+(n-1)} \tag{17}$$

If $\sigma_q = 0$, we will now have $\widetilde{x}_{1,q} + \widetilde{x}_{2,q} = 0$. Since $\widetilde{x}_{2,q} = \dot{\widetilde{x}}_{1,q}$, we can write this as

$$\dot{\widetilde{x}}_{1,q} = -\widetilde{x}_{1,q} \tag{18}$$

which will ensure that $\widetilde{x}_{1,q}$ globally exponentially converges to zero. Now, since $\widetilde{x}_{1,q} = \widetilde{\xi}_q$, the state variables $\widetilde{\xi}_q$ will also globally exponentially converge to zero if $\sigma_q = 0$. To drive the sliding surface to zero, we will use the STA with adaptive gains.

Super-twisting algorithm with adaptive gains: Here, we present the equations describing the STA with adaptive gains in detail. The STA with adaptive gains proposed in (Shtessel et al., 2010) can be written by

$$\begin{aligned} u_{\mathrm{STA}}(\sigma) &= -\alpha|\sigma|^{\frac{1}{2}}\mathrm{sgn}\;(\sigma) + v \quad \in \mathbb{R}^{6+(n-1)} \\ \dot{v} &= -\beta sgn(\sigma) \end{aligned} \tag{19}$$

where the adaptive gains are defined as

$$\dot{\alpha} = \begin{cases} \omega_1\sqrt{\dfrac{\gamma_1}{2}} & \text{if } \sigma \neq 0 \\ 0, & \text{if } \sigma = 0 \end{cases} \tag{20}$$

and

$$\beta = 2\varepsilon\alpha + \lambda + 4\varepsilon^2 \tag{21}$$

where $\varepsilon \in \mathbb{R}^{6+(n-1)}$, $\lambda \in \mathbb{R}^{6+(n-1)}$, $\gamma_1 \in \mathbb{R}^{6+(n-1)}$ and $\omega_1 \in \mathbb{R}^{6+(n-1)}$ are positive constants, and σ is the sliding surface. Note that the mathematical operations in (19), (20) and (21) are performed in an element-wise manner. For implementation purposes, a small boundary is placed on the sliding surface such that the adaptive gains can be expressed as

$$\begin{aligned}\dot{\alpha} &= \begin{cases} \omega_1\sqrt{\dfrac{\gamma_1}{2}} & \text{if } |\sigma| > \alpha_m \\ 0, & \text{if } |\sigma| \leq \alpha_m \end{cases} \\ \beta &= 2\varepsilon\alpha + \lambda + 4\varepsilon^2\end{aligned} \tag{22}$$

where the design parameter α_m is a small positive constant selected empirically. This boundary is introduced because numerically σ will never be exactly zero meaning that in practical implementations the second condition of (20) will never be met, which would make the adaptive gains increase to infinity. The STA with adaptive gains makes σ and $\dot{\sigma}$ go to zero in finite time (Shtessel et al., 2010).

Control input: Now, we want the control input $\tau(q)$ to be selected such that $\dot{\sigma}_q = u_{\mathrm{STA}}$, so that σ_q and $\dot{\sigma}_q$ will reach zero in finite time. However, since we do not know anything about the model, we choose our control input to be

$$\tau(q) = u_{\mathrm{STA}}(\sigma_q) \tag{23}$$

with σ_q defined as in (17). With this choice, u_{STA} appears in $\dot{\sigma}_q$, as desired.

3.1.2. *Control law based on the STA with adaptive gains combined with a higher-order sliding mode observer*

In this section, a control law based on the STA with adaptive gains combined with an HOSMO will be derived. Using Lyapunov theory, we will show that $(\widetilde{\xi}_e, \widetilde{\zeta}) = (0, 0)$ is an asymptotically stable equilibrium point of (1) and (7) with the proposed control law.

State observer: Because velocity measurements are not available, a state observer has to be designed. We want to use the third-order sliding mode observer (SMO) presented in (Kumari et al., 2016) as it has been proven to be finite-time stable in (Moreno, 2012). To use this third-order SMO, we introduce a change of variables. Define $x_{1,e} = \xi_e$, where ξ_e is given by (6), and $x_{2,e} = J_e(\eta_2)\zeta$, where $J_e(\eta_2)\zeta$ is given by (7). The dynamics can then be written as

$$\begin{aligned}
\dot{x}_{1,e} &= x_{2,e} \\
\dot{x}_{2,e} &= \frac{d}{dt}\left(J_e\left(\eta_2\right)\right)J_e^{-1}\left(\eta_2\right)x_{2,e} + M^{-1}\left(q\right)J_e\left(\eta_2\right)\left(-C\left(q, J_e^{-1}\left(\eta_2\right)x_{2,e}\right)J_e^{-1}\left(\eta_2\right)x_{2,e} - D(q, J_e^{-1}(\eta_2)x_{2,e})J_e^{-1}(\eta_2)x_{2,e} - g\left(q, R_B^I\right) + \tau(q)\right)
\end{aligned} \tag{24}$$

Assumption 1. We assume that $\frac{d}{dt}(J_e(\eta_2))J_e^{-1}(\eta_2)x_{2,e}$ is a small bounded disturbance, which we will call $d(t)$.

We also define $f(\cdot) = -C(q, J_e^{-1}(\eta_2)x_{2,e})J_e^{-1}(\eta_2)x_{2,e} - D(q, J_e^{-1}(\eta_2)x_{2,e})$
$J_e^{-1}(\eta_2)x_{2,e} - g(q, R_B^I)$

to reduce the space used to write the model. The model can then be written as

$$\begin{aligned}
&\dot{x}_{1,e} = x_{2,e} \\
&\dot{x}_{2,e} = d(t) + M^{-1}(q)J_e(\eta_2)(f(\cdot) + \tau(q))
\end{aligned} \tag{25}$$

By designing the third-order SMO structure as in (Kumari et al., 2016), the third-order SMO can be written as

$$\begin{aligned}
&\dot{\widehat{x}}_1 = \widehat{x}_2 + z_1 \\
&\dot{\widehat{x}}_2 = \widehat{x}_3 + z_2 + M^{-1}(q)J_e(\eta_2)\tau(q) \\
&\dot{\widehat{x}}_3 = z_3
\end{aligned} \tag{26}$$

where

$$\begin{aligned}
z_1 &= k_1\left|e_1\right|^{2/3}\mathrm{sgn}(e_1) \\
z_2 &= k_2\left|e_1\right|^{1/3}\mathrm{sgn}(e_1) \\
z_3 &= k_3\ \mathrm{sgn}(e_1)
\end{aligned} \tag{27}$$

and $k_1 \in \mathbb{R}^{6+(n-1)}$, $k_2 \in \mathbb{R}^{6+(n-1)}$ and $k_3 \in \mathbb{R}^{6+(n-1)}$ are gains to be chosen according to (Levant, 1998) and (Levant, 2003), where $e_1 = x_{1,e} - \widehat{x}_1 \in \mathbb{R}^{6+(n-1)}$. One choice of parameters that satisfies the requirements in (Levant, 1998) and (Levant, 2003) is, according to (Chalanga et al., 2016), $k_1 = 6L^{1/3}$, $k_2 = 11L^{1/2}$ and $k_3 = 6L$, where $L \in \mathbb{R}^{6+(n-1)}$ is a sufficiently large constant. Note that the mathematical operations in (26) and (27) are performed in an element-wise manner. By defining $e_2 = x_{2,e} - \widehat{x}_2$ and $e_3 = -\widehat{x}_3 + F(\cdot)$, where $F(\cdot) = d(t) + M^{-1}(q)J_e(\eta_2)f(\cdot)$, the error dynamics of the HOSMO can be written as

$$\begin{aligned}
\dot{e}_1 &= -k_1\left|e_1\right|^{2/3}\mathrm{sgn}(e_1) + e_2 \\
\dot{e}_2 &= -k_2\left|e_1\right|^{1/3}\mathrm{sgn}(e_1) + e_3. \\
\dot{e}_3 &= -k_3\ \mathrm{sgn}(e_1) + \dot{F}(\cdot)
\end{aligned} \tag{28}$$

If $\left|\dot{F}(\cdot)\right| < \Delta$, then the third-order SMO errors go to zero in finite time (Moreno, 2012). Since $F(\cdot)$ is a combination of $d(t)$, $C(q, J_e^{-1}(\eta_2)x_{2,e})J_e^{-1}(\eta_2)x_{2,e}$, $D(q, J_e^{-1}(\eta_2)x_{2,e})J_e^{-1}(\eta_2)x_{2,e}$ and $g(q, R_B^I)$, and since the AIAUV is a mechanical system, these matrices will not change infinitely fast. Therefore, assuming that $\dot{F}(p, \zeta, \eta_2)$ is bounded is a valid assumption.

Sliding surface: As mentioned previously, we have to design a sliding surface to use SMC. The sliding surface is designed in the same manner as in Sec. 3.1.1. When the velocity measurement is available, we choose the sliding surface to be

$$\sigma_e = \widetilde{x}_{1,e} + \widetilde{x}_{2,e} \quad \in \mathbb{R}^{6+(n-1)} \tag{29}$$

where $\widetilde{x}_{1,e} = \widetilde{\xi}_e$, with $\widetilde{\xi}_e$ defined as in (10), and $\widetilde{x}_{2,e} = J_e(\eta_2)\zeta - J_e(\eta_{2,d})\zeta_d$, where $J_e(\eta_2)\zeta$ is defined in (7). If $\sigma_e = 0$, we will now have $\widetilde{x}_{1,e} + \widetilde{x}_{2,e} = 0$. Since $\widetilde{x}_{2,e} = \dot{\widetilde{x}}_{1,e}$, we can write this as

$$\dot{\widetilde{x}}_{1,e} = -\widetilde{x}_{1,e} \tag{30}$$

which will ensure that $\widetilde{x}_{1,e}$ globally exponentially converges to zero. Now, since $\widetilde{x}_{1,e} = \widetilde{\xi}_e$, the state variables $\widetilde{\xi}_e$ will also globally exponentially converge to zero if $\sigma_e = 0$. When the velocity measurement is not available, the observed state values are used, and we can therefore write the sliding surface with the observed values as

$$\widehat{\sigma}_e = \widehat{\widetilde{x}}_1 + \widehat{\widetilde{x}}_2 \quad \in \mathbb{R}^{6+(n-1)} \tag{31}$$

where $\widehat{\widetilde{x}}_1 = \widehat{x}_1 - \xi_{e,d}$ and $\widehat{\widetilde{x}}_2 = \widehat{x}_2 - J_e(\eta_{2,d})\zeta_d$. Since the third-order SMO errors in (28) go to zero in finite time, $\widehat{\sigma}_e = \sigma_e$ after some finite time. Thus, if $\widehat{\sigma}_e$ goes to zero, the tracking objective will be satisfied. To drive the sliding surface to zero, we will use the STA with adaptive gains.

Control input: By designing the control input $\tau(q)$ such that $\dot{\widehat{\sigma}}_e = u_{\mathrm{STA}}$, we achieve that $\widehat{\sigma}_e$ and $\dot{\widehat{\sigma}}_e$ reach zero in finite time since the STA is finite-time stable. Taking the time derivative of (31) and substituting $\dot{\widehat{x}}_1$ and $\dot{\widehat{x}}_2$, defined in (26), we find that

$$\dot{\widehat{\sigma}}_e = \dot{\widehat{\widetilde{x}}}_1 + \dot{\widehat{\widetilde{x}}}_2 = \dot{\widehat{x}}_1 - \dot{x}_{1,d} + \dot{\widehat{x}}_2 - \dot{x}_{2,d} \quad = \widehat{x}_2 + z_1 - \dot{x}_{1,d} + \widehat{x}_3 + z_2 + M(q)^{-1}J_e(\eta_2)\tau(q) - \dot{x}_{2,d} \tag{32}$$

By choosing $\tau(q)$ to be

$$\tau(q) = J_e(\eta_2)^{-1}M(q)\left(-\widehat{x}_2 - z_1 + \dot{x}_{1,d} - \widehat{x}_3 - z_2 + \dot{x}_{2,d} + u_{\mathrm{STA}}\right) \tag{33}$$

we obtain

$$\dot{\widehat{\sigma}}_e = u_{\mathrm{STA}}\left(\widehat{\sigma}_e\right). \tag{34}$$

Stability: In this section, we perform a stability analysis of the closed-loop system and show that the tracking error converges asymptotically to zero. We consider the closed-loop system defined by (1), (7) and (33). By using the fact that $\widehat{x}_1 = x_{1,e} - e_1$ and that $\widehat{x}_2 = x_{2,e} - e_2$, from Sec. 3.1.2, (31) can be written as

$$\widehat{\sigma}_e = x_{1,e} - e_1 - x_{1,d} + x_{2,e} - e_2 - x_{2,d} = \widetilde{\xi}_e - e_1 + \dot{\widetilde{\xi}}_e - e_2 \tag{35}$$

Through rearranging, we obtain that the tracking error dynamics are expressed as

$$\dot{\widetilde{\xi}}_e = -\widetilde{\xi}_e + \widehat{\sigma}_e + e_1 + e_2 \tag{36}$$

Furthermore, the velocity tracking error $\widetilde{\zeta}$ is represented by the sliding variable $\widehat{\sigma}_e$, cf., (24) and (31). The overall closed-loop dynamics with $\tau(q)$ given by (23) is thus given by $\dot{\widehat{\sigma}}_e$ given in (34), $\dot{\widetilde{\xi}}_e$ given in (36) and the state observer error given in (28). The closed-loop dynamics are thus expressed as

$$\sum_1 \begin{cases} \dot{\widetilde{\xi}}_e = -\widetilde{\xi}_e + \widehat{\sigma}_e + e_1 + e_2 \\ \dot{\widehat{\sigma}}_e = -\alpha\left|\widehat{\sigma}_e\right|^{1/2} sgn\left(\widehat{\sigma}_e\right) + v \\ \dot{v} = -\beta \text{ sgn}\left(\widehat{\sigma}_e\right) \end{cases}$$
$$\sum_2 \begin{cases} \dot{e}_1 = -k_1\left|e_1\right|^{2/3} \text{sgn}(e_1) + e_2 \\ \dot{e}_2 = -k_2\left|e_1\right|^{1/3} \text{sgn}(e_1) + e_3 \\ \dot{e}_3 = -k_3 \text{ sgn}(e_1) + \dot{F}(\cdot) \end{cases} \tag{37}$$

Theorem 1. *Consider the closed-loop system (1), (7), and (33). Assume that the HOSMO in (26) and (27) is used to estimate $x_{1,e}$ and $x_{2,e}$, where $\left|\dot{F}(\cdot)\right| < \Delta$ is assumed, and the sliding surface is selected as in (31). Then, the complete system is represented by the cascaded system in (37), and the origin of the cascaded system is uniformly globally asymptotically stable (UGAS), which ensures the asymptotic convergence of the tracking error.*

Proof. *The proof follows along the lines of the proof of* [Borlaug et al., 2019a, Theorem 1], *but for 6DOF instead of for 2DOF. 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*

$$\sum_1 \begin{cases} \dot{\widetilde{\xi}}_e = -\widetilde{\xi}_e + \widehat{\sigma}_e \\ \dot{\widehat{\sigma}}_e = -\alpha\left|\widehat{\sigma}_e\right|^{1/2} sgn\left(\widehat{\sigma}_e\right) + v \\ \dot{v} = -\beta_{\text{STA}} \text{ sgn}\left(\widehat{\sigma}_e\right) \end{cases} \tag{38}$$

This can then be divided into two subsystems:

$$\sum_{11}\left\{\dot{\widetilde{\xi}}_e = -\widetilde{\xi}_e + \widehat{\sigma}_e\right.$$
$$\sum_{12} \begin{cases} \dot{\widehat{\sigma}}_e = -\alpha\left|\widehat{\sigma}_e\right|^{1/2} sgn\left(\widehat{\sigma}_e\right) + v \\ \dot{v} = -\beta_{\text{STA}} \text{ sgn}\left(\widehat{\sigma}_e\right) \end{cases} \tag{39}$$

where [Loría and Panteley, 2005, Lemma 2.1] can be used. Subsystem Σ_{11} with $\widehat{\sigma}_e = 0$ is analysed first, which is clearly a globally exponentially stable linear system, but since we will need a Lyapunov function to analyse this system when $\widehat{\sigma}_e \neq 0$, we use the Lyapunov function candidate $V_{11}(\widetilde{\xi}_e) = \frac{1}{2}\widetilde{\xi}_e^2$ for the analysis. The derivative of V_{11} yields

$$\dot{V}_{11}\left(\widetilde{\xi}_e\right) = \widetilde{\xi}_e\dot{\widetilde{\xi}}_e = \widetilde{\xi}_e\left(-\widetilde{\xi}_e\right) = -\widetilde{\xi}_e^2 \leq -\left|\left|\widetilde{\xi}_e\right|\right|^2 \tag{40}$$

This indicates that the Lyapunov function satisfies:

$$\begin{aligned} &k_1||x||^a \leq V_{11}(x) \leq k_2||x||^a \\ &\frac{\partial V_{11}}{\partial x}f_{11}(t,x) \leq -k_3\left|\left|x\right|\right|^a \end{aligned} \tag{41}$$

with $k_1 = k_2 = \frac{1}{2}$, $k_3 = 1$ and $a = 2$. Hence, by virtue of [Khalil, 2002, Theorem 4.10], the origin for subsystem Σ_{11} with $\widehat{\sigma}_e = 0$ is globally exponentially stable.

Subsystem Σ_{12} has the structure of the STA with adaptive gains. In (Shtessel et al., 2010), 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 Σ_{12} and that for any initial conditions, $\widehat{\sigma}_e, \dot{\widehat{\sigma}}_e \rightarrow 0$ in finite time by using the STA with adaptive gains given by Eq. (20) and Eq. (21), where ε, λ, γ_1 and ω_1 are arbitrary positive constants. The sliding surface $\widehat{\sigma}_e = 0$ is also proven to be reached in finite time. Now, since the subsystem is GFTS and autonomous, it is also UGAS by [Polyakov and Fridman, 2014, Proposition 2 and Proposition 3], which also implies that $||\widehat{\sigma}_e(t)|| < \beta_1 \forall t \geq 0$.

To verify that the solutions of Σ_1 are uniformly globally bounded (UGB), subsystem Σ_{11} must be analysed with $\widehat{\sigma}_e \neq 0$. The derivative of the Lyapunov function V_{11} is then as follows:

$$\begin{aligned} &\dot{V}_{11}\left(\widetilde{\xi}_e\right) = -\left|\left|\widetilde{\xi}_e\right|\right|^2 + \widehat{\sigma}_e\widetilde{\xi}_e \\ &\leq -\left|\left|\widetilde{\xi}_e\right|\right|^2 + \theta\left|\left|\widetilde{\xi}_e\right|\right|^2 - \theta\left|\left|\widetilde{\xi}_e\right|\right|^2 + \beta_1\left|\left|\widetilde{\xi}_e\right|\right| \\ &\leq -\left(1-\theta\right)\left|\left|\widetilde{\xi}_e\right|\right|^2 \quad \forall \quad \left|\left|\widetilde{\xi}_e\right|\right| \geq \frac{\beta_1}{\theta} \end{aligned} \tag{42}$$

where $0 < \theta < 1$. The solutions are then UGB because the conditions of [Khalil, 2002, Theorem 4.18] are satisfied. Consequently, the conditions of [Loría and Panteley, 2005, Lemma 2.1] are satisfied, which implies that the origin of subsystem Σ_1 is UGAS.

Analysis of subsystem 2: In (Moreno, 2012) a Lyapunov function is proposed for a third-order observer. The Lyapunov function is proven to be radially unbounded and positive definite and that it is a Lyapunov function for subsystem Σ_2, whose trajectories converge in finite time to the origin $e = 0$ for every value of $\left|\dot{F}(t)\right|$ as long as $\dot{F}(t)$ is bounded. Since $\dot{F}(t)$ is bounded by assumptions, the origin is GFTS for every value of $\dot{F}(t)$, which means that the origin is also UGAS by [Polyakov and Fridman, 2014, Proposition 2 and Proposition 3], which in turn implies $||e(t)|| \leq \beta_2 \forall t \geq 0$.

Analysis of the complete system: To analyse the complete system [Loría and Panteley, 2005, Lemma 2.1] is used. To check if the solutions of the complete system are UGB, the boundedness of $\widetilde{\xi}_e$ must be evaluated when $e_1 \neq 0$ and $e_2 \neq 0$, and for this, the Lyapunov function V_{11} is used. Note that the boundedness of $\widehat{\sigma}_e$ follows from Σ_{12} being UGAS because Σ_{12} is not perturbed by Σ_2.

$$\begin{aligned} &\dot{V}_{11}\left(\widetilde{\xi}_e\right) = -\left|\left|\widetilde{\xi}_e\right|\right|^2 + \left(\widehat{\sigma}_e + e_1 + e_2\right)\widetilde{\xi}_e \\ &\leq -\left|\left|\widetilde{\xi}_e\right|\right|^2 + \theta\left|\left|\widetilde{\xi}_e\right|\right|^2 - \theta\left|\left|\widetilde{\xi}_e\right|\right|^2 + (\beta_1 + 2\beta_2)\left|\left|\widetilde{\xi}_e\right|\right| \\ &\leq -\left(1-\theta\right)\left|\left|\widetilde{\xi}_e\right|\right|^2 \quad \forall \quad \left|\left|\widetilde{\xi}_e\right|\right| \geq \frac{\beta_1 + 2\beta_2}{\theta} \end{aligned} \tag{43}$$

where $0 < \theta < 1$. The solutions are then UGB because the conditions of [Khalil, 2002, Theorem 4.18] are satisfied. Consequently, the conditions of [Loría and Panteley, 2005, Lemma 2.1] are satisfied, which implies that the complete system is UGAS.

3.2. Generalized super-twisting algorithm

In this section, two tracking control laws for the AIAUV based on the GSTA will be presented: one where only the GSTA is used, and one where the GSTA is combined with an HOSMO. These control laws were derived and proven to achieve the tracking control objective in (Borlaug et al., 2020), but we briefly present them here for completeness.

3.2.1. Control law based on the GSTA

In this subsection, the tracking control law for the AIAUV using the GSTA will be presented. In (Borlaug et al., 2020), the control input that achieves the tracking control objective was found to be

$$\tau(q) = T_q^{-1}(\widetilde{p})u_{\text{GSTA}}(\sigma) \tag{44}$$

where $\widetilde{p}$ is defined as in (14), $T_q(\widetilde{p})$ is defined as in (16), σ is defined as in (17) and $u_{\text{GSTA}}(\sigma)$ is defined as in (Castillo et al., 2018):

$$\begin{aligned} u_{\text{GSTA}}(\sigma) &= -k_1\varphi_1(\sigma) + z \quad \in \mathbb{R}^{6+(n-1)} \\ \dot{z} &= -k_2\varphi_2(\sigma) \end{aligned} \tag{45}$$

where

$$\begin{aligned} \varphi_1(\sigma) &= \lceil \sigma \rfloor^{\frac{1}{2}} + \beta_{\text{GSTA}}\,\sigma \\ \varphi_2(\sigma) &= \frac{1}{2}\lceil \sigma \rfloor^{0} + \frac{3}{2}\beta_{\text{GSTA}}\lceil \sigma \rfloor^{\frac{1}{2}} + \beta_{\text{GSTA}}^2\sigma \end{aligned} \tag{46}$$

where $\lceil a \rfloor^{b} = |a|^{b}\text{sgn}(a)$, where b is a constant, and $k_1 \in \mathbb{R}^{6+(n-1)}$, $k_2 \in \mathbb{R}^{6+(n-1)}$ and $\beta_{\text{GSTA}} \in \mathbb{R}^{6+(n-1)}$ are controller gains. Note that the mathematical operations in (45) and (46) are performed in an element-wise manner. The two extra terms in $\varphi_2(\sigma)$ compared to the STA with adaptive gains in (19) provide one more parameter to adjust when designing the GSTA gains: k_1, k_2 and β_{GSTA}. The linear growth term $\beta_{\text{GSTA}}\sigma$ in φ_1 helps to counteract the effects of state-dependent perturbations, which can exponentially increase in time. In (Borlaug et al., 2020), it was proven that the control input (44) makes the closed-loop system UGAS and that the tracking errors for the complete system converge asymptotically to zero.

3.2.2. Control law based on the GSTA with a higher-order sliding mode observer

In this subsection, the tracking control law for the AIAUV using the GSTA combined with an HOSMO is presented. The HOSMO that is used is the third-order SMO presented in Sec. 3.1.2. In (Borlaug et al., 2020), the control input that achieves the tracking control objective was found to be

$$\tau(q) = J_e^{-1}(\cdot)M(\cdot)\Big(-\widehat{x}_2 - z_1 + \dot{x}_{1,d} - \widehat{x}_3 - z_2 + \dot{x}_{2,d} + u_{\text{GSTA}}(\sigma)\Big) \tag{47}$$

where σ is as in (31) and where $u_{\text{GSTA}}(\sigma)$ is defined as in (Castillo et al., 2018), i.e., (45). In (Borlaug et al., 2020), it was proven that the control input (44) makes the closed-loop system UGAS and that the tracking errors for the complete system converge asymptotically to zero.

Table 1
Eelume link properties.

Link nr.	Length [m]	Volume [m^3]	Thrusters
1	0.62	0.0158	0
2, 4, 6, 8	0.10	0.0025	0
3	0.59	0.0150	2: Z, Y
5	0.80	0.0204	3: X, X, Z
7	0.59	0.0150	2: Y, Z
9	0.37	0.0094	0

3.3. PID controller

To benchmark the two SMC algorithms, we also include a PID controller in the comparison. Therefore, in this section, we propose two tracking control laws for the AIAUV based on the PID controller: one where we assume that the velocity is known, and one where we use the estimated velocity from the HOSMO.

3.3.1. Control law based on the PID controller

In this subsection, the tracking control law for the AIAUV using the PID controller is presented. The control input when the PID controller is used is defined as

$$\tau(q) = -k_p\widetilde{\xi}_e - k_d\widetilde{\zeta} - k_i\int \widetilde{\xi}_e dt \tag{48}$$

where $\widetilde{\xi}_e$ is defined as in (10) and $\widetilde{\zeta} = \zeta - \zeta_d$, where ζ and ζ_d are defined as in (3) and (4), respectively. The constants k_p, k_d and k_i are controller gains.

3.3.2. Control law based on the PID controller combined with a higher-order sliding mode observer

In this subsection, the tracking control law for the AIAUV using the PID controller with the estimated velocity from the HOSMO is presented. The control input when the PID controller with the velocity from the third-order SMO from Sec. 3.1.2 is used can be found as

$$\tau(q) = -k_p\widetilde{\xi}_e - k_d\widehat{\widetilde{\zeta}} - k_i\int \widetilde{\xi}_e dt \tag{49}$$

where $\widetilde{\xi}_e$ is defined as in (10) and $\widehat{\widetilde{\zeta}} = J_e(\eta_2)^{-1}\widehat{x}_2 - \zeta_d$, where $\widehat{x}_2$ and ζ_d are defined as in (26) and (4), respectively. The constants k_p, k_d and k_i are controller gains.

4. Simulation results

In this section, we present the implemented simulation model, the test cases and the simulation results for trajectory tracking using the control laws presented in Sec. 3.

4.1. Implementation

The AIAUV model is implemented using the method described in (Schmidt-Didlaukies et al., 2018) in MATLAB Simulink together with the controllers. The implemented AIAUV is based on the Eelume robot, Fig. 1, which is the robot used in the experiments presented in Sec. 5. The AIAUV has $n + 1 = 9$ cylindrical links with a radius of 0.09 m, $n = 8$ revolute joints and $m = 7$ thrusters. The link frames are right-hand coordinate systems, in which the completely outstretched robot is placed such that the x-axes point forward and the z-axes point upwards. In Table 1, the properties of each link are presented. In the thrusters column, "2: Z, Y" means that the link has 2 thrusters, one imposing a force in the z-direction and one imposing a force in the y-direction. The maximum thrust of each thruster is approximately 50 N. Joints 1, 3, 5 and 7 rotate around the z-axis, and joints 2, 4, 6 and 8 rotate around the y-axis. The joint rotation occurs in the link frame of the corresponding link, i.e., joint 1 rotates around the z-axis of link 1.

To ensure a valid comparison between the simulations and the experiments, we use a P-controller for the joints. The reason for this choice is that the Eelume robot has an internal joint controller, which is a P-controller. To create a continuous trajectory, we do not use inverse kinematic control, as in (Borlaug et al., 2018) and (Borlaug et al., 2019b); rather, we use a filter to generate a continuous trajectory between setpoints. The reason is that the joint controller on the Eelume robot we used is not sufficiently precise to satisfy the assumption that $q = q_d$, which is a prerequisite for the inverse kinematic control to perform

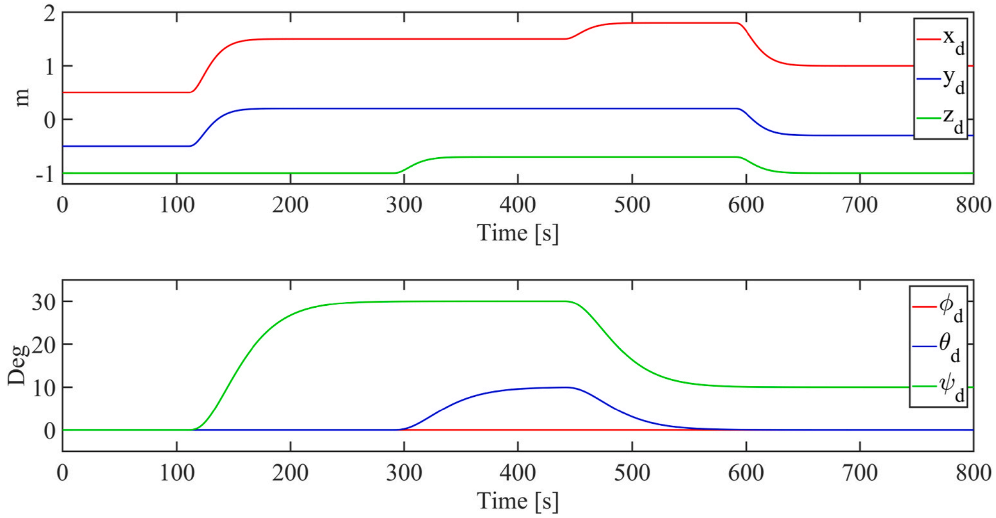


Fig. 5. Reference trajectories for position and orientation.

properly. The thruster allocation matrix is implemented as proposed in (Sverdrup-Thygeson et al., 2018). For the simulations, the ode3 fixed-step solver with a step size of 0.002 was used.

4.2. Simulation cases

In this section, we describe the different test cases considered in the simulations. To validate and compare the control laws presented in Sec. 3, we created two different test cases to represent the operation mode, a mode that is thoroughly explained in (Sverdrup-Thygeson et al., 2018). In operation mode, we perform inspection and intervention tasks; therefore, accuracy is extremely important. The robot that we used during the experiments is unactuated in roll when the AIAUV is straight (all joint angels are equal to zero), and accurate control of rolling when the robot is straight is therefore not possible. We therefore found a shape, which we call C-shape, where the robot is actuated in every DOF such that we can perform accurate control in every DOF. We therefore have one test case called C-shape, which is explained in detail in Sec. 4.2.1. We also wanted to include a test case where the robustness of the algorithms would be evaluated. We therefore wanted to introduce some disturbance in our second test. However, we do not have access to a tank where we can introduce current. We therefore introduce disturbance by using the joints. Thus, the second test case is the C-shape with a moving head, which is explained in detail in Sec. 4.2.2. This test captures operation mode where the robot performs an inspection by looking around in a circular motion with its head, and the motion also introduces a disturbance.

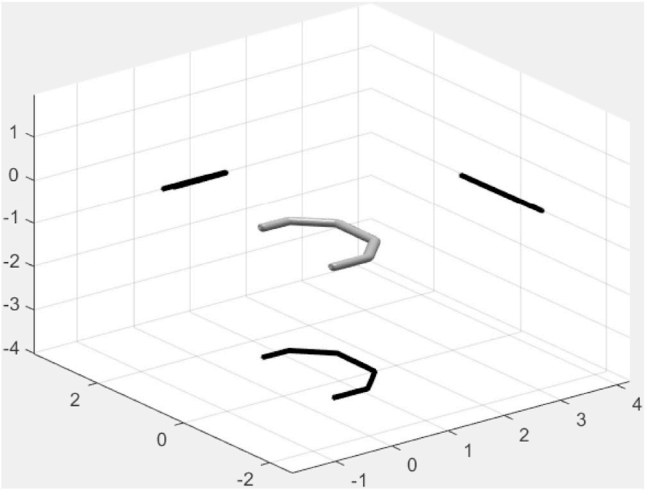


Fig. 6. Configuration in the C-shape.

The reference trajectories for the position and orientation used for the two test cases are shown in Fig. 5. The references for x, y and ψ are changed at 100 s, and at 300 s, the references for z and θ are changed. The references for x, ψ and θ are also changed at 450 s and for x, y and z at 600 s. The reference value for z is, however, slightly different from case to case because during the experiments, we observed that some of the joints were emerging from the water and creating problems for the measurement system when we started the trajectory at -1 for the C-shape with a moving head. The reference value for z is therefore as in Fig. 5 for the C-shape, i.e., the reference value starts at -1 and changes to $-$ 0.7, whereas for the C-shape with a moving head, the reference value starts at -0.7 and changes to $-$ 0.5. The z reference for the C-shape with a moving head when the control law with the HOSMO is used is also changed at 100 s, i.e., the reference value starts at $-$ 1, changes to -0.7 at 100 s, and then changes to -0.5 at 300 s.

4.2.1. C-shape

In the C-shape configuration, the AIAUV is shaped like a C. In this shape, the AIAUV is actuated in every DOF since the thrusters are oriented such that the robot can provide thrust in all 6DOFs, and the C-shape is therefore ideal for the operation mode. In the C-shape, the joint angles references are set to $q_d = [45\text{deg}\ 0\text{deg}\ 45\text{deg}\ 0\text{deg}\ 45\text{deg}\ 0\text{deg}\ 45\text{deg}\ 0\text{deg}]^T$, and in Fig. 6, the configuration of the robot is shown.

4.2.2. C-shape with a moving head

The case of the C-shape with a moving head is almost identical to the C-shape case, but in this case, we move the head of the AIAUV around. Specifically, joint nr. 7 and joint nr. 8 are moved such that the camera positioned at joint nr. 8 (the head), is looking around. The joint angles together with their reference trajectories for the C-shape with a moving head are presented in Fig. 7.

4.3. Simulation results

In this section, the results from the performed simulation test cases are presented. To ensure that the comparison between the simulations and experiments is as fair as possible, the gains used during the simulations are the gains found during the experiments. The gains for the STA with adaptive gains and the HOSMO are presented in Table 2; for the GSTA and the HOSMO, the gains are presented in Table 3; and for the

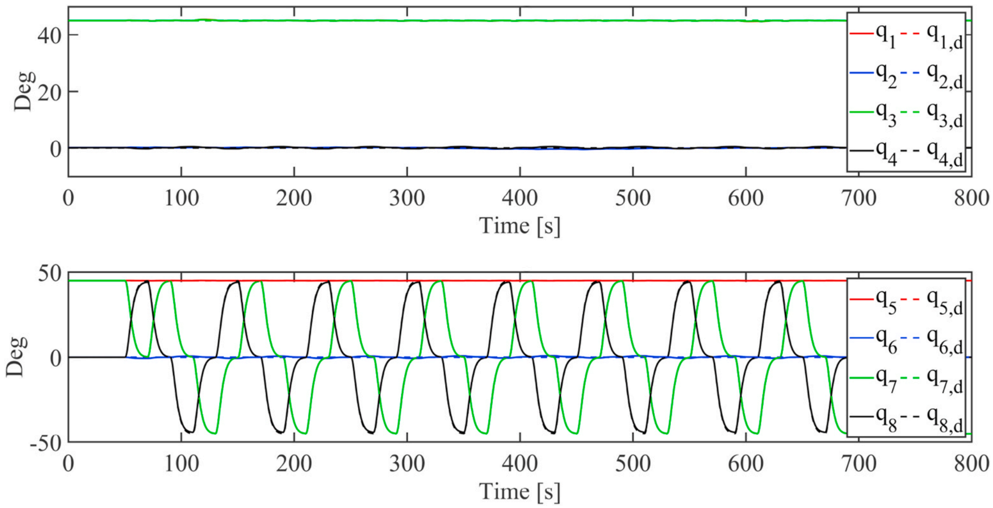


Fig. 7. Simulation: The joint angles and their reference trajectories for the C-shape with a moving head.

Table 2
Simulation: Control gains for the STA with adaptive gains.

Gains	Tests		
	C-shape	C-shape w/moving head without HOSMO	C-shape w/moving head with HOSMO
ε	$1\cdot 10^{-6}$	$1\cdot 10^{-6}$	$1\cdot 10^{-6}$
λ	0.05	0.05	0.05
γ_1	0.5	0.5	0.5
ω_1	0.4	0.4	0.2
L	0.1	HOSMO not used	0.1

Table 3
Simulation: Control gains used for the GSTA.

Gains	Tests	
	C-shape	C-shape w/moving head
k_1	1	1
k_2	0.0006	0.0006
β	24	24
L	0.1	0.1

Table 4
Simulation: Control gains used for the PID controller.

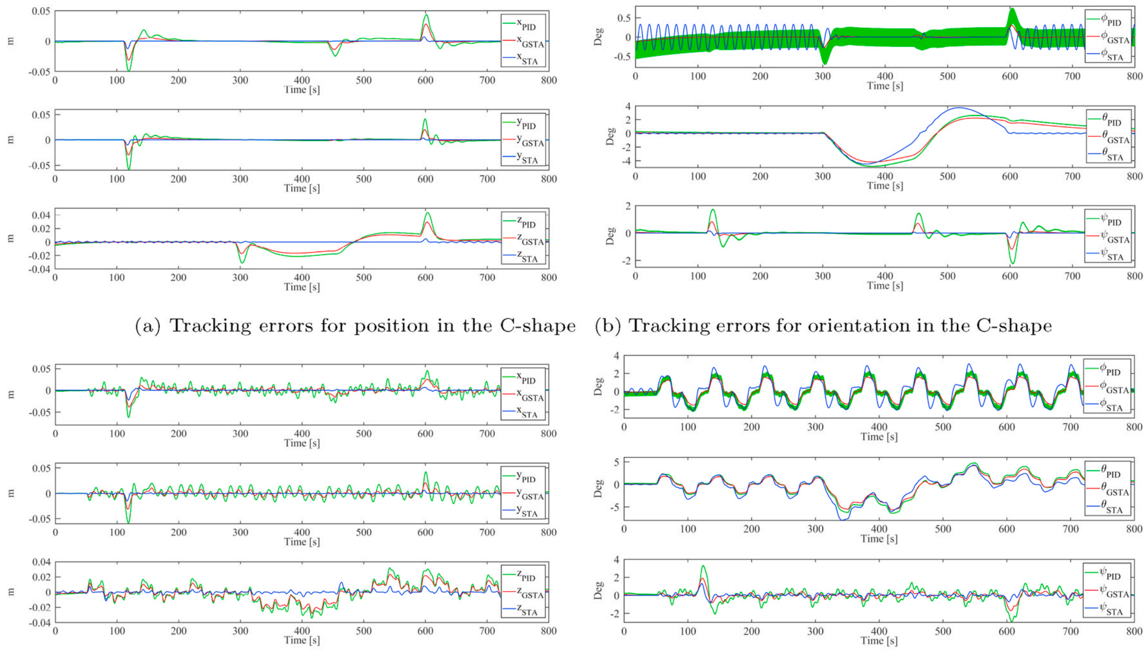
Gains	Tests	
	C-shape	C-shape w/moving head
k_p	20	20
k_d	0.4	0.4
k_i	0.1	0.1
L	0.01	0.01

PID controller, the gains are presented in Table 4. For the SMC algorithms, the gains for the HOSMO, L, are different from those used during the experiments for the C-shape and the C-shape with a moving head because we observed chattering in the simulations when we used $L = 0.01$, which was not experienced in the experiments.

The trajectory tracking results using the control laws from Sec. 3.1.1, Sec. 3.2.1 and Sec. 3.3.1, i.e., without the HOSMO, are presented in Fig. 8. The trajectory tracking results include both the position errors and the orientation errors for both test cases. The corresponding thruster forces are shown for the C-shape in Fig. 9 and for the C-shape with a moving head in Fig. 10. In Table 5, the root mean square error (RMSE) and maximum error for the position and orientation are given for each control law without the HOSMO and for both simulation test cases, i.e., the C-shape and the C-shape with a moving head. In Table 6, the root mean square (RMS) for the thruster forces and the derivative of the thruster forces are given for each control law without the HOSMO and for both simulation test cases. The RMS of the thruster forces provides an idea of how much force is used, and the RMS of the derivative of the thruster forces provides a measure of how much chattering is present in the thruster forces. The trajectory tracking results using the control laws from Sec. 3.1.2, Sec. 3.2.2 and Sec. 3.3.2, i.e., with the HOSMO, are presented in Fig. 11. The corresponding thruster forces are shown for the C-shape in Fig. 12 and for the C-shape with a moving head in Fig. 13. In Table 7, the RMSE and maximum error for the position and orientation are given for each control law with the HOSMO and for both simulation test cases. In Table 8, the RMS for the thruster forces and the derivative of the thruster forces are given for each control law with the HOSMO and for both simulation test cases. In Fig. 14 and Fig. 15 the estimation errors for the HOSMO are shown for the C-shape and the C-shape with a moving head, respectively. The first sub-plot shows $e_1 = x_{1,e} - \hat{x}_1$, and the second sub-plot shows $e_2 = x_{2,e} - \hat{x}_2$. Note that if higher gains had been used in the simulations, the results would have improved, at least for the GSTA.

4.4. Discussion

As shown in Fig. 8, the AIAUV follows the given position and orientation very well for both test cases as all errors are usually below $0.5 - 2$ cm in position and $1 - 3$deg in orientation, when the control laws from Sec. 3.1.1, Sec. 3.2.1 and Sec. 3.3.1, i.e., without the HOSMO, are used. This result indicates that all the control laws are applicable. As shown in Figs. 8a and 8b, in the C-shape case, the STA with adaptive gains provides the lowest error and therefore the best result overall, and the GSTA provides the second best results. The reason why the STA with adaptive gains provides better results than the GSTA is likely because it adaptively finds the best gains, whereas the GSTA must be tuned manually. However, if we more closely examine some of the results for φ and θ, we do find that for φ, the GSTA provides the lowest error, and for θ, GSTA and PID provide a slightly lower error than the STA with adaptive gains. From Figs. 8c and 8d, we observe the same tendencies for the C-shape with a moving head as for the C-shape case, i.e., in x, y, z and ψ, the STA with adaptive gains provides the smallest error, and the GSTA is the second best. However, for φ and θ, the errors are much more similar. Examining Table 5 confirms what we can observe from Fig. 8, i.



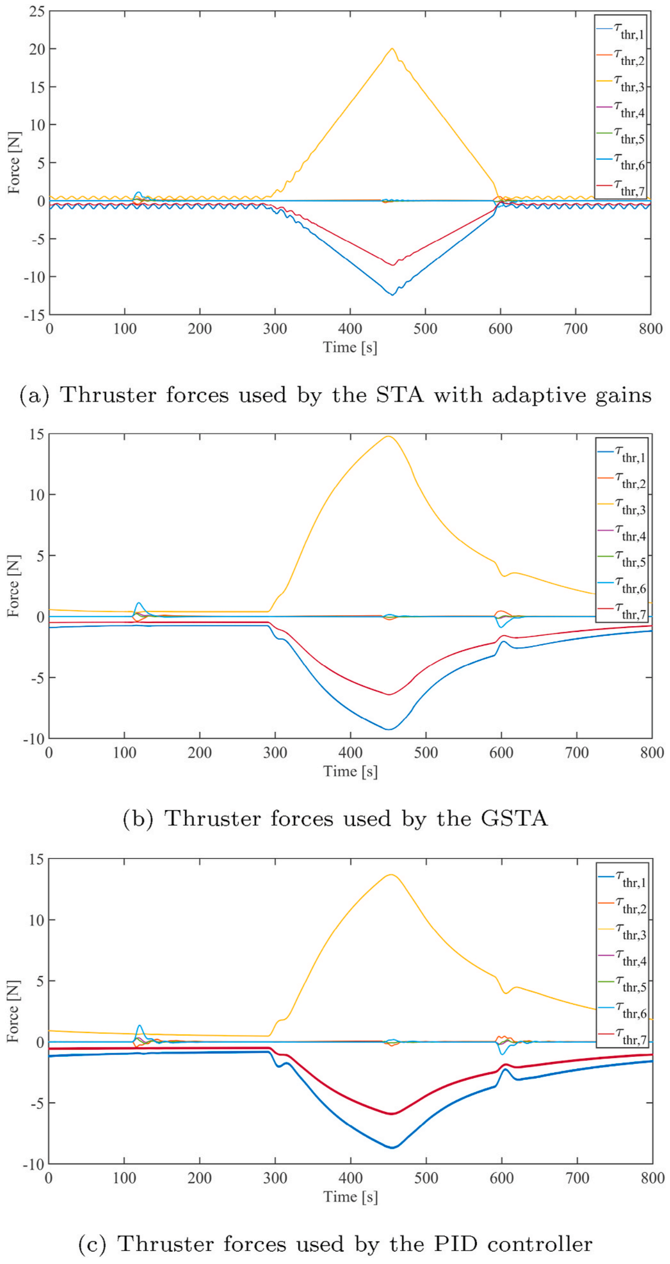
(a) Tracking errors for position in the C-shape (b) Tracking errors for orientation in the C-shape

(c) Tracking errors for position in the C-shape with a moving head (d) Tracking errors for orientation in the C-shape with a moving head

Fig. 8. Simulation: Trajectory tracking results using the control laws without the HOSMO.

e., that the STA with adaptive gains provides the smallest position errors, with an RMSE value of 0.0008 m for the C-shape and 0.0019 m for the C-shape with a moving head, and that the GSTA is the second best, with an RMSE value of 0.0054 m for the C-shape and 0.0068 m for the C-shape with a moving head, while the PID controller gives the largest position errors, with an RMSE value of 0.0087 m for the C-shape and 0.0115 m for the C-shape with a moving head. For the orientation errors however, the GSTA gives an RMSE value of 0.0114 rad for the C-shape and 0.0193 rad for the C-shape with a moving head, while the STA with adaptive gains gives an RMSE value of 0.0116 rad for the C-shape and 0.0228 rad for the C-shape with moving head, so the results are much more similar, and the GSTA actually provides slightly better results. The PID controller clearly offers the worst tracking performance for orientation, with an RMSE value of 0.0154 rad for the C-shape and 0.0249 rad for the C-shape with a moving head. We see the same results for the maximum error value. If we take the thruster use of the different algorithms into consideration, presented in Figs. 9 and 10, we find that for both test cases, the STA with adaptive gains uses more force than the GSTA and the PID controller, while the GSTA and the PID use very similar amounts of force. However, the RMS of τ_{thr} in Table 6 shows that over time, the amount of force used is not actually very different. The STA with adaptive gains uses slightly more force in the C-shape case, with an RMS value of 2.16, while the GSTA and the PID controller uses a very similar amount of force with RMS values of 1.84 and 1.80, respectively. For the case of the C-shape with a moving head, however, all the algorithms use a very similar amount of force with RMS values between $1.87-1.91$. From the RMS values of $\dot{\tau}_{thr}$, we find that the most chattering occurs in the control input from the STA with adaptive gains, with RMS value of 5.32 for the C-shape and 2.19 for the C-shape with a moving head, and that less chattering occurs in that from the GSTA, with RMS value of 5.05 for the C-shape and 0.31 for the C-shape with a moving head, while the control input from the PID controller shows the least rapid changes in the thrust forces, with RMS value of 0.74 for the C-shape and 0.67 for the C-shape with a moving head. However, the value of the RMS of $\dot{\tau}_{thr}$ is so small that the chattering that is introduced by the STA with adaptive gains is negligible.

From Figs. 14 and 15 we can see that the estimation errors for the HOSMO are quite small, the HOSMO is therefore applicable for state estimation. When the control laws with the HOSMO are used, i.e., the control laws from Sec. 3.1.2, Sec. 3.2.2 and Sec. 3.3.2, we observe the same tendencies in Fig. 11 as when the control laws without the HOSMO are used. As shown in Figs. 11a and 11b, the STA with adaptive gains provides the smallest errors for x, y, z and ψ, and the GSTA yields the second smallest errors. For φ, however, the GSTA provides the smallest error, while the STA with adaptive gains results in oscillations. For θ, the results are more similar between the STA with adaptive gains and the GSTA, while the PID controller gives rise to a larger error. As shown in Figs. 11c and 11d, the results for the C-shape with a moving head are the same as those for the C-shape case, except that the results for φ and θ are now more similar. Examining Table 7 confirms what we can observe from Fig. 11, i.e., that the STA with adaptive gains provides the best results for position, with an RMSE value of 0.0009 m for the C-shape and 0.0032 m for the C-shape with a moving head, and that the GSTA is the second best, with an RMSE value of 0.0059 m for the C-shape and 0.0075 m for the C-shape with a moving head, while the PID controller gives the largest position errors, with an RMSE value of 0.0083 m for the C-shape and 0.0112 m for the C-shape with a moving head. For the orientation errors however, the GSTA gives an RMSE value of 0.0072 rad for the C-shape and 0.0125 rad for the C-shape with a moving head, while the STA with adaptive gains gives an RMSE value of 0.0078 rad for the C-shape and 0.0198 rad for the C-shape with moving head, so the results are much more similar, and the GSTA actually provides slightly better results. The PID controller clearly offers the worst tracking performance for orientation, with an RMSE value of 0.0147 rad for the C-shape and 0.0245 rad for the C-shape with a moving head. We see the same results for the maximum error value. If we take the thruster use of the different algorithms into consideration, presented in Figs. 12 and 13, we find that the STA with adaptive gains also uses more force for both test cases, whereas the GSTA and the PID controller use quite similar amounts of force. Over time, however, as seen in Table 8 from the RMS value of τ_{thr}, the amount of force used is not actually very different. The STA with adaptive gains uses slightly more force in the C-shape case, with an RMS value of 2.25, while the GSTA uses the second most amount of force with an RMS value of 2.11, while the PID controller uses the

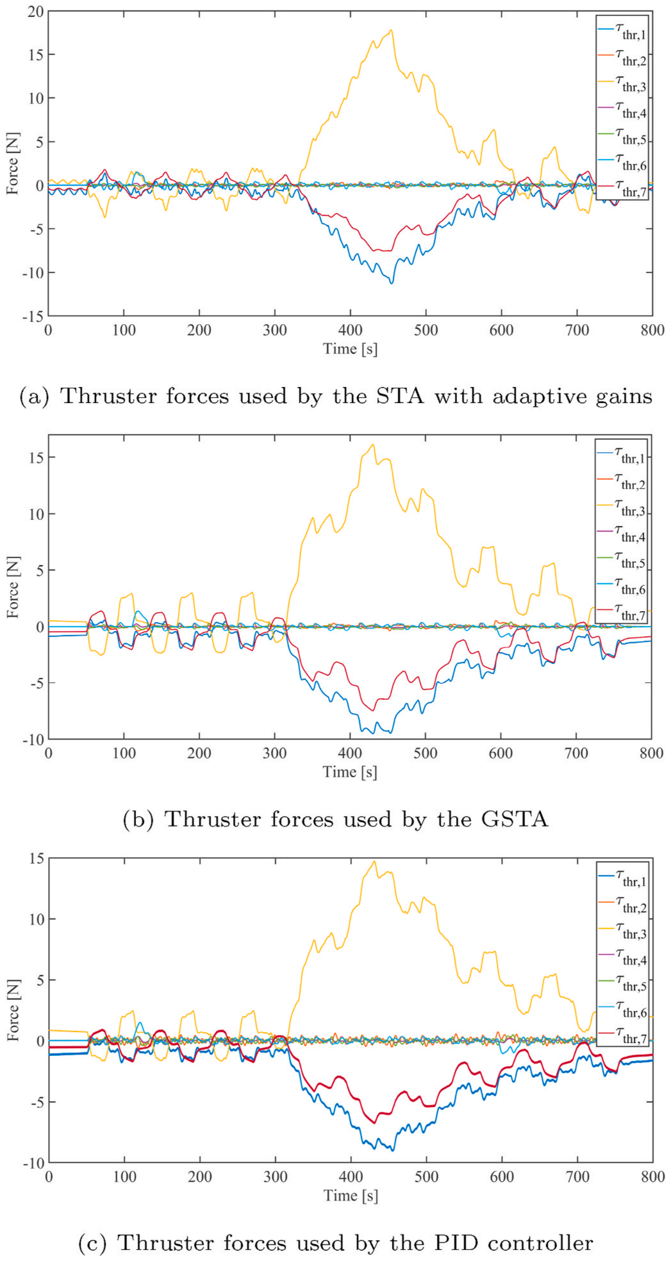


(a) Thruster forces used by the STA with adaptive gains

(b) Thruster forces used by the GSTA

(c) Thruster forces used by the PID controller

Fig. 9. Simulation: Thruster forces used in the C-shape when the control laws without the HOSMO are used.



(a) Thruster forces used by the STA with adaptive gains

(b) Thruster forces used by the GSTA

(c) Thruster forces used by the PID controller

Fig. 10. Simulation: Thruster forces used in the C-shape with a moving head when the control laws without the HOSMO are used.

least amount of force with an RMS value of 1.82. For the C-shape with a moving head however, the GSTA actually uses the most amount of force with an RMS value of 2.22, and the STA with adaptive gains uses the second most amount of force with an RMS value of 2.05. The PID controller does however, also for the C-shape with a moving head case use the least amount of force with an RMS value of 1.89. From the RMS values of $\dot{\tau}_{thr}$, we find that the most chattering occurs in the control input from the STA with adaptive gains, with RMS value of 15.78 for the C-shape and 9.48 for the C-shape with a moving head, and that less chattering occurs in that from the GSTA, with RMS value of 10.99 for the C-shape and 7.90 for the C-shape with a moving head, while the control input from the PID controller shows the least rapid changes in the thrust forces, with RMS value of 1.43 for the C-shape and 1.37 for the C-shape with a moving head. However, the value of the RMS of $\dot{\tau}_{thr}$ is so small that the chattering that is introduced by the STA with adaptive gains is negligible.

Table 5
Simulation: Comparison of the results when the control laws without the HOSMO are used.

Algorithm		Root Mean Square Error (RMSE)		Maximum error	
		C-shape	C-shape w/ moving head	C-shape	C-shape w/ moving head
STA	Position [m]	0.0008	0.0019	0.0090	0.0167
STA	Orientation [rad]	0.0116	0.0228	0.0301	0.0720
GSTA	Position [m]	0.0054	0.0068	0.0303	0.0327
GSTA	Orientation [rad]	0.0114	0.0193	0.0330	0.0535
PID	Position [m]	0.0087	0.0115	0.0509	0.0521
PID	Orientation [rad]	0.0154	0.0249	0.0455	0.0699

Table 6
Simulation: Comparison of the thruster forces used when the control laws without the HOSMO are used.

Algorithm	Root Mean Square (RMS)			
	τ_{thr}		$\dot{\tau}_{thr}$	
	C-shape	C-shape w/moving head	C-shape	C-shape w/moving head
STA	2.16	1.89	5.32	2.19
GSTA	1.84	1.91	5.05	0.31
PID	1.80	1.87	0.74	0.67

If we compare the results obtained with the control laws from Sec. 3.1.1, Sec. 3.2.1 and Sec. 3.3.1, i.e., without the HOSMO, and the results obtained with the control laws from Sec. 3.1.2, Sec. 3.2.2 and Sec. 3.3.2, i.e., when the HOSMO is used, by comparing Table 5 with Table 7, Table 6 and Table 8, we find that when the HOSMO is used, we obtain better tracking results for the orientation for all the algorithms, as well as better tracking for position when the PID controller is used. Additionally, note that the difference between the position errors, i.e., with and without the HOSMO, is smaller than the difference between the orientation errors, i.e., the control laws with the HOSMO provide the best performance overall. The reason for the better tracking performance is likely due to the increase in thruster use. The thruster use is, however, well within the boundaries of what the Eelume AIAUV can provide, which is 50 N.

To conclude, we find that we obtain the best tracking performance in position by using the STA with adaptive gains, and the GSTA provides the best tracking performance in orientation. If we simultaneously consider the errors for position and orientation, we find that the STA with adaptive gains provides the best tracking performance, but a large difference does not occur between the STA with adaptive gains and the GSTA. The reason for this difference is likely that we have to tune the GSTA manually, whereas the method with adaptive gains adaptively finds the best gains and thus has an advantage. If we use the HOSMO to estimate the linear and angular velocities in the control laws, we improve our results.

5. Experimental investigation

This section describes the Eelume robot used during the experiments, the experimental set-up and the obtained results.

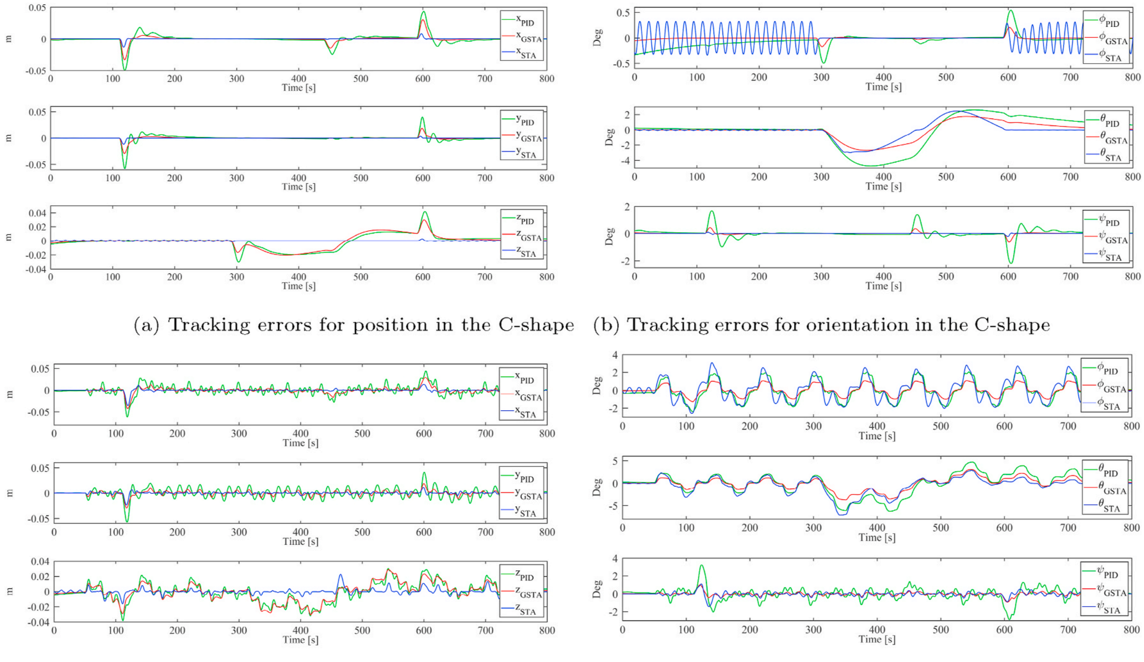
5.1. *The Eelume robot*

The AIAUV robot "Eely", which was used during the experiments, was the first AIAUV created by the Eelume company, and the AIAUV is described in detail in (Liljebäck and Mills, 2017), which is the AIAUV on which the implemented simulation model is based. The Eelume AIAUV has internal joint controllers, which are P-controllers. Therefore, during the experiments, we could only provide a desired joint reference q_d to the AIAUV. We thus also used a P-controller in the simulations, as explained in Sec. 4.1, for the fairest possible comparison between the simulations and experiments.

5.2. *Experimental set-up*

We performed the experiments at the MC-lab at NTNU, Trondheim, Norway (Marine cybernetics labora, 2018). The tank located in the MC-lab has the following dimensions: length of 40m, height of 1.5m and width of 6.45m. The underwater motion capture system from Qualisys (Qualisys-motion capture s, 2018) is installed in the basin to provide us with real-time measurements of the position and orientation of the base of the AIAUV. The Qualisys system has six identical cameras, allowing us to track reflective markers attached to the base of the AIAUV underwater, as shown in Fig. 16. The cameras allow us to track the reflective markers inside a working area with dimensions of 10m × 1.35m × 5.45m.

The system structure used during the experiments is illustrated in Fig. 17. The six cameras (indicated by blue boxes in Fig. 17), tracked reflective markers in the working area (indicated by the red box in Fig. 17). The Qualisys system or camera positioning system sent the measured position and orientation from an external computer, to which the Qualisys system was connected, through the User Datagram Protocol (UDP) to LabView 2016. LabView was then connected to the AIAUV



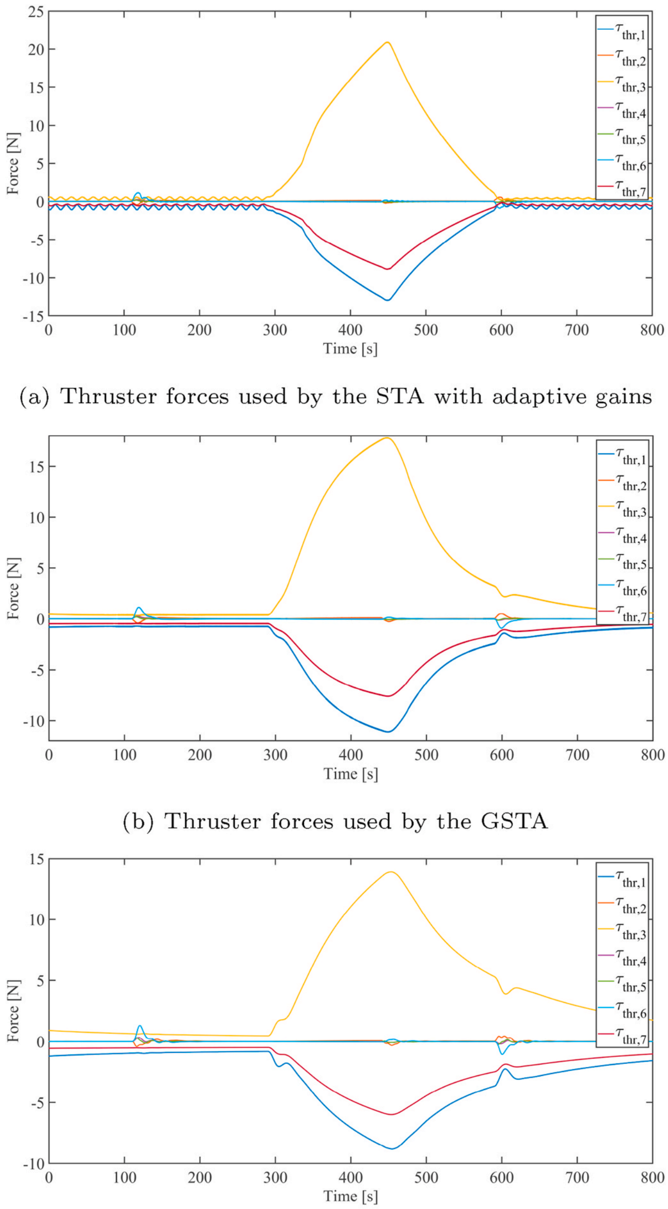
(a) Tracking errors for position in the C-shape

(b) Tracking errors for orientation in the C-shape

(c) Tracking errors for position in the C-shape with a moving head

(d) Tracking errors for orientation in the C-shape with a moving head

Fig. 11. Simulation: Trajectory tracking results using the control laws with the HOSMO.



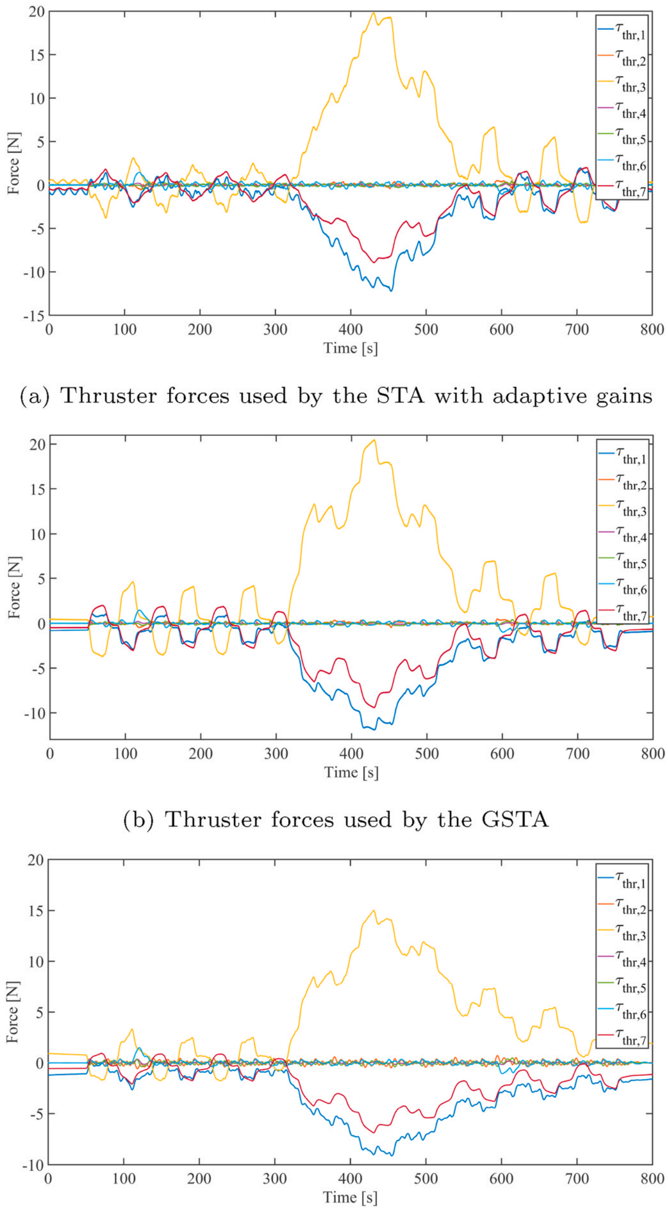
(a) Thruster forces used by the STA with adaptive gains

(b) Thruster forces used by the GSTA

(c) Thruster forces used by the PID controller

Fig. 12. Simulation: Thruster forces used in the C-shape when the control laws with the HOSMO are used.

through an optical fibre cable and through UDP to another computer running MATLAB Simulink with the dynamic controller; thruster allocation, which was implemented as in (Sverdrup-Thygeson et al., 2018); and the reference generator. The computer running MATLAB Simulink received the measured position and orientation, and if the control laws without the HOSMO were used, also an estimate of the linear and angular velocities, which were estimated by using a simple Kalman filter. When the control laws with the HOSMO were used, the linear and angular velocities were estimated by the HOSMO in the dynamic controller. The computer running MATLAB Simulink then sent the thrust commands and the desired joint angles back to LabView through UDP. These were then passed through to the AIAUV through the optical fibre cable. The trajectories for the position and orientation of the base of the AIAUV and the joints were generated by setting setpoints and then using a filter to create a continuous trajectory between them.



(a) Thruster forces used by the STA with adaptive gains

(b) Thruster forces used by the GSTA

(c) Thruster forces used by the PID controller

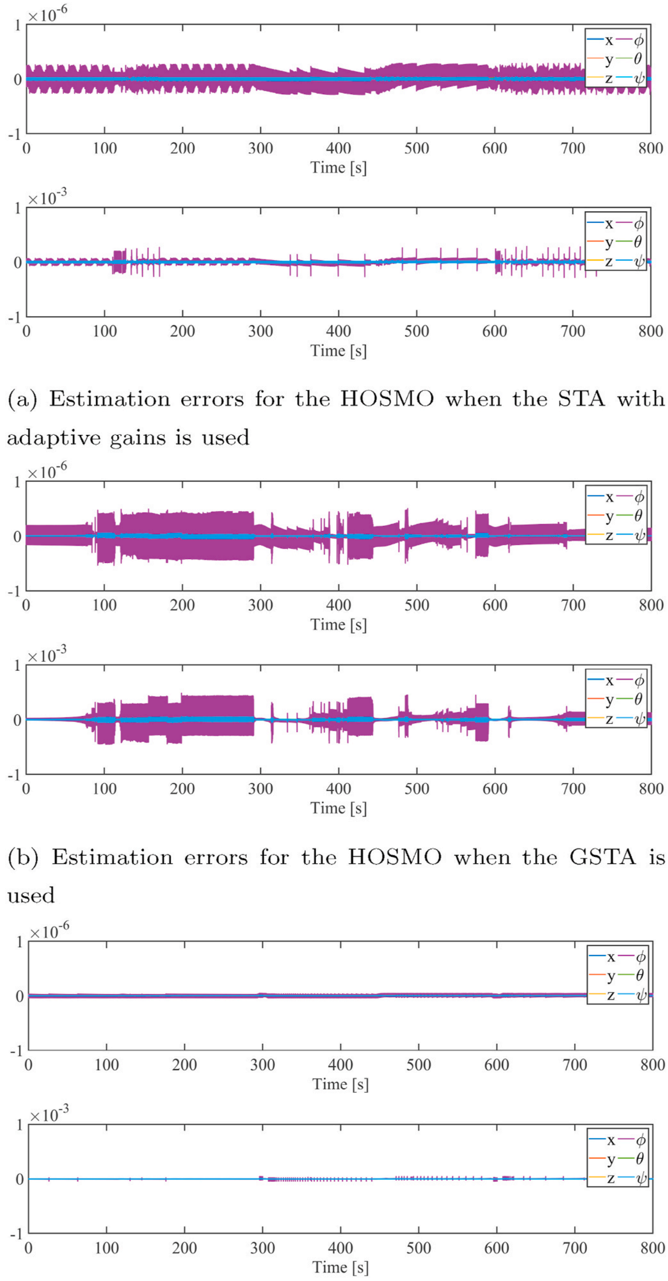
Fig. 13. Simulation: Thruster forces used in the C-shape with a moving head when the control laws with the HOSMO are used.

Table 7
Simulation: Comparison of the results when the control laws with the HOSMO are used.

Algorithm		Root Mean Square Error (RMSE)		Maximum error	
		C-shape	C-shape w/ moving head	C-shape	C-shape w/ moving head
STA	Position [m]	0.0009	0.0032	0.0091	0.0271
STA	Orientation [rad]	0.0078	0.0198	0.0206	0.0676
GSTA	Position [m]	0.0059	0.0075	0.0304	0.0341
GSTA	Orientation [rad]	0.0072	0.0125	0.0205	0.0348
PID	Position [m]	0.0083	0.0112	0.0497	0.0524
PID	Orientation [rad]	0.0147	0.0245	0.0434	0.0692

Table 8
Simulation: Comparison of the thruster forces used when the control laws with the HOSMO are used.

Algorithm	Root Mean Square (RMS)			
	τ_{thr}		$\dot{\tau}_{thr}$	
	C-shape	C-shape w/moving head	C-shape	C-shape w/moving head
STA	2.25	2.05	15.78	9.48
GSTA	2.11	2.22	10.99	7.90
PID	1.82	1.89	1.43	1.37

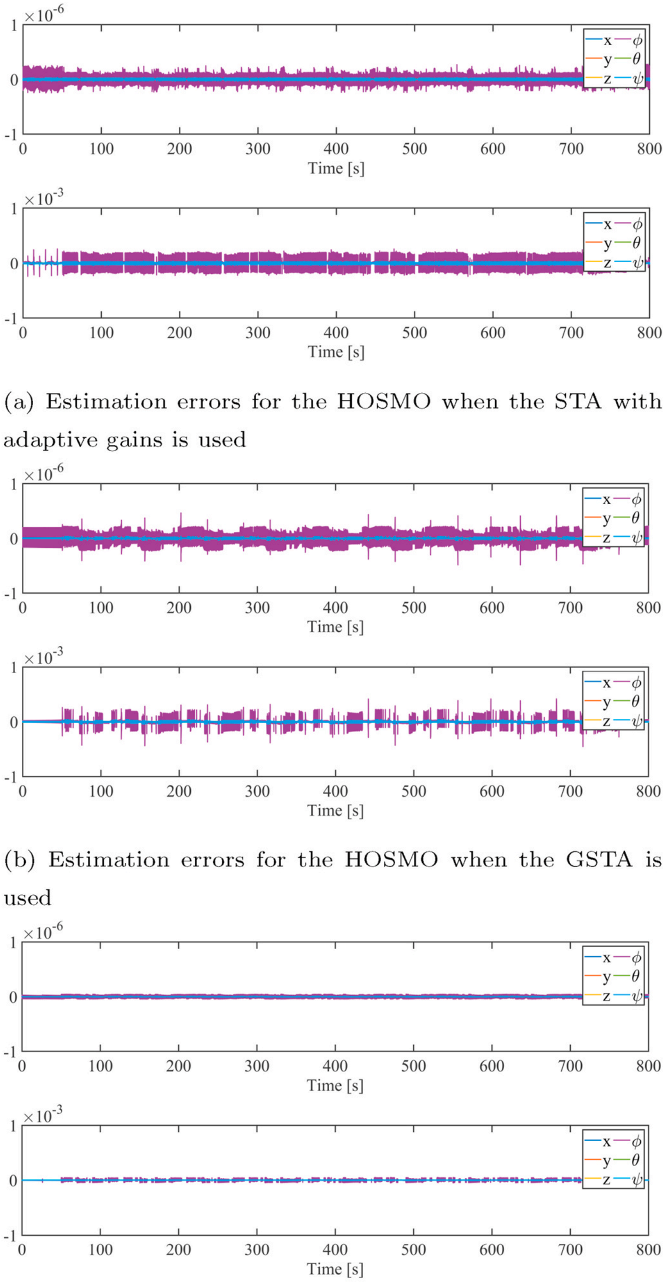


(a) Estimation errors for the HOSMO when the STA with adaptive gains is used

(b) Estimation errors for the HOSMO when the GSTA is used

(c) Estimation errors for the HOSMO when the PID controller is used

Fig. 14. Simulation: Estimation errors for the HOSMO during C-shape (the first sub-plot shows $e_1 = x_{1,e} - \hat{x}_1$, and the second sub-plot shows $e_2 = x_{2,e} - \hat{x}_2$).



(a) Estimation errors for the HOSMO when the STA with adaptive gains is used

(b) Estimation errors for the HOSMO when the GSTA is used

(c) Estimation errors for the HOSMO when the PID controller is used

Fig. 15. Simulation: Estimation errors for the HOSMO during C-shape with a moving head (the first sub-plot shows $e_1 = x_{1,e} - \hat{x}_1$, and the second sub-plot shows $e_2 = x_{2,e} - \hat{x}_2$).

5.3. Test cases

The test cases considered during the experiments are the test cases that were explained in Sec. 4.2 and reproduced in the simulations, i.e., the C-shape and the C-shape with a moving head.

5.4. Experimental results

In this section, the results from the experiments are presented. All three algorithms, i.e., the STA with adaptive gains, the GSTA and the PID controller, were easy to apply successfully in the experiments, and none of them required much effort in tuning. In the experiments, we increased

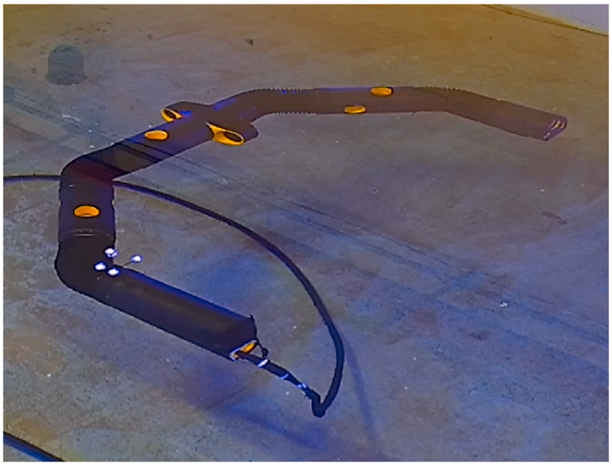


Fig. 16. The Eelume vehicle with reflective markers attached at the base.

the gains until we observed that the AIAUV started to oscillate around the reference instead of following it. The gains that we chose therefore create a small deviation from the reference rather than oscillations around the reference. If we would have chosen more aggressive (higher) gains, the AIAUV would have oscillated. These oscillations might be caused by delays in the thrusters. For all the algorithms, we attempted to use the same gains for both test cases, i.e., we attempted to find the best possible gains for the C-shape, and then we used the same for the C-shape with a moving head for a fair comparison. For the STA with adaptive gains, the choice of gains is not very important because the gains will adapt to their best possible values by themselves. We did, however, observe that if we chose the initial adaptive gains to be too high, the Eelume AIAUV started to oscillate. We therefore had to use lower gains when the HOSMO was used in the case of the C-shape with a moving head likely because when the HOSMO was used, the thruster forces increased. The gains for the STA with adaptive gains and the HOSMO are presented in Table 2; for the GSTA and the HOSMO, the gains are presented in Table 3; and for the PID controller, the gains are listed in Table 4. Note that for the SMC algorithms, the HOSMO gain L was chosen as $L = 0.01$ in the experiments.

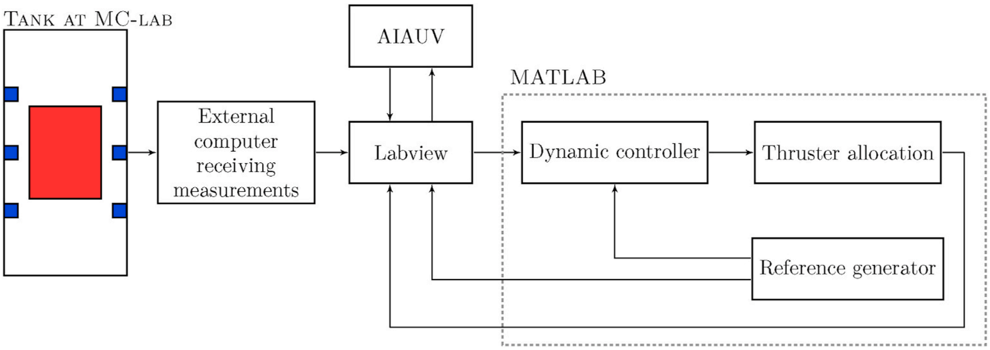
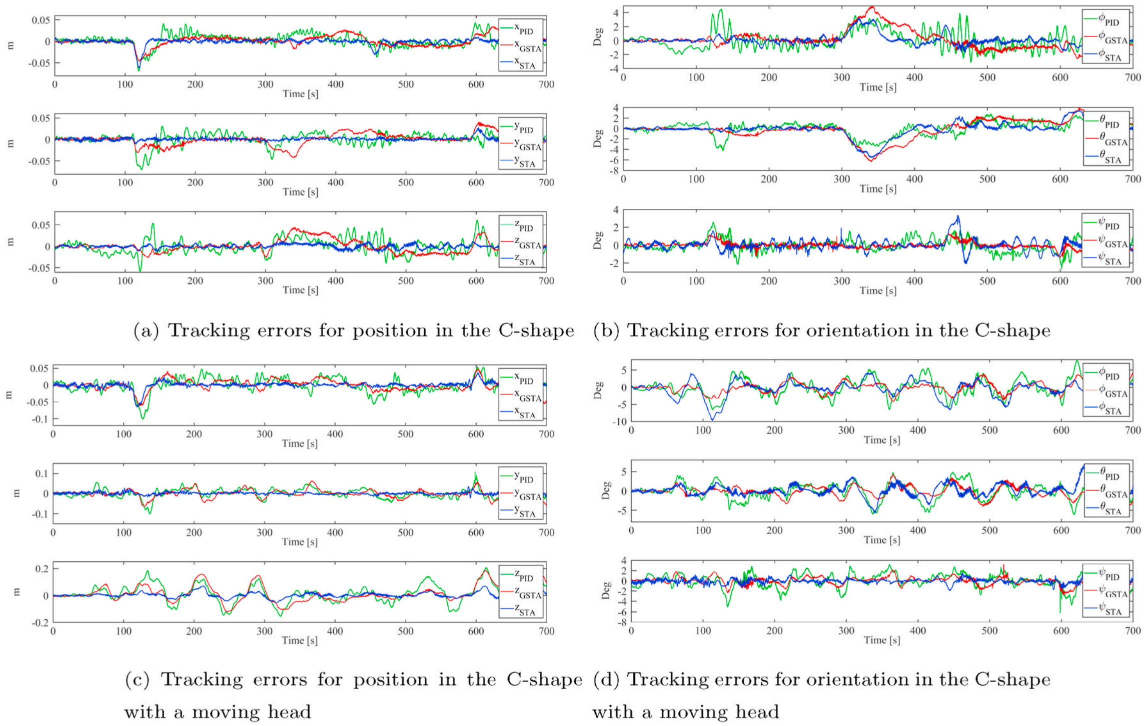


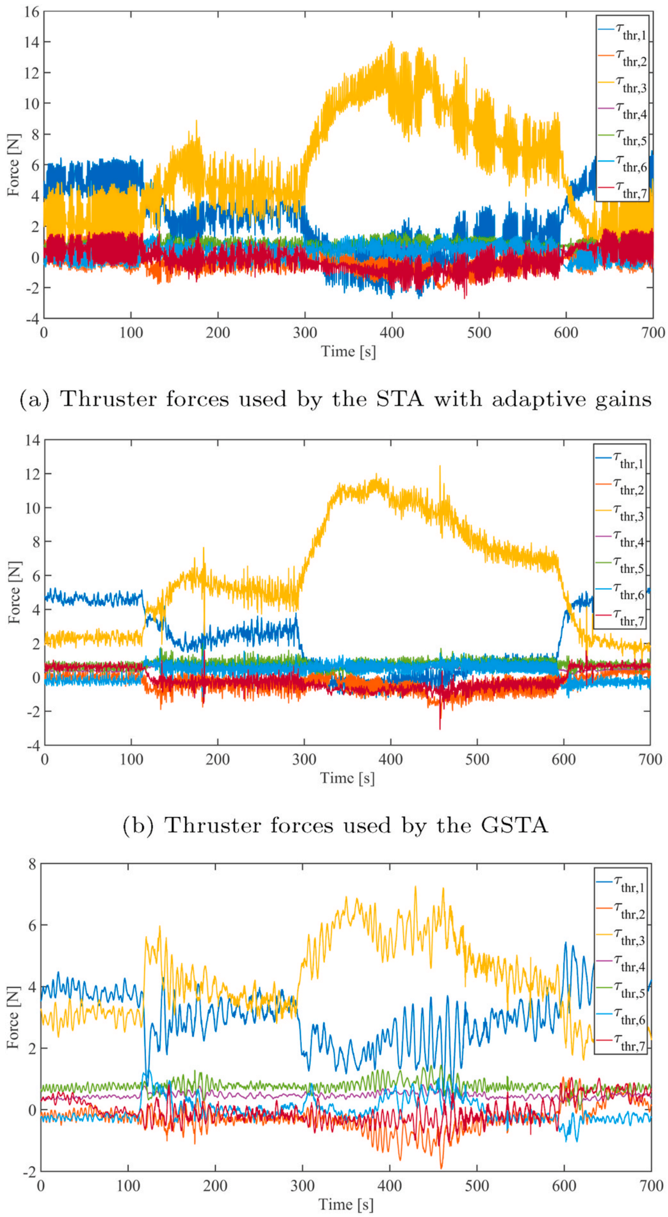
Fig. 17. Illustration of the system structure (the blue boxes represent cameras, and the red box represents the working area). (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)



(a) Tracking errors for position in the C-shape (b) Tracking errors for orientation in the C-shape

(c) Tracking errors for position in the C-shape with a moving head (d) Tracking errors for orientation in the C-shape with a moving head

Fig. 18. Experiments: Trajectory tracking results using the control laws without the HOSMO.

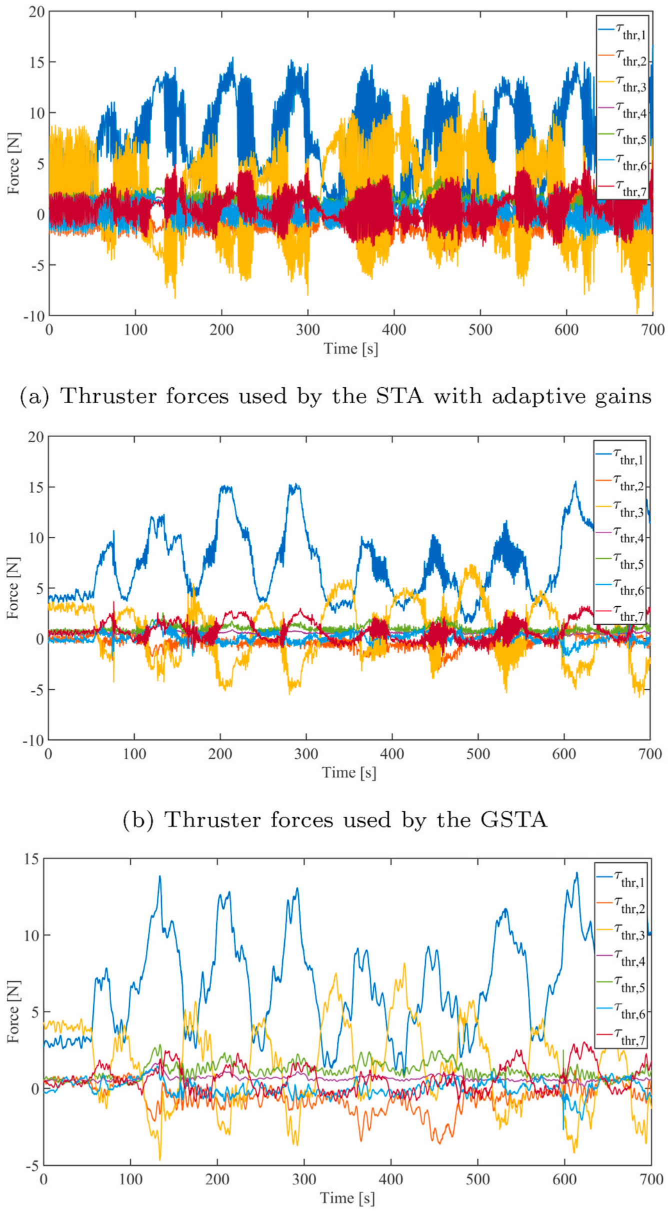


(a) Thruster forces used by the STA with adaptive gains

(b) Thruster forces used by the GSTA

(c) Thruster forces used by the PID controller

Fig. 19. Experiments: Thruster forces used in the C-shape when the control laws without the HOSMO are used.



(a) Thruster forces used by the STA with adaptive gains

(b) Thruster forces used by the GSTA

(c) Thruster forces used by the PID controller

Fig. 20. Experiments: Thruster forces used in the C-shape with a moving head when the control laws without the HOSMO are used.

The trajectory tracking results using the control laws from Sec. 3.1.1, Sec. 3.2.1 and Sec. 3.3.1, i.e., without the HOSMO, are presented in Fig. 18. The trajectory tracking results include both the position errors and the orientation errors for both test cases. The corresponding thruster forces are shown for the C-shape in Fig. 19 and for the C-shape with a moving head in Fig. 20. Note that the thruster forces shown are the output from the control scheme, i.e. the desired thruster forces, since we are not able to measure the actual force of the thrusters or the thruster speed. In Table 9, the RMSE and maximum error for the position and orientation are given for each control law without the HOSMO and for both test cases. In Table 10, the RMS for the thruster forces and the derivative of the thruster forces are given for each control law without the HOSMO and for both test cases. The trajectory tracking results using the control laws from Sec. 3.1.2, Sec. 3.2.2 and Sec. 3.3.2, i.e., with the HOSMO, are presented in Fig. 21. The corresponding thruster forces are shown for the C-shape in Fig. 22 and for the C-shape with a moving head

Table 9
Experiments: Comparison of results when the control laws without the HOSMO are used.

Algorithm		Root Mean Square Error (RMSE)		Maximum error	
		C-shape	C-shape w/ moving head	C-shape	C-shape w/ moving head
STA	Position [m]	0.0054	0.0121	0.0345	0.0579
STA	Orientation [rad]	0.0163	0.0255	0.0707	0.1074
GSTA	Position [m]	0.0133	0.0327	0.0446	0.1039
GSTA	Orientation [rad]	0.0210	0.0216	0.0740	0.0674
PID	Position [m]	0.0158	0.0392	0.0663	0.1382
PID	Orientation [rad]	0.0194	0.0362	0.0669	0.1167

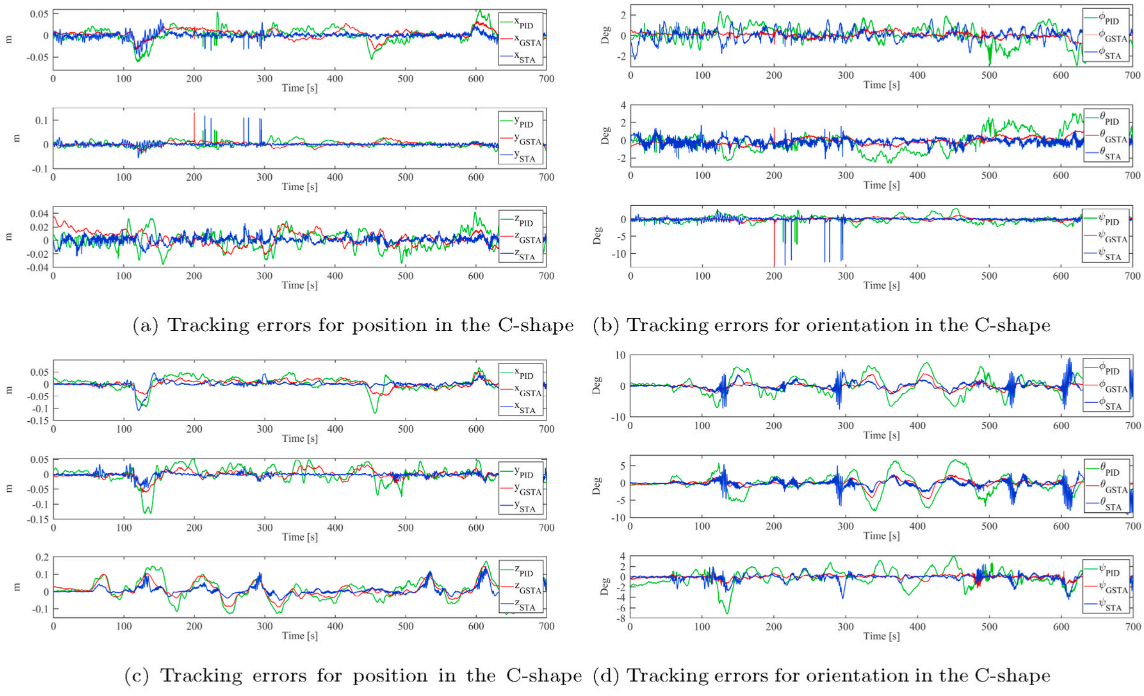
Table 10
Experiments: Comparison of the thruster forces used when the control laws without the HOSMO are used.

Algorithm	Root Mean Square (RMS)			
	τ_{thr}		$\dot{\tau}_{thr}$	
	C-shape	C-shape w/moving head	C-shape	C-shape w/moving head
STA	1.94	2.76	2.95	7.24
GSTA	1.82	2.18	1.46	1.95
PID	1.44	2.14	0.75	0.93

in Fig. 23. In Table 11, the RMSE and maximum error for the position and orientation are given for each control law with the HOSMO and for both test cases. In Table 12, the RMS for the thruster forces and the derivative of the thruster forces are given for each control law with the HOSMO and for both test cases. Note that the simulation study indicates that if higher gains could have been used without causing oscillations, then the tracking accuracy would have been improved.

5.5. Discussion

As shown in Fig. 18, the AIAUV follows the given position and orientation very well for both test cases, as all errors are usually below $1-4$ cm in position and $2-5$deg in orientation, when the control laws from Sec. 3.1.1, Sec. 3.2.1 and Sec. 3.3.1, i.e., without the HOSMO, are used. As shown in Fig. 18a, in the C-shape case, the STA with adaptive gains provides the best results for the position. The GSTA and PID controller have much more similar results for the position as the GSTA has almost as large position errors as the PID controller; however, we do observe that the position errors for the PID controller oscillate more than the position errors for the GSTA. As shown in Fig. 18b, for the orientation, the results are more similar for all algorithms. From Figs. 18c and 18d, the same tendencies are observed for the C-shape with a moving head as for the C-shape case, i.e., for the position, the STA with adaptive gains achieves the smallest error, whereas for the orientation, the tracking performance is more similar. Examining Table 9 confirms what we find from Fig. 18, i.e., that the STA with adaptive gains achieves the smallest position errors, with an RMSE value of 0.0054 m for the C-shape and 0.0121 m for the C-shape with a moving head, and that the GSTA is the second best, with an RMSE value of 0.0133 m for the C-shape and 0.0327 m for the C-shape with a moving head, while the PID controller gives the largest position errors, with an RMSE value of 0.0158 m for the C-shape and 0.0392 m for the C-shape with a moving head. For the orientation errors however, the GSTA gives an RMSE value of 0.0210 rad for the C-shape and 0.0216 rad for the C-shape with a moving head, while the STA with adaptive gains gives an RMSE value of 0.0163 rad for the C-shape and 0.0255 rad for the C-shape with moving head, so the STA with adaptive gains gives the best results for the C-shape, and the GSTA gives the best results for the C-shape with a moving head. The PID controller actually also gives better results than the GSTA for the C-shape case, with an RMSE value of 0.0194 rad, while for the C-shape with a moving head the PID controller gives the worst tracking performance in orientation with an RMSE value of 0.0362 rad. We see the same results for the maximum error value. The reason why the STA with adaptive gains outperforms the two other algorithms is likely because it has adaptive gains and is therefore robust against tuning. If we take the thruster use of the different algorithms into consideration, by looking at Figs. 19 and 20, we find that for both test cases, the STA with adaptive gains uses more force than the GSTA and the PID controller and that the GSTA uses more force than the PID controller. However, based on the RMS value of τ_{thr}, Table 10 shows that over time, the amount of force used is not actually very different. In the C-shape case, the STA with adaptive gains uses slightly more force, with an RMS value of 1.94, than the GSTA, that has an RMS value 1.82, and the GSTA uses more force than the PID controller. that has an RMS value of 1.44. In the case of the C-shape with a moving head, however, the GSTA and the PID controller uses very similar amounts of force, with RMS values of 2.18 and 2.14, respectively, while the STA with adaptive gains uses more, with an RMS value of 2.76. From the RMS values of $\dot{\tau}_{thr}$, we find that the most chattering occurs in the control input from the STA with adaptive gains, with RMS value of 2.95 for the C-shape and 7.24 for the C-shape with a moving head, and that less chattering occurs in that from the GSTA, with RMS value of 1.46 for the C-shape and 1.95 for the C-shape with a



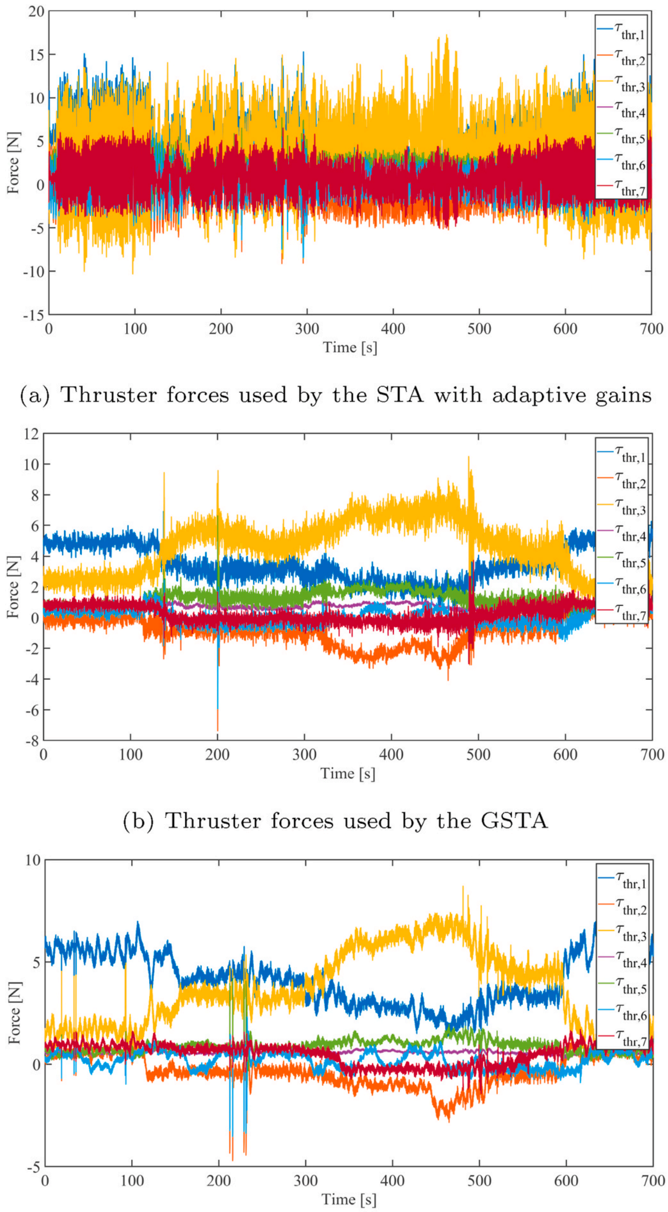
(a) Tracking errors for position in the C-shape

(b) Tracking errors for orientation in the C-shape

(c) Tracking errors for position in the C-shape with a moving head

(d) Tracking errors for orientation in the C-shape with a moving head

Fig. 21. Experiments: Trajectory tracking results using the control laws with the HOSMO.



(a) Thruster forces used by the STA with adaptive gains

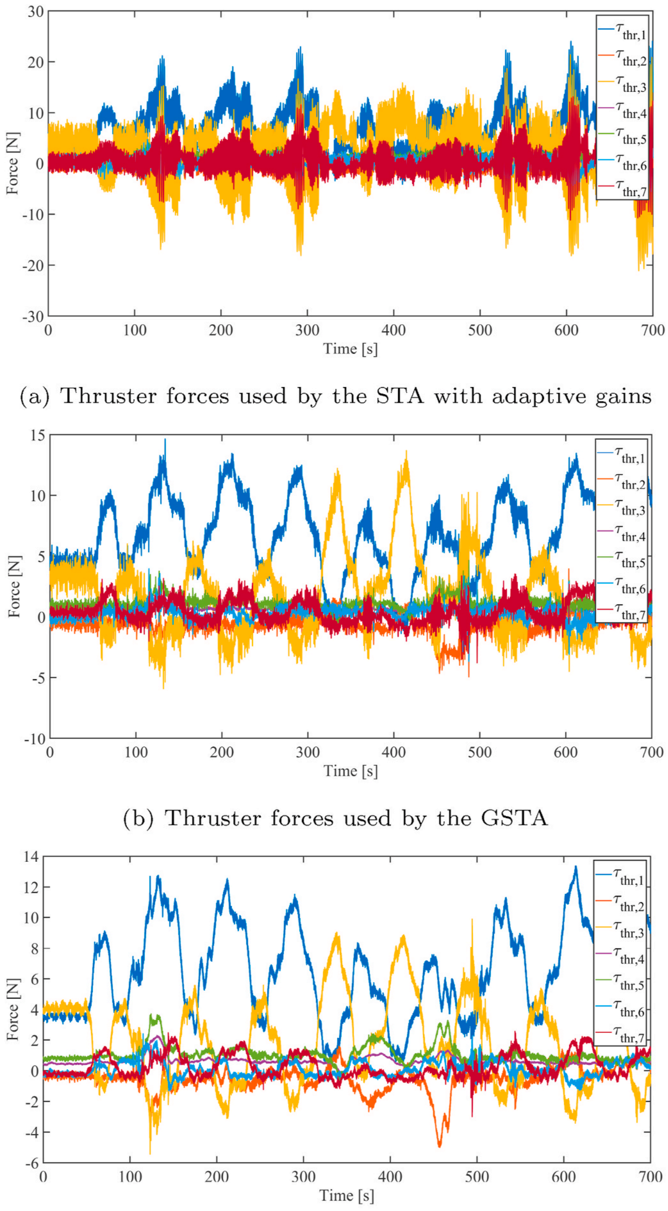
(b) Thruster forces used by the GSTA

(c) Thruster forces used by the PID controller

Fig. 22. Experiments: Thruster forces used in the C-shape when the control laws with the HOSMO are used.

moving head, while the control input from the PID controller shows the least rapid changes in the thrust forces, with RMS value of 0.75 for the C-shape and 0.93 for the C-shape with a moving head. However, the value of the RMS of $\dot{\tau}_{thr}$ is so small that the chattering that is introduced by the STA with adaptive gains is almost negligible. The reason why the chattering may appear to be considerable in the plots is because of the time scale used.

When the control laws with the HOSMO are used, i.e., from Sec. 3.1.2, Sec. 3.2.2 and Sec. 3.3.2, we observe the same tendencies in Fig. 21 as when the control laws without the HOSMO are used, but we find from Figs. 21a and 21b and from Figs. 21c and 21d that the STA with adaptive gains is no longer clearly better. We also observe that we have some more measurement noise and outliers in the measurements from the measurement system, see for instance the jumps in Figs. 21a and Fig. 21b between 200 s and 300 s. The performance of the algorithm indicates that the algorithms are robust against this noise and the outliers. As shown in Table 11, we have the same results as those when the



(a) Thruster forces used by the STA with adaptive gains

(b) Thruster forces used by the GSTA

(c) Thruster forces used by the PID controller

Fig. 23. Experiments: Thruster forces used in the C-shape with a moving head when the control laws with the HOSMO are used.

Table 11
Experiments: Comparison of the results when the control laws with the HOSMO are used.

Algorithm		Root Mean Square Error (RMSE)		Maximum error	
		C-shape	C-shape w/ moving head	C-shape	C-shape w/ moving head
STA	Position [m]	0.0064	0.0177	0.0640	0.1007
STA	Orientation [rad]	0.0087	0.0215	0.1035	0.1321
GSTA	Position [m]	0.0100	0.0252	0.0661	0.0871
GSTA	Orientation [rad]	0.0062	0.0147	0.0914	0.0618
PID	Position [m]	0.0143	0.0394	0.0549	0.1428
PID	Orientation [rad]	0.0181	0.0431	0.0779	0.1327

Table 12
Experiments: Comparison of the thruster forces used when the control laws with the HOSMO are used.

Algorithm	Root Mean Square (RMS)			
	τ_{thr}		$\dot{\tau}_{thr}$	
	C-shape	C-shape w/moving head	C-shape	C-shape w/moving head
STA	3.02	3.34	17.49	13.51
GSTA	1.85	2.23	4.77	5.05
PID	1.71	2.09	4.27	4.64

HOSMO is not used, i.e., the STA with adaptive gains provides the smallest position errors, with an RMSE value of 0.0064 m for the C-shape and 0.0177 m for the C-shape with a moving head, and that the GSTA is the second best, with an RMSE value of 0.0100 m for the C-shape and 0.0252 m for the C-shape with a moving head, while the PID controller gives the largest position errors, with an RMSE value of 0.0143 m for the C-shape and 0.0394 m for the C-shape with a moving head. For the orientation errors however, the GSTA gives an RMSE value of 0.0062 rad for the C-shape and 0.0147 rad for the C-shape with a moving head, while the STA with adaptive gains gives an RMSE value of 0.0087 rad for the C-shape and 0.0215 rad for the C-shape with moving head, so the results are much more similar, and the GSTA provides slightly better results. The PID controller clearly offers the worst tracking performance for orientation, with an RMSE value of 0.0181 rad for the C-shape and 0.0431 rad for the C-shape with a moving head. From the maximum errors in Table 11, we can see that the measurement noise impacts the maximum error results since if we would have filtered the position measurements such that the peaks/outliers in the measurements were removed, we would have obtained a lower maximum error for all algorithms, particularly the STA with adaptive gains and the GSTA, as some large peak/outliers exist in those measurements. If we take the thruster use of the different algorithms into consideration, presented in Figs. 22 and 23, we find that the STA with adaptive gains uses more force for both test cases and that the GSTA uses more force than the PID controller in the C-shape case, whereas in the case of the C-shape with a moving head, the PID controller and the GSTA use almost the same amount of force. However, as shown in Table 12, the RMS of τ_{thr} indicates that the STA does use more force, with an RMS value of 3.02 for the C-shape and 3.34 for the C-shape with a moving head, than the GSTA, with an RMS value of 1.85 for the C-shape and 2.23 for the C-shape with a moving head, and the PID controller, with an RMS value of 1.71 for the C-shape and 2.09 for the C-shape with a moving head. The GSTA and the PID controller does, however, use a very similar amount of force. From the RMS values of $\dot{\tau}_{thr}$, we find that the most chattering occurs in the control input from the STA with adaptive gains, with RMS value of 17.49 for the C-shape and 13.51 for the C-shape with a moving head, and that less chattering occurs in that from the GSTA, with RMS value of 4.77 for the C-shape and 5.05 for the C-shape with a moving head, while the control input from the PID controller shows the least rapid changes in the thrust forces, with RMS value of 4.27 for the C-shape and 4.64 for the C-shape with a moving head. However, the value of the RMS of $\dot{\tau}_{thr}$ is so small that the chattering that is introduced by the STA with adaptive gains is negligible. The reason why considerable chattering may be evident in the plots is again because of the time scale used.

If we compare the results obtained with the control laws from Sec. 3.1.1, Sec. 3.2.1 and Sec. 3.3.1, i.e., without the HOSMO, and the results obtained with the control laws from Sec. 3.1.2, Sec. 3.2.2 and Sec. 3.3.2, i.e., when the HOSMO is used, by comparing Table 9 with Table 11, Table 10 and Table 12, we find that when the HOSMO is used, we obtain better tracking results for the orientation for all the algorithms, as well as better tracking for position when the GSTA is used. Additionally, note that the difference between the position errors, i.e., with and without the HOSMO, is smaller than the difference between the orientation errors, i.e., the control laws with the HOSMO provide the overall best performance. The reason for the better tracking performance is likely due to the increase in thruster use. The thruster use is, however, well within the boundaries of what the Eelume AIAUV can provide, which is 50 N.

If we compare the simulation results (Figs. 8 and 11) and the experimental results (Figs. 18 and 21) for both test cases, we find that the results are quite similar and that all algorithms are applicable with and without the HOSMO to control the AIAUV. We also find that the same algorithms provide the best tracking results in the simulations and the experiments, i.e., the STA with adaptive gains and the GSTA in a good second place. We also observed an improvement in the tracking performance for all algorithms when the HOSMO was used in both simulations and experiments. Of course, the tracking errors, i.e., the position and orientation errors, are larger in the experiments than in the simulations, but that is to be expected because of measurement noise and outliers from the measurements system, thruster dynamics and other unmodelled dynamics. The fact that we did not have feedback from the thrusters likely also affected the errors because we did not necessarily obtain the desired forces from the thrusters.

6. Conclusions and future research

In this paper, we have presented and compared three different algorithms for solving the trajectory tracking problem for AIAUVs in 6DOF: the STA with adaptive gains, the GSTA and a PID controller. Comprehensive simulation and experimental results for two different test cases, have been presented and used to compare the performance of the three mentioned algorithms. In both the simulations and in the experiments, the STA with adaptive gains provided the best overall tracking performance, but the GSTA was not far from achieving the same tracking results. Thus, the fact that we had to tune the GSTA manually is likely the reason that we achieved lower tracking performance with this algorithm. The STA with adaptive gains is therefore better in practice than the GSTA, but the GSTA is proved to provide global finite-time stability for a larger class of systems.

Furthermore, we have also solved the 6DOF trajectory tracking problem using the three mentioned algorithms in combination with an HOSMO. Additionally, we have proven that the closed-loop system when the STA with adaptive gains is used in combination with a HOSMO is UGAS. The same two test cases were used to obtain comprehensive simulation and experimental results for the three algorithms in combination with the HOSMO. The results have been used to compare the control laws and to study the performance of the HOSMO. When the HOSMO was used to estimate the linear and angular velocities in the control laws, we improved our results, indicating that the HOSMO is indeed applicable for state estimation.

Future work includes proving when Assumption 24 holds. Moreover, identifying and including an HOSMO that works with quaternions is future work, as well as to determine whether adaptive gains can be used with the GSTA. We could then realize both the theoretical advantages afforded by the GSTA and the practical advantages afforded by having adaptive gains.

CRediT authorship contribution statement

Ida-Louise G. Borlaug: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing - original draft, Visualization. **Kristin Y. Pettersen:** Funding acquisition, Supervision, Writing - review & editing. **Jan Tommy Gravdahl:** Supervision, Writing - review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was partly supported by the Research Council of Norway through the Centres of Excellence funding scheme, project No. 223254 – NTNU AMOS and project No. 304667.

References

Antonelli, G., 2014. Underwater Robots; Vol. 96 of *Springer Tracts In Advanced Robotics*, third ed. Springer International Publishing, Cham, Switzerland.

Antonelli, G., Chiaverini, S., 1998. Singularity-free regulation of underwater vehicle-manipulator systems. In: Proc. American Control Conference. Philadelphia, Pennsylvania, pp. 399–403.

Borlaug, I.L.G., Pettersen, K.Y., Gravdahl, J.T., 2018. Trajectory tracking for an articulated intervention AUV using a super-twisting algorithm in 6DOF. IFAC PapersOnLine 51 (29), 311–316. Proc. 11th IFAC Conference on Control Applications in Marine Systems, Robotics, and Vehicles, Opatija, Croatia.

Borlaug, I.L.G., Gravdahl, J.T., Sverdrup-Thygeson, J., Pettersen, K.Y., Loría, A., 2019a. Trajectory tracking for underwater swimming manipulator using a super twisting algorithm. Asian J. Contr. 21 (1), 208–223.

Borlaug, I.L.G., Pettersen, K.Y., Gravdahl, J.T., 2019b. Tracking control of an articulated intervention AUV in 6DOF using the generalized super-twisting algorithm. In: Proc. American Control Conference. Philadelphia, USA, pp. 5705–5712.

Borlaug, I.L.G., Pettersen, K.Y., Gravdahl, J.T., 2020. Tracking control of an articulated intervention autonomous underwater vehicle in 6DOF using generalized super-twisting: theory and experiments. In: IEEE Transactions on Control Systems Technology.

Castillo, I., Fridman, L.M., Moreno, J.A., 2018. Super-Twisting Algorithm in presence of time and state dependent perturbations. Int. J. Contr. 91 (11), 2535–2548.

Chalanga, A., Kamal, S., Fridman, L.M., Bandyopadhyay, B., Moreno, J.A., 2016. Implementation of super-twisting control: super-twisting and higher order sliding-mode observer-based approaches. IEEE Trans. Ind. Electron. 63 (6), 3677–3685.

Cui, R., Zhang, X., Cui, D., 2016. Adaptive sliding-mode attitude control for autonomous underwater vehicles with input nonlinearities. Ocean. Eng. 123, 4554.

Dannigan, M.W., Russell, G.T., 1998. Evaluation and reduction of the dynamic coupling between a manipulator and an underwater vehicle. IEEE J. Ocean. Eng. 23 (3), 260–273.

Fjellstad, O.E., Fossen, T.I., 1994. Singularity-free tracking of unmanned underwater vehicles in 6 DOF. In: Proc. 33rd IEEE Conference on Decision and Control. Lake Buena Vista, Florida, pp. 1128–1133.

Fossen, T.I., 1991. Adaptive macro-micro control of nonlinear underwater robotic systems. In: Proc. 5th International Conference on Advanced Robotics. Pisa, Italy, pp. 1569–1572.

Fossen, T.I., 2011. Handbook of Marine Craft Hydrodynamics and Motion Control. John Wiley & Sons, Ltd, Chichester, UK.

Fossen, T.I., Sagatun, S., 1991. Adaptive control of nonlinear underwater robotic systems. Model. Ident. Contr. 12 (2), 95–105.

From, P.J., Gravdahl, J.T., Pettersen, K.Y., 2014. Vehicle-Manipulator Systems: Modeling for Simulation, Analysis, and Control. Advances in Industrial Control. Springer London, London.

Kelasidi, E., Pettersen, K.Y., Gravdahl, J.T., 2014. A waypoint guidance strategy for underwater snake robots. In: Proc. IEEE 22nd Mediterranean Conference on Control and Automation. Palermo, Italy, pp. 1512–1519.

Kelasidi, E., Pettersen, K.Y., Liljebäck, P., Gravdahl, J.T., 2016. Locomotion efficiency of underwater snake robots with thrusters. In: Proc. International Symposium on Safety, Security and Rescue Robotics. Lausanne, Switzerland, pp. 174–181.

Kelasidi, E., Liljeback, P., Pettersen, K.Y., Gravdahl, J.T., 2017a. Integral line-of-sight guidance for path following control of underwater snake robots: theory and experiments. IEEE Trans. Robot. 33 (3), 610–628.

Kelasidi, E., Pettersen, K.Y., Gravdahl, J.T., Strømsøyen, S., Sørensen, A.J., 2017b. Modeling and propulsion methods of underwater snake robots. In: Proc. 1st IEEE Conference on Control Technology and Applications. Kohala Coast, Hawaii, pp. 819–826.

Khalil, H.K., 2002. Nonlinear Systems, third ed. ed. Prentice Hall, Upper Saddle River, N. J.

Kohl, A.M., Kelasidi, E., Mohammadi, A., Maggiore, M., Pettersen, K.Y., 2016a. Planar maneuvering control of underwater snake robots using virtual holonomic constraints. Bioinspiration Biomimetics 11 (6), 065005.

Kohl, A.M., Pettersen, K.Y., Kelasidi, E., Gravdahl, J.T., 2016b. Planar path following of underwater snake robots in the presence of ocean currents. IEEE Robotics and Automation Letters 1 (1), 383–390.

Kohl, A.M., Pettersen, K.Y., Gravdahl, J.T., 2017a. Velocity and orientation control of underwater snake robots using absolute velocity feedback. In: Proc. 1st IEEE Conference on Control Technology and Applications. Kohala Coast, Hawaii, pp. 752–759.

Kohl, A.M., Moe, S., Kelasidi, E., Pettersen, K.Y., Gravdahl, J.T., 2017b. Set-based path following and obstacle avoidance for underwater snake robots. In: Proc. 2017 IEEE International Conference on Robotics and Biomimetics. Parisian Macao, China, pp. 1206–1213.

Kumari, K., Chalanga, A., Bandyopadhyay, B., 2016. 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. Monterey, CA, USA, pp. 873–878.

Levant, A., 1998. Robust exact differentiation via sliding mode technique. Automatica 34 (3), 379–384.

Levant, A., 2003. Higher-order sliding modes, differentiation and output-feedback control. Int. J. Contr. 76 (9–10), 924–941.

Liljebäck, P., Mills, R., 2017. Eelume: a flexible and subsea resident IMR vehicle. In: OCEANS 2017. Aberdeen, UK.

Liu, S., Liu, Y., Wang, N., 2017. Nonlinear disturbance observer-based backstepping finite-time sliding mode tracking control of underwater vehicles with system uncertainties and external disturbances. Nonlinear Dynam. 88 (1), 465–476.

Loría, A., Panteley, E., 2005. Cascaded Nonlinear Time-Varying Systems: Analysis and Design; Vol. 311 of *Lecture Notes In Control And Information Sciences*. Springer International Publishing, Cham, Switzerland.

Marine cybernetics laboratory (mc-lab). https://www.ntnu.edu/imt/lab/cybernetics, 2018.

Moreno, J.A., 2012. Lyapunov function for levant's second order differentiator. In: Proc. IEEE 51st IEEE Conference on Decision and Control. Maui, Hawaii, USA, pp. 6448–6453.

Polyakov, A., Fridman, L.M., 2014. Stability notions and lyapunov functions for sliding mode control systems. J. Franklin Inst. 351 (4), 1831–1865.

Qualisys-motion capture systems. https://www.qualisys.com, 2018.

Schmidt-Didlaukies, H.M., Sørensen, A.J., Pettersen, K.Y., 2018. Modeling of articulated underwater robots for simulation and control. In: Proc. 2018 IEEE/OES Autonomous Underwater Vehicles. Porto, Portugal, pp. 1–7.

Shtessel, Y.B., Moreno, J.A., Plestan, F., Fridman, L.M., Poznyak, A.S., 2010. Super-twisting adaptive sliding mode control: a Lyapunov design. In: Proc. 49th IEEE Conference on Decision and Control. Atlanta, GA, USA, pp. 5109–5113.

Soylu, S., Buckham, B.J., Podhorodeski, R.P., 2008. A chattering-free sliding-mode controller for underwater vehicles with fault-tolerant infinity-norm thrust allocation. Ocean. Eng. 35 (16), 1647–1659.

Sverdrup-Thygeson, J., Kelasidi, E., Pettersen, K.Y., Gravdahl, J.T., 2016. A control framework for biologically inspired underwater swimming manipulators equipped with thrusters. IFAC PapersOnLine 49 (23), 89–96. Proc. 10th IFAC Conference on Control Applications in Marine Systems, Robotics, and Vehicles, Trondheim, Norway.

Sverdrup-Thygeson, J., Kelasidi, E., Pettersen, K.Y., Gravdahl, J.T., 2018. The underwater swimming manipulator - a bioinspired solution for subsea operations. IEEE J. Ocean. Eng. 43 (2), 1–16.

von Ellenrieder, K.D., Henninger, H.C., 2019. A higher order sliding mode controller-observer for marine vehicles. IFAC PapersOnLine 52 (21), 341–346. Proc. 12th IFAC Conference on Control Applications in Marine Systems, Robotics, and Vehicles, Daejeon, Korea.

Wrzos-Kaminska, M., Pettersen, K.Y., Gravdahl, J.T., 2019. Path following control for articulated intervention-AUVs using geometric control of reduced attitude. IFAC PapersOnLine 52 (16), 192–197. Proc. 11th IFAC Symposium on Nonlinear Control Systemss, Vienna, Austria.

Xu, J., Wang, M., Qiao, L., 2015. Dynamical sliding mode control for the trajectory tracking of underactuated unmanned underwater vehicles. Ocean. Eng. 105, 54–63.

Zhu, D., Sun, B., 2013. The bio-inspired model based hybrid sliding-mode tracking control for unmanned underwater vehicles. Eng. Appl. Artif. Intell. 25 (10), 2260–2269.