Received: 7 September 2020 | Revised: 6 April 2022 | Accepted: 8 May 2022

DOI: 10.1002/rnc.6212

RESEARCH ARTICLE

WILEY

The generalized super-twisting algorithm with adaptive gains

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

Correspondence
Ida-Louise G. Borlaug, Centre for Autonomous Marine Operations and Systems, Department of Engineering Cybernetics, Norwegian University of Science and Technology (NTNU), O. S. Bragstads plass 2D, Elektroblokk D, 1. etg., Gløshaugen, 7034 Trondheim, Norway
Email: Ida.Louise.Borlaug@ntnu.no

Funding information
H2020 European Research Council, Grant/Award Number: 101017697-CRÈME; Norges Forskningsråd, Grant/Award Numbers: 223254, 304667

Abstract

In this article, a novel adaptive generalized super-twisting algorithm (GSTA) is proposed for a class of systems whose perturbations and uncertain control coefficients may depend on both time and state. The proposed approach uses dynamically adapted control gains, and it is proven that this ensures global finite-time convergence. A nonsmooth strict Lyapunov function is used to obtain the conditions for global finite-time stability. A simulation and experimental case study is performed using an articulated intervention autonomous underwater vehicle (AIAUV). It is also shown that the adaptive GSTA causes the tracking errors of the AIAUV to converge to zero in finite time. In the case study, we use the singularity-robust multiple task-priority method to create a continuous trajectory for the AIAUV to follow. The simulation and experimental results validate and verify that the proposed approach is well suited for controlling an AIAUV. We also perform a comparison with the super-twisting algorithm with adaptive gains and the original GSTA to evaluate whether adding adaptive gains to the GSTA actually improves the tracking capabilities by combining the theoretical advantages afforded by the GSTA with the practical advantages afforded by adaptive gains. Based on this comparison, the adaptive GSTA yields the best tracking results overall without increasing the energy consumption, and the simulations and experiments thus indicate that adding adaptive gains to the GSTA does indeed improve the consequent tracking results and capabilities.

KEYWORDS

autonomous underwater vehicle, experimental results, robotics, sliding mode control (variable structure systems)

1 | INTRODUCTION

Sliding mode control (SMC) is a robust and versatile nonlinear control approach that is particularly well suited for controlling perturbed systems, especially systems perturbed by matched uncertainties and disturbances.[1] These properties are achieved by means of a discontinuous control law. The discontinuous element provides robustness, but it also introduces chattering, that is, high-frequency switching in the control input. Chattering can be avoided by using a saturation or sigmoid function instead of the discontinuous signum function.[2,3] In this way, a continuous

control input is achieved, but the sliding system's trajectories are restricted to within a certain boundary around the sliding surface, thus sacrificing robustness to disturbances. Alternatively, chattering can also be avoided using higher-order sliding mode (HOSM) techniques.[4-6] Thus, a continuous control input can be achieved without losing any robustness. HOSM methods drive the sliding variable and its derivatives to zero in the presence of disturbances and uncertainties.[7]

The super-twisting algorithm (STA)[8] is one of the most powerful second-order continuous SMC algorithms. It attenuates chattering through the introduction of a dynamic extension to the system such that the discontinuous term is hidden behind an integrator. Thus, it generates a continuous control input that drives the sliding variable and its derivatives to zero in finite time in the presence of smooth matched disturbances with a bounded gradient.[9] The main drawback of this approach is that the bound on the disturbance gradient must be known. This bound is not always easy to estimate and is often overestimated, resulting in unnecessarily large control gains. Therefore, in Reference 7, an STA with adaptive gains was proposed. This approach continuously drives the sliding variable and its derivatives to zero in the presence of a bounded disturbance with an unknown bound, such that no conservative upper bound on the disturbance gradient needs to be considered to maintain sliding because of the adaptive gains. In more recent years other adaptation methods for the gains for the STA have been proposed.[10,11] These methods also allow the adaptive gains to decrease. In Reference 10 an adaptation method based on equivalent control by a first-order low-pass filter is proposed. The method reduces the control action magnitude of the STA to the minimum possible value while keeping the property of finite-time convergence. In Reference 11 an adaptation method based on a barrier function is proposed. This method ensures convergence of the output variable and maintains it in a predefined neighborhood of zero independent from the upper bound of the derivatives of the disturbances. However, it also forces the adaptive gain to decrease together with the output variable and the control input follows the absolute value of the disturbances.

In recent years, various Lyapunov functions have been designed to obtain convergence conditions and estimates of the reaching time. However, the corresponding Lyapunov proofs have been conducted under conservative assumptions: the perturbations are assumed to depend only on time,[8,12] the control coefficient is assumed to be known,[8,12-14] or the perturbations are taken to depend on both state and time, but it is assumed that their total time derivative is a priori bounded by some constant.[13,15] Therefore, Reference 9 proposed a generalized super-twisting algorithm (GSTA) that considers a more general scenario, that is, the case in which both the perturbations and the control coefficients may depend on both state and time and the control coefficients are uncertain. This approach gives rise to some additional theoretical properties relative to the regular STA proposed in Reference 8. However, it is still necessary to know the bounds on the perturbations and control coefficients to obtain bounds on the control gains that are not too conservative.

In this article, we therefore propose an adaptive GSTA for a class of systems with time- and state-dependent perturbations and uncertain control coefficients, that is, we combine the best properties of the STA with adaptive gains[7] and the GSTA.[9] The proposed approach consists of using dynamically adapted control gains to ensure global finite-time (GFT) convergence. The advantage of the GSTA is that stability is proven for a larger class of systems, that is, systems for which both the perturbations and the control coefficients may depend on both state and time and the control coefficients are uncertain. The advantage of adaptive gains is that no conservative upper bounds on the perturbations and uncertain control coefficients need to be considered to maintain sliding. We prove that the resulting closed-loop system is globally finite-time stable (GFTS). Please note that the abbreviations GFT and GFTS should not be confused. Compared to the traditional STA,[8] the main difference is therefore that the stability is proven for a larger class of systems, that is, systems for which both the perturbations and the control coefficients may depend on both state and time and the control coefficients are uncertain, and that no conservative upper bounds on the perturbations and uncertain control coefficients need to be considered to maintain sliding because of the adaptive gains.

In References 16 and 17, two alternative adaptive GSTAs have been proposed based on the GSTA presented in Reference 18, for single-input single-output (SISO) and multiple-input multiple-output (MIMO) systems, respectively. The approach proposed in this article differs from these two algorithms in that it also allows for unknown control coefficients. In Reference 19, a third alternative adaptive GSTA has also been proposed. The approach proposed in this article also differs from the algorithm in Reference 19 in that it additionally handles state-dependent disturbances, not merely time-dependent disturbances. These distinctions are important because they allow us to use the adaptive GSTA proposed here to control a larger class of systems and, in particular, to control an articulated intervention autonomous underwater vehicle (AIAUV), which was our motivation for investigating an adaptive GSTA that can handle unknown control coefficients and state-dependent disturbances.

An AIAUV is an underwater vehicle with multiple joints and multiple thrusters.[20] An AIAUV is subject to hydrodynamic and hydrostatic parameter uncertainties, uncertain thruster characteristics, unknown disturbances, unmodeled dynamics and large coupling forces caused by joint motion, and a robust control approach is therefore essential. We have previously investigated the use of SMC for trajectory tracking for an AIAUV, since trajectory tracking is essential to allow an AIAUV to move in confined spaces and to perform intervention tasks (tasks where the robot interacts or is in contact with the environment). In References 21 and 22, we investigated the use of the STA with adaptive gains to control an AIAUV, and in References 23 and 24, we investigated the use of the original GSTA for this purpose. In Reference 25, we compared these two SMC algorithms through both simulations and experiments, and we observed that the STA with adaptive gains yielded better tracking results than the GSTA, even though the GSTA has better theoretical properties as it is proven to provide GFT stability for a larger class of systems. The incorporation of adaptive gains was thus seen to be very practical and to provide certain tuning advantages. We are therefore interested in combining the practical advantages afforded by adaptive gains with the theoretical advantages of the GSTA to control an AIAUV. This was our motivation for the theoretical development of the GSTA with adaptive gains which is presented in this article. Moreover, we subsequently present a simulation and experimental case study performed using an AIAUV to show the effectiveness of the proposed adaptive GSTA. We also show that the adaptive GSTA causes the tracking errors of the AIAUV to converge to zero in finite time. In Reference 26, we combined the singularity-robust multiple task-priority (SRMTP) framework[27] with robust SMC algorithms that are finite-time stable. The finite-time convergence property allowed us to show that multiple set-point regulation tasks would converge asymptotically to zero without the strict requirement of perfect velocity control. We combined the SRMTP framework with a first-order SMC algorithm and the adaptive GSTA proposed in this article. However, in Reference 26 we only used the adaptive GSTA as an example of a SMC algorithm, and the finite-time convergence property was not proved. Therefore, in this article, we wish to use the SRMTP method to create a continuous trajectory for the AIAUV to follow. Compared to Reference 26 we prove that the adaptive GSTA is GFTS. Additionally, we use different set-point tasks and we also include experimental results. The primary task is to control the position and orientation of the front end of the AIAUV, and the secondary task is to control the position and orientation of the back end. We also present a comparison with the STA with adaptive gains[7] and the original GSTA[7] to evaluate whether adding adaptive gains to the GSTA actually improves the resulting tracking capabilities by combining the theoretical advantages afforded by the GSTA with the practical advantages afforded by adaptive gains. Additionally, we compare the results with those of a standard proportional-integral-derivative (PID) controller.

The contributions of this article can be summarized as follows. A novel adaptive GSTA is proposed for a class of systems whose perturbations and uncertain control coefficients may depend on both time and state. The proposed approach consists of using dynamically adapted control gains in a GSTA, such that no conservative upper bounds on the perturbations and uncertain control coefficients need to be considered to maintain sliding. A nonsmooth strict Lyapunov function is used to obtain conditions for GFT stability. It is also shown that the adaptive GSTA causes the tracking errors of an AIAUV to converge to zero in finite time. A simulation and experimental case study is performed to show the effectiveness of the proposed algorithm. We also present a comparison with the STA with adaptive gains[7] and the original GSTA[7] to evaluate whether adding adaptive gains to the GSTA actually improves the resulting tracking capabilities by combining the theoretical advantages afforded by the GSTA with the practical advantages afforded by adaptive gains. Additionally, we compare the results with those of a standard PID controller. Preliminary results were presented in Reference 28. In this article, moreover, we present not only simulations but also experimental results. Furthermore, the simulation results are new and we use the SRMTP method to create a continuous trajectory for the AIAUV to follow. Moreover, the comparison study is new; comparing the proposed algorithm with the STA with adaptive gains, the original GSTA and a standard PID controller through both simulations and experiments.

The remainder of this article is organized as follows. In Section 2, the problem statement and main results are given. The simulation and experimental case study of the AIAUV is reported in Section 3. In Section 4, conclusions and suggestions for future work are presented.

2 | PROBLEM STATEMENT AND MAIN RESULTS

In this section, we prove that the proposed GSTA with adaptive gains causes the system trajectories to globally converge to zero in finite time while accounting for unknown bounds on the uncertain control coefficient and perturbation term.

2.1 | System dynamics

Consider the dynamic system represented by the differential equation

$$\dot{\sigma} = \gamma(\sigma, t)u + \varphi(\sigma, t), \tag{1}$$

where $\sigma \in \mathbb{R}$ is the state vector and $u \in \mathbb{R}$ is the control input vector. The functions $\gamma(\sigma, t)$ and $\varphi(\sigma, t)$ are uncertain functions depending on the state and time. Following Reference 9, we adopt the following assumptions:

Assumption 1. The functions $\gamma(\sigma, t)$ and $\varphi(\sigma, t)$ are Lipschitz continuous functions with respect to t, and $\gamma(\sigma, t), \varphi(\sigma, t) \in C^1$ with respect to σ.

Remark 1. This means that $\gamma(\sigma, t)$ and $\varphi(\sigma, t)$ are limited in how fast they can change with time. The functions also have to be smooth (cannot contain any breaks, angles or cusps) and their derivatives with respect to σ have to exist at each point in the domain of each function. These are commonly accepted conditions as they are needed to guarantee that the control input is continuous. That is because $\sigma = 0 \rightarrow \dot{\sigma} = 0 \rightarrow u_{eq} = -\varphi(\cdot)/\gamma(\cdot)$.

Assumption 2. The uncertain control coefficient function is bounded by

$$0 < k_m \leq \gamma(\sigma, t) \leq k_M, \tag{2}$$

where k_m and k_M are positive constants.

Remark 2. This means that the uncertain control coefficient function, $\gamma(\sigma, t)$, needs to be lower and upper bounded by positive constants, that is, it needs to be positive and cannot be equal to zero or infinity. This is a commonly accepted constraint as it is needed for the control input to be valid and to guarantee that the control input is continuous. That is because $\sigma = 0 \rightarrow \dot{\sigma} = 0 \rightarrow u_{eq} = -\varphi(\cdot)/\gamma(\cdot)$.

Assumption 3. The perturbation term $\varphi(\sigma, t)$ can be split into two components:

$$\varphi(\sigma, t) = \varphi_1(\sigma, t) + \varphi_2(\sigma, t), \tag{3}$$

where the first component vanishes at the origin, that is, $\varphi_1(0, t) = 0, \forall t \geq 0$, and is bounded by

$$|\varphi_1(\sigma, t)| \leq \alpha|\phi_1(\sigma)|, \qquad \alpha > 0. \tag{4}$$

Remark 3. This means that the vanishing term of the perturbation, $\varphi_1(\sigma, t)$, needs to be upper bounded by a function that only depend on σ.

Assumption 4. The total time derivative of the nonvanishing component of the perturbation term divided by the control coefficient $\gamma(\sigma, t)$ can be represented as

$$\frac{d}{dt}\left(\gamma^{-1}(\sigma, t)\varphi_2(\sigma, t)\right) = \delta_1\sigma t)\gamma^{-1}\frac{\partial\varphi_2}{\partial t} - \gamma^{-2}\varphi_2\frac{\partial\gamma}{\partial t} + \delta_2\sigma t)\left(\gamma^{-1}\frac{\partial\varphi_2}{\partial\sigma} - \gamma^{-2}\varphi_2\frac{\partial\gamma}{\partial\sigma}\right)\dot{\sigma} = \delta_1(\sigma, t) + \delta_2(\sigma, t)\dot{\sigma}, \tag{5}$$

where $\delta_1(\sigma, t)$ and $\delta_2(\sigma, t)$ are bounded by positive constants:

$$|\delta_1(\sigma, t)| \leq \overline{\delta}_1, \qquad |\delta_2(\sigma, t)| \leq \overline{\delta}_2. \tag{6}$$

Remark 4. This means that the nonvanishing term divided by the control coefficient, $\gamma^{-1}(\sigma, t)\varphi_2(\sigma, t)$, can only grow linearly in σ. Since $\gamma(\sigma, t)$ has to be lower and upper bounded by constants (Assumption 2), this means that the nonvanishing term $\varphi_2(\sigma, t)$ can only grow at most linearly in σ.

Remark 5. Note that Assumptions 1–4 are exactly the same assumptions as made in Reference 9, which means that even though we add adaptive gains to the GSTA, we do not impose stricter conditions on the dynamic system.

2.2 | Generalized super-twisting algorithm with adaptive gains

In this section, the equations describing the adaptive GSTA are presented. The GSTA proposed in Reference 9 can be written as

$$\begin{aligned} u_{\text{AGSTA}} &= -k_1\phi_1(\sigma) + z \quad \in \mathbb{R} \\ \dot{z} &= -k_2\phi_2(\sigma) \end{aligned} \tag{7}$$

with

$$\begin{aligned} \phi_1(\sigma) &= \lceil\sigma\rfloor^{\frac{1}{2}} + \beta\sigma \\ \phi_2(\sigma) &= \frac{1}{2}\lceil\sigma\rfloor^0 + \frac{3}{2}\beta\lceil\sigma\rfloor^{\frac{1}{2}} + \beta^2\sigma, \end{aligned} \tag{8}$$

where $\lceil a\rfloor^b = |a|^b sgn(a)$ and $k_1 \in \mathbb{R}$, $k_2 \in \mathbb{R}$ and $\beta \in \mathbb{R}$ are constant controller gains. Motivated by Shtessel et al.,[7] we propose to instead let k_1 and k_2 be adaptive gains defined by the following update laws:

$$\dot{k}_1 = \begin{cases} \omega_1\sqrt{\frac{\gamma_1}{2}}, & \text{if } \sigma \neq 0, \\ 0, & \text{if } \sigma = 0, \end{cases} \tag{9a}$$

$$k_2 = 2\varepsilon k_1 + \lambda + 4\varepsilon^2, \tag{9b}$$

where $\varepsilon \in \mathbb{R}$, $\lambda \in \mathbb{R}$, $\gamma_1 \in \mathbb{R}$, and $\omega_1 \in \mathbb{R}$ are positive constants.

2.3 | Closed-loop dynamics

The closed-loop dynamics are obtained by substituting (3) and (7) into (1), yielding

$$\dot{\sigma} = -k_1\gamma(\sigma,t)\phi_1(\sigma) + \varphi_1(\sigma,t) + \gamma(\sigma,t)\left(z + \gamma^{-1}(\sigma,t)\varphi_2(\sigma,t)\right). \tag{10}$$

By defining $\sigma_1 = \sigma$ and $\sigma_2 = z + \gamma^{-1}(\sigma_1,t)\varphi_2(\sigma_1,t)$, we can represent the closed-loop dynamics as

$$\dot{\sigma}_1 = \gamma(\sigma_1,t)\left(-k_1\phi_1(\sigma_1) + \gamma^{-1}(\sigma_1,t)\varphi_1(\sigma_1,t) + \sigma_2\right) \tag{11a}$$

$$\dot{\sigma}_2 = -k_2\phi_2(\sigma_1) + \frac{d}{dt}\left(\gamma^{-1}(\sigma_1,t)\varphi_2(\sigma_1,t)\right). \tag{11b}$$

Theorem 1. *Suppose that $\gamma(\sigma_1,t)$ and $\varphi(\sigma_1,t)$ in system (1) satisfy Assumptions 1-4. Then, the closed-loop dynamics in (11) are GFTS, such that the states σ_1 and σ_2 converge to zero and z converges to $-\gamma^{-1}(0,t)\varphi_2(0,t)$, globally and in finite time, if the gains k_1 and k_2 are designed as expressed in (9), $\beta > 0$, $\lambda > 0$, $\omega_1 > 0$, $\gamma_1 > 0$, and $\varepsilon = \frac{\omega_2}{2\omega_1}\sqrt{\frac{\gamma_2}{\gamma_1}}$, where $\omega_2 > 0$ and $\gamma_2 > 0$.*

Remark 6. Note that the proof is for a one-dimensional case; however, since SMC approaches do not use model information, the n dimensions can simply be separated into n one-dimensional cases. The proof thus holds for n dimensions as long as Assumptions 1–4 hold for each dimension. This will be demonstrated in the case study presented in Section 3.

Remark 7. To make the difference between this proof and the results obtained in Reference 9 clear, and since we will use the Lyapunov function obtained in Reference 9, we start the proof of Theorem 1 by stating the results obtained in Reference 9.

Proof. From Reference 9, we have that the closed-loop system in (11) is GFTS when constant values of k_1, k_2 and $\beta > 0$ are used in (7) and the gains are chosen in accordance with Reference 9(theorem 2.1). This is proven using the Lyapunov function candidate

$$V_0 = \xi^T P \xi, \quad P = \begin{bmatrix} p_1 & -1 \\ -1 & p_2 \end{bmatrix}, \tag{12}$$

where $p_1 p_2 > 1$ and $\xi^T = \begin{bmatrix} \phi_1(\sigma_1) & \sigma_2 \end{bmatrix}$. It is shown that the derivative along the trajectory of the system is

$$\dot{V}_0 \leq -2\gamma(\sigma_1, t)\phi_1'(\sigma_1)\xi^T Q(t)\xi \leq -\mu_1 V_0^{\frac{1}{2}}(\sigma_1, \sigma_2) - \mu_2 V_0(\sigma_1, \sigma_2), \tag{13}$$

where

$$\mu_1 = \frac{k_m \epsilon \lambda_{\min}^{\frac{1}{2}}\{P\}}{\lambda_{\max}\{P\}}, \qquad \mu_2 = \beta \frac{2k_m \epsilon}{\lambda_{\max}\{P\}}, \tag{14}$$

and $Q(t)$ is positive definite if the gains are chosen in accordance with Reference 9(theorem 2.1). For the proposed adaptive GSTA, however, k_1 and k_2 are not constants. Instead, k_1 and k_2 are time-varying functions given by (9). Motivated by Shtessel et al.,[7] we use the Lyapunov function candidate defined in (12) to find a k_1 that satisfies (9a) such that $Q(t)$ is positive definite when k_2 is chosen as expressed in (9b). From Reference 9, the elements of $Q(t)$ are

$$Q(t) = \begin{bmatrix} q_1(t) & q_2(t) \\ q_2(t) & q_3(t) \end{bmatrix} = \begin{bmatrix} \tilde{k}_1\tilde{p}_1 - \tilde{k}_2 & \frac{1}{2}(p_2\tilde{k}_2 - (\tilde{k}_1\tilde{h} + \tilde{p}_1)) \\ \frac{1}{2}(p_2\tilde{k}_2 - (\tilde{k}_1\tilde{h} + \tilde{p}_1)) & \tilde{h} \end{bmatrix} \tag{15}$$

with

$$\begin{aligned}
\tilde{p}_1 &= \left(p_1 - \frac{\delta_2(\sigma_1, t)}{\phi_1'(\sigma_1)}\right) \Rightarrow \tilde{p}_1 \in [\underline{p}_1, \overline{p}_1] = \left[p_1 - \frac{\overline{\delta}_2}{\beta}, p_1 + \frac{\overline{\delta}_2}{\beta}\right], \\
\tilde{k}_2 &= \gamma^{-1}(\sigma_1, t)\left(k_2 - \frac{\delta_1(\sigma_1, t)}{\phi_2(\sigma_1)}\right) \Rightarrow \tilde{k}_2 \in [\underline{k}_2, \overline{k}_2] = \left[\frac{1}{k_M}(k_2 - 2\overline{\delta}_1), \frac{1}{k_m}(k_2 + 2\overline{\delta}_1)\right], \\
\tilde{k}_1 &= \left(k_1 - \gamma^{-1}(\sigma_1, t)\frac{\varphi_1(\sigma_1, t)}{\phi_1(\sigma_1)}\right) \Rightarrow \tilde{k}_1 \in [\underline{k}_1, \overline{k}_1] = \left[k_1 - \frac{\alpha}{k_m}, k_1 + \frac{\alpha}{k_m}\right], \\
\tilde{h} &= \left(1 - \frac{p_2\delta_2(\sigma_1, t)}{\phi_1'(\sigma_1)}\right) \Rightarrow \tilde{h} \in [\underline{h}, \overline{h}] = \left[1 - \frac{p_2\overline{\delta}_2}{\beta}, 1 + \frac{p_2\overline{\delta}_2}{\beta}\right].
\end{aligned} \tag{16}$$

For the matrix $Q(t)$ in (15) to be positive definite, we need $q_1(t) > 0$ and $\det(Q(t)) > 0$. If we choose $\tilde{k}_1 = k_1 - \frac{\alpha}{k_m}$ and $\tilde{k}_2 = \frac{1}{k_m}(k_2 + 2\overline{\delta}_1)$, we get that $q_1(t)$ and $\det(Q(t))$ are as small as possible. We therefore do not need to consider the case where $\tilde{k}_1 = k_1 + \frac{\alpha}{k_m}$ and $\tilde{k}_2 = \frac{1}{k_m}(k_2 - 2\overline{\delta}_1)$, since $q_1(t)$ and $\det(Q(t))$ then would be larger, and the matrix $Q(t)$ is thus positive definite if it is positive definite when $\tilde{k}_1 = k_1 - \frac{\alpha}{k_m}$ and $\tilde{k}_2 = \frac{1}{k_m}(k_2 + 2\overline{\delta}_1)$. With this choice of $\tilde{k}_1$ and $\tilde{k}_2$, we can rewrite Q(t) as

$$Q(t) = \begin{bmatrix} \left(k_1 - \frac{\alpha}{k_m}\right)\tilde{p}_1 - \frac{1}{k_m}(k_2 + 2\overline{\delta}_1) & q_2(t) \\ \frac{1}{2}\left(\frac{p_2}{k_m}(k_2 + 2\overline{\delta}_1) - \left(\left(k_1 - \frac{\alpha}{k_m}\right)\tilde{h} + \tilde{p}_1\right)\right) & \tilde{h} \end{bmatrix}. \tag{17}$$

By using k_2 as given in (9b) and calculating the determinant of $Q(t)$ in (17), we obtain

$$\begin{aligned}
&\det(Q(t)) = q_1(t)q_3(t) - q_2^2(t) \\
&= \left(\left(k_1 - \frac{\alpha}{k_m}\right)\tilde{p}_1 - \frac{1}{k_m}\left(2\varepsilon k_1 + \lambda + 4\varepsilon^2 + 2\overline{\delta}_1\right)\right)\tilde{h} - \frac{1}{4}\left(\frac{p_2}{k_m}\left(2\varepsilon k_1 + \lambda + 4\varepsilon^2 + 2\overline{\delta}_1\right) - \left(\left(k_1 - \frac{\alpha}{k_m}\right)\tilde{h} + \tilde{p}_1\right)\right)^2 \\
&= \tilde{h}\tilde{p}_1 k_1 - \frac{\alpha\tilde{h}\tilde{p}_1}{k_m} - \frac{2\varepsilon\tilde{h}}{k_m}k_1 - \frac{\lambda\tilde{h}}{k_m} - \frac{4\varepsilon^2\tilde{h}}{k_m} - \frac{2\overline{\delta}_1\tilde{h}}{k_m} - \frac{1}{4}\left(\frac{2\varepsilon p_2}{k_m}k_1 + \frac{\lambda p_2}{k_m} + \frac{4\varepsilon^2 p_2}{k_m} + \frac{2\overline{\delta}_1 p_2}{k_m} - \tilde{h}k_1 + \frac{\alpha\tilde{h}}{k_m} - \tilde{p}_1\right)^2.
\end{aligned} \tag{18}$$

By introducing $k_a = \frac{\alpha\tilde{h}\tilde{p}_1}{k_m} + \frac{\lambda\tilde{h}}{k_m} + \frac{4\varepsilon^2\tilde{h}}{k_m} + \frac{2\bar{\delta}_1\tilde{h}}{k_m} > 0$ and $k_b = \frac{\lambda p_2}{k_m} + \frac{4\varepsilon^2 p_2}{k_m} + \frac{2\bar{\delta}_1 p_2}{k_m} + \frac{\alpha\tilde{h}}{k_m} > 0$, where $\tilde{h} = 1 + \frac{p_2\bar{\delta}_2}{\beta}$ since this will make $\det(Q(t))$ the smallest, we can rewrite (18) as

$$\begin{aligned}
\det(Q(t)) &= \left(\left(\tilde{h}\tilde{p}_1 - \frac{2\varepsilon\tilde{h}}{k_m}\right)k_1 - k_a\right) - \frac{1}{4}\left(\left(\frac{2\varepsilon p_2}{k_m} - \tilde{h}\right)k_1 + k_b - \tilde{p}_1\right)^2 = \left(\left(\tilde{h}\tilde{p}_1 - \frac{2\varepsilon\tilde{h}}{k_m}\right)k_1 - k_a\right) \\
&- \left(\left(\frac{\varepsilon^2 p_2^2}{k_m^2} - \frac{\varepsilon\tilde{h}p_2}{k_m} + \frac{1}{4}\tilde{h}^2\right)k_1^2 + \left(\frac{\varepsilon k_b p_2}{k_m} - \frac{\varepsilon\tilde{p}_1 p_2}{k_m} - \frac{1}{2}\tilde{h}k_b + \frac{1}{2}\tilde{h}\tilde{p}_1\right)k_1 + \frac{1}{4}k_b^2 - \frac{1}{2}k_b\tilde{p}_1 + \frac{1}{4}\tilde{p}_1^2\right) \\
&= \left(-\frac{\varepsilon^2 p_2^2}{k_m^2} + \frac{\varepsilon\tilde{h}p_2}{k_m} - \frac{1}{4}\tilde{h}^2\right)k_1^2 + \left(\tilde{h}\tilde{p}_1 - \frac{2\varepsilon\tilde{h}}{k_m} - \frac{\varepsilon k_b p_2}{k_m} + \frac{\varepsilon\tilde{p}_1 p_2}{k_m} + \frac{1}{2}\tilde{h}k_b - \frac{1}{2}\tilde{h}\tilde{p}_1\right)k_1 - \frac{1}{4}k_b^2 + \frac{1}{2}k_b\tilde{p}_1 - \frac{1}{4}\tilde{p}_1^2 - k_a \\
&= (k_d - k_c)k_1^2 + (k_e - k_f)k_1 + k_h - k_g
\end{aligned} \tag{19}$$

with

$$k_c = \frac{\varepsilon^2 p_2^2}{k_m^2} + \frac{1}{4}\tilde{h}^2 > 0, \tag{20a}$$

$$k_d = \frac{\varepsilon\tilde{h}p_2}{k_m} > 0, \tag{20b}$$

$$k_e = \tilde{h}\tilde{p}_1 + \frac{\varepsilon\tilde{p}_1 p_2}{k_m} + \frac{1}{2}\tilde{h}k_b > 0, \tag{20c}$$

$$k_f = \frac{2\varepsilon\tilde{h}}{k_m} + \frac{\varepsilon k_b p_2}{k_m} + \frac{1}{2}\tilde{h}\tilde{p}_1 > 0, \tag{20d}$$

$$k_g = \frac{1}{4}k_b^2 + \frac{1}{4}\tilde{p}_1^2 + k_a > 0, \tag{20e}$$

$$k_h = \frac{1}{2}k_b\tilde{p}_1 > 0. \tag{20f}$$

A solution to

$$\det(Q(t)) = (k_d - k_c)k_1^2 + (k_e - k_f)k_1 + k_h - k_g > 0 \tag{21}$$

is then

$$k_1 > \frac{k_g - k_h}{k_e - k_f}, \tag{22}$$

where we must choose p_1 and p_2 such that $k_f < k_e$ and $k_d = k_c$. To ensure that $k_d = k_c$, we choose

$$p_2 = \frac{\tilde{h}k_m}{2\varepsilon}. \tag{23}$$

By substituting (23) into (20a) and (20b), we obtain

$$\begin{aligned}
k_c &= \frac{\varepsilon^2\left(\frac{\tilde{h}k_m}{2\varepsilon}\right)^2}{k_m^2} + \frac{1}{4}\tilde{h}^2 = \frac{1}{2}\tilde{h}^2, \\
k_d &= \frac{\varepsilon\tilde{h}\frac{\tilde{h}k_m}{2\varepsilon}}{k_m} = \frac{1}{2}\tilde{h}^2,
\end{aligned} \tag{24}$$

thus showing that $k_d = k_c$ is ensured. To ensure that $k_f < k_e$, we calculate

$$k_f < k_e$$
$$\frac{2\varepsilon\tilde{h}}{k_m} + \frac{\varepsilon k_b p_2}{k_m} + \frac{1}{2}\tilde{h}\tilde{p}_1 < \tilde{h}\tilde{p}_1 + \frac{\varepsilon\tilde{p}_1 p_2}{k_m} + \frac{1}{2}\tilde{h}k_b. \tag{25}$$

By substituting (23) into (25), we obtain

$$\begin{aligned}\frac{2\varepsilon\tilde{h}}{k_m} + \frac{\varepsilon k_b}{k_m}\frac{\tilde{h}k_m}{2\varepsilon} + \frac{1}{2}\tilde{h}\tilde{p}_1 &< \tilde{h}\tilde{p}_1 + \frac{\varepsilon\tilde{p}_1}{k_m}\frac{\tilde{h}k_m}{2\varepsilon} + \frac{1}{2}\tilde{h}k_b\\ \frac{2\varepsilon\tilde{h}}{k_m} + \frac{\tilde{h}k_b}{2} + \frac{1}{2}\tilde{h}\tilde{p}_1 &< \tilde{h}\tilde{p}_1 + \frac{\tilde{h}\tilde{p}_1}{2} + \frac{1}{2}\tilde{h}k_b\\ \frac{2\varepsilon}{k_m} &< \tilde{p}_1.\end{aligned} \tag{26}$$

This means that by choosing $\tilde{p}_1 > \frac{2\varepsilon}{k_m}$ or

$$p_1 > \frac{2\varepsilon}{k_m} + \frac{\bar{\delta}_2}{\beta}, \tag{27}$$

we can ensure that $k_f < k_e$. Therefore, by choosing k_1 as in (22), p_1 as in (27) and p_2 as in (23), we obtain $\det(Q(t)) > 0$. Now, to ensure that $q_1(t) > 0$, we calculate

$$\begin{aligned}q_1(t) &> 0,\\ \left(k_1 - \frac{\alpha}{k_m}\right)\tilde{p}_1 - \frac{1}{k_m}(2\varepsilon k_1 + \lambda + 4\varepsilon^2 + 2\bar{\delta}_1) &> 0,\\ \left(\tilde{p}_1 - \frac{2\varepsilon}{k_m}\right)k_1 - \frac{\alpha}{k_m}\tilde{p}_1 - \frac{1}{k_m}(\lambda + 4\varepsilon^2 + 2\bar{\delta}_1) &> 0.\end{aligned} \tag{28}$$

By choosing

$$k_1 > k_q = \left(\tilde{p}_1 - \frac{2\varepsilon}{k_m}\right)^{-1}\left(\frac{\alpha}{k_m}\tilde{p}_1 + \frac{1}{k_m}(\lambda + 4\varepsilon^2 + 2\bar{\delta}_1)\right), \tag{29}$$

we can ensure that $q_1(t) > 0$. By combining (22) and (29), we obtain

$$k_1 > \frac{k_g}{k_e - k_f} + k_q \tag{30}$$

which will ensure both that $\det(Q(t)) > 0$ and that $q_1(t) > 0$. Note that the term $-k_h$ that appears in the numerator in (22) is removed since $k_h > k_g$ could lead to $q_1(t) < 0$. Also note that this does not affect $\det(Q(t)) > 0$ since $\frac{k_g}{k_e - k_f} > \frac{k_g - k_h}{k_e - k_f}$ because $k_h > 0$. From this, we can conclude that if we choose k_1 as in (30), p_1 as in (27) and p_2 as in (23), then the matrix $Q(t)$ will be positive definite, thus ensuring that the closed-loop system in (11) will be GFTS when constant gains are used; in other words, we have proven that there exists a gain k_1 such that $Q(t)$ is positive definite. We now need to prove that when the adaptive gains defined in (9) are adopted, k_1 will converge such that (30) is satisfied.

Now, we will use the Lyapunov function candidate

$$V = V_0 + \frac{1}{2\gamma_1}(k_1 - k_1^\star)^2 + \frac{1}{2\gamma_2}(k_2 - k_2^\star)^2, \tag{31}$$

where $k_1^\star > 0$ and $k_2^\star > 0$ are constants, to prove that the closed-loop dynamics in (11) are also GFTS with the adaptive gains given in (9). By taking the derivative of (31), we obtain

$$\dot{V} = \dot{V}_0 + \frac{1}{\gamma_1}(k_1 - k_1^\star)\dot{k}_1 + \frac{1}{\gamma_2}(k_2 - k_2^\star)\dot{k}_2. \tag{32}$$

By using the fact that $\dot{V}_0 \leq -\mu_1 V_0^{\frac{1}{2}}(\sigma_1, \sigma_2)$ and subtracting and adding $\frac{\omega_1}{\sqrt{2\gamma_1}}|k_1 - k_1^\star| + \frac{\omega_2}{\sqrt{2\gamma_2}}|k_2 - k_2^\star|$, we can rewrite (32) as

$$\dot{V} \leq -\mu_1 V_0^{\frac{1}{2}} - \frac{\omega_1}{\sqrt{2\gamma_1}}|k_1 - k_1^\star| - \frac{\omega_2}{\sqrt{2\gamma_2}}|k_2 - k_2^\star| + \frac{1}{\gamma_1}(k_1 - k_1^\star)\dot{k}_1 + \frac{1}{\gamma_2}(k_2 - k_2^\star)\dot{k}_2 + \frac{\omega_1}{\sqrt{2\gamma_1}}|k_1 - k_1^\star| + \frac{\omega_2}{\sqrt{2\gamma_2}}|k_2 - k_2^\star|. \tag{33}$$

By using the well-known inequality

$$(x^2 + y^2 + z^2)^{1/2} \leq |x| + |y| + |z| \tag{34}$$

in (31), we obtain

$$\begin{aligned}\sqrt{V} &= \left(V_0 + \frac{1}{2\gamma_1}(k_1 - k_1^\star)^2 + \frac{1}{2\gamma_2}(k_2 - k_2^\star)^2\right)^{\frac{1}{2}} \\ &\leq V_0^{\frac{1}{2}} + \frac{1}{\sqrt{2\gamma_1}}|k_1 - k_1^\star| + \frac{1}{\sqrt{2\gamma_2}}|k_2 - k_2^\star|.\end{aligned} \tag{35}$$

We can then derive

$$-\mu_1 V_0^{\frac{1}{2}} - \frac{\omega_1}{\sqrt{2\gamma_1}}|k_1 - k_1^\star| - \frac{\omega_2}{\sqrt{2\gamma_2}}|k_2 - k_2^\star| \leq -\eta\sqrt{V}, \tag{36}$$

where $\eta = \min(\mu_1, \omega_1, \omega_2)$. Considering (36), we can rewrite (33) as

$$\dot{V} \leq -\eta V^{\frac{1}{2}} + \frac{1}{\gamma_1}(k_1 - k_1^\star)\dot{k}_1 + \frac{1}{\gamma_2}(k_2 - k_2^\star)\dot{k}_2 + \frac{\omega_1}{\sqrt{2\gamma_1}}|k_1 - k_1^\star| + \frac{\omega_2}{\sqrt{2\gamma_2}}|k_2 - k_2^\star|. \tag{37}$$

By Shtessel et al.[7(proposition 1)], we have that the adaptation law given in (9) causes the adaptive gains k_1 and k_2 to be bounded. Then, there exist positive constants $k_1^\star$ and $k_2^\star$ such that

$$k_1(t) - k_1^\star < 0, \qquad k_2(t) - k_2^\star < 0 \qquad \forall t \geq 0. \tag{38}$$

We can therefore reduce (37) to

$$\dot{V} \leq -\eta V^{\frac{1}{2}} - |k_1 - k_1^\star|\left(\frac{1}{\gamma_1}\dot{k}_1 - \frac{\omega_1}{\sqrt{2\gamma_1}}\right) - |k_2 - k_2^\star|\left(\frac{1}{\gamma_2}\dot{k}_2 - \frac{\omega_2}{\sqrt{2\gamma_2}}\right), \tag{39}$$

where we must ensure that

$$-|k_1 - k_1^\star|\left(\frac{1}{\gamma_1}\dot{k}_1 - \frac{\omega_1}{\sqrt{2\gamma_1}}\right) - |k_2 - k_2^\star|\left(\frac{1}{\gamma_2}\dot{k}_2 - \frac{\omega_2}{\sqrt{2\gamma_2}}\right) = 0 \tag{40}$$

to achieve finite-time convergence. The satisfaction of (40) should be achieved through the adaptation of the gains k_1 and k_2, that is,

$$\dot{k}_1 = \omega_1\sqrt{\frac{\gamma_1}{2}}, \tag{41a}$$

$$\dot{k}_2 = \omega_2\sqrt{\frac{\gamma_2}{2}}. \tag{41b}$$

If we select $\varepsilon = \frac{\omega_2}{2\omega_1}\sqrt{\frac{\gamma_2}{\gamma_1}}$, then (9b) and (41b) are equal, since

$$k_2 = 2\varepsilon k_1 + \lambda + 4\varepsilon^2 \quad \Rightarrow$$
$$\dot{k}_2 = 2\varepsilon \dot{k}_1 \quad \Rightarrow \quad \dot{k}_2 = \varepsilon\omega_1\sqrt{2\gamma_1} = \omega_2\sqrt{\frac{\gamma_2}{2}}. \tag{42}$$

For finite-time convergence, $k_1(t)$ must satisfy (30). This means that $k_1(t)$ must increase in accordance with (41a) until (30) is satisfied; since $k_1(t)$ increases linearly, (30) will be satisfied in finite time. This guarantees the positive definiteness of the matrix $Q(t)$. Once (30) is satisfied, finite-time convergence is guaranteed according to (39), and as nicely described in the Introduction of Reference 29, this implies that the closed-loop system in (11) is GFTS. ∎

3 | CASE STUDY: ARTICULATED INTERVENTION-AUV

In this section, we apply the theoretical results of Section 2 for the tracking control of an AIAUV in a case study to show the effectiveness of the proposed control algorithm. In this case study, the SRMTP method[27] is used to create a continuous trajectory for the AIAUV to follow. We present both simulations and experiments. The purpose of the experiments is to validate the underlying theory and the robustness of the proposed method by showing that the proposed method also works in experiments and not only in the ideal case presented in simulations. We also present a comparison with the STA with adaptive gains[7] and the original GSTA[7] to evaluate whether adding adaptive gains to the GSTA actually improves the resulting tracking capabilities by combining the theoretical advantages afforded by the GSTA with the practical advantages afforded by adaptive gains. Additionally, we compare the results with those of a standard PID controller.

3.1 | AIAUV model

In this section, we present the model of the AIAUV. The AIAUV has n links and $n-1$ motorized joints, each of which is regarded as a one-dimensional Euclidean joint. The AIAUV also has m thrusters. The AIAUV is modeled as an underwater vehicle-manipulator system, with dynamic equations given in matrix form as follows:[30,31]

$$\dot{\xi} = J(\eta_2)\zeta = \begin{bmatrix} R_{Ib}^T(\eta_2) & 0_{3\times3} & 0_{3\times(n-1)} \\ 0_{3\times3} & J_{k,o}^{-1}(\eta_2) & 0_{3\times(n-1)} \\ 0_{(n-1)\times3} & 0_{(n-1)\times3} & I_{(n-1)\times(n-1)} \end{bmatrix}\zeta, \tag{43a}$$
$$M(q)\dot{\zeta} + C(q,\zeta)\zeta + D(q,\zeta)\zeta + g(q,R_{Ib}) = \tau(q) \tag{43b}$$

with $\xi = [\eta_1^T \quad \eta_2^T \quad q^T]^T \in \mathbb{R}^{6+(n-1)}$, where $\eta_1 = [x \quad y \quad z]^T \in \mathbb{R}^3$ is the position, $\eta_2 = [\phi \quad \theta \quad \psi]^T \in \mathbb{R}^3$ is the set of Euler angles describing the orientation of the AIAUV in the inertial frame, and $q \in \mathbb{R}^{(n-1)}$ is the vector representing the joint angles; and $\zeta = [\nu^T \quad \omega^T \quad \dot{q}^T]^T \in \mathbb{R}^{6+(n-1)}$, where ν and ω are the body-fixed linear and angular velocities, respectively, and $\dot{q}$ is the vector of the joint angle velocities. The rotation matrix R_{Ib} expresses the transformation from the inertial frame to the body-fixed frame, $J_{k,o}$ is the Jacobian matrix, and $I_{(n-1)\times(n-1)}$ is the $(n-1)\times(n-1)$ identity matrix. The matrix $M(q)$ is the inertia matrix, $C(q,\zeta)$ is the Coriolis matrix, $D(q,\zeta)$ is the damping matrix, $g(q,R_{Ib})$ is the vector of gravitational and buoyant forces, and τ is the control input. The dynamic model given in (43) can be formulated with respect to a coordinate frame with its origin at an arbitrary position on the AIAUV. In this article, we assume that the model is formulated with respect to the center link of the AIAUV, such that the velocity state vector, ζ, is defined as

$$\zeta = \begin{bmatrix} \nu_{Ic}^c \\ \omega_{Ic}^c \\ \dot{q} \end{bmatrix} \in \mathbb{R}^{6+(n-1)}, \tag{44}$$

where ν_{Ic}^c and ω_{Ic}^c are the body-fixed linear and angular velocities, respectively, of the center link of the AIAUV and $\dot{q}$ is the vector of the joint velocities. The control input is given by the generalized forces $\tau(q)$:

$$\tau(q) = \begin{bmatrix} T(q) & 0_{6\times(n-1)} \\ 0_{(n-1)\times m} & I_{(n-1)\times(n-1)} \end{bmatrix} \begin{bmatrix} \tau_{thr} \\ \tau_q \end{bmatrix}, \tag{45}$$

where $T(q) \in \mathbb{R}^{6\times m}$ is the thruster configuration matrix, $\tau_{thr} \in \mathbb{R}^m$ is the vector of the thruster forces and $\tau_q \in \mathbb{R}^{(n-1)}$ represents the joint torques.

3.2 | Test case: SRMTP method

In this section, the test case considered in the simulations and experiments is explained. To generate a continuous trajectory for the AIAUV to follow, we will use the SRMTP framework.[27] This choice is based on Reference 26, where it was shown how algorithms giving finite-time convergence, as we here have proved that the adaptive GSTA gives, can be combined with the SRMTP framework. It was proved that the resulting system would make multiple set-point regulation tasks converge asymptotically to zero, with strict priority between the tasks, and moreover without the strict requirement of perfect velocity control which is often made when using SRMTP algorithms. To demonstrate the approach, we consider the following two set-point regulation tasks:

1. Control the position and orientation of the front end (link 9),
2. Control the position and orientation of the back end (link 1).

This combination of tasks corresponds to the ability of the AIAUV to move to a position of interest and then perform a double observation task by simultaneously adjusting the position and orientation of its front and back ends.

The expressions for the task error and task Jacobian for task 1 are as follows:

Task 1: Position and orientation of the front end

$$\tilde{\chi}_1 = \left[(\tilde{\eta}^f_{1,If})^T, \tilde{\eta}^T_{2,f} \right]^T,$$
$$J_1 = J_{cf},$$

where $\tilde{\eta}^f_{1,If}$ is the position deviation of the front end and $\tilde{\eta}_{2,f}$ is the orientation deviation of the frond end. The task Jacobian J_1 for task 1 is the front end Jacobian J_{cf}, which relates the body-fixed velocities of the front end to the body-fixed velocity of the center link and the joint velocities. The task can then be solved by using the position and orientation of the center link and the two double-joint modules in front of the center link.

Similarly, the second task utilizes the position and orientation of the center link and the two double-joint modules behind the center link; accordingly, the expressions for the task error and task Jacobian for task 2 are as follows:

Task 2: Position and orientation of the back end

$$\tilde{\chi}_2 = \left[(\tilde{\eta}^b_{1,Ib})^T, \tilde{\eta}^T_{2,b} \right]^T,$$
$$J_2 = J_{cb},$$

where $\tilde{\eta}^b_{1,Ib}$ is the position deviation of the back end and $\tilde{\eta}_{2,b}$ is the orientation deviation of the back end. The task Jacobian J_2 for task 2 is the back end Jacobian J_{cb}, which relates the body-fixed velocities of the back end to the body-fixed velocity of the center link and the joint velocities.

The reference velocities, ζ_r, are calculated in accordance with

$$\zeta_r = J_1^+ \Lambda_1 \tilde{\chi}_1 + N_1 J_2^+ \Lambda_2 \tilde{\chi}_2, \tag{46}$$

where $J_i^+ = J_i^T (J_i J_i^T)^{-1}$ is the right Moore–Penrose pseudo-inverse of the task Jacobian and N_1 is the null space projector for task 1. The set-points $\chi_{i,d}$ were set manually and filtered through a third-order reference filter to avoid discontinuities and large jumps in the calculated reference velocities. The gain parameters in (46) were set to $\Lambda_1 = 0.5 I_{6\times6}$ and $\Lambda_2 = 0.5 I_{6\times6}$.

3.3 | Tracking control law using the adaptive GSTA

In this section, we develop a tracking control law for the AIAUV based on the adaptive GSTA and show that the sliding surface and, therefore, the tracking errors converge to zero in finite time. Let the sliding surface be defined as

$$\sigma = \zeta - \zeta_r \quad \in \mathbb{R}^{6+(n-1)} \tag{47}$$

and let the control input be given by

$$\tau(q) = u_{\text{AGSTA}} \quad \in \mathbb{R}^{6+(n-1)}, \tag{48}$$

where u_{AGSTA} is given in (7)–(9). Note that no additional terms are added to the control input to compensate for neither the unknown nor known terms, which are dependent on both state and time, in the AIAUV model. These terms can therefore be considered as a perturbation. By differentiating (47) and inserting (43), we obtain

$$\dot{\sigma} = \dot{\zeta} - \dot{\zeta}_r = M^{-1}(\cdot)(-C(\cdot)\zeta - D(\cdot)\zeta - g(\cdot) + \tau(\cdot)) - \dot{\zeta}_r \tag{49}$$

and by using the fact that $\zeta = \zeta_r + \sigma$ from (47), we obtain the following equation describing the dynamics of σ:

$$\dot{\sigma} = M^{-1}(\cdot)(-C(\cdot)(\zeta_r + \sigma) - D(\cdot)(\zeta_r + \sigma) - g(\cdot) + \tau(\cdot)) - \dot{\zeta}_r. \tag{50}$$

Now, we can separate (50) into $6 + (n-1)$ one-dimensional equations such that Theorem 1 can be used. The dynamics of σ_i can then be described by

$$\dot{\sigma}_i = m_i^{-1}(\cdot)(-c_i(\cdot)(\zeta_{r,i} + \sigma_i) - d_i(\cdot)(\zeta_{r,i} + \sigma_i) - g_i(\cdot) + \tau_i(\cdot)) - \dot{\zeta}_{r,i} + \boldsymbol{m}_{\neq i}^{-1}(\cdot)(-\boldsymbol{c}_{\neq i}(\cdot)(\zeta_{r,\neq i} + \sigma_{\neq i}) - \boldsymbol{d}_{\neq i}(\cdot)(\zeta_{r,\neq i} + \sigma_{\neq i}) - g_{\neq i}(\cdot)), \tag{51}$$

where $\zeta_{r,i}$ and $\dot{\zeta}_{r,i}$ are the velocity reference and its derivative, respectively, corresponding to $\sigma_i = \zeta_i - \zeta_{r,i}$; $m_i^{-1}(\cdot)$, $c_i(\cdot)$ and $d_i(\cdot)$ are the ith elements on the diagonals of the matrices $M^{-1}(\cdot)$, $C(\cdot)$ and $D(\cdot)$, respectively; g_i is element i in the vector $g(\cdot)$; and $\tau_i(\cdot)$ is element i in the control input $\tau(\cdot)$. The vectors $\boldsymbol{m}_{\neq i}^{-1}(\cdot) \in \mathbb{R}^{6+(n-1)-1}$, $\boldsymbol{c}_{\neq i}(\cdot) \in \mathbb{R}^{6+(n-1)-1}$ and $\boldsymbol{d}_{\neq i}(\cdot) \in \mathbb{R}^{6+(n-1)-1}$ consist of the elements (i,j), where $j = 1 : 6 + (n-1)$ but $j \neq i$, in the matrices $M^{-1}(\cdot)$, $C(\cdot)$ and $D(\cdot)$, respectively. The vectors $\zeta_{r,\neq i} \in \mathbb{R}^{6+(n-1)-1}$, $\sigma_{\neq i} \in \mathbb{R}^{6+(n-1)-1}$ and $g_{\neq i}(\cdot) \in \mathbb{R}^{6+(n-1)-1}$ consists of the elements j, where $j = 1 : 6 + (n-1)$ but $j \neq i$, in the velocity reference vector ζ_r, the state vector σ and the vector $g(\cdot)$, respectively.

Now, by introducing $\varphi_i(\sigma_i, t) = \varphi_{1,i}(\sigma_i, t) + \varphi_{2,i}(\sigma_i, t)$, where $\varphi_{1,i}(0, t) = 0$, and $\gamma_i(\cdot) = m_i^{-1}(\cdot)$ and substituting $\tau_i(\cdot)$ from (48) and (7)-(9) into the above expression, we obtain

$$\dot{\sigma}_i = -k_{1,i}\gamma_i(\cdot)\phi_1(\sigma_i) + \varphi_{1,i}(\sigma_i, t) + \gamma_i(\cdot)(z_i + \gamma_i^{-1}(\cdot)\varphi_{2,i}(\sigma_i, t)), \tag{52}$$

where $\varphi_{1,i}(\sigma_i, t) = \gamma_i(\cdot)(-c_i(\cdot)\sigma_i - d_i(\cdot)\sigma_i)$ and $\varphi_{2,i}(\sigma_i, t) = \gamma_i(\cdot)(-c_i(\cdot)\zeta_{r,i} - d_i(\cdot)\zeta_{r,i} - g_i(\cdot) - m_i(\cdot)\dot{\zeta}_{r,i}) + \boldsymbol{m}_{\neq i}^{-1}(\cdot)(-\boldsymbol{c}_{\neq i}(\cdot)(\zeta_{r,\neq i} + \sigma_{\neq i}) - \boldsymbol{d}_{\neq i}(\cdot)(\zeta_{r,\neq i} + \sigma_{\neq i}) - g_{\neq i}(\cdot))$. By setting $\sigma_{1,i} = \sigma_i$ and $\sigma_{2,i} = z_i + \gamma_i^{-1}(\cdot)\varphi_{2,i}(\sigma_i, t)$, we can then write the dynamics as

$$\begin{aligned} \dot{\sigma}_{1,i} &= -k_{1,i}\gamma_i(\cdot)\phi_1(\sigma_{1,i}) + \varphi_{1,i}(\sigma_{1,i}, t) + \gamma_i(\cdot)\sigma_{2,i}, \\ \dot{\sigma}_{2,i} &= -k_{2,i}\phi_2(\sigma_{1,i}) + \frac{d}{dt}\left(\gamma_i^{-1}(\cdot)\varphi_{2,i}(\sigma_{1,i}, t)\right). \end{aligned} \tag{53}$$

Now, if $\gamma(\cdot)_i$, $\varphi_{1,i}(\cdot)$ and $\varphi_{2,i}(\cdot)$ satisfy Assumptions 1–4, then Theorem 1 will be satisfied, and the dynamics in (53) will be GFTS. The sliding surface $\sigma_i = 0$ is then a GFTS equilibrium point, which means that σ_i converges to zero in finite time.

Now, to prove that $\gamma_i(\cdot)$, $\varphi_{1,i}(\cdot)$ and $\varphi_{2,i}(\cdot)$ satisfy Assumptions 1–4, we need to adopt the following assumptions regarding the AIAUV:

Assumption 5. The AIAUV is neutrally buoyant.

Assumption 6. The AIAUV has only revolute joints. The following properties then hold:[30]

1. Property 1: $0 < \lambda_{\min}(M) \leq ||M|| \leq \lambda_{\max}(M)$.

2. Property 2: $M = M^T > 0$.
3. Property 3: $\dot{M} = C + C^T$ and $\zeta^T(\dot{M} - 2C)\zeta = 0 \quad \forall \quad \zeta \in \mathbb{R}^{6+(n-1)}$.

Assumption 7. The reference trajectory and its derivatives are continuous and bounded by design.

Remark 8. Assumptions 5–7 are valid due to the design of the AIAUV and the reference trajectories. Note that Assumptions 5 and 6 are common underwater vehicle manipulator system modeling assumptions.

Assumption 8. The following conditions hold:

1. $|\gamma_i(\cdot)c_i(\cdot)| \leq k_{\alpha,c}$, where $k_{\alpha,c}$ is a positive constant,
2. $|\gamma_i(\cdot)d_i(\cdot)| \leq k_{\alpha,d}$, where $k_{\alpha,d}$ is a positive constant.

Assumption 9. The following conditions hold:

1. $|\frac{d}{dt}(c_i(\cdot))| \leq k_{\delta_1,c_1} + k_{\delta_2,c_1}|\dot{\sigma}_i|$, where k_{δ_1,c_1} and k_{δ_2,c_1} are positive constants,
2. $|\frac{d}{dt}(d_i(\cdot))| \leq k_{\delta_1,d_1} + k_{\delta_2,d_1}|\dot{\sigma}_i|$, where k_{δ_1,d_1} and k_{δ_2,d_1} are positive constants,
3. $|\frac{d}{dt}(g_i(\cdot))| \leq k_{\delta_1,g} + k_{\delta_2,g}|\dot{\sigma}_i|$, where k_{δ_1,g_1} and k_{δ_2,g_1} are positive constants,
4. $|\frac{d}{dt}\left(\gamma_i^{-1}(\cdot)(\boldsymbol{m}_{\neq i}^{-1}(\cdot)\boldsymbol{c}_{\neq i}(\cdot))\right)| \leq k_{\delta_1,c_2} + k_{\delta_2,c_2}|\dot{\sigma}_i|$, where k_{δ_1,c_2} and k_{δ_2,c_2} are positive constants,
5. $|\frac{d}{dt}\left(\gamma_i^{-1}(\cdot)(\boldsymbol{m}_{\neq i}^{-1}(\cdot)\boldsymbol{d}_{\neq i}(\cdot))\right)| \leq k_{\delta_1,d_2} + k_{\delta_2,d_2}|\dot{\sigma}_i|$, where k_{δ_1,d_2} and k_{δ_2,d_2} are positive constants,
6. $|\frac{d}{dt}\left(\gamma_i^{-1}(\cdot)(\boldsymbol{m}_{\neq i}^{-1}(\cdot)g_{\neq i}(\cdot))\right)| \leq k_{\delta_1,g_2} + k_{\delta_2,g_2}|\dot{\sigma}_i|$, where k_{δ_1,g_2} and k_{δ_2,g_2} are positive constants.

Remark 9. For all practical purposes, Assumptions 8 and 9 will be satisfied since the AIAUV is a mechanical system and has a limited control input, which will cause the velocities of the system to also be bounded. For theoretical purposes, it should be proven that these assumptions hold; however, this proof will be left as a topic for future work. Some of the assumptions can also be discarded when the adaptive GSTA is extended to MIMO systems; this is also a topic for future work.

Now, Assumption 2 holds because of Property 1 in Assumption 6, while Assumptions 3 and 4 hold because of Assumptions 8 and 9, respectively. The conditions of Theorem 1 are thus satisfied in each dimension, and therefore, by Theorem 1, the dynamics in (53) are GFTS. The sliding surface $\sigma = 0$ is therefore a GFTS equilibrium point, which means that σ converges to zero in finite time.

3.4 | Simulation results

In this section, the simulation implementation, set-up and results are presented.

3.4.1 | Implementation

The motion of the AIAUV was simulated using MATLAB Simulink. The model was implemented using the method described in Reference 32 and is based on the Eelume 2020 robot, which is the robot used in the experiments reported in Section 3.5. The 2020 version of the Eelume robot has a length of 6.15 m and weighs 194.1 kg, and it is shown in Figure 1. The AIAUV has $n = 9$ cylindrical links with a radius of 0.1 m, $n - 1 = 8$ revolute joints and $m = 12$ thrusters. Note that the robot has only 5 visible links in Figure 1; however, the links are interconnected by joint modules that allow rotation about two axes (y and z). Accordingly, we model each joint as two revolute joints interconnected by a small link; this is why we define the robot as possessing $n = 9$ links and $n - 1 = 8$ revolute joints. The properties of each link are presented in Table 1. In the thrusters column, "4: (X,Z), (X,-Z), Y, -Y" means that the link has four thrusters: two thrusters work in both the x-direction and the positive/negative z-direction in the link frame, and two thrusters work in the positive/negative y-direction. The thrusters that work in both the x- and z-directions are tilted by 45 deg to allow them to work in both directions. Joints 1, 3, 5, and 7 rotate about the z-axis, and joints 2, 4, 6, and 8 rotate about the y-axis. Joint rotation occurs in the link frame of the corresponding link, for example, joint 1 rotates about the z-axis of link 1. The link frames are right-handed coordinate systems in which the completely outstretched robot is oriented such that the x-axes point

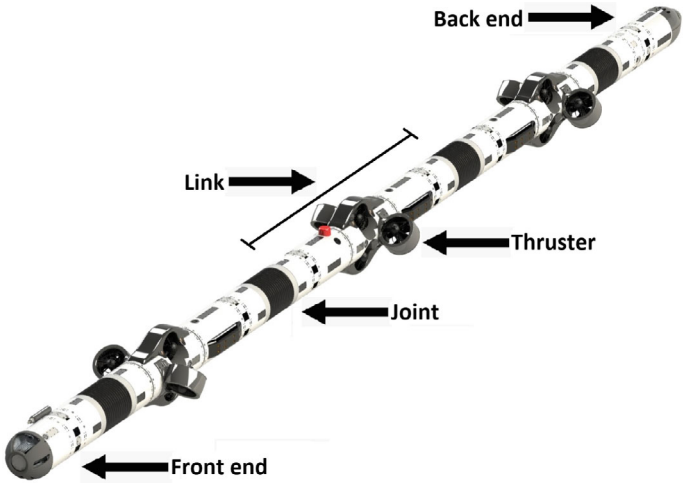


FIGURE 1 The Eelume 2020 vehicle (Courtesy: Eelume)

TABLE 1 Eelume 2020 link properties

Link no.	Length (m)	Mass (kg)	Thrusters
1	0.84	26.4	None
2, 4, 6, 8	0.08	2.6	None
3	1.38	43.5	4: (X,Z), (X,-Z), Y, -Y
5	1.63	51.3	4: (X,Z), (X,-Z), Y, -Y
7	1.38	43.5	4: (X,-Z), (X,Z), -Y, Y
9	0.60	19.0	None

TABLE 2 Physical parameters used in the simulations

Physical parameter	Value
Added mass coefficient for the cross section	1
Nonlinear drag coefficient in surge	0.2
Nonlinear drag coefficient in roll	0.1
Nonlinear crossflow drag coefficient	0.5
Linear cross-sectional drag coefficient	0.1
Added mass ratio in surge/heave for a link	0.2
Linear drag parameter in surge	0.1
Linear drag parameter in roll	0.1

forwards and the z-axes point upwards. In the simulations, the AIAUV was assumed to be neutrally buoyant. A thruster allocation matrix was implemented as proposed in Reference 32. The maximum thrust of each thruster is approximately 60 N, while the limit on the joint torques are 16 Nm. The physical parameters used in the simulations are shown in Table 2.

For implementation purposes, a small bound was placed on σ such that the adaptive gains could be expressed as

$$\dot{k}_{1,i} = \begin{cases} \omega_{1,i}\sqrt{\frac{\gamma_{1,i}}{2}}, & \text{if } |\sigma_i| > \alpha_{m,i} \\ 0, & \text{if } |\sigma_i| \leq \alpha_{m,i} \end{cases} \tag{54a}$$

$$k_{2,i} = 2\varepsilon_i k_{1,i} + \lambda_i + 4\varepsilon_i^2, \tag{54b}$$

where the design parameter $\alpha_m \in \mathbb{R}^{6+(n-1)}$ is a small, empirically chosen positive constant. This boundary was introduced because numerically σ will never be exactly zero, meaning that in practical implementations the second condition of (9a) will never be met, which would make the adaptive gains increase to infinity.

3.4.2 | Simulations

We used the simulation model presented in Section 3.4.1 to simulate the test case explained in Section 3.2. For the simulations, the ode3 fixed-step solver with a step size of 0.01 was used. We were not able to directly use the gains obtained during the experiments; it was necessary to make some small changes to the experimental gains to allow them to work properly in the simulations. However, we used the experimental gains as the starting point and attempted to modify them as little as possible. The gains used for the GSTA with adaptive gains are presented in Table 3, where e_i is a $1 \times i$ vector of ones. For the adaptive GSTA, it was necessary only to change the λ gain for the joints from 0.1 to 1. The results obtained when using the control law proposed in Section 3.3 are presented in Figure 2, and the evolution of the adaptive gains over time for the adaptive GSTA can be seen in Figure 3. In Figure 2F label q_i corresponds to the torque used for joint i, and in Figure 3 label x corresponds to the evolution of the adaptive gains $k_1(t)$ and $k_2(t)$ over time for the x-dimension, label y corresponds to the evolution of the adaptive gains $k_1(t)$ and $k_2(t)$ over time for the y-dimension, etc.

For comparison, we also obtained results using the previously mentioned algorithms, that is, the STA with adaptive gains from Reference 7 and the original GSTA from Reference 7. The STA with adaptive gains proposed in Reference 7 can be expressed as

$$\begin{aligned} u_{\mathrm{STA}}(\sigma) &= -\alpha|\sigma|^{1/2}sgn(\sigma) + v, \\ \dot{v} &= -\beta_{\mathrm{STA}}sgn(\sigma), \end{aligned} \tag{55}$$

where σ is the sliding surface and the adaptive gains are defined as

$$\dot{\alpha} = \begin{cases} \omega_1\sqrt{\frac{\gamma_1}{2}} & \text{if } \sigma \neq 0 \\ 0, & \text{if } \sigma = 0 \end{cases} \tag{56}$$

and

$$\beta_{\mathrm{STA}} = 2\varepsilon\alpha + \lambda + 4\varepsilon^2, \tag{57}$$

where ε, λ, γ_1, and ω_1 are positive constants. For implementation purposes, a small bound was placed on the sliding surface such that the adaptive gains could be expressed in the same manner as those for the adaptive GSTA, that is, as shown in (54). The GSTA proposed in Reference 9 can be expressed as

$$\begin{aligned} u_{\mathrm{GSTA}} &= -k_1\phi_1(\sigma) + z, \\ \dot{z} &= -k_2\phi_2(\sigma) \end{aligned} \tag{58}$$

TABLE 3 Simulation: control gains for the adaptive GSTA

Gain	Values
ε	$1 \times 10^{-5}e_{14}^T$
λ	e_{14}^T
γ_1	e_{14}^T
ω_1	$[e_6 \quad 0.1e_8]^T$
β	$[3e_6 \quad e_8]^T$
α_m	$0.05e_{14}^T$

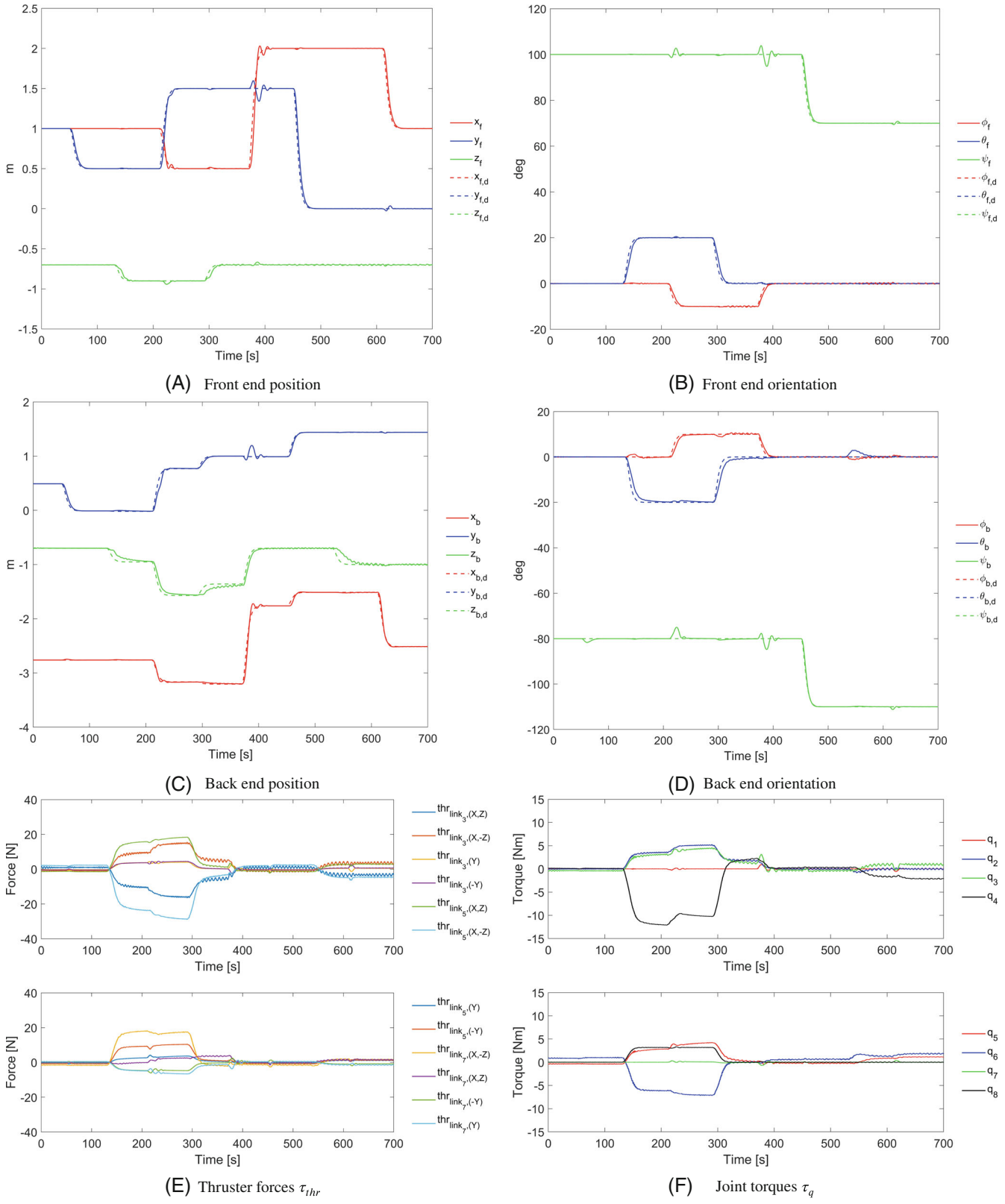
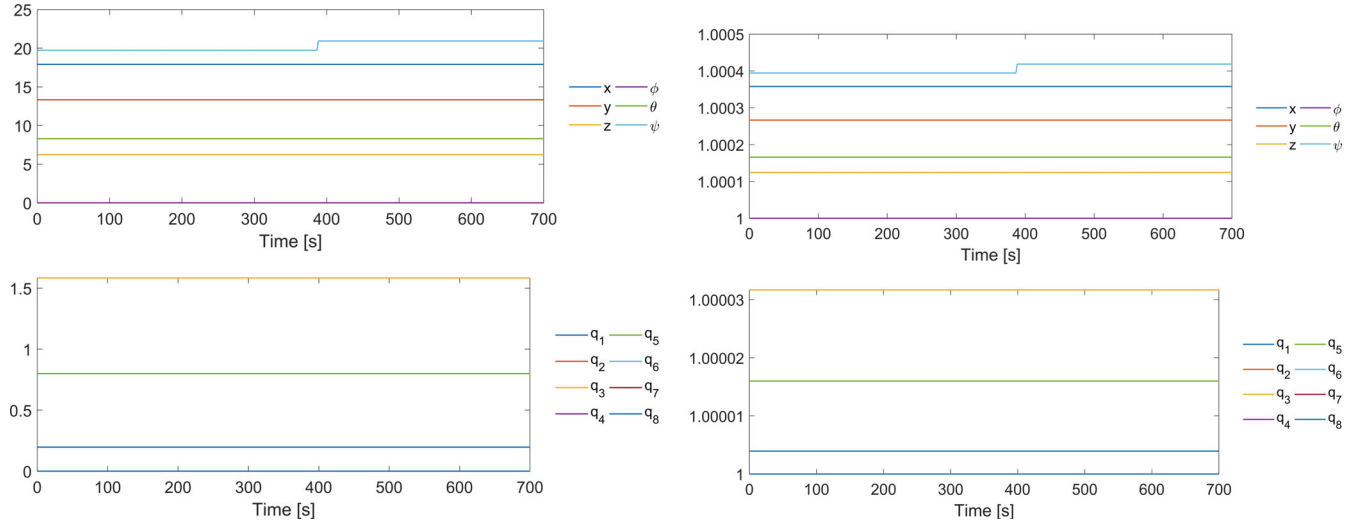


FIGURE 2 Simulation results obtained using the adaptive GSTA for the task-priority case



(A) Evolution of $k_1(t)$ over time

(B) Evolution of $k_2(t)$ over time

FIGURE 3 Simulation: evolution of the adaptive gains over time for the adaptive GSTA

with

$$\phi_1(\sigma) = \lceil \sigma \rfloor^{\frac{1}{2}} + \beta_{\text{GSTA}}\sigma,$$
$$\phi_2(\sigma) = \frac{1}{2}\lceil \sigma \rfloor^0 + \frac{3}{2}\beta_{\text{GSTA}}\lceil \sigma \rfloor^{\frac{1}{2}} + \beta_{\text{GSTA}}^2\sigma, \tag{59}$$

where k_1, k_2 and β_{GSTA} are the controller gains and σ is the sliding surface. For both the STA with adaptive gains and the GSTA, we used the sliding surface defined in (47). We also compare the results with those of a standard PID controller. The reason we chose to include a PID controller as a representative standard control method is because it is one of the most widely used types of controllers and is known to yield good results if tuned correctly. However, it does not provide any stability guarantee. The PID controller considered in this comparison is defined as

$$\tau = -k_p\tilde{\zeta} + k_d\dot{\zeta}_r - k_i\int \tilde{\zeta}dt, \tag{60}$$

where $\tilde{\zeta} = \zeta - \zeta_r$ and k_p, k_d, and k_i are the controller gains.

As mentioned previously, we were not able to use the control gains found during the experiments directly. The gains used in the simulations are presented in Table 4A for the STA with adaptive gains, in Table 4B for the GSTA and in Table 4C for the PID controller. For the STA with adaptive gains, we needed only to change the λ gain for the joints from 0.1 to 1. For the GSTA, it was necessary to change the β gains from $[80e_6 \quad 15e_8]$ to $[25e_6 \quad 45e_8]$, and for the PID controller, we needed to change the k_p gain for the joints from 20 to 40 and the k_i gain for the joints from 10 to 15, to get the control algorithms to work properly in the simulations. The tracking errors for all algorithms are presented in Figure 4. In Figure 5, we present the thruster forces and joint torques for all algorithms except the adaptive GSTA, for which the forces and torques have been previously presented in Figure 2. For the STA with adaptive gains, the evolution of the adaptive gains over time can be seen in Figure 6. In Table 5, the root mean square errors (RMSEs) and maximum errors of the position and orientation for each task are given for all algorithms. In Table 6, the root mean square (RMS) values of the thruster forces, the derivatives of the thruster forces, the joint torques, and the derivatives of the joint torques are given for all algorithms. The RMS of the thruster forces and the joint torques provides a measure of how much control effort that is used, and the RMS of the derivative of the thruster forces and the derivative of the joint torques provides a measure of how much chattering that is present in the thruster forces and the joint torques.

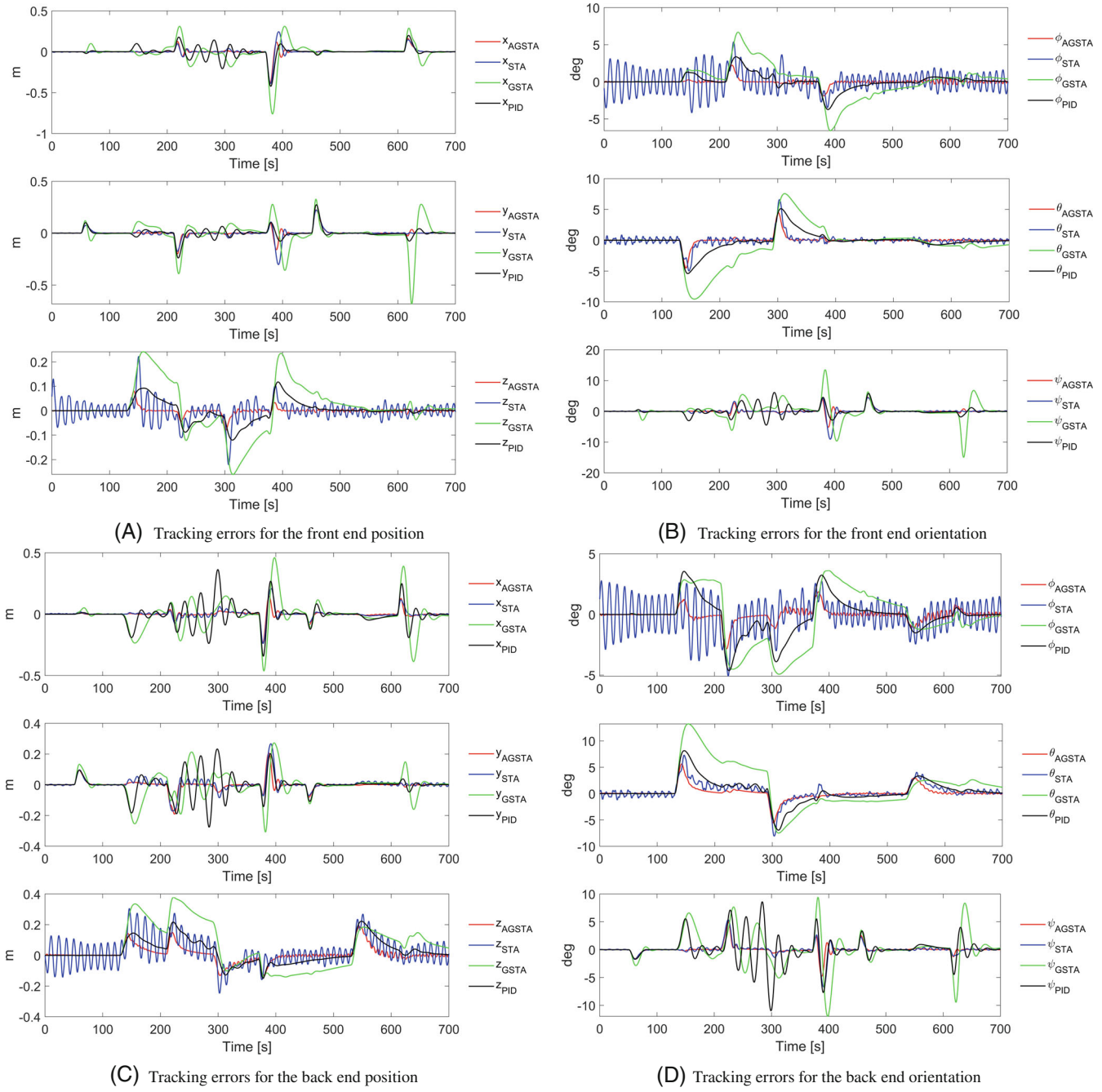
TABLE 4 Simulation: control gains

(A) For the STA with adaptive gains	
Gain	**Values**
ϵ	$1 \times 10^{-5} e_{14}^T$
λ	e_{14}^T
γ_1	e_{14}^T
ω_1	$[e_6 \quad 0.1e_8]^T$
α_m	$0.05e_{14}^T$
(B) For the GSTA	
Gain	**Values**
k_1	$[3e_6 \quad e_8]^T$
k_2	$[0.004e_6 \quad 0.002e_8]^T$
β	$[25e_6 \quad 45e_8]^T$
(C) For the PID controller	
Gain	**Values**
k_p	$[250e_6 \quad 40e_8]^T$
k_d	$[80e_6 \quad 10e_8]^T$
k_i	$[15e_6 \quad 15e_8]^T$

3.4.3 | Discussion

Figures 2A and 2B shows the commanded and actual position and orientation of the front end corresponding to task 1 when the GSTA with adaptive gains (7) is used. For task 1, we can see some small transient deviations at approximately 200 s and 400 s in the position and in the yaw orientation. These transient errors correspond to a roll movement of the front end, which may indicate that movement in the roll direction interferes with movement in the other degrees of freedom. The results for task 2 are shown in Figure 2C and 2D. As shown, the combination of the SRMTP method and the dynamic control law enables all tasks to be performed simultaneously, as indicated by Borlaug et al.[26(theorem 1)] in combination with Section 3.3; however, some transient deviations can also be observed for task 2. We see some small transient errors around 200 and 400 s in position and yaw that correspond to roll movements for task 1 and task 2. We also see small transient errors in the roll and pitch directions at approximately 550 s, which correspond to movement in the x-direction for task 1. These deviations occur because task 1 is the primary task, whereas task 2 is the secondary task. The first term in (46) does not consider the task errors for task 2. The attempt to achieve the desired front end position and orientation will consequently introduce errors in the back end position and orientation. These deviations disappear as soon as the second term in (46) compensates for these task errors. From the deviations for task 1, we can also conclude that the subtasks of task 1 interfere with each other. The simulation results therefore support the theoretical results obtained in Reference 26 as well as in Theorem 1 and Section 3.3, as the finite-time convergence property of the adaptive GSTA is essential for ensuring that all set-point tasks can be performed simultaneously. From Figure 2E, we can see that the forces used are smooth and well below 60 N, which is approximately the limit of the thrusters. From Figure 2F, we can see that the joint torques are smooth and below 16 Nm, which is the limit on the joint torques. The control inputs are therefore feasible. From Figure 3A and 3B, we can also see that the gains $k_1(t)$ and $k_2(t)$ increase linearly and converge to suitable values, which agrees with the theoretical results we found in Section 2. Note that even though the resulting $k_2(t)$ in this case is small, it is both necessary and practical to have both the gains $k_1(t)$ and $k_2(t)$ adaptive. It is necessary to complete the proof and practical as we did not know in advance that k_2 would be small. By having both k_1 and k_2 adaptive we are able to use the algorithm for a larger class of system, and we do not need to have any a prior knowledge of the size of k_1 and k_2.

From Figure 4, we can clearly see that there are some points that exhibit transient errors introduced by the SRMTP method. These transient errors appear when the set-point for task 1 or 2 is changed. It can also see that the magnitudes



(A) Tracking errors for the front end position

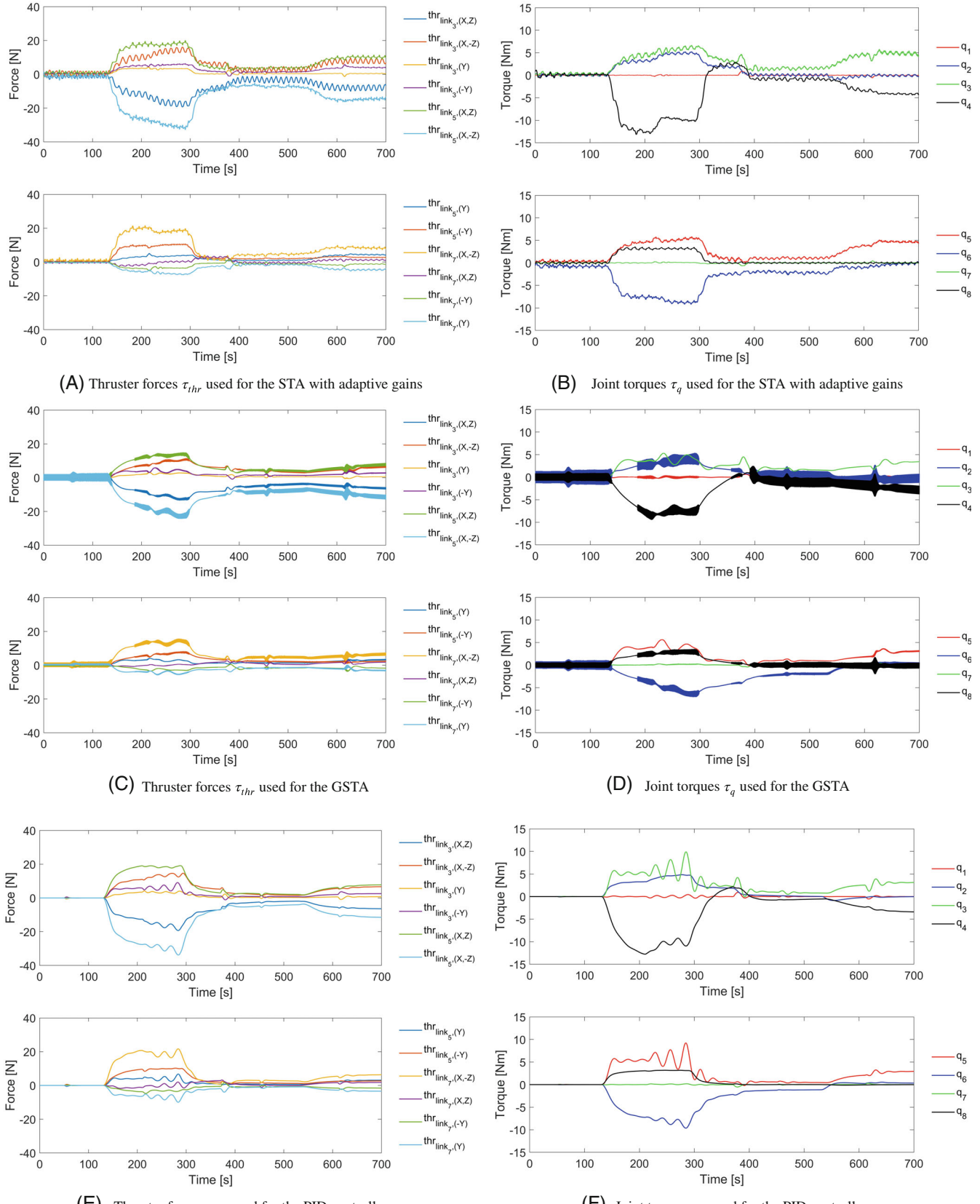
(B) Tracking errors for the front end orientation

(C) Tracking errors for the back end position

(D) Tracking errors for the back end orientation

FIGURE 4 Simulation: tracking errors for the task-priority case

of these transient errors are different for the different control algorithms. From Figure 4A, we can see that the adaptive GSTA clearly results in the smallest position errors for task 1, while the STA with adaptive gains yields the second smallest errors. However, the STA with adaptive gains also introduces some oscillation in z. The PID controller achieves the third smallest errors, while the GSTA results in the largest errors. Figure 4B shows that this is also the case for the orientation errors for task 1. However, it appears that the PID controller actually yields better results than the STA with adaptive gains. The STA with adaptive gains also introduces some oscillations into the states. When we consider the position errors for task 2 in Figure 4C, we see that the STA with adaptive gains again yields better results than the PID controller, but the STA with adaptive gains also introduces oscillation in z, while the adaptive GSTA still results in the lowest errors and best results. When we additionally consider the orientation errors for task 2 in Figure 4D, we see that here also, the adaptive GSTA achieves the lowest errors. The STA with adaptive gains again introduces some oscillations into the states; however, it also yields the second smallest errors. These findings are further supported by Table 5, which shows that the adaptive



(A) Thruster forces τ_{thr} used for the STA with adaptive gains

(B) Joint torques τ_q used for the STA with adaptive gains

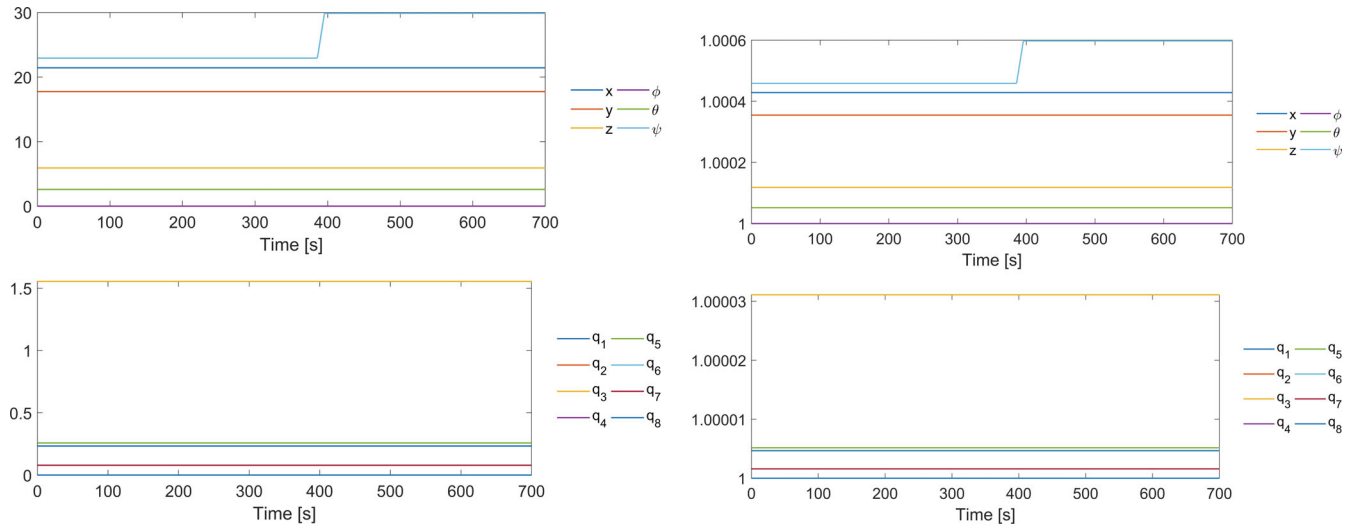
(C) Thruster forces τ_{thr} used for the GSTA

(D) Joint torques τ_q used for the GSTA

(E) Thruster forces τ_{thr} used for the PID controller

(F) Joint torques τ_q used for the PID controller

FIGURE 5 Simulation: thruster forces τ_{thr} and joint torques used for the task-priority case



(A) Evolution of $\alpha(t)$ over time (B) Evolution of $\beta(t)$ over time

FIGURE 6 Simulation: evolution of the adaptive gains over time for the STA with adaptive gains

TABLE 5 Simulation: comparison of the tracking results

		RMSE		Maximum error	
Algorithm		**Task 1**	**Task 2**	**Task 1**	**Task 2**
AGSTA	Position	0.0313	0.0382	0.2295	0.2124
	Orientation	0.0113	0.0146	0.0698	0.0787
STA	Position	0.0467	0.0581	0.3057	0.2713
	Orientation	0.0216	0.0239	0.1227	0.1158
GSTA	Position	0.1106	0.1177	0.5683	0.3832
	Orientation	0.0483	0.0550	0.1816	0.1755
PID	Position	0.0512	0.0763	0.2732	0.2882
	Orientation	0.0227	0.0356	0.0891	0.1385

TABLE 6 Simulation: comparison of the control inputs

	RMS			
Algorithm	τ_{thr}	$\dot{\tau}_{thr}$	τ_q	$\dot{\tau}_q$
AGSTA	4.7933	0.4191	1.9777	0.4487
STA	5.8351	0.7371	2.4834	0.8704
GSTA	4.3251	107.6121	1.8879	83.8284
PID	5.4231	0.1324	2.3276	0.0778

GSTA results in the smallest RMSE values and maximum error values in both position and orientation for both tasks. We can therefore conclude that the adaptive GSTA exhibits the best tracking performance overall, that is, the smallest errors. The STA with adaptive gains comes in second overall, even though the PID controller has smaller maximum error values for task 1. The PID controller thus takes third place, while the GSTA comes fourth.

Regarding the thruster and torque use of the different algorithms, we can see from Figures 2E, 2F and 5 that the force use is quite similar for all of the considered algorithms. However, the GSTA uses less force than the other algorithms and introduces some chattering, which may be the reason why it results in the highest tracking errors. From Table 6, we can

see that there are some small differences among the other algorithms, as well. However, the RMS values of the thruster forces and the joint torques indicate that the adaptive GSTA uses the least force after the GSTA, even though it produces the best tracking performance. Nevertheless, the STA with adaptive gains and the PID controller also use very similar amounts of force. Furthermore, from the RMS values of $\dot{\tau}_{thr}$ and $\dot{\tau}_q$, we can see that the only algorithm that introduces chattering is the GSTA.

3.5 | Experimental investigation

The purpose of the experiments was to validate the underlying theory and the robustness of the control approach by showing that the proposed approach also works in practice and not only in the ideal case presented in simulations. In the following subsections, we describe the Eelume robot, the experimental set-up employed to validate the proposed adaptive GSTA controller for trajectory tracking, and the obtained results.

3.5.1 | Eelume robot

The Eelume robot used in the experiments was the 2020 version of the robot, which is described by the simulation model introduced in Section 3.4.1. This new robot has torque control on the joints, which means that we could use our control algorithms to also control the joint angles by means of the torque, and not just the position and orientation of the vehicle by means of the thrusters.

3.5.2 | Experimental set-up

We performed the experiments at the Marine Cybernetics Laboratory (MC-Lab) at the Norwegian University of Science and Technology (NTNU), Trondheim, Norway.[33] The tank in the MC-Lab has the following dimensions: a length of 40 m, a height of 1.5 m and a width of 6.45 m. We used an underwater motion capture system from Qualisys,[34] which allowed us to track reflective markers attached to the center link of the AIAUV within a working area with dimensions of approximately 10 m × 1.35 m × 5.45 m. The Qualisys system provided us with real-time measurements of the position of the center link of the AIAUV. Figure 7 shows the 2020 version of the robot with the reflective markers attached to the center link.

The system structure used during the experiments is illustrated in Figure 8. The Qualisys system, or camera positioning system, sent the measured position and orientation from an external computer, to which the Qualisys system was connected, via the UDP to a different computer running LabVIEW 2016. The LabVIEW computer was then connected via UDP to another computer running MATLAB Simulink, in which the SRMTP method, the dynamic controllers and the thruster allocation were implemented. The computer running MATLAB Simulink was also running Eelume Suite, which is the program developed by Eelume to connect to the robot. From the Eelume Suite program, we obtained the orientation of the vehicle and the joint angles. We sent the joint torques and thrust commands from MATLAB Simulink to Eelume Suite. Eelume Suite then sent and received information to and from the robot through an optical fibre cable.

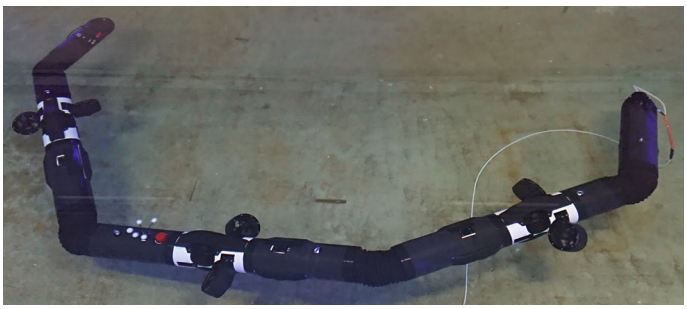


FIGURE 7 The Eelume 2020 vehicle with reflective markers attached to the center link

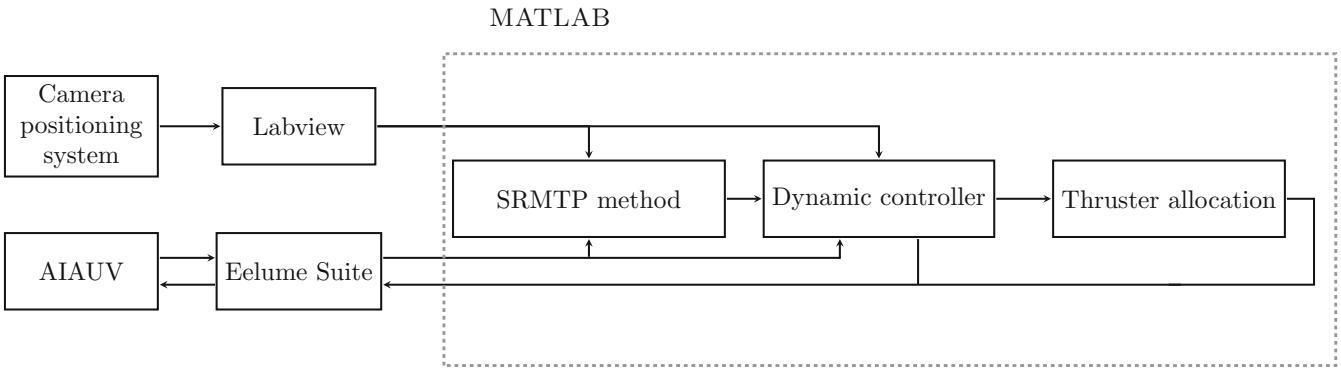


FIGURE 8 Illustration of the system structure

TABLE 7 Experiments: control gains for the adaptive GSTA

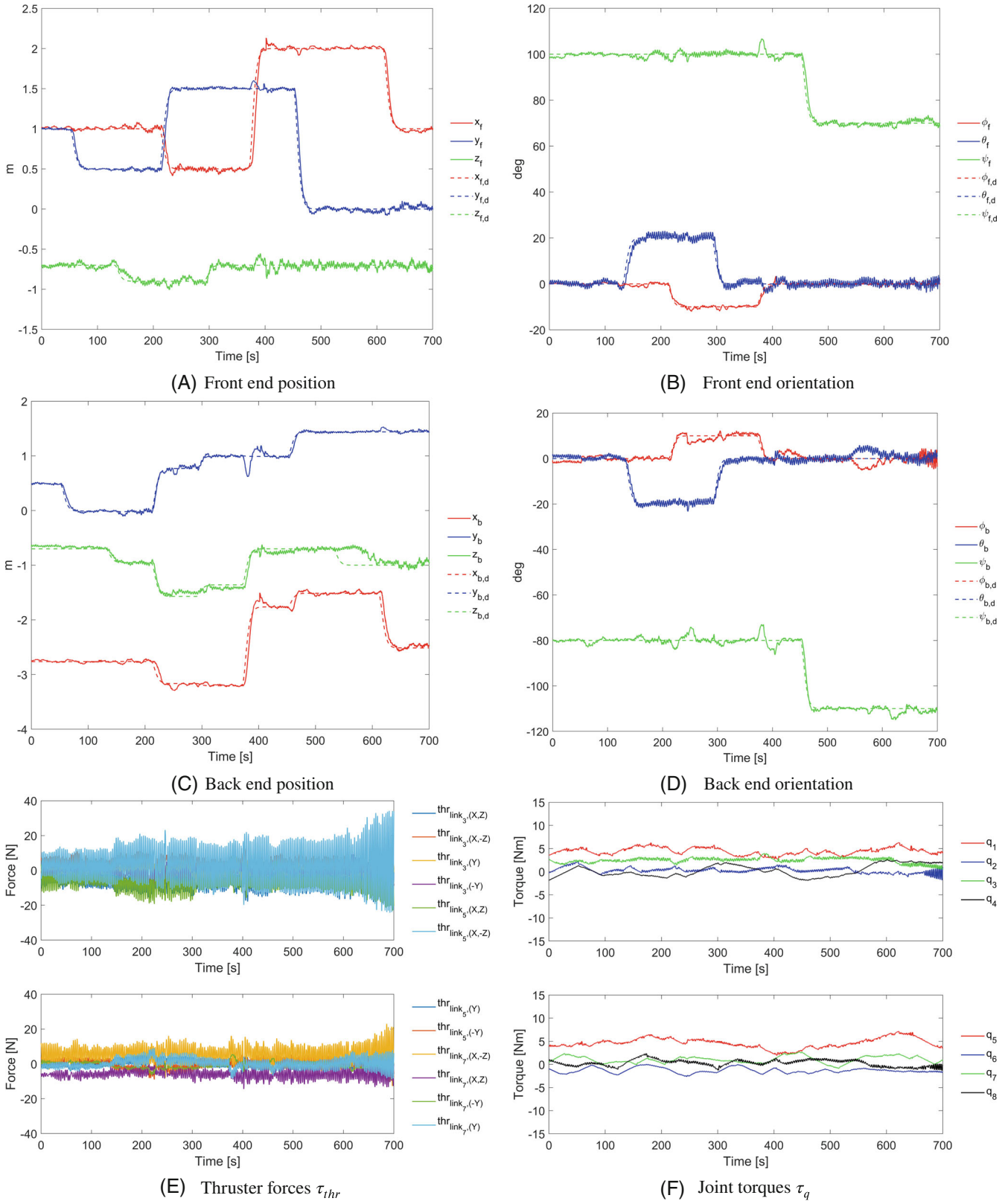
Gain	Values
ε	$1 \times 10^{-5} e_{14}^T$
λ	$[e_6 \quad 0.1e_8]^T$
γ_1	e_{14}^T
ω_1	$[e_6 \quad 0.1e_8]^T$
β	$[3e_6 \quad e_8]^T$
α_m	$0.05e_{14}^T$

To estimate the linear and angular velocities of the vehicle and the joint velocities, we used an extended Kalman filter based on the kinematic model, which was also implemented in MATLAB Simulink. As inputs to the Kalman filter, we used the position measurements from the Qualisys system and the orientation and joint measurements from the robot, as the orientation measurements from the robot were more accurate than those from the Qualisys system. To control the thrusters, we used current control. The thruster reference was thus proportional to the motor current, that is, we used a linear mapping to calculate the commanded reference sent from MATLAB Simulink based on the desired force. The commanded reference was a value in the range of ±100, corresponding to ±23 A on the motor and a force range of approximately ±60 N.

3.5.3 | Experimental results

In this subsection, the experimental results from the test described in Section 3.2 are presented. The gains for the GSTA with adaptive gains are presented in Table 7. For the adaptive GSTA, the choice of the gains should not be very important since the gains will autonomously adapt to suitable values, and indeed, it was observed during the experiments that it was easy to find a starting point for the adaptive gains that yielded good tracking results. The results obtained using the control law proposed in Section 3.3 are presented in Figure 9, and the evolution of the adaptive gains over time for the adaptive GSTA can be seen in Figure 10.

For comparison, we also obtained results using the previously mentioned algorithms, that is, the STA with adaptive gains (55), the original GSTA (58) and a PID controller (60). For the SMC algorithms, we used the sliding surface defined in (47). The gains found for the STA with adaptive gains, the GSTA and the PID controller during the experiments are presented in Table 8A–C. The tracking errors for all algorithms are presented in Figure 11. Figure 12 shows the thruster forces and joint torques for all algorithms except the adaptive GSTA, for which the forces and torques have been previously presented in Figure 9. For the STA with adaptive gains, the evolution of the adaptive gains over time can be seen in Figure 13. In Table 9, the RMSEs and maximum errors of the position and orientation for each task are given for all algorithms. In Table 10, the RMS values of the thruster forces, the derivatives of the thruster forces, the joint torques, and the derivatives of the joint torques are given for all algorithms.

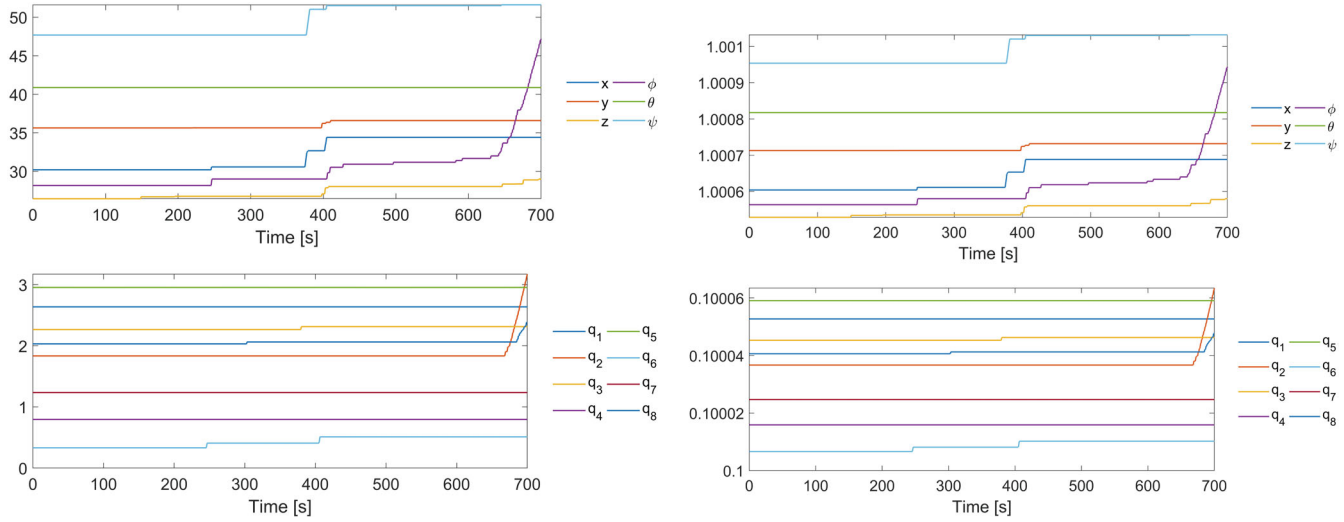


(A) Front end position (B) Front end orientation

(C) Back end position (D) Back end orientation

(E) Thruster forces τ_{thr} (F) Joint torques τ_q

FIGURE 9 Experimental results using the adaptive GSTA for the task-priority case



(A) Evolution of $k_1(t)$ over time

(B) Evolution of $k_2(t)$ over time

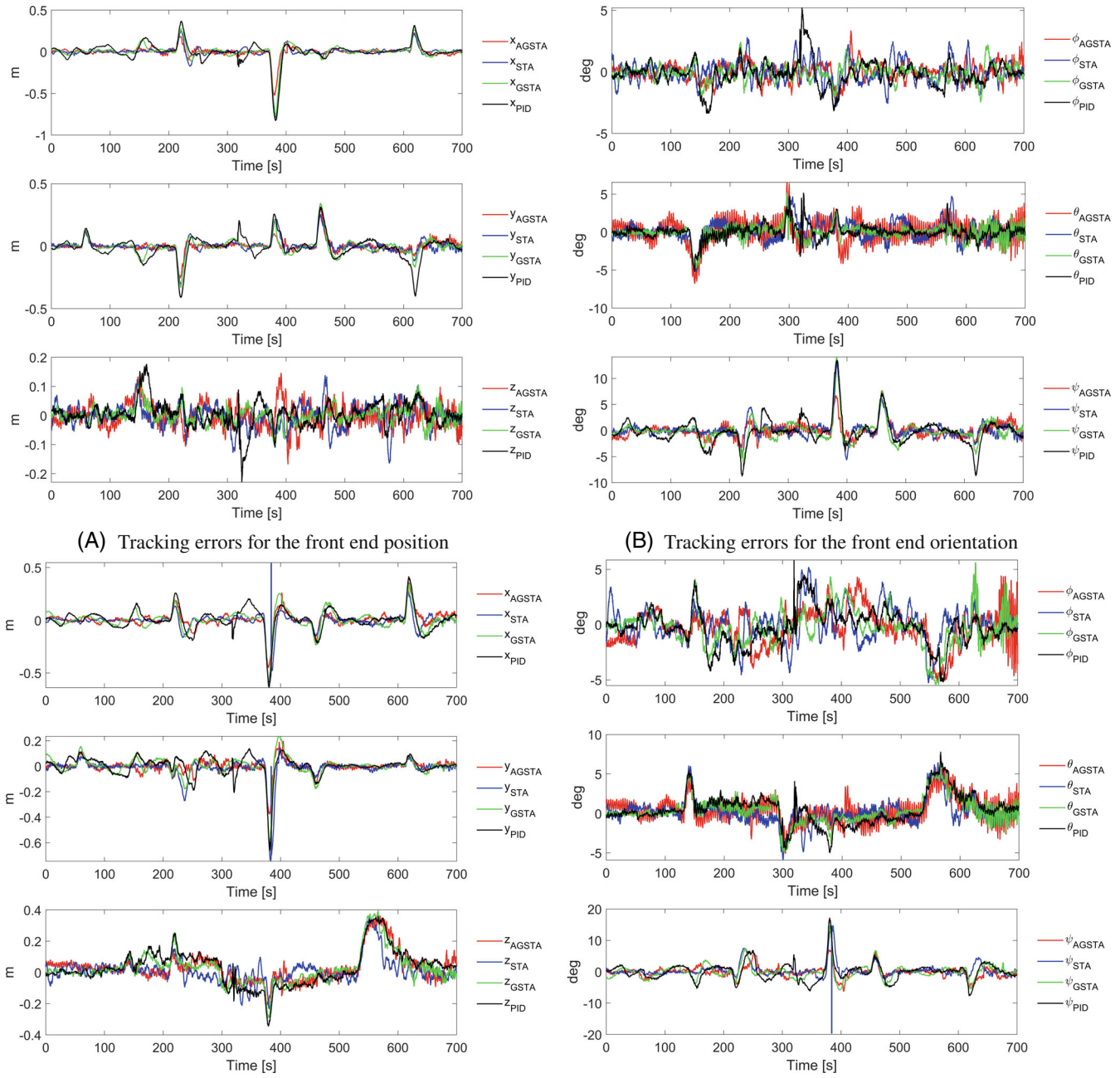
FIGURE 10 Experiments: evolution of the adaptive gains over time for the adaptive GSTA

TABLE 8 Experiments: control gains

(A) For the STA with adaptive gains	
Gain	**Values**
ε	$1 \times 10^{-5} e_{14}^T$
λ	$[e_6 \quad 0.1e_8]^T$
γ_1	e_{14}^T
ω_1	$[e_6 \quad 0.1e_8]^T$
α_m	$0.05e_{14}^T$
(B) For the GSTA	
Gain	Values
k_1	$[3e_6 \quad e_8]^T$
k_2	$[0.004e_6 \quad 0.002e_8]^T$
β	$[80e_6 \quad 15e_8]^T$
(C) For the PID controller	
Gain	Values
k_p	$[250e_6 \quad 20e_8]^T$
k_d	$[80e_6 \quad 10e_8]^T$
k_i	$[15e_6 \quad 10e_8]^T$

3.5.4 | Discussion

Figures 9A and 9B shows the commanded and actual position and orientation of the front end corresponding to task 1 when the GSTA with adaptive gains (7) is used. For task 1, we can see a small transient deviation at approximately 400 s in the yaw orientation. This transient error corresponds to a roll movement of the front end, which may indicate that movement in the roll direction interferes with movement in the yaw direction. This corresponds with the simulation results. Some other small oscillations are also evident, but these can simply be attributed to experimental uncertainties. The results for task 2 are shown in Figures 9C and 9D. It is seen that the combination of the SRMTP method and the



(A) Tracking errors for the front end position

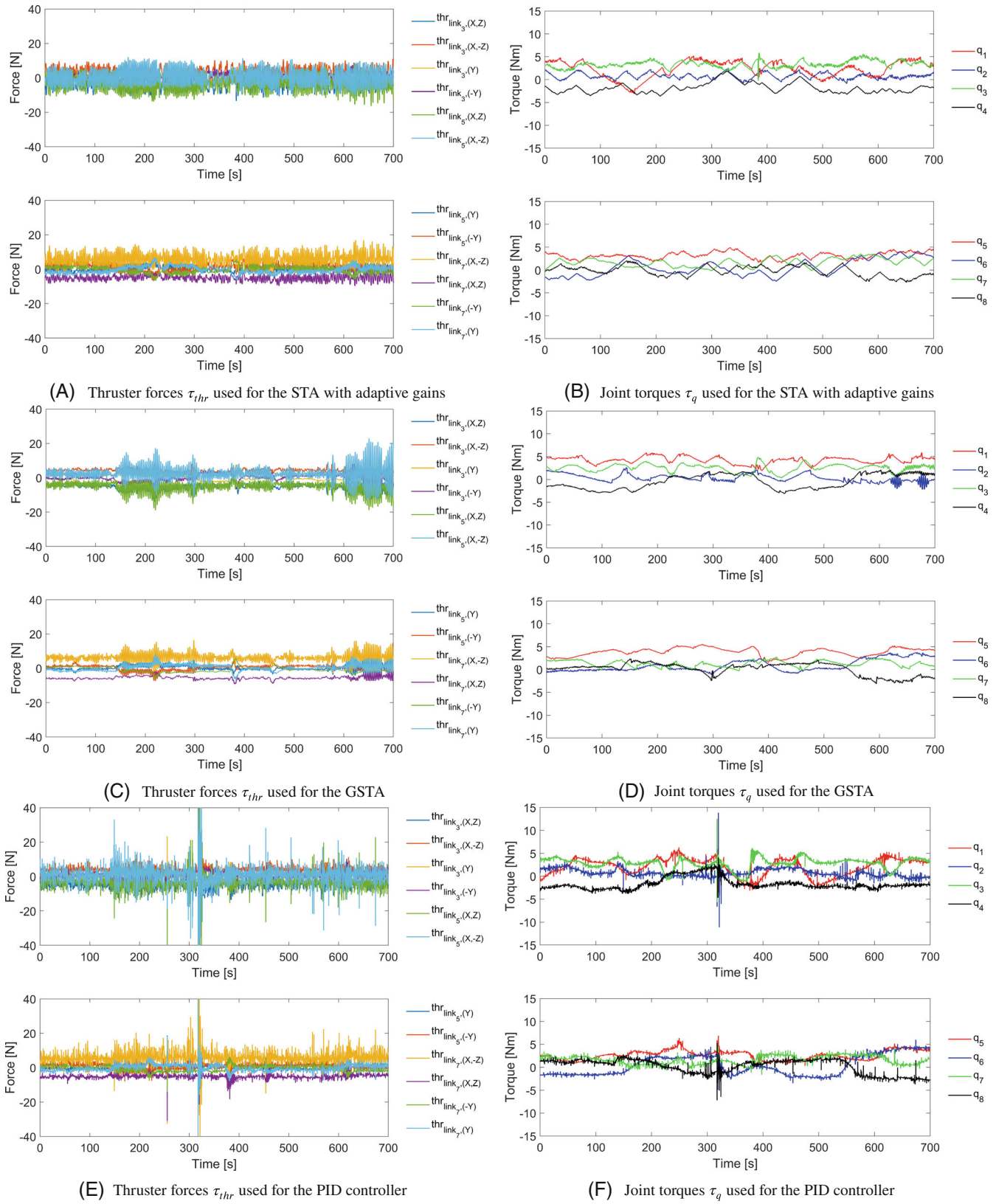
(B) Tracking errors for the front end orientation

(C) Tracking errors for the back end position

(D) Tracking errors for the back end orientation

FIGURE 11 Experiments: tracking errors for the task-priority case

dynamic control law allows all tasks to be performed simultaneously, as indicated by Borlaug et al.[26(theorem 1)] in combination with Section 3.3. Some transient deviations can be observed also for task 2. We see some small transient errors at approximately 200 and 400 s in position and yaw that correspond to roll movements for task 1 and task 2. We also see small transient errors in the roll and pitch directions at approximately 550 s, which correspond to movement in the x-direction for task 1. This also corresponds with the simulation results. The reason why we are unable to eliminate these deviations is because they are introduced by the SRMTP method, as discussed in Section 3.4.3. The experimental results therefore support the theoretical results, and we find that all set-point tasks are achieved. From Figure 9E, we can see that the forces used are well below 60 N, which is approximately the limit of the thrusters. From Figure 9F, we can see that the joint torques are smooth and below 16 Nm, which is the limit on the joint torques. The control inputs are therefore feasible. From Figures 10A and 10B, we can see that the gains $k_1(t)$ and $k_2(t)$ converge to suitable values.



(A) Thruster forces τ_{thr} used for the STA with adaptive gains

(B) Joint torques τ_q used for the STA with adaptive gains

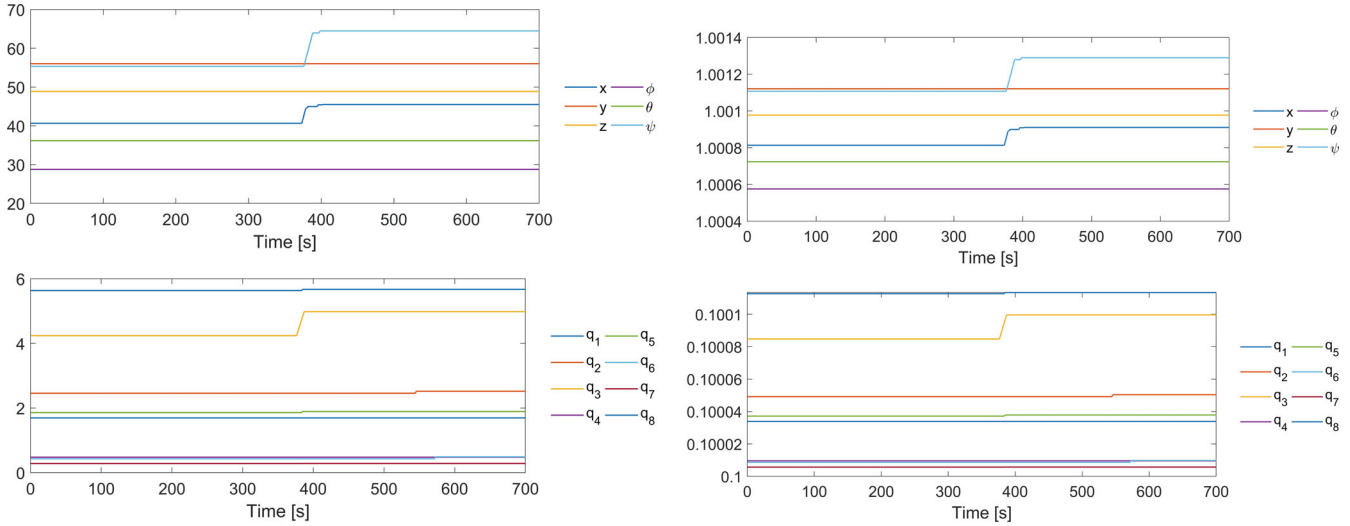
(C) Thruster forces τ_{thr} used for the GSTA

(D) Joint torques τ_q used for the GSTA

(E) Thruster forces τ_{thr} used for the PID controller

(F) Joint torques τ_q used for the PID controller

FIGURE 12 Experiments: thruster forces τ_{thr} and joint torques τ_q used for the task-priority case



(A) Evolution of $\alpha(t)$ over time

(B) Evolution of $\beta(t)$ over time

FIGURE 13 Experiments: evolution of the adaptive gains over time for the STA with adaptive gains

TABLE 9 Experiments: comparison of the tracking results

Algorithm		RMSE		Maximum error	
		Task 1	Task 2	Task 1	Task 2
AGSTA	Position	0.0521	0.0760	0.3323	0.4011
	Orientation	0.0220	0.0296	0.1040	0.1054
STA	Position	0.0651	0.0839	0.4213	0.5638
	Orientation	0.0235	0.0321	0.1253	0.1831
GSTA	Position	0.0649	0.0915	0.4154	0.5304
	Orientation	0.0215	0.0328	0.1267	0.1560
PID	Position	0.0851	0.1020	0.4900	0.5545
	Orientation	0.0263	0.0366	0.1394	0.1797

TABLE 10 Experiments: comparison of the control inputs

	RMS			
Algorithm	τ_{thr}	$\dot{\tau}_{thr}$	τ_q	$\dot{\tau}_q$
AGSTA	5.0325	7.8897	2.1488	0.3898
STA	3.6394	5.7950	2.2093	0.3645
GSTA	3.6527	3.3307	2.2126	0.3398
PID	3.2469	42.1036	2.1960	5.3478

By comparing the simulation results in Figure 2 with the experimental results in Figure 9, we can see that the same transient errors introduced by the SRMTP method occur in both the simulations and experiments. In the experiments, we also see some more oscillations and larger deviations; however, this is to be expected because of measurement noise, the thruster dynamics, the joint dynamics and other unmodeled dynamics that inevitably affect the experiment. A comparison of the thruster forces and joint torques reveals that the greatest difference between the simulations and experiments is that there is a stronger tendency toward chattering in the control inputs in the experiments; in addition, less force is used in the simulations. Nevertheless, the experimental results and

the simulation results are quite similar, indicating that the adaptive GSTA is indeed applicable for controlling the AIAUV.

From the position error plots in Figure 11, we can clearly see that there are some points that exhibit transient errors introduced by the SRMTP method. This behavior is less clear in the orientation error plots, although some tendency toward such a pattern is also seen here. It is also evident that the magnitudes of the transient errors are different for the different control algorithms, although it is not as easy to distinguish the different algorithms from each other as it is in the simulation results. We can see from Figure 11A that the adaptive GSTA yields the smallest position errors in x and y for task 1, although it is difficult to determine which algorithm performs the best in z. The STA with adaptive gains performs second best, while the performances of the GSTA and the PID controller are not easy to distinguish. In Figure 11B, it is also difficult to distinguish among the different algorithms based on the results in the roll and pitch directions for task 1, as the errors are quite small for all algorithms. For the yaw direction, the adaptive GSTA is seen to result in the smallest error for task 1, and the STA with adaptive gains yields the second best results; however, it is again difficult to distinguish between the GSTA and the PID controller. When we consider the position errors for task 2 in Figure 11C, we can see that for the x and y directions, the adaptive GSTA yields the smallest errors, while the results for z show little distinction. When we additionally consider the orientation errors for task 2 in Figure 11D, we see that here also, it is difficult to distinguish among the different algorithms based on the results for the roll and pitch directions; however, the adaptive GSTA is seen to result in the lowest error in yaw, although the other algorithms are still difficult to distinguish from each other in the yaw direction. These findings are also supported by Table 9, from which we can see that the adaptive GSTA results in the smallest RMSE values and maximum error values in both position and orientation for both tasks, with the exception of the RMSE orientation results for task 1, for which it is actually the GSTA that produces the lowest value. We can therefore conclude that the adaptive GSTA achieves the best tracking performance overall, that is, the smallest errors. In terms of the RMSE values and the maximum errors, the STA with adaptive gains and the GSTA actually yield very similar results. The GSTA usually achieves the best results for task 1, while the STA with adaptive gains achieves the best results for task 2. By contrast, the PID controller exhibits the highest RMSE values for both tasks and the highest maximum error values for task 1. Meanwhile, the highest maximum error values for task 2 correspond to the STA with adaptive gains. In Figure 11C,D, an outlier is evident in the results for the STA with adaptive gains; this outlier is probably the reason why the maximum error for task 2 is so high for the STA with adaptive gains.

Regarding the thruster and torque use of the different algorithms, we can see from Figures 9E and 9F and 12 that there are no large differences in the amounts of force used. Nevertheless, Table 10 reveals some small differences among the algorithms. The RMS values of the thruster forces indicate that the adaptive GSTA uses the greatest amount of force, which may be the reason why the adaptive GSTA achieves the best overall performance. Nevertheless, the other algorithms also use very similar amounts of thruster force. The RMS values of the joint torques also are all very similar. From the RMS values of $\dot{\tau}_{thr}$, we can see that there is some chattering in the thruster inputs for all algorithms; however, the most rapid changes are observed with the PID controller. From the $\dot{\tau}_q$ results, we can see that there is no chattering in the torque control inputs of the SMC controllers, while the PID controller introduces some rapid changes.

By comparing Figure 4 with Figure 11, we can see that overall, the tracking errors are smaller in the simulations, which can also be confirmed by comparing Table 5 with Table 9. This is to be expected because of measurement noise, the thruster dynamics, the joint dynamics and other unmodeled dynamics that inevitably affect the experiment. However, the GSTA actually shows better tracking performance in the experiments. This probably means that we could have found gains in the simulations that would have yielded better results. When we compare the forces used, by comparing Figures 2E and 2F and 5 with Figures 9E and 9F and 12, we can see evidence of more chattering in the control inputs during the experiments, which can also be confirmed by comparing Table 6 with Table 10. One other interesting observation is that while the adaptive GSTA uses less force in the simulations than in the experiments, the other algorithms actually use more force in the simulations than in the experiments. This may be because we tuned the gains of the controllers up in the simulations to obtain results similar to those obtained in the experiments.

One thing worth noting is that the results of this performance comparison between the STA with adaptive gains, the original GSTA and the PID controller are similar to those previously obtained in Reference 25. This strengthens the evidence from Reference 25 that adaptive gains are beneficial, as the results we obtain do not depend on our tuning capabilities. We also see that adding adaptive gains to the GSTA, thereby combining the theoretical advantages afforded by the GSTA and the practical advantages afforded by adaptive gains, improves the consequent tracking results and capabilities.

4 | CONCLUSIONS AND FUTURE RESEARCH

A novel adaptive GSTA is proposed for a class of systems whose perturbations and uncertain control coefficients may depend on both time and state. The proposed approach consists of using dynamically adapted control gains, such that no conservative upper bounds on the perturbations and uncertain control coefficients need to be considered to maintain sliding. We prove that the resulting closed-loop SISO system is GFTS. A simulation and experimental case study performed using an AIAUV is reported. It is also shown that the proposed adaptive GSTA causes the tracking errors of the AIAUV to converge to zero in finite time. In the case study, the SRMTP method is used to create a continuous trajectory for the AIAUV to follow. The primary task is to control the position and orientation of the front end of the AIAUV, and the secondary task is to control the position and orientation of the back end. The simulation and experimental results validate and verify that the proposed approach is well suited for controlling an AIAUV. The results are almost equally good between the simulations and experiments. The tracking errors are larger in the experiments than in the simulations; however, this is to be expected because of measurement noise, the thruster dynamics, the joint dynamics and other unmodeled dynamics that inevitably affect the experiment.

We also present a comparison with the STA with adaptive gains and the original GSTA to evaluate whether adding adaptive gains to the GSTA actually improves the tracking capabilities by combining the theoretical advantages afforded by the GSTA with the practical advantages afforded by adaptive gains. Based on this comparison, the adaptive GSTA yields the best tracking results overall without increasing the energy consumption, and the simulations and experiments thus indicate that adding adaptive gains to the GSTA does indeed improve the consequent tracking results and capabilities.

Future work will include investigating an adaptation method that allows the adaptive gains to be decreased and extending the adaptive GSTA to MIMO systems.

ACKNOWLEDGMENTS

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. The work has also received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme, through the ERC Advanced Grant 101017697-CRÈME.

CONFLICT OF INTEREST

There is no conflict of interest.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

ORCID

Ida-Louise G. Borlaug https://orcid.org/0000-0001-5269-3038

REFERENCES

1. Utkin VI. *Sliding Modes in Control and Optimization*. 1st ed. Springer-Verlag; 1992.
2. Burton JA, Zinober ASI. Continuous approximation of variable structure control. *Int J Syst Sci*. 1986;17(6):875-885.
3. Slotine JJ, Li W. *Applied Nonlinear Control*. 1st ed. Prentice-Hall; 1991.
4. Levant A. Higher-order sliding modes, differentiation and output-feedback control. *Int J Control*. 2003;76(9-10):924-941.
5. Levant A. Homogeneity approach to high-order sliding mode design. *Automatica*. 2005;41(5):823-830.
6. Shtessel YB, Shkolnikov IA, Levant A. Smooth second-order sliding modes: missile guidance application. *Automatica*. 2007;43(8):1470-1476.
7. Shtessel YB, Moreno JA, Plestan F, Fridman LM, Poznyak AS. Super-twisting adaptive sliding mode control: a Lyapunov design. Proceedings of the 49th IEEE Conference on Decision and Control; 2010:5109-5113; Atlanta, GA.
8. Levant A. Sliding order and sliding accuracy in sliding mode control. *Int J Control*. 1993;58(6):1247-1263.
9. Castillo I, Fridman LM, Moreno JA. Super-twisting algorithm in presence of time and state dependent perturbations. *Int J Control*. 2018;91(11):2535-2548.
10. Utkin VI, Poznyak AS. Adaptive sliding mode control with application to super-twist algorithm: equivalent control method. *Automatica*. 2013;49(1):39-47.
11. Obeid H, Laghrouche S, Fridman L, Chitour Y, Harmouche M. Barrier function-based adaptive super-twisting controller. *IEEE Trans Automat Contr*. 2020;65(11):4928-4933.

12. Moreno JA. Lyapunov approach for analysis and design of second order sliding mode algorithms. In: Fridman L, Moreno J, Iriarte R, eds. *Sliding Modes After the First Decade of the 21st Century*. *Lecture Notes in Control and Information Sciences*. Vol 412. Springer; 2012:113-149.
13. Gonzalez T, Moreno JA, Fridman L. Variable gain super-twisting sliding mode control. *IEEE Trans Automat Contr*. 2012;57(8): 2100-2105.
14. Guzmàn E, Moreno JA. Super-twisting observer for second-order systems with time-varying coefficient. *IET Control Theory Appl*. 2015;9(4):553-562.
15. Shtessel Y, Taleb M, Plestan F. A novel adaptive-gain super-twisting sliding mode controller: methodology and application. *Automatica*. 2012;48(5):759-769.
16. Wang Z. Adaptive smooth second-order sliding mode control method with application to missile guidance. *Trans Inst Meas Control*. 2017;39(6):848-860.
17. Wei J, Yuan J, Wang Z. Adaptive multivariable generalized super-twisting algorithm based robust coordinated control for a space robot subjected to coupled uncertainties. *Proc Inst Mech Eng G J Aerosp Eng*. 2019;233(9):3244-3259.
18. Moreno JA. A linear framework for the robust stability analysis of a generalized super-twisting algorithm. Proceedings of the 6th International Conference on Electrical Engineering, Computing Science and Automatic Control; 2009:1-6.
19. Guerrero J, Torres J, Creuze V, Chemori A. Trajectory tracking for autonomous underwater vehicle: an adaptive approach. *Ocean Eng*. 2019;172:511-522.
20. Sverdrup-Thygeson J, Kelasidi E, Pettersen KY, Gravdahl JT. The underwater swimming manipulator - A bioinspired solution for subsea operations. *IEEE J Ocean Eng*. 2018;43(2):1-16.
21. Borlaug ILG, Gravdahl JT, Sverdrup-Thygeson J, Pettersen KY, Loría A. Trajectory tracking for underwater swimming manipulator using a super twisting algorithm. *Asian J Control*. 2019;21(1):208-223.
22. Borlaug ILG, Pettersen KY, Gravdahl JT. Trajectory tracking for an articulated intervention AUV using a super-twisting algorithm in 6DOF. *IFAC PapersOnLine*. 2018;51(29):311-316. Proceedings of the 11th IFAC Conference on Control Applications in Marine Systems, Robotics, and Vehicles, Opatija, Croatia.
23. Borlaug ILG, Pettersen KY, Gravdahl JT. Tracking control of an articulated intervention AUV in 6DOF using the generalized super-twisting algorithm. Proceedings of the American Control Conference; 2019:5705-5712; Philadelphia.
24. Borlaug ILG, Pettersen KY, Gravdahl JT. Tracking control of an articulated intervention autonomous underwater vehicle in 6DOF using generalized super-twisting: theory and experiments. *IEEE Trans Control Syst Technol*. 2021;29(1):353-369.
25. Borlaug ILG, Pettersen KY, Gravdahl JT. Comparison of two second-order sliding mode control algorithms for an articulated intervention AUV: theory and experimental results. *Ocean Eng*. 2021;222:108480.
26. Borlaug ILG, Pettersen KY, Gravdahl JT. Combined kinematic and dynamic control of vehicle-manipulator systems. *Mechatronics*. 2020;69:102380.
27. Chiaverini S. Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators. *IEEE Trans Robot Autom*. 1997;13(3):398-410.
28. Borlaug ILG, Pettersen KY, Gravdahl JT. The generalized super-twisting algorithm with adaptive gains. Proceedings of the European Control Conference; 2020:1624-1631; Saint Petersburg, Russia. Best Student Paper Award Finalist.
29. Shtessel Y, Edwards C, Fridman L, Levant A. *Sliding Mode Control and Observation Control Engineering*. Birkhäuser; 2014.
30. Antonelli G. *Underwater Robots. Springer Tracts in Advanced Robotics*. Vol 96. 3rd ed. Springer International Publishing; 2014.
31. From PJ, Gravdahl JT, Pettersen KY. *Vehicle-Manipulator Systems: Modeling for Simulation, Analysis, and Control*. Advances in Industrial Control. Springer; 2014.
32. Schmidt-Didlaukies HM, Sørensen AJ, Pettersen KY. Modeling of articulated underwater robots for simulation and control. Proceedings of the 2018 IEEE/OES Autonomous Underwater Vehicles; 2018:1-7; Porto, Portugal.
33. Marine cybernetics laboratory (MC-lab); 2018. https://www.ntnu.edu/imt/lab/cybernetics
34. Qualisys-motion capture systems; 2018. https://www.qualisys.com

How to cite this article: Borlaug I-LG, Pettersen KY, Gravdahl JT. The generalized super-twisting algorithm with adaptive gains. *Int J Robust Nonlinear Control*. 2022;32(13):7240-7270. doi: 10.1002/rnc.6212