Planar Maneuvering Control of Underwater Snake Robots Using Virtual Holonomic Constraints

Anna M. Kohl[1], Eleni Kelasidi[1], Alireza Mohammadi[2], Manfredi Maggiore[2,3], and Kristin Y. Pettersen[1]

[1] Centre for Autonomous Marine Operations and Systems (NTNU-AMOS), Department of Engineering Cybernetics, NTNU, Norwegian University of Science and Technology, NO-7491 Trondheim, Norway

[2] Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON M5S 3G4, Canada

[3] This research was carried out while M. Maggiore was on sabbatical leave at the Laboratoire des Signaux et Systèmes, Centrale Supélec, Gif sur Yvette, France

E-mail: `anna.kohl@itk.ntnu.no, eleni.kelasidi@itk.ntnu.no, alireza.mohammadi@mail.utoronto.ca, maggiore@control.utoronto.ca, kristin.y.pettersen@itk.ntnu.no`

May 2016

Abstract. This paper investigates the problem of planar maneuvering control for bio-inspired underwater snake robots that are exposed to unknown ocean currents. The control objective is to make a neutrally buoyant snake robot which is subject to hydrodynamic forces and ocean currents converge to a desired planar path and traverse the path with a desired velocity. The proposed feedback control strategy enforces virtual constraints which encode biologically inspired gaits on the snake robot configuration. The virtual constraints, parametrized by states of dynamic compensators, are used to regulate the orientation and forward speed of the snake robot. A two-state ocean current observer based on relative velocity sensors is proposed. It enables the robot to follow the path in the presence of unknown constant ocean currents. The efficacy of the proposed control algorithm for several biologically inspired gaits is verified both in simulations for different path geometries and in experiments.

Keywords: feedback control of underwater snake robots, biologically inspired gaits, virtual holonomic constraints, hierarchical control, current estimation.

1. Introduction

In the contest of marine engineering, increasing autonomy results in more economic and efficient offshore operations such as exploration, monitoring, and maintenance. Species that have adapted to subsea conditions are extremely efficient for underwater propulsion and maneuvering. Bio-inspired underwater robots are capable of efficient animal-like

locomotion in aquatic environments because they have morphologies similar to species found in nature. Bio-inspired robotic systems are therefore potential candidates for providing robust, agile, and versatile autonomous solutions.

Fish-like robots are among the earliest robotic systems developed for underwater swimming [1, 2, 3, 4]. Fish-like robot locomotion is relatively easy to control due to the small number of degrees of freedom. Such robots, however, do not have extensive manipulation capabilities. A class of bio-inspired robots that combines locomotion and manipulation capabilities is that of underwater snake robots. These robots propel themselves forward by eel-like swimming [3]. Because of their slender structure, underwater snake robots can easily operate in narrow environments and are thus suitable for inspection and maintenance of offshore structures such as subsea oil installations. Research on robotic snakes started with land-based snake robots [5, 6, 7, 8], and was later extended to amphibious and underwater robots [9, 10, 11]. Recently, virtual holonomic constraints (VHCs) have been introduced for locomotion control of land-based snake robots [12, 13, 14, 15]. The VHC framework relies on enforcing time-independent relations that encode biologically inspired gaits on the configuration of the robot. VHC-based controllers have been employed for ape-like brachiating [16] and biped walking robots [17, 18]. In the context of snake robotics, the VHC framework makes the control design amenable to hierarchical synthesis, where the bio-inspired gaits are enforced at the lowest level and path planning is done at the highest level of hierarchy [12, 13]. Compared to other path following control strategies for snake robots [19, 10, 20, 21], the approach in [13] can be used for curved paths and not only straight lines and considers speed control in addition to direction control.

The contributions of this article are the following. The first contribution is extending the VHC framework that was developed for land-based snake robots in [12, 13] to underwater snake robots under the influence of hydrodynamic forces and ocean currents. The analysis shows that the VHC control scheme works for the underwater snakes as it is, without significantly changing the controllers, as long as the ocean currents are compensated for. The second contribution is the design of a reduced order observer for ocean current velocity estimation, thus maintaining the performance of the path following controller in the presence of currents. In addition to the lateral undulatory gait that was investigated in [12, 13], it is shown in this article that the VHC framework can also be used along with a more general class of biologically inspired gaits including eel-like motion. Finally, the VHC framework is experimentally validated. In particular, the control system was successfully tested for straight line path following for different gaits.

This article is organized as follows. In Section 2, biologically inspired gaits and implementation methods for snake and eel-like robots are reviewed. The model of the underwater snake robot which will be used in this paper is reviewed in Section 3. Our control strategy is presented in Section 4. The proposed control system is validated by extensive simulations in Section 5 and an experimental study in Section 6. The article ends with a discussion of the results in Section 7 and concluding remarks in Section 8.

2. Robotic implementation of snake locomotion gaits

In this section we review the biologically inspired snake robot gait and locomotion control paradigms for snake robots.

2.1. A discrete approximation of a biological gait

The focus of this article is on the important biological gait lateral undulation, where snakes achieve forward propulsion by a sinusoidal wave that travels through their body from head to tail. It is suitable for achieving amphibious forward propulsion, i.e., propulsion both on land and in water. Indeed, the lateral undulatory gait demonstrated by land-based snakes is the most similar to the underwater anguilliform gait, also called eel-like motion. A discrete approximation of a moving snake's body is frequently used to enforce an approximate undulatory gait on snake robots consisting of rigid links (see, e.g., [22, 6]). This lateral undulatory gait is achieved by making the single joints track the reference

$$\phi_{i,\text{ref}}(t) = \alpha \sin\big(\omega t - (i-1)\delta\big) + \phi_0(t), \tag{1}$$

where $i \in \{1, \cdots, N-1\}$ is an index indicating which joint the reference signal is used for; α is the maximal amplitude of the joint angles; ω is the frequency of the body oscillations; δ is the phase shift between the adjacent joints of the robot; finally, ϕ_0 is an offset used to induce turning motion of the robot. In [11], the robotic lateral undulatory gait (1) was later generalized to

$$\phi_{i,\text{ref}}(t) = \alpha g(i) \sin\big(\omega t - (i-1)\delta\big) + \phi_0(t), \tag{2}$$

where $g(i)$ is a scaling function that is used to vary the amplitude along the snake robot body. This generalized gait includes, but is not limited to, lateral undulation for land-based snake robots, eel-like motion for underwater snake robots, and other sinusoidal forms of motion, like the gaits proposed in [23].

Different methods have been used to control bio-inspired snake and eel-like robots according to the gaits (1) and (2). The control strategies fall into two general classes: open-loop and closed-loop strategies.

2.2. Open-loop strategies

Central pattern generators   A popular method for the locomotion of snake robots is inspired by neural circuits found in animals, termed central pattern generators (CPG). CPGs are biological neural networks that generate rhythmic patterned outputs without sensory feedback. They have been employed in land-based snake robot locomotion [24] and underwater snake robot locomotion [25] (see [26] for a detailed review of CPGs in animal and robot locomotion). However, CPG-based control is essentially an open-loop method capable of only local motion planning. As mentioned in [26], there does not exist a theoretical foundation for analyzing the stability of the complete CPG-robot system.

2.3. Closed-loop strategies

Time reference trajectory tracking In this family of control strategies, dynamic controllers such as PID and PD controllers are employed to make the joints of a snake robot track time-dependent reference signals. These reference signals are typically generated by a motion planning algorithm [22, 6, 10]. This strategy is a popular choice for the implementation of physical robots [22, 27, 9, 28]. In particular, the joint offset $\phi_0(t)$ in the generalized lateral undulatory gait (2) is designed in a way that makes the snake robot follow a certain path. Introducing the time-varying reference signal $\phi_{i,\mathrm{ref}}(t), i = 1, \cdots, N - 1$, however, complicates the motion planning and mathematical analysis, and decreases the robustness of the control system.

Virtual holonomic constraints Another family of closed-loop strategies is the virtual holonomic constraints (VHC) framework which was developed for underactuated mechanical system control [29, 30, 31]. A VHC is a time-independent relation involving the configuration variables of a mechanical system that does not physically exist in the system, but is enforced via feedback. Under the influence of a VHC, the controlled mechanical system behaves as if there was a physical constraint on the configuration variables. Instead of feeding a time-dependent reference signal to the controller, VHCs enforce time-independent relations on the snake robot configuration. These relations encode the gait that propels the robot forward. Control methodologies based on VHCs have been successfully employed in underactuated robotic biped locomotion [17, 18, 32, 33, 34].

In the context of snake robotics, biologically inspired VHCs come from adapting the reference signal for the single joints (2) in the following way [12, 13, 14]:

$$\phi_{i,\mathrm{ref}}(\lambda, \phi_0) = \alpha g(i) \sin\big(\lambda + (i-1)\delta\big) + \phi_0, \tag{3}$$

where λ and ϕ_0 are states of controllers

$$\ddot{\lambda} = u_\lambda, \quad \ddot{\phi}_0 = u_{\phi_0}, \tag{4}$$

with the new control inputs u_λ, u_{ϕ_0}. The state-dependent relations $\phi_i = \phi_{i,\mathrm{ref}}(\lambda, \phi_0), i = 1, \cdots, N - 1$ are the proposed VHCs. The state ϕ_0 is used to control the orientation, while the state λ is used to control the speed of the snake robot. The reference joint angles in (3) and thereby the VHCs can be enforced with any type of controller, for example a PD-controller. Note that in (3), the time signal t no longer appears explicitly. Instead, the dynamic gait time evolution is governed by the state of the compensators in (4) and the new inputs u_λ and u_{ϕ_0}. VHCs make the control design amenable to a hierarchical synthesis, where the biological gaits are enforced at the lowest level of hierarchy and path planning is done for a point-mass abstraction of the snake robot at the highest level of hierarchy [12, 13].
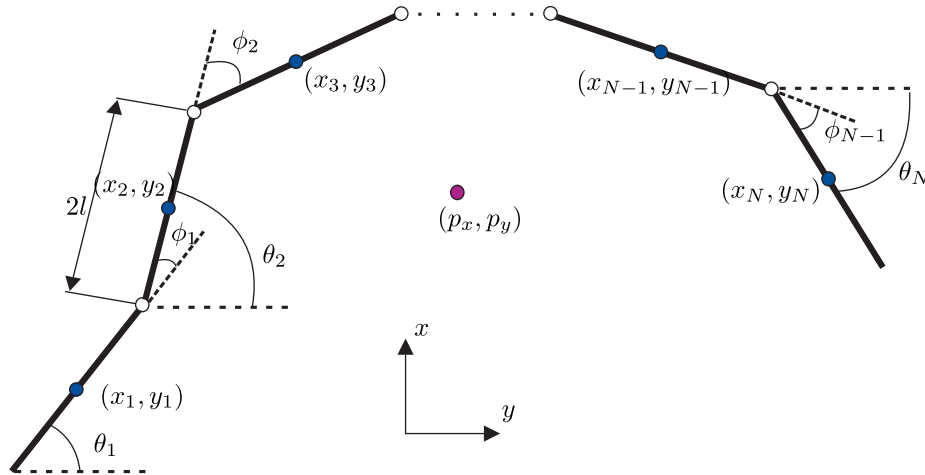
Figure 1. The kinematics of the underwater snake robot.

3. The model of the underwater snake robot

In this section we review the dynamic model of underwater snake robots developed in [10]. The snake robot is assumed to be neutrally buoyant and to move in a fully submerged plane. The kinematic structure of the snake robot is depicted in Figure 1. As it can be seen from the figure, the robot consists of N rigid links connected serially by $N-1$ revolute joints. The links have the length $2l$, mass m, and moment of inertia J. The orientation of each individual link i is denoted by θ_i. The joint angles are denoted by $\phi_i = \theta_i - \theta_{i+1}$. We define the vectors of link and joint angles to be $\boldsymbol{\theta} \in \mathbb{R}^N$ and $\boldsymbol{\phi} \in \mathbb{R}^{N-1}$, respectively. The position of the center of mass (CM) of the robot is $\mathbf{p} = [p_x, p_y]^T$, and its velocity in the inertial frame is $\dot{\mathbf{p}}$. In addition, the velocity components of the CM that are tangential and normal with respect to the head orientation of the robot are defined as $[v_t, v_n]^T = \mathbf{R}_{\theta_N}^T \dot{\mathbf{p}}$ with the rotation matrix $\mathbf{R}_{\theta_N}$ defined in [13]. With regards to the fluid effects on the robot, the following assumptions are made:

Assumption 1 *The robot is exposed to an unknown constant irrotational ocean current* $\mathbf{v}_c = [V_x, V_y]^T$.

Assumption 2 *Added mass effects can be disregarded, i.e. the fluid forces are described with a model for linear drag and nonlinear drag.*

Remark 1 *Disregarding added mass effects is a common assumption for slowly moving underwater vehicles [35]. In particular, it is an assumption that is frequently made for bio-inspired robots [3].*

Table 1 summarizes the parameters of the robot, and provides the numerical values of the parameters for the physical snake robot Mamba [28]. In accordance with Assumption 2, the dynamic equations developed in [10] simplify as follows:

$$\mathbf{M}_\theta \ddot{\boldsymbol{\theta}} + \mathbf{W}_\theta \dot{\boldsymbol{\theta}}^2 + \boldsymbol{\Lambda}_2 \dot{\boldsymbol{\theta}} + \boldsymbol{\Lambda}_3 |\dot{\boldsymbol{\theta}}| \dot{\boldsymbol{\theta}} - l\mathbf{S}\mathbf{C}_\theta^T \mathbf{f}_D = \mathbf{D}^T \mathbf{u},$$
$$\ddot{\mathbf{p}} = \frac{1}{Nm} \mathbf{E}^T \mathbf{f}_D,$$

(5)

Table 1. The parameters of the snake robot.

Symbol	Description	Numerical values
N	Number of links	10
$2l$	Length of a link	0.18 m
m	Mass of a link	1.56 kg
J	Moment of inertia of a link	0.0042 kgm^2
c_t	Tangential drag parameter	4.45
c_n	Normal drag parameter	17.3
λ_2	Linear rotational damping parameter	0.0120
λ_3	Nonlinear rotational damping parameter	8.1160e-04
V_x	x-component of the ocean current	-0.01 m/s
V_y	y-component of the ocean current	0.01 m/s

where $\mathbf{M}_\theta = J\mathbf{I}_N + ml^2\mathbf{S}_\theta\mathbf{V}\mathbf{S}_\theta + ml^2\mathbf{C}_\theta\mathbf{V}\mathbf{C}_\theta$ is the rotational inertia matrix, $\mathbf{W}_\theta = ml^2\mathbf{S}_\theta\mathbf{V}\mathbf{C}_\theta - ml^2\mathbf{C}_\theta\mathbf{V}\mathbf{S}_\theta$ is a generalized Coriolis and centripetal force matrix, $\mathbf{\Lambda}_2 = \lambda_2\mathbf{I}_N$ and $\mathbf{\Lambda}_3 = \lambda_3\mathbf{I}_N$ are the linear and nonlinear fluid damping matrices, and $\mathbf{f}_D = \mathbf{f}_D^{I} + \mathbf{f}_D^{II}$ are the linear and nonlinear fluid drag forces

$$
\mathbf{f}_D^{I} = l\mathbf{Q}_\theta\mathbf{S}\mathbf{C}_\theta\dot{\boldsymbol{\theta}} + \mathbf{Q}_\theta\mathbf{E}\mathbf{v}_r,
$$

$$
\mathbf{f}_D^{II} = - \begin{bmatrix} c_t\mathbf{C}_\theta & -c_n\mathbf{S}_\theta \\ c_t\mathbf{S}_\theta & c_n\mathbf{C}_\theta \end{bmatrix} \mathrm{sgn}\left(\begin{bmatrix} \mathbf{V}_{r,x} \\ \mathbf{V}_{r,y} \end{bmatrix} \right) \begin{bmatrix} \mathbf{V}_{r,x} \\ \mathbf{V}_{r,y} \end{bmatrix}^2, \tag{6}
$$

respectively. In the above equations, $\mathbf{v}_r = \dot{\mathbf{p}} - \mathbf{v}_c$ is the relative velocity of the CM in the inertial frame and $\mathbf{V}_{r,x}$, $\mathbf{V}_{r,y}$ are vectors whose components are the relative velocities of the single links i with respect to the ocean current $\mathbf{v}_c$. The definitions of the remaining vectors and matrices in (5) and (6) are found in Appendix A. At lower velocities, the linear drag forces $\mathbf{f}_D^{I}$ are the dominant forces, while the nonlinear drag forces $\mathbf{f}_D^{II}$ become dominant at higher velocities.

Remark 2 *The model of the underwater snake robot simplifies to the model of a land-based snake robot when the non-linear fluid drag forces are neglected, and the drag parameters in the linear drag forces are replaced by viscous friction coefficients. In this case, the ocean current will be zero, and the relative velocities will be equal to the absolute velocities.*

4. Maneuvering control for underwater snake robots in the presence of ocean current

In this section the control system for maneuvering control of underwater snake robots in the presence of ocean current is presented. The control framework for land-based robots from [13], which the approach for underwater robots is based on, is reviewed in the first part. In the second part we propose the new control system for maneuvering control of underwater snake robots.
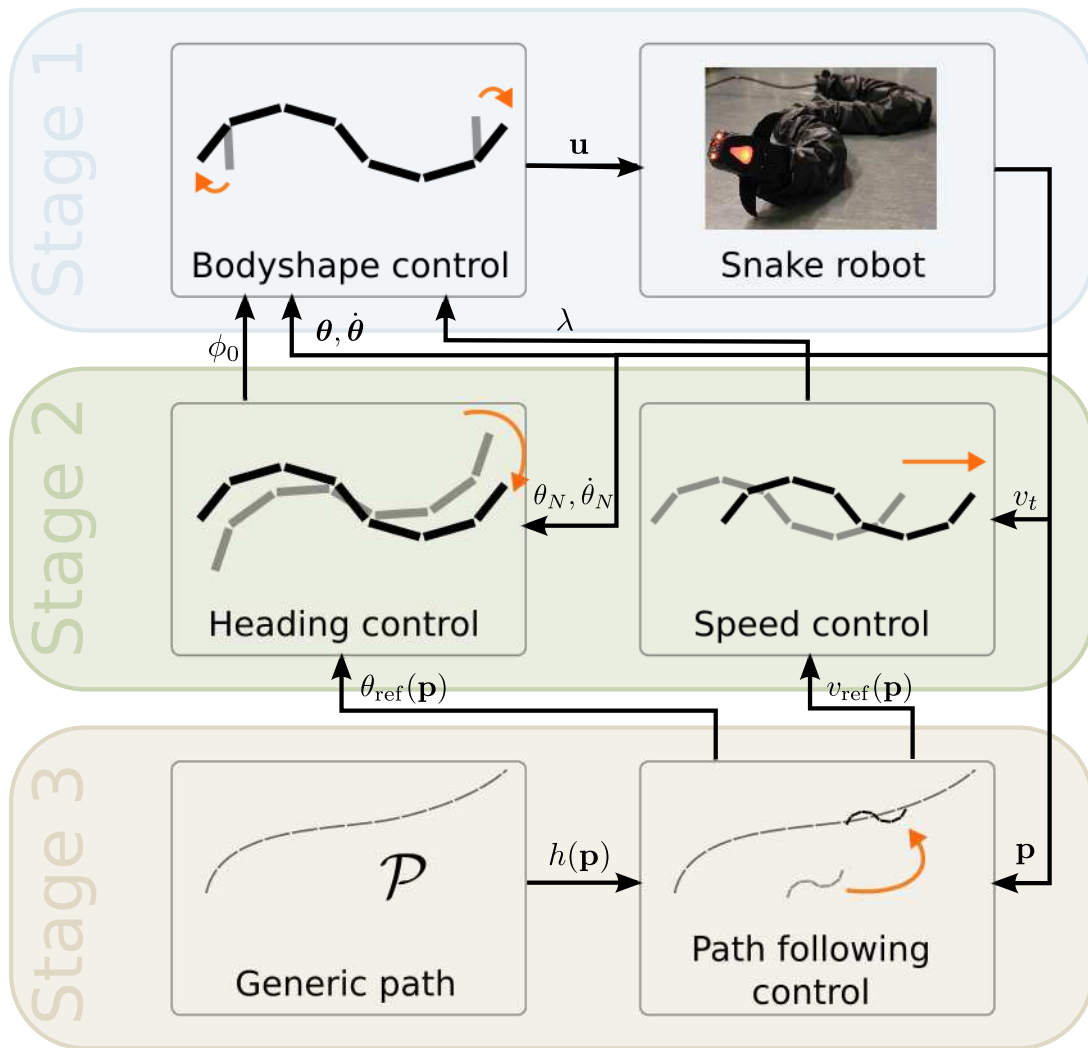
Figure 2. The structure of the control system in [13].

In [13], a formal stability proof for VHC-based maneuvering control of land-based snake robots was presented. In this article, the proof details that are analogous to those developed for land-based snake robots in [13] are not presented because of the strong similarity. Instead, a sketch of the proof is provided focusing on the new components of the VHC-based control strategy that are specifically designed for underwater snake robots.

4.1. The VHC framework for snake robot control

The aim of the following paragraphs is to convey a general understanding of the control system that was developed for land-based snake robots in [13]. The structure of the system will therefore be explained in an intuitive and non-technical way. For the complete equations, detailed derivations, and mathematical proofs the reader is referred to [13].

The control design approach in [13] is hierarchical in the sense that the design has three main stages corresponding to three prioritized control specifications. Figure 2 depicts the hierarchical structure of the control strategy.

- **Stage 1: Body shape control.** This stage represents the inner control loop and has highest priority. The control torque **u** of the snake robot is used to stabilize the VHCs (3). In [13], this objective is achieved by means of an input-output feedback-linearizing controller that zeros the outputs $e_i = \phi_i - \phi_{i,\text{ref}}(\lambda, \phi_0), i = 1, \cdots, N-1$. Once the VHCs are enforced, the configuration variables of the snake robot satisfy the relations in (3), and the states λ and ϕ_0 can be interpreted as new inputs to be assigned in the second stage of the control design.

- **Stage 2: Velocity Control.** At this stage, the middle layer of the control hierarchy, the dynamic compensator u_{ϕ_0} is designed to make the head angle θ_N of the snake robot converge arbitrarily close to a state dependent reference heading $\theta_{\text{ref}}(\mathbf{p})$ to be assigned. Similarly, the design of u_λ ensures that the forward speed v_t converges arbitrarily close to a reference tangential speed $v_{\text{ref}}(\mathbf{p})$. The references $\theta_{\text{ref}}(\mathbf{p})$ and $v_{\text{ref}}(\mathbf{p})$ are assigned by the third control stage.

- **Stage 3: Path Following Control.** At the last stage of the control hierarchy, the reference signals for Stage 2, $\theta_{\text{ref}}(\mathbf{p})$ and $v_{\text{ref}}(\mathbf{p})$ are designed so as to make the robot approach the path and follow it with a desired speed. For underwater applications, where the snake robot is exposed to ocean currents, the third stage of the control hierarchy will have to be modified compared to [13] in order to take into account the perturbing effect of ocean currents.

Remark 3 *When a variable is controlled such that it converges to an arbitrarily small neighbourhood of its desired value through a suitable choice of control parameters, we say that the variable is* practically stabilized. *A formal definition can be found for instance in [36]. This terminology indicates that the variable does not converge exactly to the desired value, but can be made to converge close enough such that it is "practically" stable.*

4.2. Control system design

We now describe an enhancement of the control methodology outlined in Section 4.1 to handle underwater snake robots and compensate for the presence of constant unknown ocean currents.

4.2.1. Stage 1: Body shape controller In order to enforce the biologically inspired VHCs (3) on the snake robot configuration, the PD feedback law

$$\mathbf{u} = -k_p\big(\mathbf{D}\boldsymbol{\theta} - \boldsymbol{\Phi}(\lambda) - \mathbf{b}\phi_0\big) - k_d\big(\mathbf{D}\dot{\boldsymbol{\theta}} - \boldsymbol{\Phi}'(\lambda)\dot{\lambda} - \mathbf{b}\dot{\phi}_0\big) + k_{u_\lambda}\boldsymbol{\Phi}'(\lambda)u_\lambda + k_{u_\phi}\mathbf{b}u_{\phi_0} \quad (7)$$

can be used, where the control gains k_p, k_d, k_{u_λ} and k_{u_ϕ} are positive design constants, $\mathbf{b} = [1, \cdots, 1]^T \in \mathbb{R}^{N-1}$, and the vector $\boldsymbol{\Phi}(\lambda)$ contains the sinusoidal part of (3). We can

see that the new control inputs u_λ, u_{ϕ_0} appear in (7). The control law (7) practically stabilizes the VHCs $\phi_i = \phi_{i,\text{ref}}(\lambda, \phi_0), i = 1, \cdots, N-1$ and is easy to implement. The feedback law (7) requires the knowledge of the link angles $\boldsymbol{\theta}$ and their derivatives $\dot{\boldsymbol{\theta}}$. In practice it suffices to equip the robot with a sensor in each joint measuring the joint angles ϕ_i and velocities $\dot{\phi}_i$, and a positioning system that measures the absolute angle θ_i and velocity $\dot{\theta}_i$ of one of the links. The single link angles are then obtained by the kinematics.

Remark 4 *In [13], the offset ϕ_0 has only been used to control the orientation of the last link, i. e. $\mathbf{b} = [0, \cdots, 0, 1]^T \in \mathbb{R}^{N-1}$. In this article, however, we add ϕ_0 to all of the joint angles, i. e. $\mathbf{b} = [1, \cdots, 1]^T \in \mathbb{R}^{N-1}$ in order to have faster turning motion and thus faster convergence towards the path.*

4.2.2. Stage 2: Velocity controller In the second stage of the control system, feedback laws for the new control inputs u_{ϕ_0}, u_λ will be designed. The objective is to achieve a velocity vector whose direction is characterized by a desired angle $\theta_{\text{ref}}(\cdot)$ to be assigned later. While in [13] the *absolute* velocity was considered, due to the ocean currents we here consider the *relative* velocity of the swimming snake robot with respect to the ocean current

$$\mathbf{v}_{\text{rel}} = \begin{bmatrix} v_{t,\text{rel}} \\ v_{n,\text{rel}} \end{bmatrix} = \mathbf{R}_{\theta_N}^T \big(\dot{\mathbf{p}} - \mathbf{v}_c\big). \tag{8}$$

Geometrically, $v_{t,\text{rel}}$ is the component of the relative velocity vector parallel to the head link of the robot, while $v_{n,\text{rel}}$ is the component of the relative velocity orthogonal to the head link. The control objective is to make the relative velocity in (8) follow the desired velocity when the head link orientation θ_N is stabilized to a given reference $\theta_{\text{ref}}(\cdot)$ with the control input u_{ϕ_0}, while the forward speed $v_{t,\text{rel}}$ is stabilized to a desired value $v_{\text{ref}}(\cdot)$ and the normal speed $v_{n,\text{rel}}$ is close to zero. The references $\theta_{\text{ref}}(\cdot)$ and $v_{\text{ref}}(\cdot)$ are designed in Section 4.2.3 below. In light of (8), the practical stabilization of θ_N to $\theta_{\text{ref}}(\cdot)$, v_t to $v_{\text{ref}}(\cdot)$, and v_n to 0 is equivalent to the practical stabilization of $\dot{\mathbf{p}} - \mathbf{v}_c$ to $\mathbf{R}_{\theta_N}[v_{\text{ref}}(\cdot), 0]^T$. This fact is exploited in Section 4.2.3.

The heading controller The first task of the velocity controller is to use the control input u_{ϕ_0} to make the angle of the head link θ_N converge to a neighbourhood of the reference heading $\theta_{\text{ref}}(\cdot)$. In other words, the heading error $\widetilde{\theta}_N = \theta_N - \theta_{\text{ref}}(\cdot)$ is practically stabilized to zero. It is shown in [13] that this can be achieved by the control law

$$u_{\phi_0} = \frac{1}{\epsilon}\big(\dot{\widetilde{\theta}}_N + k_n \widetilde{\theta}_N\big) - k_1 \phi_0 - k_2 \dot{\phi}_0. \tag{9}$$

The controller parameters k_n, k_1, and k_2 are positive constants and ϵ is a small positive parameter that controls how small the error $\widetilde{\theta}_N$ gets asymptotically. For this feedback law, the required measurements are the head link angle θ_N and velocity $\dot{\theta}_N$.

The speed controller The second task of the velocity controller is to use the control input u_λ to make the forward and normal components $v_{t,\mathrm{rel}}$ and $v_{n,\mathrm{rel}}$ defined in (8) converge to the reference speed $v_{\mathrm{ref}}(\cdot)$ and a small neighbourhood of the origin, respectively. The feedback law

$$u_\lambda = -k_z\big(\dot\lambda + k_\lambda \Delta v_{t,\mathrm{rel}}\big) \tag{10}$$

achieves this control objective. In (10), the control gains k_z and k_λ are positive constants, and $\Delta v_{t,\mathrm{rel}} = v_{t,\mathrm{rel}} - v_{\mathrm{ref}}(\cdot)$ is the velocity error. In order to implement this controller on a physical robot, relative velocity measurements are required. These can for instance be extracted from measurements provided by pressure sensors [4] or from Doppler velocity logs (DVLs) without bottom lock [37].

Remark 5 *The speed controller presented in this article is a simplified version of the speed controller in [13]. It can be shown that the feedback law (10) stabilizes the relative velocities to their desired values if the control gain k_λ is chosen sufficiently large.*

4.2.3. Stage 3: Path following controller and ocean current observer For underwater snake robots, the third control stage needs to be adapted compared to [13] in order to compensate for the perturbing effect of ocean currents. The control stage therefore consists of two parts, the modified path following controller from [13], and an ocean current observer.

We start with reviewing the path following controller from [13], which corresponds to the case when the ocean current is negligible. The third control stage objective is to practically stabilize the robot to a planar curve described implicitly as $\mathcal{P} = \{\mathbf{p} \in \mathbb{R}^2 : h(\mathbf{p}) = 0\}$, while controlling the speed along $\mathcal{P}$. The path following control design in [13] relies on the observation that once the speed and heading controllers have converged, in the absence of ocean current one has $\dot{\mathbf{p}} \approx \mathbf{R}_{\theta_{\mathrm{ref}}}[v_{\mathrm{ref}}(\mathbf{p}), 0]^T$. Letting $\widetilde{\boldsymbol{\mu}}(\mathbf{p}) := \mathbf{R}_{\theta_{\mathrm{ref}}}[v_{\mathrm{ref}}(\mathbf{p}), 0]^T$, $\widetilde{\boldsymbol{\mu}}(\mathbf{p})$ is viewed as a control input and designed to make $\mathbf{p}$ converge to the path $\mathcal{P}$. The resulting control law is

$$\widetilde{\boldsymbol{\mu}}(\mathbf{p}) = \underbrace{-\frac{dh_p^T}{\|dh_p\|^2}k_{\mathrm{tran}}h(\mathbf{p})}_{\boldsymbol{\mu}^\perp(\mathbf{p})} + \underbrace{\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}dh_p^T\frac{v}{\|dh_p\|}}_{\boldsymbol{\mu}^\|(\mathbf{p})}, \tag{11}$$

where k_{tran} is a positive constant, v is the desired speed along the path, and $dh_p^T = \nabla h(\mathbf{p})$ is the vector normal to the level sets of $h(\cdot)$. The geometrical interpretation of the reference velocity vector in (11) is the following. The reference velocity vector $\widetilde{\boldsymbol{\mu}}(\mathbf{p})$ has two components, namely a normal component $\boldsymbol{\mu}^\perp(\mathbf{p})$ that makes the robot converge to the path, and a parallel component $\boldsymbol{\mu}^\|(\mathbf{p})$ that regulates the robot speed along the path $\mathcal{P}$ (see also Figure 3).

We now enhance the ideas presented above to handle the presence of the unknown ocean current. To begin with, the identity in (8) may be re-written as

$$\dot{\mathbf{p}} = \mathbf{R}_{\theta_N}\begin{bmatrix} v_{t,\mathrm{rel}} \\ v_{n,\mathrm{rel}} \end{bmatrix} + \mathbf{v}_{\mathrm{c}}. \tag{12}$$
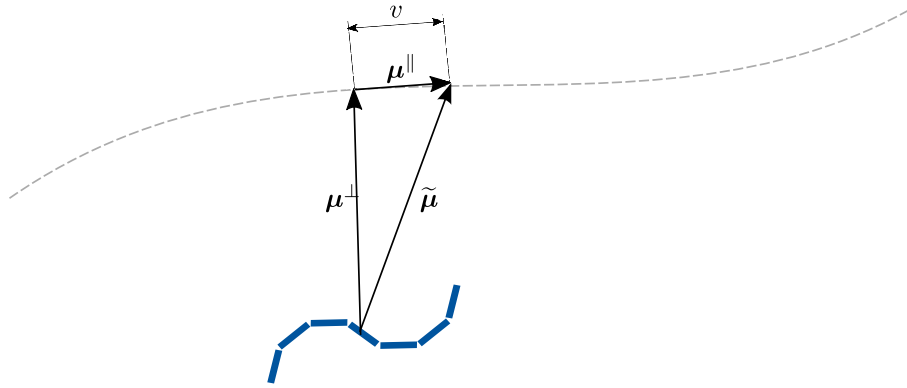
Figure 3. The path following controller from [13]. The velocity reference vector $\widetilde{\boldsymbol{\mu}}$ is composed of the normal component $\boldsymbol{\mu}^{\perp}$, used to steer the robot towards the path, and the parallel component $\boldsymbol{\mu}^{\parallel}$, used to regulate the speed of the robot along the path.

An ocean current observer for marine vehicles based on this equation was first presented in [38], where a Luenberger observer was implemented. It was later used for path following in [39, 40]. In this article, we propose a reduced order observer based on (12), which provides an exponentially stable estimate of the ocean current velocities. Letting $\mathbf{x} = [\mathbf{p}, \mathbf{v}_{\text{c}}]^{T}$, we may re-write (12) as

$$
\begin{aligned}
\dot{\mathbf{x}} &= \begin{bmatrix} \mathbf{0} & \mathbf{I}_2 \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \mathbf{x} + \begin{bmatrix} \mathbf{R}_{\theta_N} \\ \mathbf{0} \end{bmatrix} \mathbf{v}_{\text{rel}}, \\
\mathbf{y} &= \begin{bmatrix} \mathbf{I}_2 & \mathbf{0} \end{bmatrix} \mathbf{x}.
\end{aligned}
\tag{13}
$$

Assuming that $\mathbf{p}$ and $\mathbf{v}_{\text{rel}}$ are available for measurement, the reduced-order observer for $\mathbf{v}_{\text{c}}$ is given by (see e. g. [41])

$$
\dot{\mathbf{z}} = -k_o \mathbf{z} - k_o^2 \mathbf{p} - k_o \mathbf{R}_{\theta_N} \mathbf{v}_{\text{rel}},
\tag{14}
$$

where the gain k_o is a positive constant. The current estimate is then obtained by

$$
\widehat{\mathbf{v}}_{\text{c}} = \mathbf{z} + k_o \mathbf{p}.
\tag{15}
$$

The reduced order observer requires only two states $\mathbf{z} = [z_1, z_2]^{T}$ instead of the four required by a full order ocean current observer. The intuition behind using (12) is that the control system has access to its position while knowing its velocity with respect to the surrounding fluid. According to this relative velocity the robot will "expect" a certain change in position, which it then compares to the position measurements. From the difference in between, the magnitude and direction of the ocean current can be estimated.

We now turn to the design of a path following controller. We rewrite (12) as $\mathbf{p} = \boldsymbol{\mu} + \mathbf{v}_{\text{c}} + \Delta$, where

$$
\boldsymbol{\mu} = \mathbf{R}_{\theta_{\text{ref}}} \begin{bmatrix} v_{\text{ref}} \\ 0 \end{bmatrix},
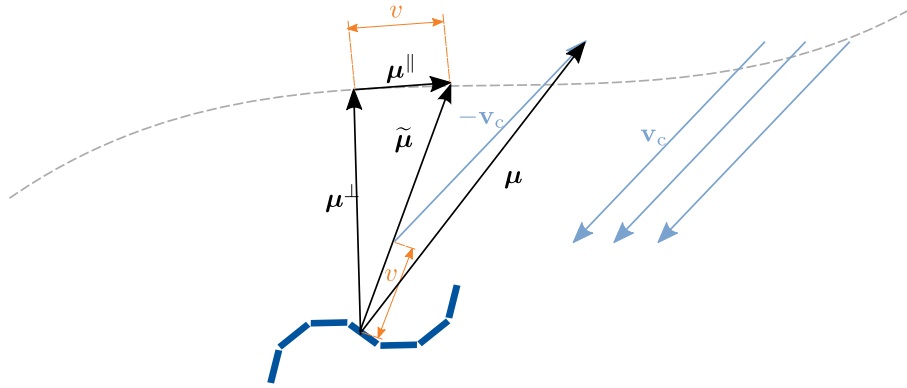\tag{16}
$$

Figure 4. The path following controller with current compensation. In order to avoid a very large reference velocity far away from the path, the original reference $\widetilde{\boldsymbol{\mu}}$ is scaled to have the length v that was originally assigned to the along path component $\boldsymbol{\mu}^{\|}$ (cmp. Figure 3 and the derivation in [13]). The current velocity is subtracted from this new reference such that the robot can make up for it by tracking the new reference velocity vector $\boldsymbol{\mu}$.

and $\Delta = \mathbf{R}_{\theta_N}[v_{t,\mathrm{rel}}, v_{n,\mathrm{rel}}]^T - \mathbf{R}_{\theta_{\mathrm{ref}}}[v_{\mathrm{ref}}, 0]^T$. In light of the control design in Section 4.2.2, the quantity Δ is practically stabilized to zero. We thus view it as a disturbance. We design $\boldsymbol{\mu}$ to make $\mathbf{p}$ practically converge to the path $\mathcal{P}$. The function $\boldsymbol{\mu}$ is an enhancement of the control law $\widetilde{\boldsymbol{\mu}}(\mathbf{p})$ in (11) and is given by

$$\boldsymbol{\mu}(\mathbf{p}, \xi, \mathbf{z}) = v \frac{\boldsymbol{\mu}_0(\mathbf{p}, \xi)}{\|\boldsymbol{\mu}_0(\mathbf{p}, \xi)\|} - \widehat{\mathbf{v}}_{\mathrm{c}}(\mathbf{p}, \mathbf{z}). \tag{17}$$

In (17), $\widehat{\mathbf{v}}_{\mathrm{c}}(\mathbf{p}, \mathbf{z})$ is the estimation of the ocean current and $\boldsymbol{\mu}_0(\mathbf{p}, \xi)$ is an alteration of $\widetilde{\boldsymbol{\mu}}(\mathbf{p})$:

$$\boldsymbol{\mu}_0(\mathbf{p}, \xi) = -\frac{dh_p^T}{\|dh_p\|^2}\left(k_{\mathrm{tran}} h(\mathbf{p}) + k_{\mathrm{int}}\xi\right) + \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} dh_p^T \frac{v}{\|dh_p\|}, \tag{18}$$

$$\dot{\xi} = h(\mathbf{p}).$$

The integrator compensates for the fact that the speed controller only makes the velocity error arbitrarily small instead of forcing it to zero. If it was not for the integral action, this would result in an offset from the path, since the current component that is normal to the path would only be partly compensated. Furthermore, in (17) the length of the original velocity reference is changed to v in order to avoid a very high reference speed far away from the desired path $\mathcal{P}$. Finally, the reference signals for the heading and speed controllers in the second stage of the control hierarchy are obtained from the definition of $\boldsymbol{\mu}(\mathbf{p}, \xi, \mathbf{z})$ in (16) by the following polar conversion:

$$\theta_{\mathrm{ref}}(\mathbf{p}, \xi, \mathbf{z}) = \arctan\left(\frac{\boldsymbol{\mu}_y}{\boldsymbol{\mu}_x}\right),$$

$$v_{\mathrm{ref}}(\mathbf{p}, \xi, \mathbf{z}) = \|\boldsymbol{\mu}(\cdot)\|_2. \tag{19}$$

Note that we now provide a reference for the *relative* velocity instead of the *absolute* velocity. The new path following control system is depicted in Figure 4, and a block diagram can be found in Figure 5.
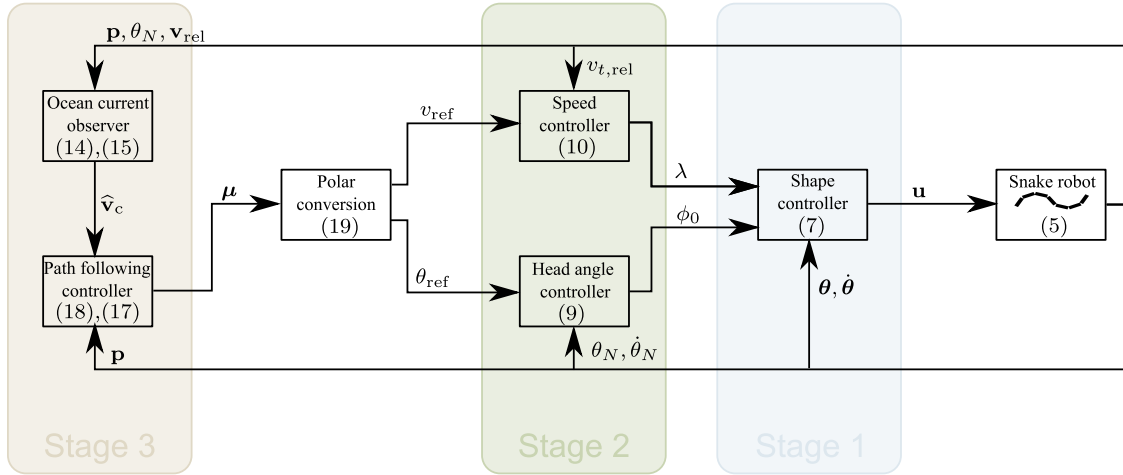
Figure 5. Block diagram of the control system.

5. Simulation study

In order to validate the approach, the control system was implemented in Matlab and extensive simulations were performed. The performance of the VHC-based controller was investigated in various scenarios. This section presents the results and discusses the performance of the robot. In the first part, the simulation parameters are provided. Next, three maneuvering scenarios based on the theoretical model in Section 3 are presented.

5.1. Simulation parameters

The simulations were carried out in Matlab R2014b. The `ode15s` solver was used with a relative and absolute error tolerance of 1e-5. The parameters of the underwater snake robot are presented in Table 1. These values correspond to the parameters of the snake robot Mamba [28] and were experimentally validated in [10]. The gains of the control system were chosen according to Table 2.

5.2. Simulation results

Three different scenarios were tested for the theoretical model in Section 3 and the control system (7),(9),(10),(17). The first one is straight line path following using eel-like motion, the second one is tracking of a circle using lateral undulation, and the third one is sinusoidal path following using a variation of eel-like motion.

5.2.1. Straight line path following with eel-like motion   The straight reference path that the robot should follow was the x-axis of the global coordinate system: $h(\mathbf{p}) = p_y$. The initial position of the robot was set to $p_x = -2$ m, $p_y = 1$ m. The state $\dot{\lambda}$ was initialized with $\dot{\lambda}(0) = \pi/2$ and the states of the current estimator with $\mathbf{z}(0) = [0.025, 0.001]^T$ m/s.

Table 2. The parameters of the control system.

	Value	Description	Stage
α	30°/ 40°/ 50°	Max. amplitude of the joints	
δ	50°	Offset between the joints	
k_p	20		
k_d	5		1
k_{u_λ}	0.01	Controller gains for Stage 1	
k_{u_ϕ}	0.01		
ϵ	0.01		
k_n	10		
k_1	0.1	Controller gains for the heading controller	
k_2	0.1		2
k_z	1		
k_λ	22.5	Controller gains for the speed controller	
k_{tran}	0.25	Transversal gain	
v	0.08	Forward velocity along the path	
k_o	0.05	Observer gain	3
k_{int}	0.002	Integral gain	

All other initial values were set to zero, i.e. the robot was aligned with the desired path. In order to achieve eel-like motion in accordance with [11], the scaling function in the gait (3) was set to $g(i) = \frac{N-i}{N+1}$. The maximum joint amplitude was chosen to be $\alpha = 50°$.

The results of the simulation are depicted in Figure 6. It can be seen that the robot converges to the reference path and continues its motion along it. Note that there are some overshoots until the robot moves on the straight line. This results from the fact that the control system is tuned to follow generic paths, which are in general curved paths. The system is therefore not specialized to converge towards a straight line very fast. Oscillating behaviour like this would be a problem for conventional maritime robots, where behaviour like this should be avoided in order to spare the rudders. For biomimetic systems like underwater snake robots, oscillations are an inherent aspect of locomotion, so introducing additional oscillations by the path following controller does not affect the physical system. Note that the overshoots can still be decreased by adapting the control system to do straight line path following. Both the head link angle and the current estimate converge to the desired values. The body shape adapts closely to the reference signal, as the PD controller makes the error small. The joint offset ϕ_0, the oscillation frequency $\dot{\lambda}$, and the control torques stay within reasonable bounds. Note that there is a trade-off between the accuracy of the speed controller and the control gains, which is in accordance with the practical stability property of the controller. In Figure 6(c) it can be seen that a constant offset remains between the mean of the forward velocity $v_{t,\text{rel}}$ and its reference $v_{\text{ref}} = \|\boldsymbol{\mu}\|$. This offset can be made small by increasing the control gain k_λ, but this will result in larger control torques.
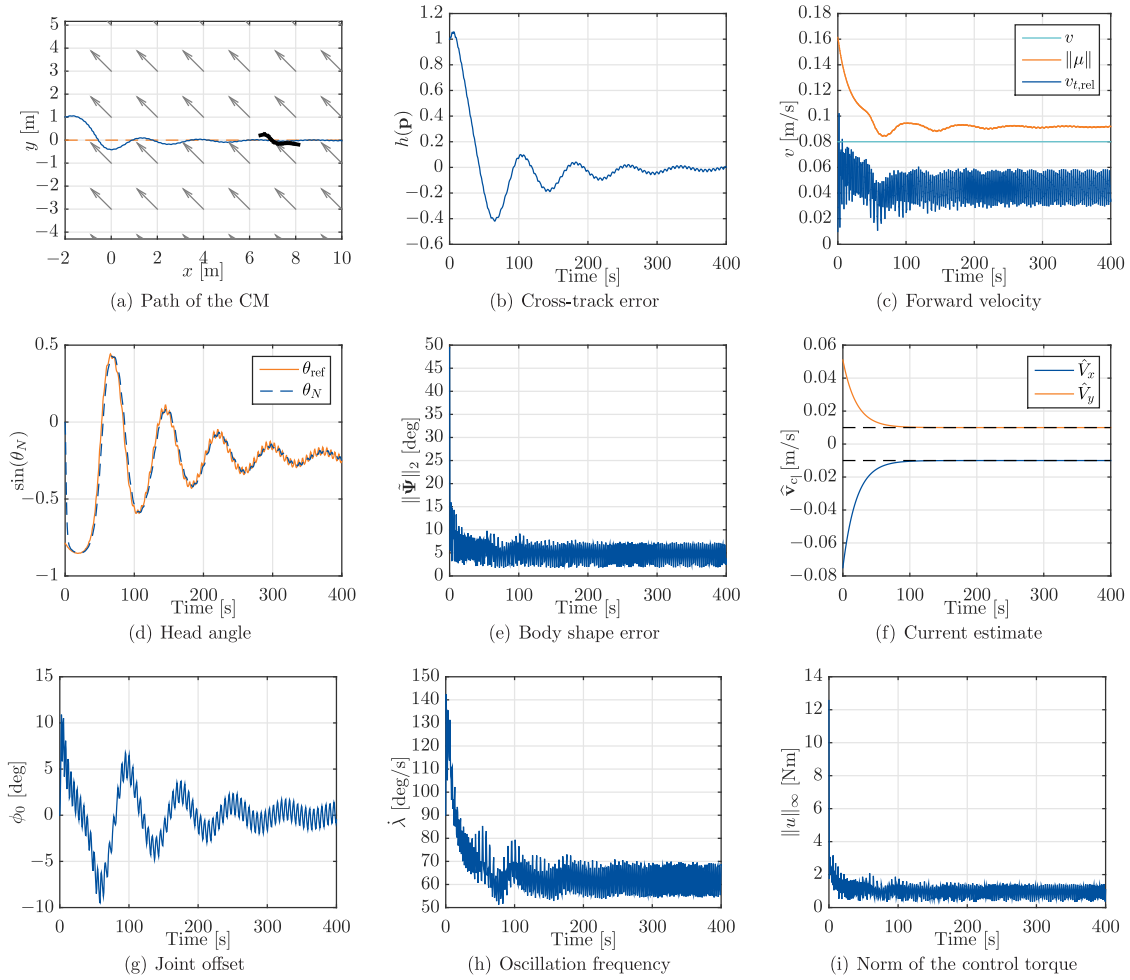
Figure 6. Simulation results for straight line path following with eel-like motion: (a) The path of the underwater snake robot (b) The path following error (c) The forward velocity $v_{t,\mathrm{rel}}$ and the reference velocity v_{ref} (d) The head link angle θ_N and the reference heading θ_{ref} (e) The error in the single joint angles: the PD controller makes the error small (f) The states of the exponentially stable ocean current observer (g) The joint offset ϕ_0 which is used to control the turning rate (h) The frequency of the undulatory gait is governed by $\dot{\lambda}$ (i) The norm of the vector of control torques $\mathbf{u}$

5.2.2. Tracking a circular reference path with lateral undulation The reference path for the underwater snake robot was a circle with radius 3 m and origin in (4,0): $h(\mathbf{p}) = p_y^2 + (p_x - 4)^2 - 9$. The joint amplitude for the lateral undulation was set to $\alpha = 30°$, and the scaling function to $g(i) = 1$. All initial conditions were chosen in accordance with the example in Section 5.2.1, which means that the robot was initially headed towards the path.

The results of the simulation are depicted in Figure 7. We see that the underwater snake robot converges to and tracks the circle. Both the head link angle and the current estimate follow their desired values. The VHCs are approximately enforced, i. e. the error stays small. The joint offset ϕ_0, the oscillation frequency $\dot{\lambda}$, and the control torques remain bounded. Compared to the previous example in Section 5.2.1, the
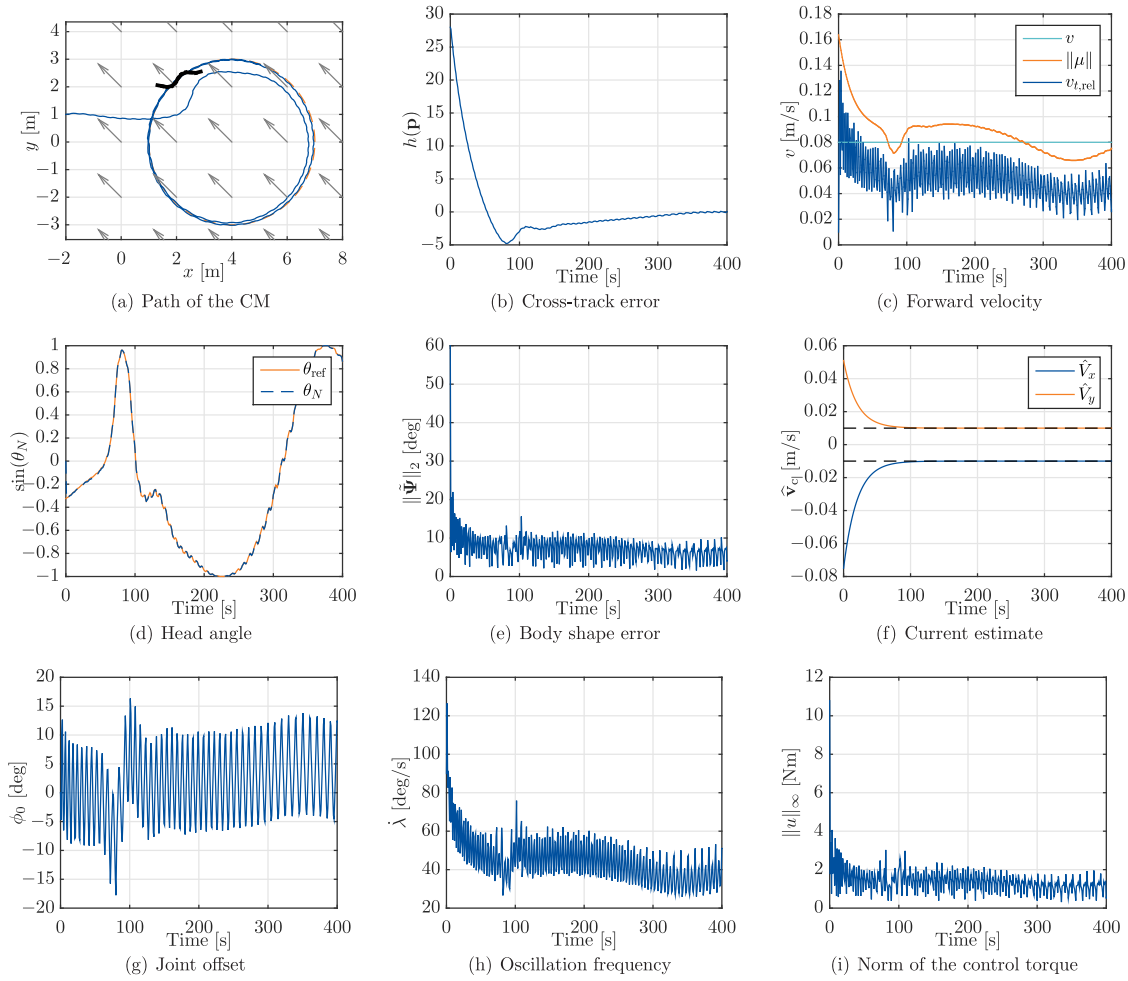
Figure 7. Simulation results for tracking a circular reference path with lateral undulation: (a) The path of the underwater snake robot (b) The path following error (c) The forward velocity $v_{t,\mathrm{rel}}$ and the reference velocity v_{ref} (d) The head link angle θ_N and the reference heading θ_{ref} (e) The error in the single joint angles: the PD controller makes the error small (f) The states of the exponentially stable ocean current observer (g) The joint offset ϕ_0 which is used to control the turning rate (h) The frequency of the undulatory gait is governed by $\dot{\lambda}$ (i) The norm of the vector of control torques $\mathbf{u}$

reference velocity $v_{\mathrm{ref}} = \|\boldsymbol{\mu}\|$ now oscillates about the predefined value v, as shown in Figure 7(c). This is a consequence from the circular path, because the ocean current now changes direction with respect to the path. The robot therefore has to compensate for an oscillating current component. Just like for the straight path, a small offset remains between the forward velocity and its reference, as predicted by the practical stability result.

5.2.3. Following a sinusoidal path with eel-like motion The sinusoidal reference path for the robot was defined as $h(\mathbf{p}) = p_y - 2\sin(2\pi/12 p_x)$. The variation of eel-like motion that was tested in this case study uses the following scaling function defined in [23]: $g(i) = e^{-0.1 l i}$. The maximum joint amplitude was set to $\alpha = 40°$. The initial conditions
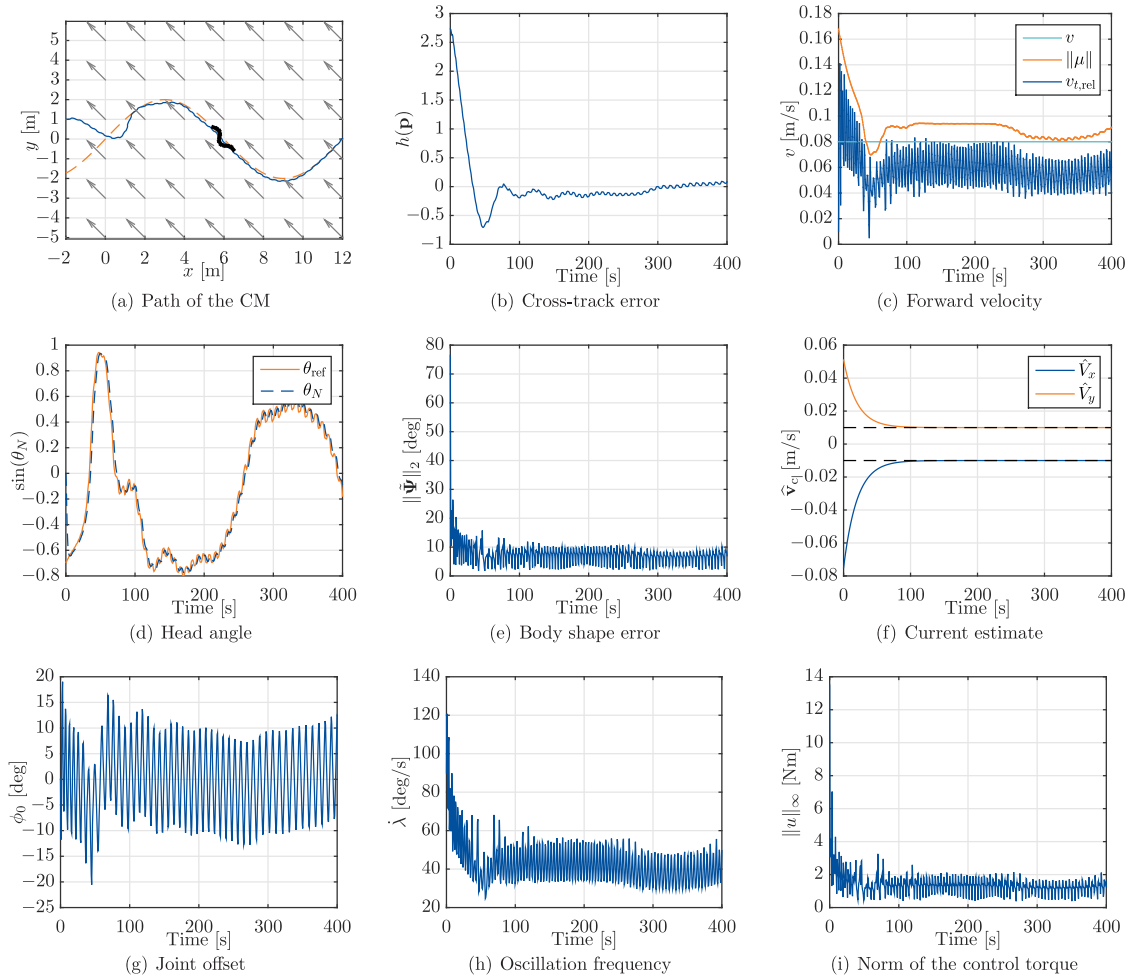
Figure 8. Simulation results for following a sinusoid path with eel-like motion: (a) The path of the underwater snake robot (b) The path following error (c) The forward velocity $v_{t,\text{rel}}$ and the reference velocity v_{ref} (d) The head link angle θ_N and the reference heading θ_{ref} (e) The error in the single joint angles: the PD controller makes the error small (f) The states of the exponentially stable ocean current observer (g) The joint offset ϕ_0 which is used to control the turning rate (h) The frequency of the undulatory gait is governed by $\dot{\lambda}$ (i) The norm of the vector of control torques $\mathbf{u}$

were chosen the same as in Section 5.2.1.

The results of the simulation are depicted in Figure 8. The robot converges to the sinusoidal path and moves along the path while the cross track error remains small. Both the head link angle and the current estimate converge to and follow their references. The errors of the joint angles remain bounded. The joint offset ϕ_0, the oscillation frequency $\dot{\lambda}$, and the control torques stay within small bounds. Just like for the other studied cases and according to the theoretical property of practical stability, a constant offset remains between the mean of the forward velocity $v_{t,\text{rel}}$ and the reference $v_{\text{ref}} = \|\boldsymbol{\mu}\|$.

Table 3. Performance analysis of the control system by means of the different criteria.

Case	(i)	(ii)	(iii)	(iv)
1	73.3 s	2.20 cm	1.84 Nm	4.89 cm/s
2	26.4 s	0.94 m^2	2.59 Nm	3.46 cm/s
3	21.0 s	9.71 cm	2.63 Nm	3.41 cm/s

5.3. Comparison of the different cases

The following criteria were used for comparing the single case studies:

 (i) The settling time: the largest time after which the path following error $h(\mathbf{p})$ remains within one third of its initial value.

 (ii) The rms value of the path following error $h(\mathbf{p})$ in steady state.

(iii) The rms value of the 2-norm of the torque $\|\mathbf{u}\|_2$ in steady state.

(iv) The rms value of the velocity error $\Delta v_{t,\mathrm{rel}}$ defined in Section 4.2.2 in steady state.

The system was considered to be in steady state from the instant of time after which $h(\mathbf{p})$ remained within one tenth of its initial value. The performance criteria are evaluated in Table 3. We see that the settling time was lowest for the sinusoidal reference path, and largest for the straight line. This is consistent with the tuning procedure: the system was tuned such that it would perform well on the sinusoidal path, whereas the other path geometries were not included in the tuning process. The rms value of the path following error $h(\mathbf{p})$ has different interpretations for the different path geometries: for the straight line in Case 1, it is equal to the rms value of the distance between the robot and the path. Compared to the body length of the robot $N2l = 1.8$ m, the distance is very small. For the other cases, the rms value of the path following error cannot be interpreted directly as the distance. The rms value of the 2-norm of the torque $\|\mathbf{u}\|_2$ is related to the energy consumption of the system. The lowest value is achieved for Case 1, which is consistent with the observation in [11] that eel-like motion is the most energy efficient. Finally, the performance criterion related to the forward velocity shows that the speed controller performed slightly better for the curved reference paths than for the straight line.

6. Experimental study

In this section we present experimental results for the control system proposed in Section 4. At first, the heading and speed controllers presented in (9) and (10) were implemented and tested separately. Subsequently, the maneuvering controller displayed in Figure 5 was tested for straight line path following first without, then with a known ocean current.

6.1. The set-up of the experiments

The experiments were performed in the North Sea Centre Flume Tank [42] operated by SINTEF Fisheries and Aquaculture in Hirtshals, Denmark. The tank can be seen in Figure 9(a). It is 30 m long, 8 m wide, and 6 m deep and is equipped with four propellers in order to generate a circulating flow of up to 1 m/s. On the back side, the tank is furthermore equipped with 9 cameras of the Qualisys underwater motion capture system [43] in order to accurately measure positions within the basin. Figure 9(b) shows a screen shot of the Qualisys QTM software. During the experimental trials the global coordinate frame was rotated by 45° with respect to the tank, such that the generated current, which is aligned with the long side of the tank, had both an x and a y component.

The snake robot Mamba [28] was used to test the proposed control system. Mamba is a modular snake robot developed at NTNU that can be operated both on land and in water. It has $N - 1 = 9$ horizontal joints and is connected to a power source and communication unit with a thin, positively buoyant cable. Its single joints are waterproof down to 5 m, and equipped with a servo motor, a microcontroller card, and various sensors. The joint angles are controlled by a proportional controller that is implemented on the microcontroller card, which communicates over a CAN bus. This internal joint position control replaced (7) in the practical implementation of the maneuvering control system. More details on the physical robot can be found in [28]. The robot is depicted in Figure 9(c). The reflective markers for the Qualisys motion capture system are attached to the head link of the robot, and it is equipped with a synthetic skin for additional waterproofing. More information about the skin is provided in [10]. The amount of air inside the skin can be varied in order to change the buoyancy of the robot. For the two-dimensional maneuvering controller presented in this article, the buoyancy was slightly positive, so that the robot would stay close the surface and thus not require depth control.

The robot was controlled from a laptop that runs LabVIEW 2013. The angle and position measurements of the reflective markers were obtained from a second laptop, on which both Qualisys Track Manager (QTM) and LabVIEW 2013 are installed. The data was sent to the laptop that controls the robot in LabVIEW 2013 via UDP in real-time at a sampling frequency of 10 Hz. The orientation θ_N of the head link was obtained directly from the Qualisys system, while the CM position of the robot was calculated from the QTM data and the joint angles from the internal sensors of the robot by using the kinematics analogously to the procedure in [10]. The control system (9),(10),(17),(18) was implemented in LabVIEW. Details about the implementation and the choice of control parameters will be explained in the following for the respective experimental tests.
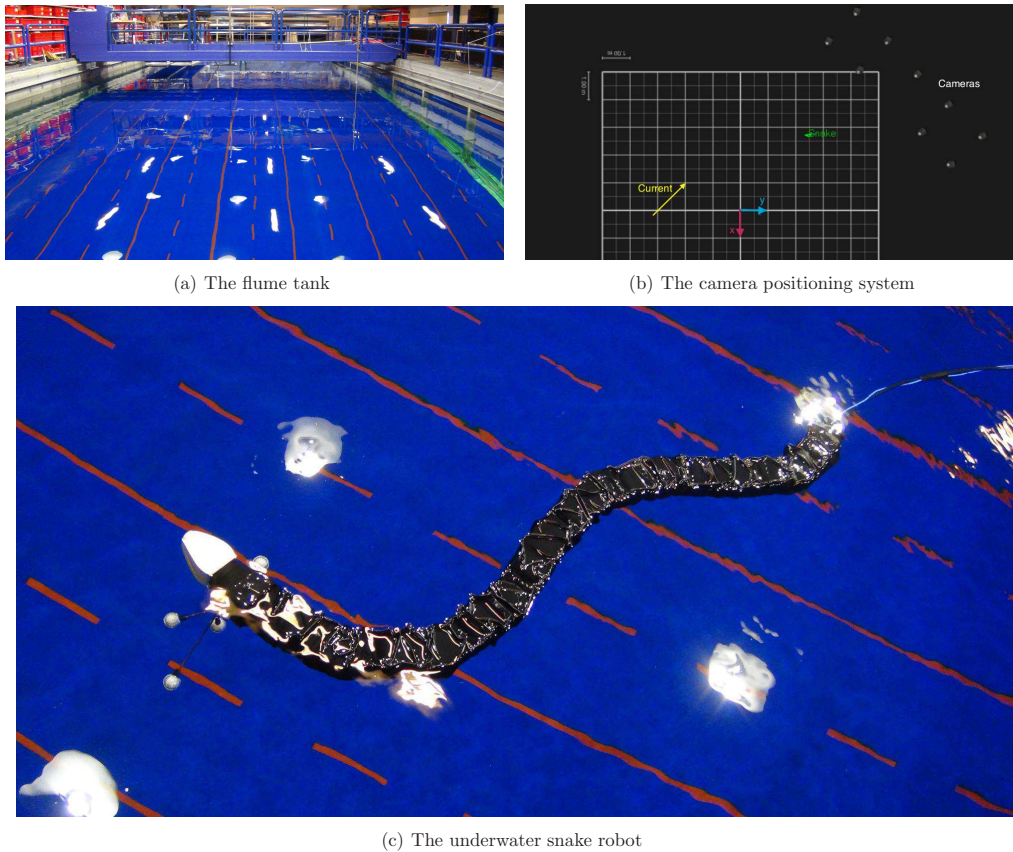
(a) The flume tank

(b) The camera positioning system

(c) The underwater snake robot

Figure 9. The experimental set-up: (a) The North Sea Centre Flume Tank operated by SINTEF Fisheries and Aquaculture in Hirtshals, Denmark. (b) A screenshot of QTM: the coordinate frame has been rotated with respect to the basin and the cameras. The arrows indicate the flow direction. (c) The snake robot Mamba with an attachment for the reflective markers on the head link.

6.2. Heading controller

As a first step, the heading controller (9) was implemented in LabVIEW and tested with the robot. The required derivative $\dot{\theta}_N$ was obtained by numerically differentiating the head angle θ_N in LabVIEW. In order to test only the heading controller, a constant body oscillation was imposed on the robot by removing the dynamic compensator u_λ, and thereby the speed control, and instead imposing a constant $\dot{\lambda} = \omega$. The robot was then controlled to perform lateral undulation according to (3), with the function $g(i) = 1$ and the joint offset ϕ_0 as an input to induce turning motion. The propellers of the tank were not active, so there was no current effect. In addition, the control input for the heading, ϕ_o, was saturated at $\phi_{0,\text{max}} = \pm 20°$. The system was tuned according to Table 4 and the behaviour of the robot was studied for different reference headings θ_{ref}. The results for two cases can be seen in Figure 10 and Figure 11. In the first presented case, the robot was initially straight and heading towards $\theta_N = -13.2°$. The reference orientation was set to $\theta_{\text{ref}} = 30°$, so the robot had to perform a turn in the positive direction. In the second case, the initial orientation of the straightened robot was $\theta_N = -33.5°$, and

Table 4. The parameters of the heading controller.

	Value	Description
α	30°	Max. amplitude of the joints
ω	90°/s	Constant body oscillation frequency
δ	50°	Offset between the joints
ϵ	100	
k_n	20	Controller gains for the heading controller
k_1	1	
k_2	1	

(a) Heading (b) Joint offset (c) Joint angle ϕ_5
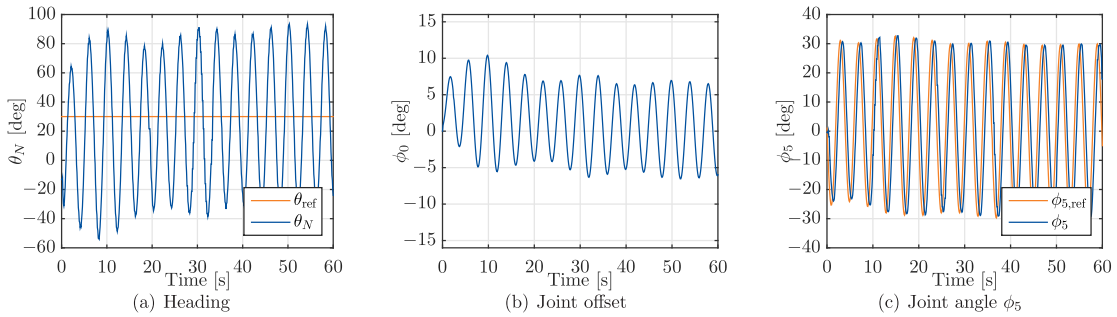
Figure 10. Experimental results for the heading controller with $\theta_{\mathrm{ref}} = 30°$: (a)The head link angle θ_N and the reference heading θ_{ref} (b) The joint offset ϕ_0 which is used to control the turning rate (c) The joint angle ϕ_5 and its reference

(a) Heading (b) Joint offset (c) Joint angle ϕ_5
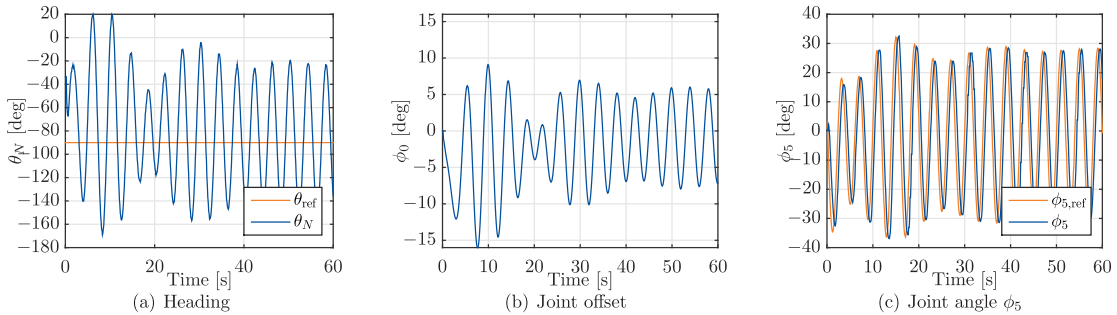
Figure 11. Experimental results for the heading controller with $\theta_{\mathrm{ref}} = -90°$: (a)The head link angle θ_N and the reference heading θ_{ref} (b) The joint offset ϕ_0 which is used to control the turning rate (c) The joint angle ϕ_5 and its reference

the set point was $\theta_{\mathrm{ref}} = -90°$, so a turn in the negative direction was required. Both times, the head link angle θ_N quickly approached a state of oscillating about the given reference. Once θ_N had reached this state, the dynamic compensator ϕ_0, which induces turning motion, oscillated about zero.

Table 5. The parameters of the speed controller.

	Value	Description
α	30°	Max. amplitude of the joints
δ	50°	Offset between the joints
k_z	1	Controller gains for the speed controller
k_λ	0.6	

6.3. Speed controller

In order to implement the speed controller (10), velocity measurements were required. Since the snake robot Mamba is not equipped with velocity sensors, the measurements had to be obtained from the position data of the external motion capture system. In the first step, the speed controller was tested without any current effects, which means that the relative velocity that was used for feedback was the same as the absolute velocity, which was obtained from the displacement of the CM position. The theoretical controller in Section 4 uses the component of the CM velocity vector which is tangential to the head link for feedback. It is not possible to measure this signal directly, and its computation would include terms with the head link angle θ_N, which is highly oscillating due to the sinusoidal motion of the snake robot. For the practical implementation of the controller, the forward velocity was therefore approximated by the total speed

$$v_t \approx u = \sqrt{\dot{p}_x^2 + \dot{p}_y^2}. \tag{20}$$

This choice is also supported by the observation that the component of the velocity vector which is normal to the head link remains close to zero [13]. In order to obtain $\dot{p}_x$ and $\dot{p}_y$, the position of the CM, given by p_x and p_y, was filtered with a LabVIEW built in lowpass filter, with a sampling frequency of 20 Hz and a cutoff frequency of 0.2 Hz and subsequently numerically differentiated with the LabVIEW point-by-point derivative function. In order to initialize the numerically differentiated signals, the processing of the measured data was started 10 seconds prior to the control system.

The speed controller was tested for lateral undulation according to (3) with $g(i) = 1$ and the control gains in Table 5. The heading controller was deactivated by setting $\phi_0 = 0$. The robot was initially kept in a sinusoidal shape until the controller was started after 10 seconds, and the state $\dot{\lambda}$ was initialized with an arbitrary number. Three different reference velocities v_{ref} were given to the controller: 3, 5, and 7 cm/s. The results of these test runs are plotted in Figure 12, Figure 13, and Figure 14, respectively. Consistently with the simulation results presented in Section 5, the mean of the velocity reached a constant offset to the reference velocity. A discussion of these findings will be presented in Section 7. Another observation can be made when looking at Figure 14(b) and Figure 14(c). When the speed controller requires a large oscillation frequency at around 150°/sec (Figure 14(b)), the physical system is no longer able to follow its reference and starts to lag behind, as can be seen from the joint angle
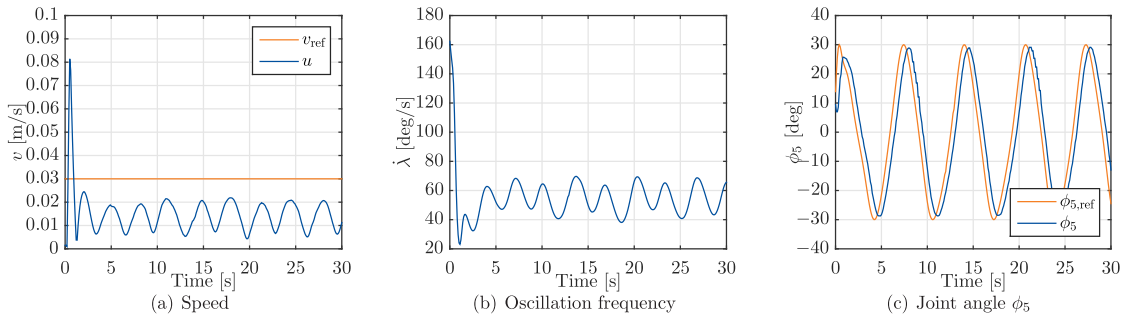
Figure 12. Experimental results for the speed controller with $v_{ref} = 3$ cm/s and $k_\lambda = 0.6$: (a) The velocity u and the reference velocity v_{ref} (b) The frequency of the undulatory gait is governed by $\dot{\lambda}$ (c) The joint angle ϕ_5 and its reference
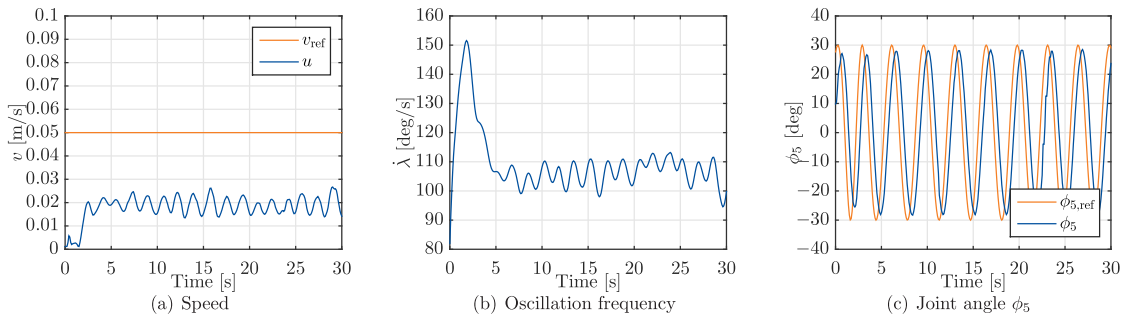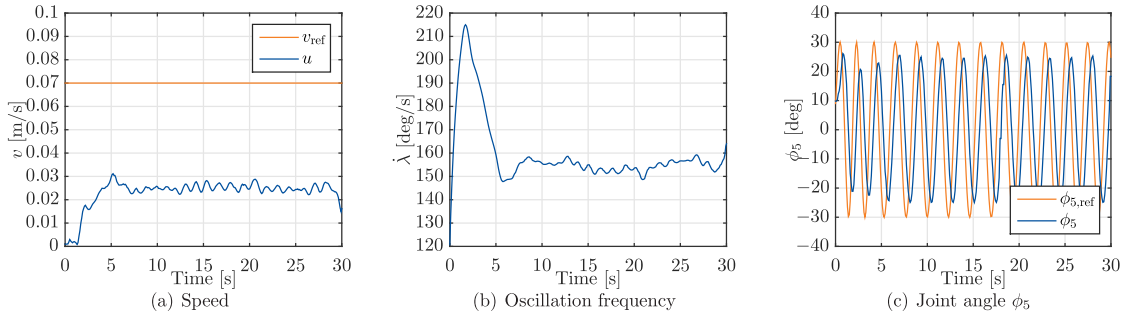

Figure 13. Experimental results for the speed controller with $v_{ref} = 5$ cm/s and $k_\lambda = 0.6$: (a) The velocity u and the reference velocity v_{ref} (b) The frequency of the undulatory gait is governed by $\dot{\lambda}$ (c) The joint angle ϕ_5 and its reference

Figure 14. Experimental results for the speed controller with $v_{ref} = 7$ cm/s and $k_\lambda = 0.6$: (a) The velocity u and the reference velocity v_{ref} (b) The frequency of the undulatory gait is governed by $\dot{\lambda}$ (c) The joint angle ϕ_5 and its reference

plotted in Figure 14(c). The same effect can be observed when increasing the control gain k_λ, as displayed in Figure 15. Here, the test for $v_{ref} = 5$ cm/s was repeated for a higher $k_\lambda = 0.8$. The mean velocity error in steady state was decreased in comparison to the results in Figure 13 with the smaller k_λ, which agrees with the theoretical property of practical stability. However, this was achieved at the cost of a higher body oscillation frequency. It can be seen from Figure 15(c) that the single joints did not quite reach the
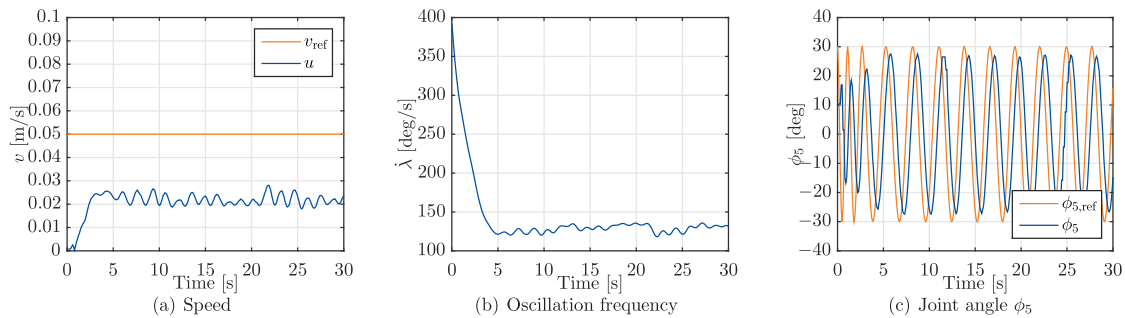
(a) Speed   (b) Oscillation frequency   (c) Joint angle ϕ_5

Figure 15. Experimental results for the speed controller with $v_{\mathrm{ref}} = 5$ cm/s and $k_\lambda = 0.8$: (a) The velocity u and the reference velocity v_{ref} (b) The frequency of the undulatory gait is governed by $\dot{\lambda}$ (c) The joint angle ϕ_5 and its reference

maximal reference amplitude already, an effect that would become even stronger with a further increase of k_λ.

6.4. Straight line path following without current

After establishing the heading and speed controller, the two were combined with the path following controller (17),(18) according to the structure in Figure 5. It was not feasible to obtain accurate measurements of the markers for a full rotation and thus the angular range was limited. This is a result of the camera setup and the geometry of the attachment of the reflective markers used during the experimental trails. The path that the robot was supposed to follow was therefore chosen to be a straight line, since this set-up did not require the robot to make big turns out of the angular range. As a first step, the flow in the tank was set to zero, so there were no current effects. The resulting control problem is very similar to the maneuvering control problem of land-based snake robots, which has been solved theoretically in [13], but not yet been tested experimentally. The control gain k_{int} in the path following controller was set to zero in order to obtain the corresponding path following controller. All other control gains were tuned according to Table 6 in order to ensure smooth convergence for the straight line path. As explained in Section 6.3, the velocity estimation was started 10 seconds prior to the actual control system. During these 10 seconds, the robot was moving slowly with a sinusoidal gait at a very low frequency.

Remark 6 *The controller parameters used in our experimental tests are different than the ones used in simulation. There are several reasons for this discrepancy. First, the theoretical controller tested in simulation includes a torque control loop which has the function of enforcing a suitable VHC. In the experiments, this control loop is replaced by decentralized proportional controllers for each servomotor that were tuned previously. These decentralized controllers introduce lags and imperfections in the VHC stabilization. Second, the controller tested in our simulations was tuned to perform well with curved paths, while the experimental controller was only tested on straight lines. Finally, the simulation model ignores unmodelled experimental effects such as the power cable and*

Table 6. The parameters of the control system.

	Value	Description
α	30° / 40°	Max. amplitude of the joints
δ	50°	Offset between the joints
ϵ	100	
k_n	15 / 20	
k_1	1	Controller gains for the heading controller
k_2	1	
k_z	1	
k_λ	0.6	Controller gains for the speed controller
k_{tran}	0.035 / 0.025	Transversal gain
v	0.05	Desired velocity along the path
k_{int}	0.005	Integral gain

the reflective markers for the motion capture system, that affected the robot during the experimental tests.

The control system was successfully tested for the two gaits lateral undulation and eel-like motion. For the first step without any current effects, we present the results for eel-like motion in the following. The straight reference path that the robot should follow was the x-axis of the global coordinate system: $h(\mathbf{p}) = p_y$. The control gains were tuned according to Table 6. In particular, for the case of eel-like motion, the scaling function in the gait (3) was set to $g(i) = \frac{N-i}{N+1}$, the maximal joint amplitude was chosen as $\alpha = 40°$, the control gain k_n of the heading controller was set to $k_n = 20$, and the transversal gain of the path controller was $k_{\mathrm{trans}} = 0.025$. The results for eel-like motion are presented in Figure 16. It can be seen that the robot converged to the reference path and stayed there, while the cross-track error oscillated about zero, thus the path following control objective is fulfilled. The head link angle approached the desired value and followed it, which means that the heading control objective was also met. Since there was no ocean current to compensate, the reference velocity stayed constant, and the measured speed oscillated about a constant offset, as had already been observed in simulations and Section 6.3. This is in accordance with the practical stability of the speed controller. The joint offset ϕ_0 and the oscillation frequency $\dot{\lambda}$ remained within reasonable values and the joint angles tracked their references.

6.5. Straight line path following with known current

In contrast to the previously presented tests of the speed controller in Section 6.3 where no current effects were influencing the robot, in the presence of current, we had to extract the relative velocity for the feedback in the speed controller from the absolute velocity that can be obtained from the displacement of the CM. Since it is possible to accurately set the current speed in the flume tank, this knowledge could be used to
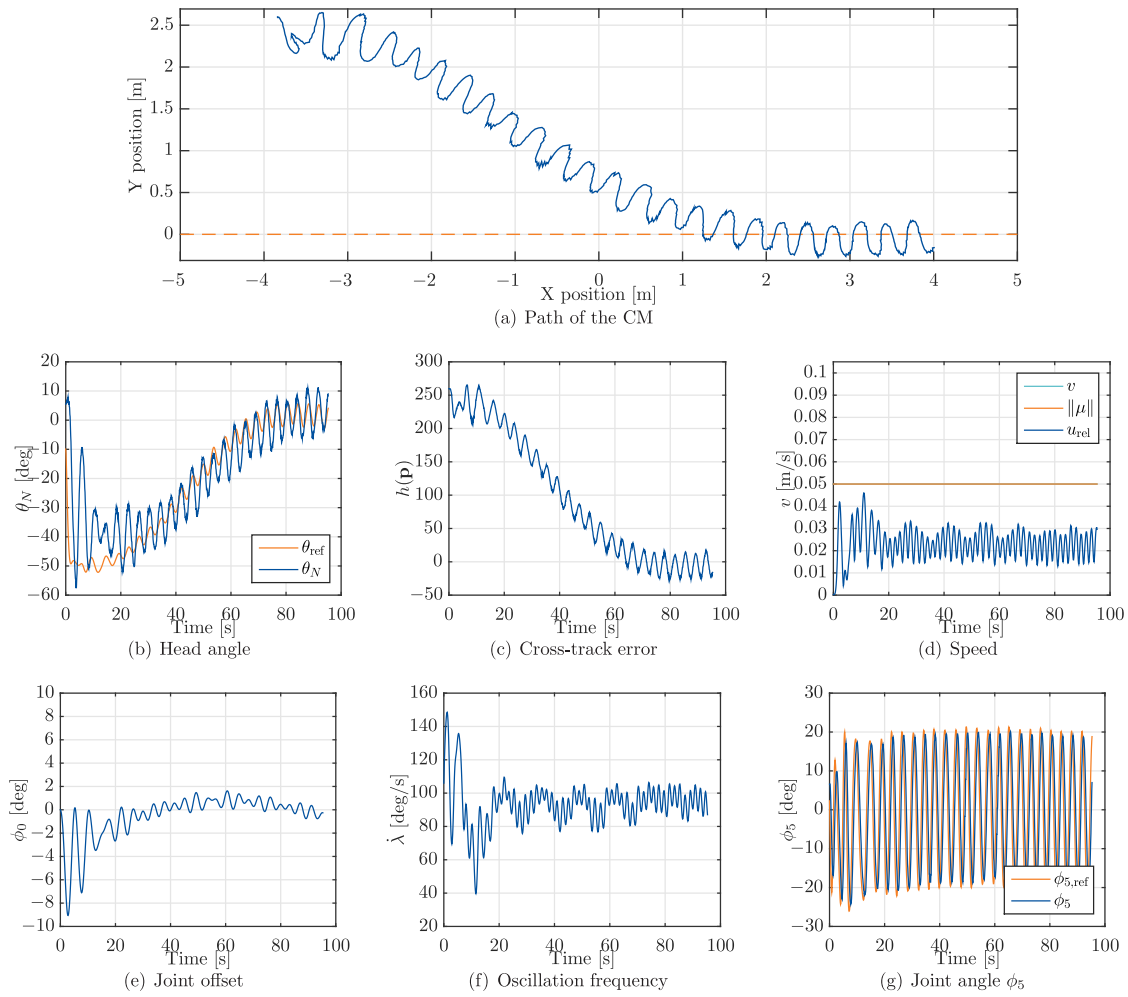
Figure 16. Experimental results for straight line path following with eel-like motion and zero ocean current: (a) The path of the underwater snake robot (b) The head link angle θ_N and the reference heading θ_{ref} (c) The path following error (d) The velocity u and the reference velocity v_{ref} (e) The joint offset ϕ_0 which is used to control the turning rate (f) The frequency of the undulatory gait is governed by $\dot{\lambda}$ (g) The joint angle ϕ_5 and its reference

approximate the relative velocity by

$$v_{t,\text{rel}} \approx u_{\text{rel}} = \sqrt{(\dot{p}_x - V_x)^2 + (\dot{p}_y - V_y)^2}, \tag{21}$$

analogously to the absolute velocity in Section 6.3. The current V_x and V_y were obtained from the set flow speed V by $V_x = -V \cos(45°)$ and $V_y = V \sin(45°)$, according to the rotation of the coordinate frame in Figure 9(b).

Remark 7 *The test platform that was used for this experimental study, the snake robot Mamba, is not equipped with any internal sensors to measure position or velocity. In a lab environment, these data can be obtained with the external motion capture system and the knowledge of the flow speed. For an industrial application within a*

subsea installation, the required measurements can be obtained from commercial systems like underwater acoustic positioning systems for position measurements [44] and DVLs without bottom lock for relative velocity measurements [37]. The components of the ocean current entering the control system in (18) are then extracted from the position and relative velocity measurements by the observer (14),(15). Implementing the observer (14),(15) in the experimental set-up for this study would have resulted in a loop, where position measurements and the knowledge of the current would first be used to approximate the relative velocity, which then subsequently would be combined with the position measurements one more time to obtain a current estimate. Therefore the knowledge of the exact current speed was used in this experimental study for the control law (18), and the observer (14),(15) was omitted.

In addition, for the practical implementation of the control system, anti-windup was added to the integral action in the path following control stage by replacing the integrator update law in (18) by

$$\dot{\xi} = \frac{\frac{h(\mathbf{p})}{k_{\mathrm{tran}}}}{(h(\mathbf{p}) + k_{\mathrm{int}}\xi)^2 + k_{\mathrm{tran}}^{-2}}. \tag{22}$$

The approach was tested for the two gaits lateral undulation and eel-like motion.

Lateral undulation The flow speed in the flume tank was set to $V = 0.05$ m/s. For lateral undulation, the joint amplitude was set to $\alpha = 30°$, and the scaling function of the gait (3) to $g(i) = 1$. The control gains were tuned slightly differently than for eel-like motion in order to compensate for the stronger oscillations of the head link: the control gain k_n of the heading controller was $k_n = 15$, and the transversal gain $k_{\mathrm{trans}} = 0.035$. All other gains were chosen in accordance with Table 6. Figure 17 shows the resulting data. We see that the underwater snake robot converged to the path and then oscillated about it, thus making the cross-track error small and fulfilling the path following control objective. The head link angle oscillated about its reference. Compared to the case with zero current in Section 6.4 it can be seen in Figure 17(b) that θ_{ref} now oscillates about a non-zero value in steady state, i.e. it automatically finds the steady state side-slip angle that is necessary in order to compensate for the current. In particular, this happened because the reference velocity vector $\boldsymbol{\mu}$ was no longer aligned with the path, since it had to compensate for the sideways component of the current. Compared to the previous example, the reference velocity $v_{\mathrm{ref}} = \|\boldsymbol{\mu}\|$ now changed, as the orientation of the robot changed. This can be explained by the fact that the robot was started with an orientation that directly opposed the current, and therefore had to make up for a stronger effect in the beginning than after convergence to the path. As in the previous example, the measured velocity did not quite reach the reference. Just like for the simulations and the case without current, an offset remained between the mean of the forward velocity and its reference due to the trade-off that has to be made for the practically stable speed controller between high control gains and a small offset.
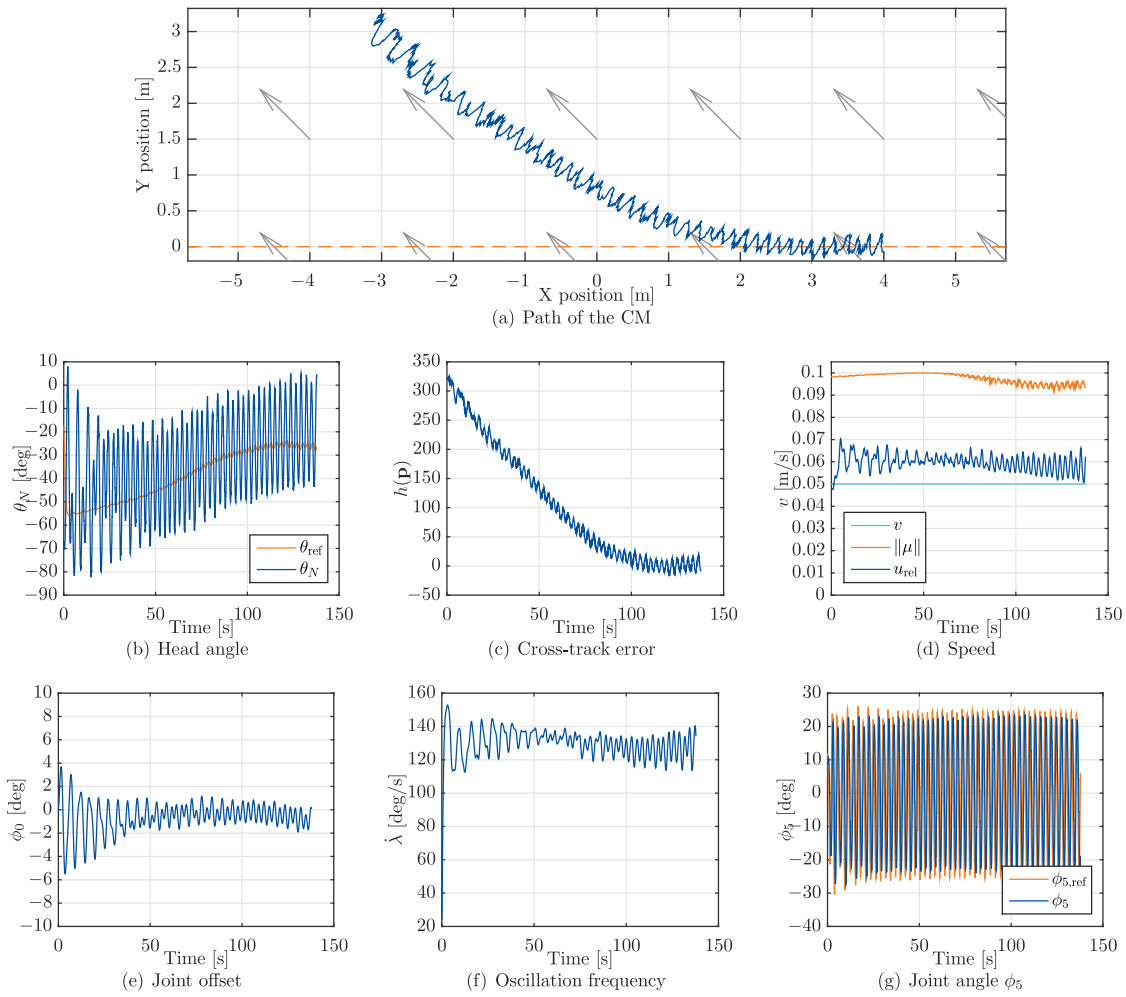
Figure 17. Experimental results for straight line path following with lateral undulation and a known ocean current: (a) The path of the underwater snake robot (b) The head link angle θ_N and the reference heading θ_{ref} (c) The path following error (d) The velocity u and the reference velocity v_{ref} (e) The joint offset ϕ_0 which is used to control the turning rate (f) The frequency of the undulatory gait is governed by $\dot{\lambda}$ (g) The joint angle ϕ_5 and its reference

This drawback is compensated for by the integral action in the path following control stage (18), thus ensuring convergence to the path despite the velocity error. The joint offset ϕ_0 and the oscillation frequency $\dot{\lambda}$ remained bounded and the joint angles tracked their respective reference signals.

Eel-like motion For eel-like motion, the scaling function in the gait (3) was set to $g(i) = \frac{N-i}{N+1}$ just like in the current-free scenario in Section 6.4. Just like for lateral undulation, the flow speed was $V = 0.05$ m/s, and the integral gain was chosen according to Table 6. The other control gains were chosen analogously to the current-free case in Section 6.4. The results of the test run are displayed in Figure 18. It can be seen that the robot approached and followed the reference path, and the cross-track error
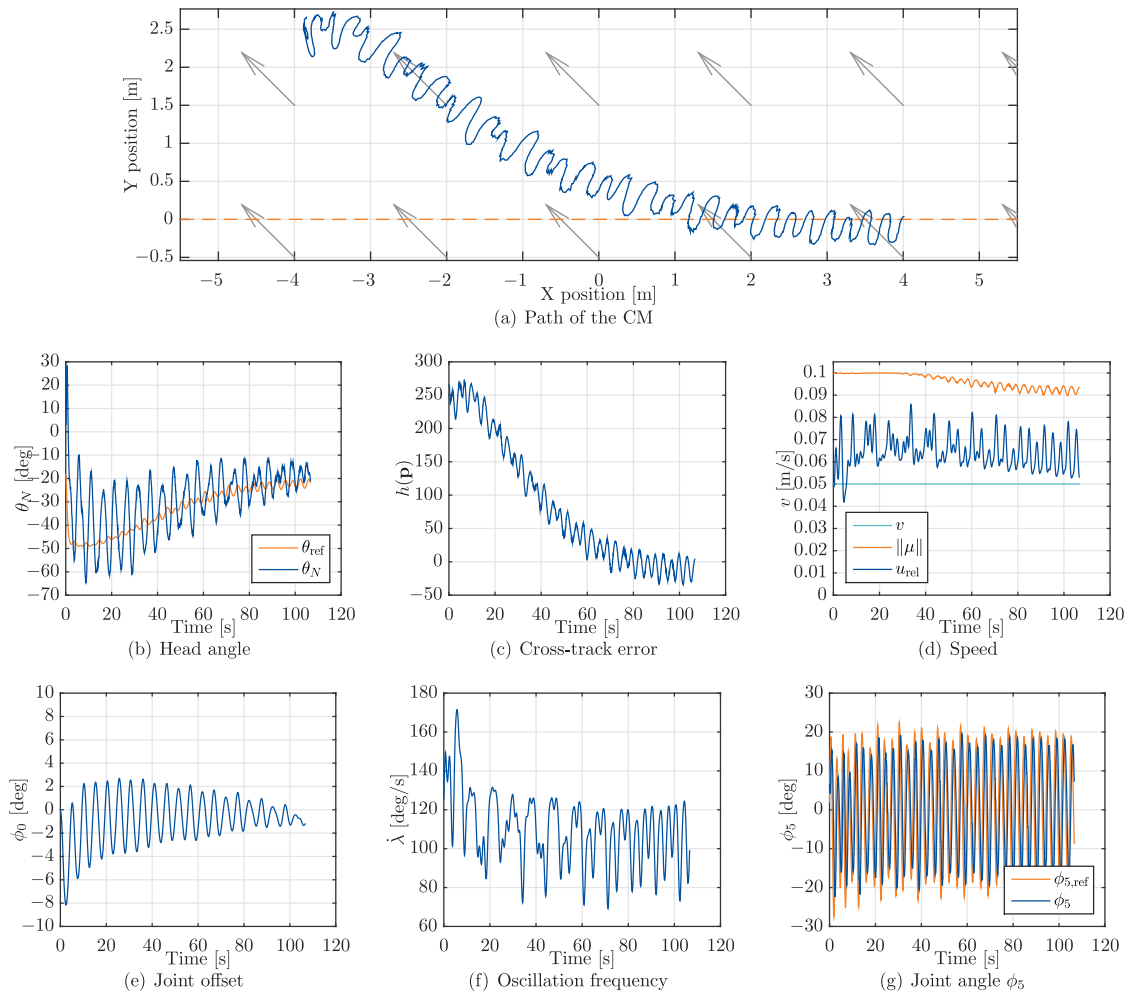
Figure 18. Experimental results for straight line path following with eel-like motion and a known ocean current: (a) The path of the underwater snake robot (b) The head link angle θ_N and the reference heading θ_{ref} (c) The path following error (d) The velocity u and the reference velocity v_{ref} (e) The joint offset ϕ_0 which is used to control the turning rate (f) The frequency of the undulatory gait is governed by $\dot{\lambda}$ (g) The joint angle ϕ_5 and its reference

approached zero. The head link angle stayed close to the reference, which reached a steady oscillation about a non-zero value in order to compensate for the sideways current. Just like in the other cases, an offset remained between the reference velocity and the mean of its actual value. Again, this is a consequence of the practical stability result, which implies that a trade-off has to be made between the control gain and the accuracy. The joint offset ϕ_0 and the oscillation frequency $\dot{\lambda}$ remained bounded and the joint angles followed the sinusoidal reference. The physical robot is displayed in Figure 19, where it can be seen how it approaches and follows the path.
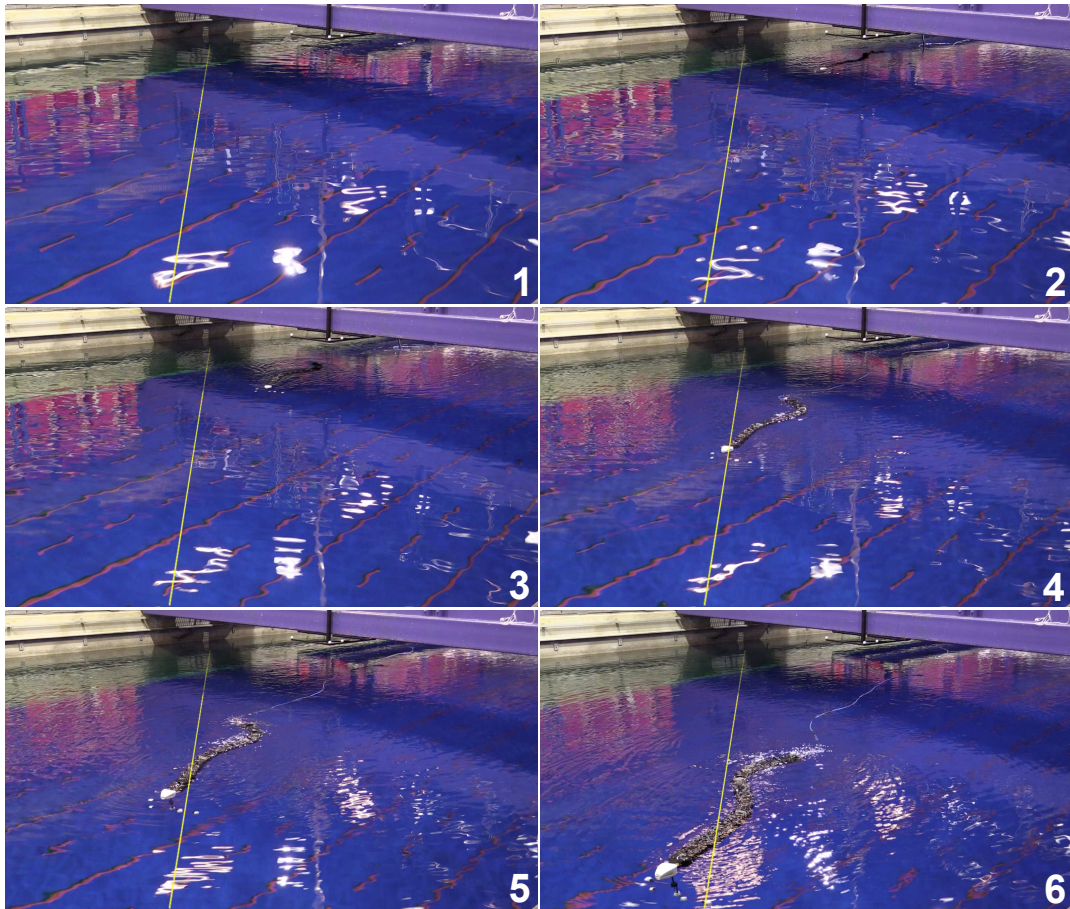
Figure 19. The snake robot Mamba: 1) The robot is launched with an offset to the path. 2) The robot turns towards the path. 3),4) The robot approaches the path smoothly. 5),6) The robot stays on the path. These frames are taken from a movie that has been attached to this article as supplementary data.

7. Discussion

The control system presented in this article enables underwater snake robots to track generic smooth reference paths with a desired forward speed. It is in essence the generalization of an analogous control system for land-based snake robots presented in [13].

Extensive simulation studies demonstrate that the control system performs well for different undulatory gaits and path geometries with only small changes compared to the system in [13]. Due to the general nature of the control system, the performance for one specific problem could be improved by selective tuning of the control parameters. This would however result in lower performance for the general case. Another trade-off must be made when adapting the body shape controller and the speed controller. In particular, the accuracy of these control layers can be improved by increasing the control gains of those stages. However, this will also increase the control torques and body oscillations and thus result in higher energy consumption and possibly even actuator

saturation.

In addition to the simulations, the control system was tested in an experimental study. In the experimental trials, only one path geometry, a straight line, was tested due to the limited angular range of the motion capture system. It was therefore possible to tune the control gains of the path following controller for that specific scenario, which was tested for two different undulatory gaits. Since the robot that was used for the tests consists of position controlled joints with readily tuned microcontrollers, the body shape controller was not in the focus of this study.

Regarding the speed controller, on the other hand, the experimental tests show what has been predicted by the simulation study: due to actuator constraints it is not feasible to tune the control gains sufficiently high to make the offset between the forward velocity and its reference negligible. This observation agrees with the practical stability property of the speed controller, which implies that there is a trade-off between the control gains and the remaining offset. However, both the simulation study and the experimental trials show that the integral action of the path following controller in (19) compensates for this imperfection. In future work we will investigate the possibility to reduce the error in the speed controller by adding the integral action directly to the speed controller, thus making the robot converge completely to the reference velocity and eliminating the need to compensate for the velocity error in the path following controller.

8. Conclusion

This article studied the problem of planar maneuvering control for bio-inspired underwater snake robots in the presence of unknown ocean currents. The control objective was to stabilize the robot to a planar reference path and track the path with a desired velocity. This was achieved by enforcing virtual constraints on the body of the robot, thus encoding a biologically inspired gait with the additional option of regulating the orientation and forward speed of the snake robot. The desired orientation and forward speed were provided by the top layer path following controller: A two-state ocean current observer was added to a path following control system designed for land-based snake robots in order to account for the additional disturbance by the current. A simulation study validated that the proposed control system performs well for different gaits and path geometries. The control system was additionally validated experimentally for the special case of a straight path and under the assumption of a known current. Future work will include a further improvement of the speed controller and an extension to the three-dimensional case.

Appendix A. Notation

In this article, the following vectors and operators are used:

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 0 & \dots & 0 \\ 0 & 1 & 1 & \dots & 0 \\ & \dots & & & \\ 0 & 0 & \dots & 1 & 1 \end{bmatrix} \in \mathbb{R}^{(N-1)\times N}, \quad \mathbf{D} = \begin{bmatrix} 1 & -1 & \dots & 0 & 0 \\ 0 & 1 & -1 & \dots & 0 \\ & \dots & & & \\ 0 & 0 & \dots & 1 & -1 \end{bmatrix} \in \mathbb{R}^{(N-1)\times N},$$

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ 0 & 1 & \dots & 1 \\ & \dots & & \\ 0 & 0 & \dots & 1 \\ 0 & 0 & \dots & 0 \end{bmatrix} \in \mathbb{R}^{N\times(N-1)}, \qquad \mathbf{I}_N = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & \dots & \\ & & & 1 \end{bmatrix} \in \mathbb{R}^{N\times N},$$

$$\mathbf{e} = \begin{bmatrix} 1 & \dots & 1 \end{bmatrix}^T \in \mathbb{R}^N, \qquad \bar{\mathbf{e}} = \begin{bmatrix} 1 & \dots & 1 \end{bmatrix}^T \in \mathbb{R}^{N-1},$$

$$\boldsymbol{\theta} = \begin{bmatrix} \theta_1 & \dots & \theta_N \end{bmatrix}^T \in \mathbb{R}^N, \qquad \mathbf{E} = \begin{bmatrix} \mathbf{e} & \mathbf{0}_{N\times 1} \\ \mathbf{0}_{N\times 1} & \mathbf{e} \end{bmatrix} \in \mathbb{R}^{2N\times 2},$$

$$\sin\boldsymbol{\theta} = \begin{bmatrix} \sin\theta_1 & \dots & \sin\theta_N \end{bmatrix}^T \in \mathbb{R}^N, \qquad \cos\boldsymbol{\theta} = \begin{bmatrix} \cos\theta_1 & \dots & \cos\theta_N \end{bmatrix}^T \in \mathbb{R}^N,$$

$$\mathbf{S}_{\boldsymbol{\theta}} = \operatorname{diag}(\sin\boldsymbol{\theta}) \in \mathbb{R}^{N\times N}, \qquad \mathbf{C}_{\boldsymbol{\theta}} = \operatorname{diag}(\cos\boldsymbol{\theta}) \in \mathbb{R}^{N\times N},$$

$$\dot{\boldsymbol{\theta}}^2 = \begin{bmatrix} \dot\theta_1^2 & \dots & \dot\theta_N^2 \end{bmatrix}^T \in \mathbb{R}^N, \qquad \mathbf{SC}_{\theta} = \begin{bmatrix} \mathbf{K}^T\mathbf{S}_{\boldsymbol{\theta}} \\ -\mathbf{K}^T\mathbf{C}_{\boldsymbol{\theta}} \end{bmatrix},$$

$$\mathbf{V} = \mathbf{A}^T(\mathbf{D}\mathbf{D}^T)^{-1}\mathbf{A}, \qquad \mathbf{K} = \mathbf{A}^T(\mathbf{D}\mathbf{D}^T)^{-1}\mathbf{D},$$

$$\mathbf{Q}_{\theta} = -\begin{bmatrix} c_t(\mathbf{C}_{\boldsymbol{\theta}})^2 + c_n(\mathbf{S}_{\boldsymbol{\theta}})^2 & (c_t - c_n)\mathbf{S}_{\boldsymbol{\theta}}\mathbf{C}_{\boldsymbol{\theta}} \\ (c_t - c_n)S_{\boldsymbol{\theta}}\mathbf{C}_{\boldsymbol{\theta}} & c_t(\mathbf{S}_{\boldsymbol{\theta}})^2 + c_n(\mathbf{C}_{\boldsymbol{\theta}})^2 \end{bmatrix}.$$

References

[1] N. Kato. Control performance in the horizontal plane of a fish robot with mechanical pectoral fins. *IEEE Journal of Oceanic Engineering*, 25(1):121–129, 2000.

[2] K. A. Morgansen, V. Duindam, R. J. Mason, J. W. Burdick, and R. M. Murray. Nonlinear control methods for planar carangiform robot fish locomotion. In *Proc. 2001 IEEE Int. Conf. Robotics and Automation*, Seoul, Korea, May 2001.

[3] J. E. Colgate and K. M. Lynch. Mechanics and control of swimming: A review. *IEEE J. Ocean. Eng.*, 29(3):660–673, 2004.

[4] M. Kruusmaa, P. Fiorini, W. Megill, M. de Vittorio, O. Akanyeti, F. Visentin, L. Chambers, H. El Daou, M. C. Fiazza, J. Ježov, M. Listak, L. Rossi, T. Salumae, G. Toming, R. Venturelli, D. S. Jung, J. Brown, F. Rizzi, A. Qualtieri, J. L. Maud, and A. Liszewski. FILOSE for Svenning: A flow sensing bioinspired robot. *IEEE Robotics & Automation Magazine*, 21(3):51–62, 2014.

[5] S. Hirose. *Biologically Inspired Robots: Snake-Like Locomotors and Manipulators*. Oxford University Press, Oxford, 1993.

[6] P. Liljebäck, K. Y. Pettersen, Ø. Stavdahl, and J. T. Gravdahl. *Snake Robots: Modelling, Mechatronics, and Control*. Advances in Industrial Control. Springer London, 2012.

[7] P. Liljebäck, K. Y. Pettersen, Ø. Stavdahl, and J. T. Gravdahl. A review on modelling, implementation, and control of snake robots. *Robotics and Autonomous Systems*, 60:29–40, 2012.

[8] C. D. Onal and D. Rus. Autonomous undulatory serpentine locomotion utilizing body dynamics of a fluidic sort robot. *Bioinspiration & Biomimetics*, 8:026003, 2013.

[9] M. Porez, F. Boyer, and A. J. Ijspeert. Improved lighthill fish swimming model for bio-inspired robots: Modeling, computational aspects and experimental comparisons. *Int. J. Robot. Res.*, 33(10):1322–1341, 2014.

[10] E. Kelasidi, P. Liljebäck, K. Y. Pettersen, and J. T. Gravdahl. Innovation in underwater robots: Biologically inspired swimming snake robots. *IEEE Robotics Automation Magazine*, 23(1):44–62, March 2016.

[11] E. Kelasidi, P. Liljebäck, K. Y. Pettersen, and J. T. Gravdahl. Experimental investigation of efficient locomotion of underwater snake robots for lateral undulation and eel-like motion patterns. *Robotics and Biomimetics*, 2(8):1–27, 2015.

[12] A. Mohammadi, E. Rezapour, M. Maggiore, and K. Y. Pettersen. Direction following control of planar snake robots using virtual holonomic constraints. In *Proc. IEEE 53rd Conf. Decision and Control*, Los Angeles, CA, USA, December 2014.

[13] A. Mohammadi, E. Rezapour, M. Maggiore, and K. Y. Pettersen. Maneuvering control of planar snake robots using virtual holonomic constraints. *IEEE Trans. Control Syst. Technol.*, 24(3):884 – 899, 2015.

[14] E. Rezapour, K. Y. Pettersen, P. Liljebäck, J. T. Gravdahl, and E. Kelasidi. Path following control of planar snake robots using virtual holonomic constraints: theory and experiments. *Robotics and Biomimetics*, 1(3):1–15, 2014.

[15] E. Rezapour, Hofmann A., K. Y. Pettersen, A. Mohammadi, and M. Maggiore. Virtual holonomic constraint based direction following control of planar snake robots described by a simplified model. In *Proc. IEEE Conf. Control Applications*, Antibes, France, Oct. 2014.

[16] J. Nakanishi, T. Fukuda, and D.E. Koditschek. A brachiating robot controller. *IEEE Transactions on Robotics and Automation*, 16(2):109–123, 2000.

[17] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. H. Choi, and B. Morris. *Feedback control of dynamic bipedal robot locomotion*. Control and automation. CRC Press, Boca Raton, 2007.

[18] C. Chevallereau, J. W. Grizzle, and C. L. Shih. Asymptotically stable walking of a five-link underactuated 3D bipedal robot. *IEEE Trans. Robot.*, 25(1):37–50, 2008.

[19] P. Liljebäck, I. U. Haugstuen, and K. Y. Pettersen. Path following control of planar snake robots using a cascaded approach. *IEEE Trans. Control Syst. Technol.*, 20(1):111–126, 2012.

[20] E. Kelasidi, K. Y. Pettersen, P. Liljebäck, and J. T. Gravdahl. Integral line-of-sight for path following of underwater snake robots. In *Proc. IEEE Conf. Control Applications*, Antibes, France, Oct. 2014.

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

[22] M. Saito, M. Fukaya, and T. Iwasaki. Serpentine locomotion with robotic snakes. *IEEE Control Systems Magazine*, 22(1):64–81, 2002.

[23] A. J. Wiens and M. Nahon. Optimally efficient swimming in hyper-redundant mechanisms: control, design, and energy recovery. *Bioinspiration & Biomimetics*, 7(4), 2012.

[24] A. J. Ijspeert and A. Crespi. Online trajectory generation in an amphibious snake robot using a lamprey-like central pattern generator model. In *Proc. IEEE Int. Conf. Robotics and Automation*, Rome, Italy, April 2007.

[25] A. Crespi and A. J. Ijspeert. AmphiBot II: An amphibious snake robot that crawls and swims using a central pattern generator. In *Proc. 9th Int. Conf. Climbing and Walking Robots*, Brussels, Belgium, Sep. 2006.

[26] A. J. Ijspeert. Central pattern generators for locomotion control in animals and robots: A review. *Neural Networks*, 21:642–653, 08.

[27] K. A. McIsaac and J. P. Ostrowski. Motion planning for anguilliform locomotion. *IEEE Trans. Robot. Autom.*, 19(4):637–652, 2003.

[28] P. Liljebäck, Ø. Stavdahl, K.Y. Pettersen, and J.T. Gravdahl. Mamba - A waterproof snake robot with tactile sensing. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Chicago, IL, Sep. 2014.

[29] C. Canudas-de Wit. On the concept of virtual constraints as a tool for walking robot control and balancing. *Annual Reviews in Control*, 28(2):157–166, 2004.

[30] A. Shiriaev, J. W. Perram, and C. Canudas de Wit. Constructive tool for orbital stabilization of underactuated nonlinear systems: virtual constraints approach. *IEEE Trans. Autom. Control*, 50(8):1164–1176, 2005.

[31] M. Maggiore and L. Consolini. Virtual holonomic contstraints for Euler-Lagrange systems. *IEEE Trans. Autom. Control*, 58(4):1001–1008, 2013.

[32] C. Chevallereau, G. Abba, Y. Aoustin, F. Plestan, E. R. Westervelt, C. Canudas-de Wit, and J. W. Grizzle. Rabbit: A testbed for advanced control theory. *IEEE Control Systems Magazine*, 23(5):57–79, 2003.

[33] J. W. Grizzle, J. Hurst, B. Morris, H. Park, and K. Sreenath. Mabel, a new robotic bipedal walker and runner. In *American Control Conference, 2009. ACC'09.*, pages 2030–2036. IEEE, 2009.

[34] J. W. Grizzle, C. Chevallereau, R. W. Sinnet, and A. D. Ames. Model, feedback control, and open problems of 3D bipedal robotic walking. *Automatica*, 50:1955–1988, 2014.

[35] T. I. Fossen. *Handbook of Marine Craft Hydrodynamics and Motion Control*. Wiley, 2011.

[36] Andrew Teel and Laurent Praly. Tools for semiglobal stabilization by partial state and output feedback. *SIAM Journal on Control and Optimization*, 33(5):1443–1488, 1995.

[37] R.D. Christ and R.L. Wernli. *The ROV Manual: A User Guide for Remotely Operated Vehicles*. Elsevier Science, 2013.

[38] A. P. Aguiar and A. M. Pascoal. Dynamic position and way-point tracking of underactuated AUVs in the presence of ocean currents. In *Proc. 41st IEEE Conf. Decision and Control*, Las Vegas, NV, USA, Dec. 2002.

[39] S. Moe, W. Caharija, K. Y. Pettersen, and I. Schjølberg. Path following of underactuated marine surface vessels in the presence of unknown ocean currents. In *Proc. American Control Conf.*, Portland, OR, USA, June 2014.

[40] C. Paliotta and K. Y. Pettersen. Geometric path following with ocean current estimation for ASVs and AUVs. accepted and to be presented at Amercian Control Conference 2016.

[41] C.-T. Chen. *Linear system theory and design*. Oxford University Press, Inc., 1995.

[42] The North Sea Centre Flume Tank–managed and operated by SINTEF Fisheries and Aquaculture.

[43] Qualisys–motion capture systems.

[44] H.-P. Tan, R. Diamant, W. K. G. Seah, and M. Waldmeyer. A survey of techniques and challenges in underwater localization. *Ocean Engineering*, 38(14-15):1663–1676, 2011.