Anna Magdalena Kohl

Guidance and Control of Underwater Snake Robots Using Planar Sinusoidal Gaits

Thesis for the degree of Philosophiae Doctor
Trondheim, October 2017

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

NTNU
Norwegian University of
Science and Technology

NTNU
Norwegian University of Science and Technology

Thesis for the degree of Philosophiae Doctor

Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

ISBN 978-82-326-2652-6 (printed version)
ISBN 978-82-326-2653-3 (electronic version)
ISSN 1503-8181
ITK Report 2017-16-W

Doctoral theses at NTNU, 2017:292

Printed by Skipnes Kommunikasjon as

Summary

This thesis presents different approaches for guidance and motion control of underwater snake robots. The robots considered in this thesis are neutrally buoyant and move slowly with a planar, biologically inspired sinusoidal gait. The proposed guidance systems are designed such that they can reject environmental disturbances by an ocean current or avoid collision with stationary obstacles. The control approaches developed in this thesis are model-based, which enables formal stability analyses.

As a basis for the development of model-based control methods and in order to create a fundamental understanding of underwater snake robot locomotion, one chapter of this thesis deals with modelling and locomotion analysis of underwater snake robots. More specifically, two models are presented for underwater snake robots that move in a plane. The first one is based on first principles and formalized using Newton-Euler equations. Based on this model, an analysis of planar sinusoidal locomotion is presented, revealing several fundamental properties. These properties allow some simplifying assumptions, based on which the second, control-oriented model is derived. The two models are compared in an extensive simulation study, where the advantages and limitations of the simplified control-oriented model become clear. The chapter is concluded by an averaging analysis of the velocity dynamics of the control-oriented model during sinusoidal gaits.

In the subsequent chapters, two approaches for guidance and motion control of underwater snake robots are developed. The first one is based on the control-oriented model and employs a sinusoidal gait to achieve a positive velocity without a velocity feedback controller in the inner control loop. As an outer control loop, a model-based orientation controller is designed, which steers the robot towards and along a straight path in the presence of unknown ocean currents. This is achieved by employing an integral line-of-sight guidance law. The control system is experimentally validated and the origin of the error dynamics is shown to be uniformly semi-globally exponentially stable using cascaded systems theory. The second approach for guidance and motion control is based on the original, more complex model and employs a manoeuvring controller that considers both heading and velocity control, and enables the robot to follow a generic path. It is an extension of a similar method for terrestrial snake robots, that was formally shown to practically stabilize the states to their references. The proposed feedback control strategy enforces virtual constraints on the snake robot configuration. The constraints encode biologically inspired gaits and are parametrized by states of dynamic compensators that are used to regulate the heading and forward velocity of the snake robot. In

order to adapt the control approach for underwater robots, a two-state ocean current observer, based on relative velocity sensors, is proposed. It enables the robot to achieve its path-following control objectives in the presence of ocean currents. The efficacy of the proposed control algorithm is verified for several biologically inspired gaits both in simulations for different path geometries, and in experiments.

The control approaches described above both rely on the availability of relative velocity measurements. In an additional chapter, this thesis presents a method for orientation and velocity control of a robot that has access to absolute velocity measurements instead. This controller also utilizes virtual constraints in order to impose a sinusoidal gait on the body shape of the robot and is based on an adaptive backstepping design. A stability analysis is presented, where it is shown that joint and velocity control objectives are guaranteed to be satisfied, and further analysis challenges are outlined. The findings are illustrated with simulations.

Finally, a guidance strategy for path-following with obstacle avoidance for underwater snake robots is presented. The guidance strategy is an adaptation of a guidance system for surface vessels, that consists of two modes, path-following mode and obstacle avoidance mode. The guidance scheme is made suitable for snake robots by introducing a new, purely kinematic switching condition It is shown that the guidance strategy guarantees obstacle avoidance for non-overlapping obstacles, which is in addition validated experimentally.

Preface

This thesis is submitted in partial fulfillment of the requirements for the degree of philosophiae doctor (PhD) at the Norwegian University of Science and Technology (NTNU). The work was carried out from August 2014 to 2017 at the Centre for Autonomous Marine Operations and Systems and the Department of Engineering Cybernetics under the supervision of Professor Kristin Y. Pettersen and the co-supervision of Professor Jan Tommy Gravdahl and Dr Eleni Kelasidi.

Acknowledgments

This work was supported by the Research Council of Norway through its Centres of Excellence funding scheme, project no. 223254–NTNU AMOS.

The last three years during which the work for this thesis was carried out have gone by incredibly fast, and while it has been a very intense time, I have learned a lot, and I had the opportunity to work together with many inspiring people, which I am very grateful for.

First of all, I would like to thank Professor Kristin Ytterstad Pettersen for her scientific advice and excellent supervision during the past three years. She has managed to keep the perfect balance between providing new input and guidance while giving me enough freedom and being supportive of my own ideas. Her advice and feedback have been immensely helpful and she has always found time for me and my problems. I also thank my co-supervisor Professor Jan Tommy Gravdahl for providing to-the-point advice in exactly the right moments and keeping calm during the hectic days before hard deadlines, and my co-supervisor Dr Eleni Kelasidi for all the hard effort she put into the experiments that we carried out as a part of this work.

As part of my PhD work, I was lucky enough to collaborate with a group from the University of Toronto, namely Professor Manfredi Maggiore and Dr Alireza Mohammadi. I have learned a lot during our fruitful joint work, their scientific advice has been invaluable, and it was a great pleasure to getting to know them during their visit at NTNU.

During the last three years I also had the privilege to work alongside the most awesome colleagues one could hope for. I am truly thankful for my office mates Albert Sans Muntadas and Signe Moe. It has been a pleasure to share offices with them and to have somebody to discuss a new thought with as soon as it popped into my head. On that note, I also want to thank Pål Liljebäck, Jørgen Sverdrup-Thygeson, Dennis Belleter, Claudio Paliotta, Mikkel Sørensen, Mikkel Cornelius

Nielsen, Kristian Klausen, and Martin Syre Wiig for all the technical discussions we had over the years. There are so many more who have contributed to making the last three years special to me, I will always remember the coffee breaks at AMOS and ITK, our semi-regular girls' nights, the weekly cageball games, and of course the times we spent travelling together.

The environment that we are working in would certainly not be the same without the administrative and technical staff at ITK. It has been a privilege to be surrounded by so many friendly and competent people who have been incredibly supportive in every way. I want to thank the administrative personnel for making my life a lot easier during the last years, especially Tove Kristin Blomset Johnsen and also Sigrid Bakken Wold from AMOS. My thanks also go to Daniel Bogen, Stefano Bertelli, Terje Haugen and the other technical staff at ITK for their invaluable support regarding IT and hardware. I am especially thankful to Glenn Angell for all the help during the experiments in Denmark. In that regard I would also like to acknowledge the support of Nina A. H. Madsen and Anders Nielsen in the flume tank in Hirtshals, Martin Holmberg from Qualisys, and Torgeir Wahl in the MC-lab.

Last but not least, my sincerest thanks go to my family and friends: the new friends that I have made here at NTNU, who have contributed so much to the good time I had both while at work and while engaging in all sorts of non-work-related activities; my friends back in Munich, Kulmbach, and anywhere else, for always welcoming me back as if I had never left and for travelling to Trondheim in flocks; my grandparents from Unternesselbach and Neustadt, who were always supportive and curious about what I was up to, my aunts Lore and Elke, and of course Marianne, who came all the way to Trondheim for visiting me, my sister Linda and my brother Stefan, and my parents Gertrud and Jürgen, who have raised me to be who I am, for believing in me, and encouraging me. Finally, I want to thank Jonas, the person who has been there for me throughout these past three years, for taking on this adventure together with me.

Contents

Nomenclature

Abbreviations

AS	Asymptotically stable
AUV	Autonomous underwater vehicle
CM	Center of mass
DVL	Doppler velocity log
IMR	Inspection, maintenance, and repair
LOS	Line-of-sight
PE	Persistently exciting
ROV	Remotely operated vehicle
UAS	Uniformly asymptotically stable
UGAS	Uniformly globally asymptotically stable
UGES	Uniformly globally exponentially stable
ULES	Uniformly locally exponentially stable
US	Uniformly stable
USGES	Uniformly semi-globally exponentially stable
VHC	Virtual holonomic constraint

Model conventions

$\bar{\theta}$	Average orientation of the robot
$\mathbf{n} \in \mathbb{R}^N$	Vector that contains all n_i
$\mathbf{p} \in \mathbb{R}^2$	CM position
$\mathbf{t} \in \mathbb{R}^N$	Vector that contains all t_i
$\mathbf{v}_\mathrm{c} \in \mathbb{R}^2$	Current velocity in the inertial frame
$\mathbf{v}_\mathrm{rel} \in \mathbb{R}^2$	Relative velocity of the CM in the body-aligned frame

$\mathbf{v}_\mathrm{r} \in \mathbb{R}^2$	Relative velocity of the CM in the inertial frame
$\mathbf{X} \in \mathbb{R}^N$	Vector that contains all x_i
$\mathbf{Y} \in \mathbb{R}^N$	Vector that contains all y_i
$\boldsymbol{\phi} \in \mathbb{R}^{N-1}$	Vector that contains all joint angles or coordinates
$\boldsymbol{\theta} \in \mathbb{R}^N$	Vector that contains all link angles
ϕ_i	Joint angle or joint coordinate of joint i
θ	Orientation of the control-oriented model
θ_i	Link angle of link i
θ_N	Heading of the robot
$v_{n,\mathrm{rel}}$	Relative velocity component of the CM that is normal to the orientation θ
$v_{t,\mathrm{rel}}$	Relative velocity component of the CM that is aligned with the orientation θ
n_i	n-coordinate of the CM position of link i
p_x	x-coordinate of the CM position
p_y	y-coordinate of the CM position
t_i	t-coordinate of the CM position of link i
V_c	Current magnitude
v_n	Velocity component of the CM that is normal to the heading θ_N or orientation θ
v_t	Velocity component of the CM that is aligned with the heading θ_N or orientation θ
V_x	x-component of the current velocity
V_y	y-component of the current velocity
x_i	x-coordinate of the CM position of link i
y_i	y-coordinate of the CM position of link i

Model parameters

λ_1	Rotational damping coefficient control-oriented model
λ_2	Rotational coupling coefficient control-oriented model
μ_θ	Rotational added inertia parameter
μ_n	Normal added inertia parameter
c_p	Propulsion coefficient control-oriented model

c_n	Normal drag parameter
c_t	Tangential drag parameter
c_θ^{II}	Nonlinear rotational damping parameter
c_θ^{I}	Linear rotational damping parameter
J	Moment of inertia of a link
L	Length of a link
l	Half the length of a link
m	Mass of a link
N	Number of links

Matrix and vector operators

$\bar{\mathbf{e}} \in \mathbb{R}^{N-1}$	Column vector of ones and length $N-1$
$\mathbf{A} \in \mathbb{R}^{(N-1)\times N}$	Adds adjacent elements of a vector
$\mathbf{C}_{\boldsymbol{\theta}}$	Diagonal matrix composed of the elements of $\cos(\boldsymbol{\theta})$
$\mathbf{D} \in \mathbb{R}^{(N-1)\times N}$	Subtracts adjacent elements of a vector
$\mathbf{E} \in \mathbb{R}^{2N\times 2}$	Summation matrix
$\mathbf{e} \in \mathbb{R}^{N}$	Column vector of ones and length N
$\mathbf{I}_N \in \mathbb{R}^{N\times N}$	Unity matrix
$\mathbf{K} \in \mathbb{R}^{N\times N}$	Operator obtained from $\mathbf{A}$ and $\mathbf{D}$
$\mathbf{R}_\theta \in \mathbb{R}^{2\times 2}$	Rotation matrix with rotation angle θ
$\mathbf{S}_{\boldsymbol{\theta}}$	Diagonal matrix composed of the elements of $\sin(\boldsymbol{\theta})$
$\mathbf{V} \in \mathbb{R}^{N\times N}$	Operator obtained from $\mathbf{A}$ and $\mathbf{D}$
$\cos(\boldsymbol{\theta})$	Element-wise cosine of the vector $\boldsymbol{\theta}$
$\mathrm{diag}(\boldsymbol{\theta})$	Diagonal matrix composed of the elements of the vector $\boldsymbol{\theta}$
$\bar{\mathbf{D}} \in \mathbb{R}^{N\times(N-1)}$	Pseudo-inverse of $\mathbf{D}$
$\mathrm{sgn}(\boldsymbol{\theta})$	Element-wise signum of the vector $\boldsymbol{\theta}$
$\sin(\boldsymbol{\theta})$	Element-wise sine of the vector $\boldsymbol{\theta}$
$\boldsymbol{\theta}^2$	Point-wise square of the elements of $\boldsymbol{\theta}$

Chapter 1

Introduction

1.1 Motivation

There is a large potential for enhancing efficiency and reducing costs by increasing the level of autonomy in technological solutions. This includes improving traditional industrial robots as well as developing new robotic solutions for sectors that have so far relied on conventional technology. Subsea inspection, maintenance, and repair (IMR) operations is a field of technology with a large potential for autonomous robotics to increase the efficiency, reduce operation costs, and provide safer and more environmentally friendly solutions. Already today, remotely operated vehicles (ROVs) are replacing divers in subsea IMR operations. Employing fully autonomous underwater vehicles (AUVs) in addition to or instead of ROVs has the potential to further improve the safety and cost-effectiveness of such operations. This requires the development of new solutions for IMR operations that are more robust, agile, and versatile than existing technology. When designing solutions with such properties, inspiration can be found in nature, where a variety of species has adapted to subsea conditions over millions of years, and are thus specialized in propelling and manoeuvring underwater. Underwater snake robots, a class of robots that propel their slender bodies by mimicking the swimming motion of eels, can be considered as self-propelled, hyperredundant manipulator arms and are therefore promising candidates for autonomous IMR solutions in the future. In particular, they provide a high transport range similar to survey AUVs, excellent access capabilities, similar to observation ROVs, and light intervention capabilities, like standard ROVs.

Moreover, studying such robotic systems is interesting in itself. New insights in the biology of snakes and eels are required and encouraged, and investigating the locomotion of animals is supported by studying their robotic counterparts [35]. Furthermore, the development process of such robots is challenging both because of the harsh environment and their high number of degrees of freedom. This drives the advance of new methods and solutions, both in hardware and software development. Better battery solutions and new communication and localization techniques for underwater environments are just a few examples of technology that is required for further advancement of autonomous subsea operations and will also benefit the use of biologically inspired AUVs. Other aspects include mathematical models

and analyses, and the development of efficient and robust control algorithms for such robots. Recently, the physical implementation [66, 87], the modelling and analysis [38, 87], and control of underwater snake robots have received increased attention. This thesis considers the development of guidance and motion control approaches for slowly swimming underwater snake robots that propel themselves with planar sinusoidal gaits. The proposed methods are obtained using model-based nonlinear design methods and both mathematical analyses of the methods as well as simulation studies and experimental results that demonstrate their performance are presented.

1.2 Background and literature overview

This section gives an introduction to previous research that is relevant for this thesis. An overview of the development, modelling, analysis, and locomotion of biologically inspired robots is given in the first part. Guidance and control methods that have originated mainly from marine control and can be applied for biologically inspired robots are introduced in the second part. The purpose of this section is not to give an exhaustive literature review, surveys and review papers on the single fields are referred to in the respective paragraphs, but rather putting the results of this thesis into a wider perspective.

1.2.1 Biologically inspired robots using sinusoidal gaits

The following paragraphs give an overview of the origins and development of (underwater) snake robots, different modelling and analysis methods, and robotic sinusoidal gaits.

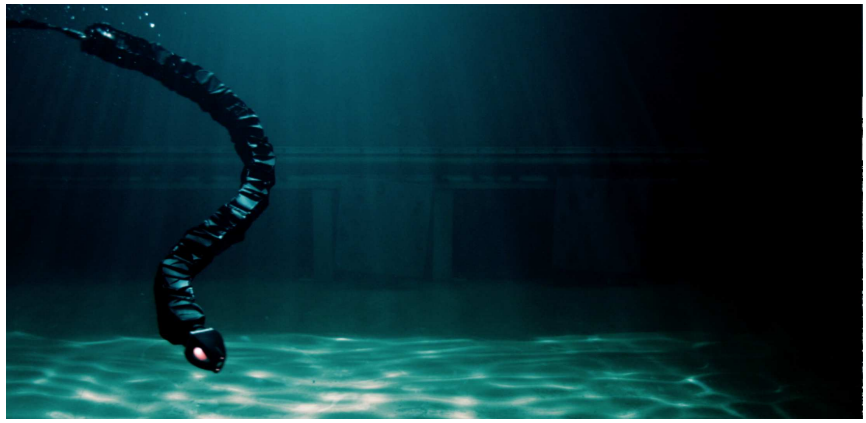
From biological snakes and fish to robots

A basis for the development of underwater snake robots was formed by Gray, who studied the locomotion mechanisms of both fish, with a special focus on eels [27], and terrestrial snakes [28]. Biological snakes demonstrate various gaits such as concertina, serpentina, rectilinear, and sidewinding locomotion [65]. In aquatic environments, biological fish locomote with other types of gaits such as anguilliform, thunniform, carangiform, and subcarangiform gaits (see [98] for a detailed study of various fish locomotory gaits). The focus of this thesis is on the important type of sinusoidal, undulating locomotion, namely serpentina and anguilliform gaits. It is suitable for achieving amphibious forward propulsion, i. e. propulsion both on land and in water. Indeed, the lateral undulatory gait demonstrated by terrestrial snakes is the most similar to the underwater anguilliform gait, which is typical for eels. It has been observed that eels use a gait similar to lateral undulation when they are forced to perform terrestrial motion [26].

Research on snake robots was initiated in the 1970s by Hirose, who developed the first terrestrial snake robot prototypes [31]. In the meantime, numerous different robot designs have emerged (see [33, 64] and references therein). Several examples can be seen in Figure 1.1. Some of these robots can even climb stairs [106] or trees [121], as can be seen in Figure 1.1(b). Most importantly for the context of this

(a) The AmphiBot III [87]. With permission by Prof. Auke Jan Ijspeert.

(b) The modular snake robot Uncle Sam. With permission by Prof. Howie Choset.



(c) The snake robot Mamba [66].

Figure 1.1: Different snake robot prototypes.

thesis, amphibious and underwater snake robots have been developed [18, 87, 123]. The amphibious snake robot Mamba (see Figure 1.1(c) and Appendix B) served as an experimental platform for the control approaches developed in this thesis.

Underwater snake robots are closely related to robotic fish, see examples in [3, 80, 112], and are sometimes even considered a special kind of fish robots [88]. Surveys on fish inspired robots and their control can be found in [17] and [88, 96]. Other related research fields include continuum robots [114] and soft robots [82, 110]. It was recently pointed out that studying biologically inspired robots such as the ones described above can, in turn, contribute to a better understanding

of animal locomotion from a biological and physical perspective [2, 35]. From an engineering point of view, snake robots are prospective candidates for solutions within space inspection [84], fire fighting [60], search and rescue [119, 120], medical surgeries [111], pipe inspection [93] and manipulation [89]. Very recently it was proposed to employ underwater snake robots for subsea IMR operations [103].

Mathematical modelling and analysis of snake robot locomotion

Different models for terrestrial snake robots and underwater snake robots can be found in the literature. Modelling approaches range from formulations based on classical Newton-Euler equations to modelling the robot body as a continuous curve [109], see also [64]. In [65], a model for two-dimensional terrestrial snake robot locomotion is obtained using first principles, based on earlier derivations in [68, 95]. From a control design perspective, it is often beneficial to use simplified approaches in order to develop a model that facilitates control design and analysis. On this note, using a virtual chassis for modelling terrestrial snake robot locomotion is proposed in [21, 94]. This strategy employs a body frame that is aligned with the principle moments of inertia and thus associates a clearly defined (virtual) orientation with the snake robot. The model from [21, 94] is mainly used to study different gaits and less for control design. A different simplified modelling approach can be found in [62, 65], which relies on simplifying the equations of motion by approximating the revolute link motion with prismatic joints. The control-oriented model from [62, 65] is developed especially for snake robots that move with the gait lateral undulation and is intended for control design and analysis.

For underwater snake robots and eel-like robots, several mathematical models have been developed in the literature [10, 71, 116]. Most recently, analytical models were developed in [87], based on improving Lighthill's large amplitude elongated body theory [59], and in [45], based on including fluid-dynamic effects into the model for terrestrial snake robots from [65]. Because of the similarity to the model in [65], the model in [45] has the advantage that it facilitates the advancement of model-based control approaches from terrestrial to underwater snake robots. However, due to the high complexity of the fluid-dynamical model, the model equations in [45] are still quite complicated and make an analysis from a control systems theoretical point of view quite difficult. A simplified, control-oriented modelling approach that captures the overall, qualitative behaviour of the robot is therefore desirable. A first step towards such a simplified model was taken in [40], by relying on the same simplifying assumptions as the model in [62, 65], and disregarding ocean currents.

Due to the complexity of the dynamics of a snake robot, all these models have in common that they are highly non-linear. Furthermore, it was shown in [65] that a control law that stabilizes snake robot locomotion has to be time-varying. For these reasons, an analysis of snake robot locomotion in general, and swimming snake robot locomotion specifically, is very challenging to perform. Because of the sinusoidal nature of snake locomotion, averaging is an appealing method for its analysis. It has been applied for the analysis of biologically inspired robots several times in the literature, for example for terrestrial snake robots [61, 65], fish robots [79, 113], and for underwater snake robots [41, 70].

Robotic implementation of snake locomotion gaits

The first mathematical representation of the snake body shape during lateral undulation was given by Hirose [31], who defined the serpenoid curve

$$\begin{bmatrix} x(s) \\ y(s) \end{bmatrix} = \int_0^s \begin{bmatrix} \cos\big(a\cos(b\sigma) + c\sigma\big) \\ \sin\big(a\cos(b\sigma) + c\sigma\big) \end{bmatrix} \mathrm{d}\sigma, \tag{1.1}$$

where s is the arc length along the snake body and a, b, c are positive real constants. A discrete approximation of the serpenoid curve is frequently used as a gait for snake robots that consist of rigid links (see e.g. [65, 95]). This gait is called lateral undulation and is achieved by making the single joints track the reference

$$\phi_{i,\mathrm{ref}}(t) = \alpha \sin\big(\omega t + (i-1)\delta\big) + \phi_0(t), \tag{1.2}$$

where $i \in \{1, \ldots, N-1\}$ is an index that indicates which joint the reference signal is used for, α is the amplitude of the joint angle motion, ω is the frequency of the body oscillations, δ is the phase shift between the adjacent joints of the robot, and ϕ_0 is an offset that induces a turning motion of the robot. In [44], the robotic lateral undulatory gait (1.2) was later generalized to

$$\phi_{i,\mathrm{ref}}(t) = \alpha g(i) \sin\big(\omega t + (i-1)\delta\big) + \phi_0(t). \tag{1.3}$$

The function $g : \mathbb{Z} \mapsto [0, 1]$ scales the amplitude of the single joints ϕ_i, and can therefore be used to vary the undulation amplitude along the body. This generalized gait includes, but is not limited to, lateral undulation for terrestrial snake robots, eel-like motion for underwater snake robots, and other sinusoidal forms of motion, like the gaits proposed in [116]. More specifically, the gait *lateral undulation* in (1.2) can be described by choosing $g(i) = 1$, and *eel-like motion* can for instance be achieved by the choice $g(i) = \frac{N-i}{N+1}$ [42].

Different methods have been used to control the gait of bio-inspired snake and eel-like robots. The control strategies fall into two general classes: open-loop and closed-loop strategies.

- **An open-loop strategy: Central pattern generators (CPGs)** are a popular method for the locomotion of snake robots, which is inspired by neural circuits found in animals. CPGs are biologically inspired neural networks that generate rhythmic patterned outputs without sensory feedback. They have been employed in terrestrial snake robot locomotion [36, 122] and underwater snake robot locomotion [18] (see [34] 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 [34], there does not exist a theoretical foundation for analysing the stability of the complete CPG-robot system.
- **A closed-loop strategy: Time reference trajectory tracking.** In this family of control strategies, feedback 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 [45, 65, 95]. This strategy is a popular choice for the implementation of physical robots

[66, 70, 87, 95]. In particular, the joint offset $\phi_0(t)$ in the generalized sinusoidal gait (1.3) is often designed in a way that makes the snake robot follow a certain path. Introducing the time-varying reference signal $\phi_{i,\text{ref}}(t), i = 1, \ldots, N-1$, however, complicates the motion planning and mathematical analysis.

- **A closed-loop strategy: Virtual holonomic constraints (VHCs)** are the basis of another closed-loop strategy that was developed for underactuated mechanical system control [13, 69, 99]. A VHC is virtual because it 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 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 [15, 29, 115]. In the context of snake robotics, biologically inspired VHCs come from adapting the reference signal for the single joints (1.2) in the following way [77, 78, 92]:

$$\phi_{i,\text{ref}}(\lambda, \phi_0) = \alpha \sin\big(\lambda + (i-1)\delta\big) + \phi_0, \tag{1.4}$$

where λ and ϕ_0 are states of dynamic compensators

$$\ddot{\lambda} = u_\lambda, \quad \ddot{\phi}_0 = u_{\phi_0}, \tag{1.5}$$

with the new control inputs u_λ and u_{ϕ_0}. The state-dependent relations $\phi_i = \phi_{i,\text{ref}}(\lambda, \phi_0)$, $i = 1, \ldots, N-1$ are the proposed VHCs. The new control state ϕ_0 is used to control the orientation, while the control state λ is used to control the velocity of the snake robot. The reference joint angles in (1.4) and thereby the VHCs can be enforced with any type of controller. Note that in (1.4), the time signal t no longer appears explicitly. Instead, the dynamic gait time evolution is governed by the state of the compensators in (1.5) and the new inputs u_λ and u_{ϕ_0}.

1.2.2 Guidance and control of marine vehicles and robots

This section introduces methods from marine control that have served as a basis for the developments in this thesis. Furthermore, these methods are linked to snake robotics research by discussing previous applications for snake robot control.

Guidance, navigation, and control

A marine motion control system can be divided into three separate, interconnected blocks, namely guidance, navigation, and control, which is illustrated in Figure 1.2. The three components are categorized as follows [22]:

- **Guidance** is the action of continuously providing a reference position, velocity, and acceleration to be used by the motion control system.

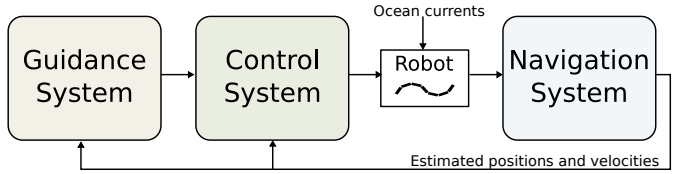


Figure 1.2: Guidance, navigation, and control [22]

- **Navigation** is the art of determining the position and attitude, course, travelled distance, velocity, and acceleration of a vehicle.
- **Control** is the action of determining necessary control inputs in order to satisfy the control objective, which often involves following a reference signal given by the guidance system.

This thesis is concerned with guidance and motion control of underwater snake robots in the presence of unknown environmental disturbances. More specifically, we present several approaches to tackle the manoeuvring problem for such robots. According to [100], the manoeuvring problem can be approached by solving the following two tasks:

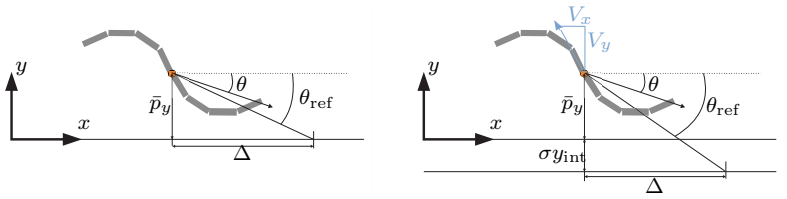
- **Geometric Task**: converge to a desired path
- **Dynamic Task**: converge to a desired speed

The geometric task can be interpreted as a path-following control problem, and the dynamic task as a velocity control problem. Environmental disturbances that typically have to be taken into account when designing marine control systems arise from waves, wind, and ocean current. For submerged vehicles that move far enough underneath the surface, waves and wind do not play a role, but ocean currents will still have a considerable effect on the robot.

The navigation block in Figure 1.2 presents a challenge in itself, in particular because many sensors and navigation techniques that are used on the surface are not available underwater. Position measurements underwater can for instance be obtained using acoustic sensors [104] or from communicating with a unit on a surface vessel that can determine its position using GPS [5]. Alternatively, vision-based approaches have been proposed [57]. Velocity measurements with respect to the earth can for instance be obtained with an acoustic Doppler velocity log (DVL) with bottom lock [16] while velocity data with respect to the ocean current is obtained from a DVL without bottom lock [16] or pressure sensors [3].

Path-following guidance of marine vehicles

One strategy for path-following is to apply the well-known line-of-sight (LOS) guidance [22], to determine a reference heading for the control system of the robot. The guidance scheme is visualized in Figure 1.3(a): the robot steers towards a point that is located on the path, at the look-ahead distance Δ ahead of the robot. For straight line paths it is possible to prove that the robot will stay on the path once it has reached it, if there is no ocean current. In the presence of ocean currents that have a component transverse to the path, this strategy will result in a steady



(a) Classical LOS guidance (b) LOS guidance with integral action

Figure 1.3: The line-of-sight guidance scheme. The tuning parameters are the look-ahead distance Δ and the integral gain σ.

state offset. This problem can be solved by augmenting the guidance law with integral action, which makes the robot target a point at the look-ahead distance Δ along a displaced path that lies upstream of the desired path, as illustrated in Figure 1.3(b). A formulation of the integral LOS guidance scheme with a strategy to prevent significant integral wind-up can be found in [11]. Based on the stability analysis of LOS guidance in [23], integral LOS guidance was shown to render the origin of the error coordinates uniformly semi-globally exponentially stable for surface vessels in [117].

There exist more general approaches to handle unknown ocean currents that can be combined with guidance strategies other than LOS. If both the absolute velocity of the robot with respect to the earth frame and the relative velocity with respect to the surrounding fluid is known, one can directly calculate the ocean current components and compensate them in the guidance law. If less information is available, other strategies exist. For instance, when position and relative velocity measurements are available, one can employ techniques like observers [1, 83] or filters [118] to estimate the unknown ocean current from the difference between the expected change of position obtained from integrating the relative velocity and the actual position change obtained from sensor data. Alternatively, when only absolute velocity measurements are available, adaptive controllers (see e. g. [56]) have been used to compensate for the unknown current effect in the dynamics [11, 76]. It should be noted that adaptive control provides robustness against uncertainties in general [24] and has recently been applied for underwater snake robots with model uncertainties [124].

Path-following control of snake and fish robots

In the context of snake and fish robotics, research on path-following control systems is quite limited. In [79], a fish robot is controlled to follow a straight line, and the averaged system is shown to stay on to the line. A motion planning strategy that is similar to LOS guidance is presented for an eel-like robot in [70]. In [58], a virtual-target guidance law is proposed for path-following of an eel-like robot. Another approach is presented in [30], where a LOS guidance law is employed in order to make a fish-like robot head towards predefined waypoints. However, none of the above consider currents. In [3], trajectory tracking in the presence of cur-

rents is performed with a fish robot with flow sensors, but no theoretical analysis is provided. The LOS guidance scheme for path-following of snake robots according to Figure 1.3 has been used both on land and in water. For terrestrial snake robots, the guidance strategy is investigated in [65] in combination with two different controllers. First, the strategy is implemented in combination with a proportional controller that steers the robot towards the path. Secondly, the orientation is controlled by a model-based strategy, which enables an analysis that formally shows stability. It has been proved to κ-exponentially stabilize the robot to a straight path in [63, 65]. This work was extended to include velocity control in [91], based on a similar strategy like an earlier velocity controller for snake robots in [95]. A manoeuvring control approach for terrestrial snake robots based on a different guidance strategy is proposed in [78]. The control system is based on a first-principle model, it considers both velocity and path-following control for generic paths, and is formally shown to practically stabilize the system states to their references. For swimming snake robots, the LOS guidance scheme in combination with a proportional orientation controller is experimentally investigated in [45]. The augmented integral LOS strategy with the same orientation controller is successfully tested in [46]. However, a formal stability analysis of the closed-loop system based on this orientation controller is not presented for the general case.

Obstacle avoidance of snake and fish robots

Another important guidance problem besides path-following is collision or obstacle avoidance. When following a path through an underwater installation, the snake robot has to be able to avoid obstacles that might interfere with its planned path. Examples of obstacles are other vehicles or stationary installed constructions that block the shortest path between the robot and its goal. There is a lot of literature on obstacle avoidance that is not specific for snake robots, as can be seen for instance in the reviews on mobile robots [37], unmanned aerial vehicles [4], or autonomous ships [102].

For underwater snake robots and fish robots, obstacle avoidance strategies have been presented in [39, 81, 86]. The work in [81] presents experimental results for obstacle recognition and avoidance for a fish-like robot using neural networks. The focus of the work in [81] lies mostly on obstacle recognition and avoidance and less on following a predefined path. Similarly, another study in [86] uses an electric exteroceptive feedback loop in a fish-like robot to detect and avoid obstacles. Again, also [86] focuses only on the obstacle avoidance rather than path-following. Another approach is presented in [39] for underwater snake robots, where an artificial potential field is used to plan a safe path towards a target within a cluttered environment. While [81, 86] consider mainly local obstacle detection, the study in [39] requires a global map of the obstacle locations a priori.

For terrestrial snake robots, obstacle avoidance strategies have been considered in [32, 72, 105]. In [32], a control strategy for a snake robot with passive wheels is presented, where obstacles in the way of the robot are avoided by altering the sinusoidal shape that the robot adapts to. This approach considers local avoidance of obstacles, but no path-following strategy is proposed. A similar strategy for moving obstacles is proposed and experimentally validated in [105], where a robot

is remotely controlled to move forward and avoids obstacles on the way by lifting links that are about to collide sideways. In [72], a method for both trajectory planning and obstacle avoidance is proposed. However, the method in [72] is for snake robots with active wheels, and thus not applicable for underwater snake robots. In contrast to these approaches, it has been shown that instead of avoiding obstacles, terrestrial snake robots can actually use them for forward propulsion, a strategy called obstacle-aided locomotion [65, 108]. This concept is, however, only applicable when there are sufficiently many obstacles to use as push points, and less suitable if the robot needs to travel through an environment with only a few obstacles.

1.3 Contributions and organization of this thesis

1.3.1 Scope

The scope of this thesis entitled "Guidance and Control of Underwater Snake Robots Using Planar Sinusoidal Gaits" can be narrowed down within the two categories "Guidance and Control" and "Planar Sinusoidal Gaits".

Guidance and control

This thesis is concerned with the theoretical development of guidance and motion control methods for underwater snake robots. More specifically, guidance laws for path-following and manoeuvring in the presence of unknown environmental disturbances and obstacle avoidance are proposed, as well as model-based control laws that ensure that the robot achieves the desired tasks. The stability of the proposed control systems is analysed theoretically and the performance is tested in simulations and/or experiments. The thesis does not focus on challenges related to aspects like the use of sensor data or limited availability of energy or communication range. For the development of feedback laws in this thesis it will therefore be assumed that sufficiently accurate measurements are available. We do furthermore not consider the energy efficiency of the proposed approaches.

The mathematical models that are used in this thesis are intended for control design, which requires a model structure that is easy to implement and analyse. The fluid effects that act on the robot are therefore modelled in a simplified way. In particular, the robot is assumed to be neutrally buoyant, the environmental forces are described by a fluid drag model, and environmental disturbances are represented by constant, irrotational currents. Furthermore, we only consider scenarios where effects like waves, lift forces, boundary layer effects close to the sea bottom, or vortices and turbulence do not need to be taken into account.

Planar sinusoidal gaits

This thesis considers underwater snake robots that propel themselves with periodic, planar bio-inspired gaits of a sinusoidal shape. There are alternative and complementary ways of achieving propulsion, like thrusters or biologically inspired features like fins, that can be attached to the robot in order to achieve faster or

more efficient propulsion. Control algorithms for robots with attributes of this sort are not taken into account in this thesis, although the proposed controllers may still be able to steer such robots to a certain degree, as long as they move slowly and use sinusoidal gaits. However, for faster motion with thrusters, fins, or alternative gaits like flapping gaits, added mass effects and lift forces should be taken into account in the models. Another alternative are three-dimensional gaits, which are achieved by adding a vertical motion component to the horizontal one that is already contained in the planar gait. Such three-dimensional gaits are not fundamentally different from the planar ones we consider in this thesis, since they are simply a superposition of two planar waves. We believe that the fundamental principles of such locomotion can be understood by using a planar perspective.

1.3.2 Contributions and outline

In this thesis, different approaches are developed for guidance and motion control of underwater snake robots. As a basis for this, underwater snake robot locomotion based on sinusoidal gaits is analysed, and a control-oriented model that captures the most important effects is developed. The proposed approaches include control systems for heading or orientation control and velocity control, path-following controllers based on closed-loop heading control and open-loop velocity control by using a predefined biologically inspired gait, manoeuvring control for path-following with a desired velocity, and a guidance strategy for path-following with obstacle avoidance. The stability properties of the proposed control systems are analysed theoretically and their performance is tested in simulations and/or experiments. The particular topics, contributions and methods of the single chapters are summarized in Figure 1.4 and outlined in the following.

Chapter	Modelling and locomotion analysis	Guidance	Model-based control	Theoretical analysis	Simulation study	Experimental study
2	✓		✓	✓	✓	
3		✓	✓	✓	✓	✓
4		✓	✓		✓	✓
5			✓	✓	✓	
6		✓		✓		✓

Figure 1.4: Topics and methods of the single chapters.

Chapter 2

This chapter lays a foundation for the control design in the following chapters by introducing two models for underwater snake robots. In addition, the type of snake robot locomotion that this thesis focusses on is analysed, revealing some fundamental properties and insights. While the chapter is used to prepare the reader for the remainder of this thesis by introducing the basic notation and reviewing a model from the literature, it also contains several new contributions.

At first, a simplified version of the complex model for underwater snake robots in [45] is introduced. The propulsive forces acting on the complex model during

sinusoidal locomotion in the presence of ocean currents are analysed and some fundamental locomotion properties are presented. These properties validate the simplification in the complex model and serve as a basis for the derivation of the second model. As a second contribution, a hydrodynamic model is derived to be used in combination with the simplified control-oriented modelling approach from [65]. As opposed to a similar, previous model for underwater snake robots from [40], the new model takes into account ocean currents. The comparison of the two different modelling approaches from [40, 65] is extended in this chapter by deriving mathematical relations for mapping the joint amplitudes of the different models. These relations enable a simulation study, in which the two models are compared a lot more extensively than the previous models for terrestrial robots and underwater robots that are not exposed to currents. Finally, the velocity dynamics of an underwater snake robot exposed to ocean currents are analysed using averaging theory, which gives relationships between the gait parameters and the steady-state velocity.

Chapter 3

This chapter presents a model-based control approach for straight line path-following of underwater snake robots. We develop a control system based on a cascaded design, that enables the robot to converge to and follow a planar straight path in the presence of ocean currents.

A LOS guidance law is employed in the outer control loop in order to provide an orientation reference for the robot. Ocean currents are accounted for in the guidance scheme by augmenting it with integral action in order to compensate for the steady state error. An orientation controller is designed that exponentially stabilizes the orientation of the robot towards the reference angle given by the integral LOS guidance law. The inner control loop is closed with a feedback-linearizing joint controller, which enforces a sinusoidal gait and thus guarantees forward propulsion of the robot. Unlike previous model-based methods for terrestrial snake robots [63, 65], this control approach is able to handle the disturbance produced by an ocean current. The second contribution of this chapter compared to previous work [43] is the stability analysis. Using cascaded systems theory, it is formally proved that under the assumption of a constant positive forward velocity, the cross-track error between the robot and the desired path is guaranteed to converge to zero. The control system will thus fulfil the control objective. Finally, we present simulations and an experimental study that validate the theoretical findings.

Chapter 4

This chapter investigates the problem of planar manoeuvring control for biologically inspired underwater snake robots that are exposed to unknown ocean currents. The control objective is to make a snake robot converge to a desired planar path and traverse the path with a desired velocity.

We propose a feedback control strategy that enforces VHCs, which encode biologically inspired gaits, on the snake robot configuration. The virtual constraints, parametrized by the states of two dynamic compensators, are used to regulate the

heading and forward velocity of the snake robot. The control framework is an extension of the work that was developed for terrestrial snake robots in [77, 78] to underwater snake robots under the influence of hydrodynamic forces and ocean currents. The analysis shows that the control scheme works for the underwater snakes as it is, without significantly changing the controllers, as long as the ocean currents are compensated for. To this end, we design a two-state ocean current observer based on relative velocity sensors. It enables the robot to follow the path in the presence of unknown constant ocean currents. In addition to the lateral undulatory gait that was investigated in [77, 78], it is shown in this chapter that the control framework can also be used along with a more general class of biologically inspired gaits. The efficacy of the proposed control algorithm for several biologically inspired gaits is verified both in simulations for different path geometries and in experiments.

Chapter 5

In this chapter, we take a first step towards the control of underwater snake robots with absolute velocity measurements as feedback. We present a control system for velocity and orientation control of underwater snake robots exposed to ocean currents. The VHC approach has the advantage that it makes the control design amenable to a hierarchical synthesis, where the gait is enforced at the lowest level and velocity and orientation control are done at a higher level. To this end, we employ the hierarchical control design methodology from [20], that was used for control design for ships in [9] and terrestrial snake robots in [78]. In order to make the feedback independent of the unknown relative velocities that enter the dynamic equations via the fluid drag forces, we make use of adaptive backstepping control [56].

More specifically, we propose a joint controller that asymptotically stabilizes the VHCs with absolute velocity measurements available for feedback, as opposed to the joint controller design based on relative velocity measurements in Chapter 3. The gait that is encoded in the VHCs is modified in this chapter to take into account actuator constraints. The velocity controller and the orientation controller are designed subsequently. In doing so, we prioritize the velocity control higher than in previous approaches for terrestrial snake robots, a change of paradigm that removes a singularity from the orientation controller that was an issue in the design for terrestrial robots [90, 91]. The velocity controller is designed along the lines of adaptive backstepping. The design is not a straightforward application of the method, because the system is non-autonomous and unknown terms enter the equations at every stage of the backstepping procedure. By postponing the design of the adaptive update law until the last step, we are able to compensate for the effect of these signals in the closed-loop system, and show asymptotic stability of the origin of the velocity error dynamics. In the last step, the orientation controller is designed as a feedback-linearizing controller utilizing the adaptive term of the velocity controller as a current estimate. We show that the zero dynamics of the closed-loop system is bounded for the proposed controller. Simulation results of a snake robot that is exposed to an unknown ocean current are presented.

Chapter 6

The set-based path-following and obstacle avoidance guidance that we propose in this chapter is motivated by the set-based guidance strategy for autonomous ships that was recently proposed in [73]. The strategy is based on the results for set-based control in [75], which facilitate a theoretical stability analysis. It is shown in [73] that the strategy guarantees obstacle avoidance, which is illustrated with simulations, but has not been tested experimentally yet. The main idea is to make the robot converge to and follow a straight path and only leave it in order to circumvent an obstacle that is in the way. In that case, the guidance switches into obstacle avoidance mode, and the robot follows a circle around the obstacle. As soon as the obstacle is passed, the robot converges to the original straight path again.

In this chapter, the switching conditions are reformulated in a more general, purely kinematic manner. This makes the switching strategy independent of the dynamic model, and thus applicable to a more general class of systems, including snake robots. Furthermore, the new switching strategy is combined with a general guidance law for snake robots from [78]. This guidance controller is suitable for generic paths, and can thus be applied both for the straight line path-following mode, and the obstacle avoidance mode, which requires a circular reference path. In [78], the guidance law is shown to practically stabilize the robot to the path. In this chapter, the guidance law is modified in order to allow for circular path-following in both directions, thus enabling the robot to choose the shortest way around the obstacles blocking its path. We show that the modifications to the guidance law still preserve the stability properties. Finally, the obstacle avoidance strategy is experimentally tested with a swimming snake robot for different stationary obstacles. The results show the efficacy of the proposed set-based path-following and obstacle avoidance guidance scheme, and are the first experimental validation of the set-based control strategy with a floating-base robot.

Chapter 7

This chapter presents concluding remarks and suggestions for future research.

1.3.3 Underlying publications

The underlying studies of this thesis have resulted in the following list of publications. It contains publications in journals, a book chapter, and several peer-reviewed conference papers. Some publications that are related to this thesis, but not a part of it, are also included in the list.

Journal papers and book chapter

- 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.

- A. M. Kohl, E. Kelasidi, A. Mohammadi, M. Maggiore, and K. Y. Pettersen. Planar maneuvering control of underwater snake robots using virtual holonomic constraints. *Bioinspiration & Biomimetics*, 11(6):065005, 2016.
- A. M. Kohl, E. Kelasidi, K. Y. Pettersen, and J. T. Gravdahl. Model-Based LOS Path-Following Control of Planar Underwater Snake Robots. In *Sensing and Control for Autonomous Vehicles – Applications to Land, Water and Air Vehicles*, Lecture Notes in Control and Information Sciences 474, eds. Thor I. Fossen, Kristin Y. Pettersen, and Henk Nijmeijer, pp. 43–363, Springer International Publishing 2017.

Peer-reviewed conference papers

- A. M. Kohl, E. Kelasidi, K. Y. Pettersen, and J. T. Gravdahl. A control-oriented model of underwater snake robots exposed to currents. In *Proceedings of the 2015 IEEE Conference on Control Applications (CCA)*, Sydney, Australia, September 2015.
- A. M. Kohl, K. Y. Pettersen, E. Kelasidi, and J. T. Gravdahl. Analysis of underwater snake robot locomotion based on a control-oriented model. In *Proceedings of the 2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Zhuhai, China, December 2015.
- A. M. Kohl, K. Y. Pettersen, and J. T. Gravdahl. Velocity and orientation control of underwater snake robots using absolute velocity feedback. *Proceedings of the 2017 IEEE Conference on Control Applications (CCTA)*, Kohala Coast, Hawaii, USA, August 2017.
- A. M. Kohl, S. Moe, E. Kelasidi, K. Y. Pettersen, and J. T. Gravdahl. Set-based path following and obstacle avoidance for underwater snake robots. *Accepted for the 2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Macau, China, December 2017.

Publications not part of this thesis

- E. Kelasidi, A. M. Kohl, K. Y. Pettersen, and J. T. Gravdahl. Waypoint guidance control for underwater snake robots exposed to ocean currents. In *Proceedings of the 24th Mediterranean Conference on Control and Automation (MED)*, Athens, Greece, June 2016.
- E. Kelasidi, K. Y. Pettersen, A. M. Kohl, and J. T. Gravdahl. An Experimental Investigation of Path Following for an Underwater Snake Robot with a Caudal Fin. *Proceedings of the 20st IFAC World Congress*, Toulouse, France, July 2017.

Chapter 2

Modelling and Analysis of Underwater Snake Robot Locomotion

This chapter presents two models that are used for control design in this thesis. The first model is based on first principles and analytical fluid-dynamics. This model will be referred to as the *complex* model in the following. Based on this model, fundamental properties of locomotion with planar sinusoidal gaits are derived. The second model is derived from the complex model by utilizing an observation about snake robot locomotion to make a simplifying assumption regarding the joint motion. This model will be denoted the *simplified* or *control-oriented* model. Based on this model, the velocity dynamics of snake robot locomotion is analysed using averaging theory.

Contributions of this chapter The first contribution of this chapter is to review the complex model for underwater snake robots in [45], and an adaptation of the notation to be consistent with the notation used in [65] for terrestrial snake robots. In addition, the model is analysed during sinusoidal motion in the presence of ocean currents and some fundamental locomotion properties are summarized. As a second contribution, a hydrodynamic model is derived to be used in combination with the simplified control-oriented modelling approach from [65]. As opposed to a previous model for underwater snake robots from [40], which used the same approach, the model in this chapter takes into account ocean currents. Furthermore, the comparison of the two different modelling approaches from [40, 65] is extended in this chapter by deriving mathematical relations for mapping the joint amplitudes of the different models. To this end, an extensive simulation study is performed, in which the two models are compared. As a last contribution, the velocity dynamics of an underwater snake robot exposed to ocean currents are analysed using averaging theory, which gives relationships between the gait parameters and the steady-state velocity.

Organization of this chapter This chapter is structured as follows. Some basic notation that will be used throughout this thesis is introduced in Section 2.1. Section 2.2 presents the complex model for underwater snake robots. Some fundamental properties of locomotion with sinusoidal gaits are given in Section 2.3. The control-oriented model is derived in Section 2.4. Finally, the averaging analysis of the velocity dynamics during sinusoidal gaits is presented in Section 2.5 and the chapter is summarized in Section 2.6.

Publications This chapter is based on [49, 50].

2.1 Basic notation

Throughout this thesis, the following vectors and matrices are used:

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

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

$$\mathbf{I}_N = \begin{bmatrix} 1 & & & 0 \\ & 1 & & \\ & & \dots & \\ 0 & & & 1 \end{bmatrix} \in \mathbb{R}^{N\times N}, \qquad \mathbf{E} = \begin{bmatrix} \mathbf{e} & \mathbf{0}_{N\times 1} \\ \mathbf{0}_{N\times 1} & \mathbf{e} \end{bmatrix} \in \mathbb{R}^{2N\times 2},$$

$$\mathbf{V} = \mathbf{A}^T(\mathbf{D}\mathbf{D}^T)^{-1}\mathbf{A} \in \mathbb{R}^{N\times N}, \qquad \mathbf{K} = \mathbf{A}^T(\mathbf{D}\mathbf{D}^T)^{-1}\mathbf{D} \in \mathbb{R}^{N\times N}.$$

The matrices $\mathbf{A}$ and $\mathbf{D}$ add and subtract adjacent elements of a vector, respectively. The matrix $\mathbf{I}_N$ is the unity matrix and the vectors $\mathbf{e}$ and $\bar{\mathbf{e}}$ as well as the matrix $\mathbf{E}$ are summation operators. We will also make use of the pseudo-inverse $\bar{\mathbf{D}} = \mathbf{D}^T\left(\mathbf{D}\mathbf{D}^T\right)^{-1} \in \mathbb{R}^{N\times(N-1)}$. Furthermore, attributes of the single robot joints and links, like for example the link orientation, will often be assembled into a vector like $\boldsymbol{\theta} = [\theta_1, \dots, \theta_N]^T \in \mathbb{R}^N$. The following operators are defined for the vector $\boldsymbol{\theta} \in \mathbb{R}^N$ with the operator $\mathrm{diag}(\cdot)$ spanning a diagonal matrix:

$$\sin(\boldsymbol{\theta}) = \begin{bmatrix} \sin\theta_1 & \dots & \sin\theta_N \end{bmatrix}^T \in \mathbb{R}^N, \qquad \mathbf{S}_{\boldsymbol{\theta}} = \mathrm{diag}(\sin(\boldsymbol{\theta})) \in \mathbb{R}^{N\times N},$$

$$\cos(\boldsymbol{\theta}) = \begin{bmatrix} \cos\theta_1 & \dots & \cos\theta_N \end{bmatrix}^T \in \mathbb{R}^N, \qquad \mathbf{C}_{\boldsymbol{\theta}} = \mathrm{diag}(\cos(\boldsymbol{\theta})) \in \mathbb{R}^{N\times N},$$

$$\mathrm{sgn}(\boldsymbol{\theta}) = \begin{bmatrix} \mathrm{sgn}(\theta_1) & \dots & \mathrm{sgn}(\theta_N) \end{bmatrix}^T \in \mathbb{R}^N, \qquad \boldsymbol{\theta}^2 = \begin{bmatrix} \theta_1^2 & \dots & \theta_N^2 \end{bmatrix}^T \in \mathbb{R}^N.$$

Finally, we define the rotation matrix $\mathbf{R}_\theta = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \in \mathbb{R}^{2\times 2}$.

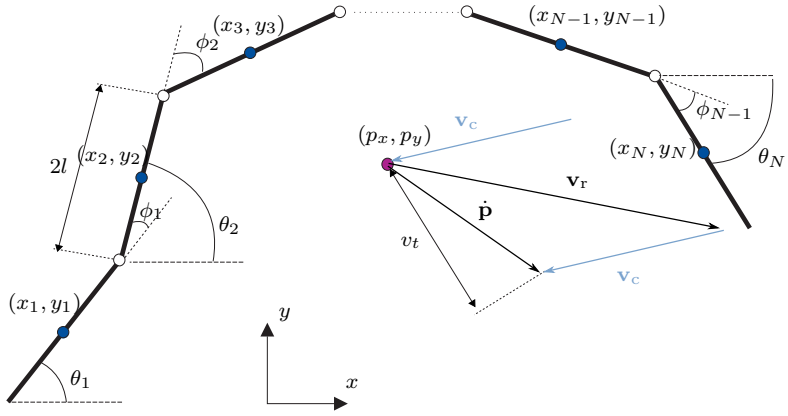


Figure 2.1: The kinematics of the underwater snake robot.

2.2 Modelling of underwater snake robots

This section presents a first-principle model of a neutrally buoyant underwater snake robot. The topic of this thesis are underwater snake robots that employ planar sinusoidal gaits for propulsion. This type of locomotion, together with the neutral buoyancy of the robot, makes it possible to capture the behaviour of the robot by considering a purely two-dimensional model similar to the model of a terrestrial snake robot presented in [65]. The only thing that differs in the underwater model is the model of the external forces, which are now obtained from a hydrodynamic model instead of a friction model.

The complex model that is presented in this section is not a novel contribution of this work, but included for completeness. The two-dimensional kinematics of the snake robot are based on [65], where a new notation of an earlier model from [95] was introduced. The hydrodynamic model used in this thesis is a simplified version of the model presented in [45]. In this thesis, however, the notation for the hydrodynamic model is changed with respect to [45] in order to be consistent with the notation in [65]. In the remainder of this thesis, the model presented here will be referred to as the *complex model*, as opposed to the *simplified* or *control-oriented model* that will be derived in Section 2.4.

2.2.1 The parameters and kinematics of the robot

The robot consists of N identical rigid links that are serially connected by $N-1$ revolute, motorized joints. All links have the same length $2l$, uniformly distributed mass m, and moment of inertia J. The total mass of the robot adds up to Nm, and it is assumed that the robot is neutrally buoyant. The joints are motorized and frictionless. The robot is furthermore exposed to a constant ocean current $\mathbf{v}_c = [V_x, V_y]^T$.

A snake robot is a kinematic chain without a fixed base. Figure 2.1 illustrates the following kinematic definitions.

Definition 1: The orientation of each individual link i is denoted by $\theta_i,\ i = 1, \ldots, N$ and is defined as the angle between the global axis and link i with counter-clockwise positive direction. The link angles are assembled into the vector $\boldsymbol{\theta} = [\theta_1, \ldots, \theta_N]^T \in \mathbb{R}^N$.

Definition 2: The joint angles between adjacent links are denoted by ϕ_i and are defined as

$$\phi_i = \theta_{i+1} - \theta_i, \quad i = 1, \ldots N-1. \tag{2.1}$$

The joint angles are assembled into the vector $\boldsymbol{\phi} = [\phi_1, \ldots, \phi_{N-1}]^T \in \mathbb{R}^{N-1}$.

Remark 2.1. Note that in this thesis we follow the joint angle convention from [68] instead of the convention from [95] that was used in [65]. This choice makes sure that all angles are defined with counter-clockwise positive direction.

Furthermore, the position of the CM of link i is given by (x_i, y_i), which are assembled into the vectors $\mathbf{X} = [x_1, \ldots, x_N]^T \in \mathbb{R}^N$ and $\mathbf{Y} = [y_1, \ldots, y_N]^T \in \mathbb{R}^N$. The position of the CM of the robot is then obtained by

$$\mathbf{p} = \begin{bmatrix} p_x \\ p_y \end{bmatrix} = \frac{1}{N} \begin{bmatrix} \mathbf{e}^T\mathbf{X} \\ \mathbf{e}^T\mathbf{Y} \end{bmatrix}. \tag{2.2}$$

The vector $\mathbf{e}$ is a summation operator and was defined in Section 2.1.

A snake robot has no clearly defined heading or orientation, since each link may have a different orientation at all time. There are different ways of circumventing this problem, for instance the virtual chassis approach in [94], where the orientation is defined based on the singular value decomposition of the positions of the links with respect to the CM of the robot. Throughout this thesis, we will make use of two different ways to obtain a measure of the heading or orientation of the robot. The first way is motivated by [65], where the average of all link angles is used as a measure of the robot orientation.

Definition 3: The (average) orientation of the snake robot is given by $\bar{\theta} \in \mathbb{R}$ and is defined as the average of all link angles:

$$\bar{\theta} = \frac{1}{N} \sum_{i=1}^{N} \theta_i. \tag{2.3}$$

The second way follows the approach in [78], where the angle of the head link, θ_N, is considered as the defining angle.

Definition 4: The heading of the robot is defined as the angle of the head link, θ_N.

In order to avoid ambiguity in this thesis, the angle θ_N is referred to as *heading* of the robot, whereas $\bar{\theta}$ is referred to as *(average) orientation* of the robot.

Remark 2.2. The approaches in Definition 3 and Definition 4 have different purposes. Using the average orientation of the robot considers the system as a whole vehicle, whereas the heading according to Definition 4 facilitates control design where the robot head is of special importance, for instance if it is equipped with sensors or tools.

The velocity of the CM of the robot in the inertial frame is given by $\dot{\mathbf{p}} = [\dot{p}_x, \dot{p}_y]^T$. In accordance with Definitions 3 and 4 and making use of the rotation matrix

$$\mathbf{R}_\theta = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \in \mathbb{R}^2 \tag{2.4}$$

for some $\theta \in \mathbb{R}$, we define the following velocities:

Definition 5: The velocity components of the CM that are tangential and normal with respect to the heading of the robot are denoted by v_t and v_n, respectively, and are defined as

$$\begin{bmatrix} v_t \\ v_n \end{bmatrix} = \mathbf{R}_{\theta_N}^T \dot{\mathbf{p}}. \tag{2.5}$$

Finally, the velocities of the single links are given by [65]:

$$\begin{aligned} \dot{\mathbf{X}} &= l\mathbf{K}^T \mathbf{S}_{\boldsymbol{\theta}} \dot{\boldsymbol{\theta}} + \mathbf{e}\dot{p}_x, \\ \dot{\mathbf{Y}} &= -l\mathbf{K}^T \mathbf{C}_{\boldsymbol{\theta}} \dot{\boldsymbol{\theta}} + \mathbf{e}\dot{p}_y. \end{aligned} \tag{2.6}$$

The hydrodynamic effects that will be described in the next section depend on the velocity of the robot with respect to the surrounding fluid, i. e. the *relative* velocity, as opposed to the *absolute* velocity given by $\dot{\mathbf{p}}$ and the velocities defined in (2.5). The relative velocity of the CM of the robot in the inertial frame is denoted by $\mathbf{v}_\mathrm{r}$ and obtained by

$$\mathbf{v}_\mathrm{r} = \dot{\mathbf{p}} - \mathbf{v}_\mathrm{c} = \begin{bmatrix} \dot{p}_x - V_x \\ \dot{p}_y - V_y \end{bmatrix}, \tag{2.7}$$

and the relative velocity components of the CM that are tangential and normal with respect to the heading of the robot are denoted by $\mathbf{v}_\mathrm{rel} = \mathbf{R}_{\theta_N}^T(\dot{\mathbf{p}} - \mathbf{v}_\mathrm{c})$.

2.2.2 The hydrodynamic model

In analogy with the assumptions in [45], the following assumptions are made with regards to the fluid effects on the robot:

Assumption 1: The fluid is viscid, incompressible, and irrotational in the inertial frame.

Assumption 2: The ocean current, $\mathbf{v}_\mathrm{c} = [V_x, V_y]^T$, is constant in the inertial frame and irrotational. Its magnitude $V_\mathrm{c} = \sqrt{V_x^2 + V_y^2}$ is bounded by $V_{\mathrm{c,max}} \geq V_\mathrm{c}$.

According to [45], we can represent the hydrodynamic model by fluid drag forces, added mass forces, resistive torques, and added inertia torques. The drag

forces acting on link i depend on the relative velocity of the CM of the link and are given by

$$f_{\mathrm{D},i}^{\mathrm{I}} = -\mathbf{R}_{\theta_i} \begin{bmatrix} c_t & 0 \\ 0 & c_n \end{bmatrix} \mathbf{R}_{\theta_i}^T \begin{bmatrix} \dot{x}_i - V_x \\ \dot{y}_i - V_y \end{bmatrix}, \tag{2.8a}$$

$$f_{\mathrm{D},i}^{\mathrm{II}} = -\mathbf{R}_{\theta_i} \begin{bmatrix} c_t & 0 \\ 0 & c_n \end{bmatrix} \mathrm{diag}\left(\mathrm{sgn}\left(\mathbf{R}_{\theta_i}^T \begin{bmatrix} \dot{x}_i - V_x \\ \dot{y}_i - V_y \end{bmatrix}\right)\right)\left(\mathbf{R}_{\theta_i}^T \begin{bmatrix} \dot{x}_i - V_x \\ \dot{y}_i - V_y \end{bmatrix}\right)^2. \tag{2.8b}$$

The added mass forces that act on link i depend on the relative acceleration of the CM of the link [47] and are obtained as

$$f_{\mathrm{A},i} = -\mathbf{R}_{\theta_i} \begin{bmatrix} 0 & 0 \\ 0 & \mu_n \end{bmatrix} \frac{\mathrm{d}}{\mathrm{d}t}\left(\mathbf{R}_{\theta_i}^T \begin{bmatrix} \dot{x}_i - V_x \\ \dot{y}_i - V_y \end{bmatrix}\right). \tag{2.9}$$

The hydrodynamic torques acting on link i are

$$\tau_{\mathrm{R},i} = -c_\theta^{\mathrm{I}} \dot{\theta}_i - c_\theta^{\mathrm{II}} \mathrm{sgn}(\dot{\theta}_i)\dot{\theta}_i^2 \tag{2.10}$$

and

$$\tau_{\mathrm{A},i} = -\mu_\theta \ddot{\theta}_i. \tag{2.11}$$

In (2.8) and (2.10), c_t and c_n are the drag parameters of the links in tangential and normal direction, respectively, and c_θ^{I} and c_θ^{II} are rotational damping parameters. The added mass parameters are μ_n in (2.9) and μ_θ in (2.11). Their numerical values depend on the fluid properties and the link geometry. For details on their derivation, the reader is referred to [45]. An important property for snake robot locomotion is to have a higher resistive force normal to the links than along the links [65]. This property can be expressed by the equation

$$c_n > c_t \tag{2.12}$$

and is an important prerequisite for propulsion with sinusoidal gaits [65]. At lower velocities, the linear drag forces $f_{\mathrm{D},i}^{\mathrm{I}}$ in (2.8a) are the dominant forces, while the nonlinear drag forces $f_{\mathrm{D},i}^{\mathrm{II}}$ in (2.8b) become dominant at higher velocities. The single forces and torques can be assembled into matrix form:

$$\mathbf{f}_{\mathrm{D}} = \begin{bmatrix} \mathbf{f}_{\mathrm{D},x} \\ \mathbf{f}_{\mathrm{D},y} \end{bmatrix} = \mathbf{f}_{\mathrm{D}}^{\mathrm{I}} + \mathbf{f}_{\mathrm{D}}^{\mathrm{II}} \in \mathbb{R}^{2N}, \tag{2.13a}$$

$$\mathbf{f}_{\mathrm{D}}^{\mathrm{I}} = -\begin{bmatrix} c_t(\mathbf{C}_{\boldsymbol{\theta}})^2 + c_n(\mathbf{S}_{\boldsymbol{\theta}})^2 & (c_t - c_n)\mathbf{S}_{\boldsymbol{\theta}}\mathbf{C}_{\boldsymbol{\theta}} \\ (c_t - c_n)\mathbf{S}_{\boldsymbol{\theta}}\mathbf{C}_{\boldsymbol{\theta}} & c_t(\mathbf{S}_{\boldsymbol{\theta}})^2 + c_n(\mathbf{C}_{\boldsymbol{\theta}})^2 \end{bmatrix} \begin{bmatrix} \dot{\mathbf{X}} - \mathbf{e}V_x \\ \dot{\mathbf{Y}} - \mathbf{e}V_y \end{bmatrix}, \tag{2.13b}$$

$$\begin{aligned} \mathbf{f}_{\mathrm{D}}^{\mathrm{II}} = -\begin{bmatrix} c_t\mathbf{C}_{\boldsymbol{\theta}} & -c_n\mathbf{S}_{\boldsymbol{\theta}} \\ c_t\mathbf{S}_{\boldsymbol{\theta}} & c_n\mathbf{C}_{\boldsymbol{\theta}} \end{bmatrix} \mathrm{diag}\left(\mathrm{sgn}\left(\begin{bmatrix} \mathbf{C}_{\boldsymbol{\theta}} & \mathbf{S}_{\boldsymbol{\theta}} \\ -\mathbf{S}_{\boldsymbol{\theta}} & \mathbf{C}_{\boldsymbol{\theta}} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{X}} - \mathbf{e}V_x \\ \dot{\mathbf{Y}} - \mathbf{e}V_y \end{bmatrix}\right)\right) \\ \left(\begin{bmatrix} \mathbf{C}_{\boldsymbol{\theta}} & \mathbf{S}_{\boldsymbol{\theta}} \\ -\mathbf{S}_{\boldsymbol{\theta}} & \mathbf{C}_{\boldsymbol{\theta}} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{X}} - \mathbf{e}V_x \\ \dot{\mathbf{Y}} - \mathbf{e}V_y \end{bmatrix}\right)^2, \end{aligned} \tag{2.13c}$$

$$\mathbf{f}_{\mathrm{A}} = \begin{bmatrix} \mathbf{f}_{\mathrm{A},x} \\ \mathbf{f}_{\mathrm{A},y} \end{bmatrix} = -\begin{bmatrix} 0 & -\mu_n\mathbf{S}_{\boldsymbol{\theta}} \\ 0 & \mu_n\mathbf{C}_{\boldsymbol{\theta}} \end{bmatrix} \frac{\mathrm{d}}{\mathrm{d}t}\left(\begin{bmatrix} \mathbf{C}_{\boldsymbol{\theta}} & \mathbf{S}_{\boldsymbol{\theta}} \\ -\mathbf{S}_{\boldsymbol{\theta}} & \mathbf{C}_{\boldsymbol{\theta}} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{X}} - \mathbf{e}V_x \\ \dot{\mathbf{Y}} - \mathbf{e}V_y \end{bmatrix}\right) \in \mathbb{R}^{2N}, \tag{2.14}$$

Table 2.1: The parameters of the underwater snake robot.

Symbol	Description
N	Number of links
l	Half the length of a link
m	Mass of a link
J	Moment of inertia of a link
c_t	Tangential drag parameter
c_n	Normal drag parameter
c_θ^{I}	Linear rotational damping parameter
c_θ^{II}	Nonlinear rotational damping parameter
V_x	x-component of the ocean current
V_y	y-component of the ocean current

with

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\left(\begin{bmatrix}\mathbf{C}_{\boldsymbol{\theta}} & \mathbf{S}_{\boldsymbol{\theta}}\\ -\mathbf{S}_{\boldsymbol{\theta}} & \mathbf{C}_{\boldsymbol{\theta}}\end{bmatrix}\begin{bmatrix}\dot{\mathbf{X}}-\mathbf{e}V_x\\ \dot{\mathbf{Y}}-\mathbf{e}V_y\end{bmatrix}\right)\\ =\begin{bmatrix}\mathbf{C}_{\boldsymbol{\theta}} & \mathbf{S}_{\boldsymbol{\theta}}\\ -\mathbf{S}_{\boldsymbol{\theta}} & \mathbf{C}_{\boldsymbol{\theta}}\end{bmatrix}\begin{bmatrix}\ddot{\mathbf{X}}\\ \ddot{\mathbf{Y}}\end{bmatrix}+\begin{bmatrix}-\mathbf{S}_{\boldsymbol{\theta}} & \mathbf{C}_{\boldsymbol{\theta}}\\ -\mathbf{C}_{\boldsymbol{\theta}} & -\mathbf{S}_{\boldsymbol{\theta}}\end{bmatrix}\begin{bmatrix}\mathrm{diag}(\dot{\boldsymbol{\theta}})(\dot{\mathbf{X}}-\mathbf{e}V_x)\\ \mathrm{diag}(\dot{\boldsymbol{\theta}})(\dot{\mathbf{Y}}-\mathbf{e}V_y)\end{bmatrix},\end{aligned} \tag{2.15}$$

and

$$\boldsymbol{\tau}_{\mathrm{R}} = -c_\theta^{\mathrm{I}}\dot{\boldsymbol{\theta}} - c_\theta^{\mathrm{II}}\mathrm{diag}\big(\mathrm{sgn}\big(\dot{\boldsymbol{\theta}}\big)\big)\dot{\boldsymbol{\theta}}^2 \in \mathbb{R}^N, \tag{2.16}$$

$$\boldsymbol{\tau}_{\mathrm{A}} = -\mu_\theta\ddot{\boldsymbol{\theta}} \in \mathbb{R}^N. \tag{2.17}$$

For the control design in this thesis, we will consider only slowly moving underwater snake robots, where hydrodynamic damping dominates [22]. The primary motivation for the models presented in this chapter is to serve as a basis for control design. It is therefore considered less important to model the hydrodynamic effects quantitatively accurately, but rather to capture the qualitative behaviour of the system. We therefore make an additional simplifying assumption regarding the hydrodynamic effects:

Assumption 3: Added mass effects can be disregarded such that the effect of the fluid on a robot link is completely described by the linear and nonlinear drag forces on the CM of the link (2.13) and the resistive fluid torque (2.16).

Remark 2.3. As was pointed out in [17], simplifications of this kind are often made in models of bio-inspired robots that may be used for motion planning and real-time control, see for instance [10, 70]. In Section 2.3 we will see an analysis of the propulsive forces that provides additional reasons to disregard added mass effects during sinusoidal locomotion.

2.2.3 The dynamic model

Table 2.1 summarizes the parameters of the robot and the surrounding fluid. The robot has $N+2$ degrees of freedom, that are defined by the link angles $\boldsymbol{\theta} \in \mathbb{R}^N$

and the CM position $\mathbf{p} \in \mathbb{R}^2$. The equations of motion of the robot are formulated in terms of the angular acceleration of the links, $\ddot{\boldsymbol{\theta}}$, and the acceleration of the CM, $\ddot{\mathbf{p}}$. They can be found by using first principles, formulating the force and torque balance for each link, assembling the expressions into matrix form, and solving for $\ddot{\boldsymbol{\theta}}$ and $\ddot{\mathbf{p}}$. With Assumption 3, the derivation of the dynamic equations is identical to the derivation in [65], except that the ground friction forces are replaced by the fluid drag forces in (2.13), and the resistive fluid torque (2.16) has to be taken into account additionally. The dynamic model of the underwater snake robot is then obtained by

$$\mathbf{M}_{\boldsymbol{\theta}}\ddot{\boldsymbol{\theta}} + \mathbf{W}_{\boldsymbol{\theta}}\dot{\boldsymbol{\theta}}^2 - \boldsymbol{\tau}_{\mathrm{R}} - l\mathbf{S}_{\boldsymbol{\theta}}\mathbf{K}\mathbf{f}_{\mathrm{D},x} + l\mathbf{C}_{\boldsymbol{\theta}}\mathbf{K}\mathbf{f}_{\mathrm{D},y} = -\mathbf{D}^T\mathbf{u}, \tag{2.18a}$$

$$Nm\ddot{\mathbf{p}} = \mathbf{E}^T\mathbf{f}_{\mathrm{D}}, \tag{2.18b}$$

where $\mathbf{M}_{\boldsymbol{\theta}} = J\mathbf{I}_N + ml^2\mathbf{S}_{\boldsymbol{\theta}}\mathbf{V}\mathbf{S}_{\boldsymbol{\theta}} + ml^2\mathbf{C}_{\boldsymbol{\theta}}\mathbf{V}\mathbf{C}_{\boldsymbol{\theta}}$ is the rotational inertia matrix, $\mathbf{W}_{\boldsymbol{\theta}} = ml^2\mathbf{S}_{\boldsymbol{\theta}}\mathbf{V}\mathbf{C}_{\boldsymbol{\theta}} - ml^2\mathbf{C}_{\boldsymbol{\theta}}\mathbf{V}\mathbf{S}_{\boldsymbol{\theta}}$ is a generalized Coriolis and centripetal force matrix, $\boldsymbol{\tau}_{\mathrm{R}}$ are the resistive fluid torques given in (2.16), $\mathbf{f}_{\mathrm{D}}$ are the fluid drag forces defined in (2.13), and $\mathbf{u} \in \mathbb{R}^{N-1}$ is a vector that contains the motor torques of the single joints.

Remark 2.4. The model of the underwater snake robot simplifies to the model of a terrestrial snake robot when the resistive fluid torques and 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.

2.3 Locomotion analysis during sinusoidal gaits

The linear drag forces $\mathbf{f}_{\mathrm{D}}^{\mathrm{I}}$ given in (2.13b) are the dominating drag effect for slow motion, whereas the nonlinear drag forces only have a strong effect for faster motion. As a basis for deriving the control-oriented model in Section 2.4, this section investigates how a snake robot can achieve forward propulsion with slow undulatory swimming according to (1.3) in the presence of a constant, irrotational current based on an analysis of the linear drag force (2.13b) and the added mass force (2.14). The analysis of the added mass effects (2.14) during sinusoidal locomotion is included in order to provide additional backup for Assumption 3. In addition, some other important properties of snake robot locomotion according to (1.3) will be summarized.

A similar analysis of the forces that propel a terrestrial snake robot was already introduced in [62] and extended to an underwater robot that is not exposed to currents in [40]. In this section, the analysis is generalized to robots that are under the influence of ocean currents, and we obtain new properties regarding the added mass effects. For the analysis, it is assumed without loss of generality that the forward direction of the robot is aligned with the global x-axis. The propulsive force is then the sum of all external forces in x-direction. It is obtained by the sum

of the x-components in (2.13b) and (2.14):

$$
\begin{aligned}
Nm\ddot{p}_x = &- \sum_{i=1}^{N} \underbrace{(c_t \cos^2\theta_i + c_n \sin^2\theta_i)\dot{x}_i}_{\text{I}} && - \sum_{i=1}^{N} \underbrace{(c_t - c_n)\sin\theta_i \cos\theta_i \dot{y}_i}_{\text{II}} \\
&+ \sum_{i=1}^{N} \underbrace{(c_t \cos^2\theta_i + c_n \sin^2\theta_i)V_x}_{\text{III}} && + \sum_{i=1}^{N} \underbrace{(c_t - c_n)\sin\theta_i \cos\theta_i V_y}_{\text{IV}} \\
&- \sum_{i=1}^{N} \underbrace{\mu_n \sin^2\theta_i \ddot{x}_i}_{\text{V}} && + \sum_{i=1}^{N} \underbrace{\mu_n \sin\theta_i \cos\theta_i \ddot{y}_i}_{\text{VI}} \\
&- \sum_{i=1}^{N} \underbrace{\mu_n \sin\theta_i \cos\theta_i \dot{\theta}_i \dot{x}_i}_{\text{VII}} && - \sum_{i=1}^{N} \underbrace{\mu_n \sin^2\theta_i \dot{\theta}_i \dot{y}}_{\text{VIII}} \\
&+ \sum_{i=1}^{N} \underbrace{\mu_n \sin\theta_i \cos\theta_i \dot{\theta}_i V_x}_{\text{IX}} && + \sum_{i=1}^{N} \underbrace{\mu_n \sin^2\theta_i \dot{\theta}_i V_y}_{\text{X}}
\end{aligned}
\tag{2.19}
$$

Terms I and II in (2.19) have already been analysed in [65] and the results will be summarized briefly here. Both terms contribute to the forward propulsion when they are negative. Under the assumption that the parameters c_t and c_n are positive, their properties are:

- **Term I:** The expression $(c_t \cos^2\theta_i + c_n \sin^2\theta_i)$ is always positive, i. e. under the assumption that all links move with a positive velocity $\dot{x}_i > 0$, Term I is positive.
- **Term II:** Under the assumption that all link angles are $|\theta_i| \leq 90°$ and with (2.12), Term II is negative when $\text{sgn}(\theta_i) = \text{sgn}(\dot{y}_i)$.

Compared to previous studies, Terms III-X have to be considered additionally when a current and added mass effects are taken into account:

- **Term III:** The sign of this term is always determined by V_x, since $(c_t \cos^2\theta_i + c_n \sin^2\theta_i) > 0$.
- **Term IV:** Under the assumption that all link angles are $|\theta_i| \leq 90°$ and with (2.12), this term has the same sign as V_y if $\theta_i < 0$ and the opposite sign if $\theta_i > 0$.
- **Term V:** This term always opposes the acceleration of the link. It contributes to the propulsion when the link is slowing down, $\ddot{x}_i < 0$, and vice versa.
- **Term VI:** Under the assumption that all link angles are $|\theta_i| \leq 90°$, this term contributes to propulsion if $\text{sgn}(\theta_i) = \text{sgn}(\ddot{y}_i)$ and opposes it otherwise.
- **Term VII:** Under the assumption that all link angles are $|\theta_i| \leq 90°$ and that all links move with a positive velocity, $\dot{x}_i > 0$, this term contributes to propulsion if $\text{sgn}(\theta_i) \neq \text{sgn}(\dot{\theta}_i)$ and opposes it otherwise.
- **Term VIII:** This term contributes to propulsion if $\text{sgn}(\dot{\theta}_i) \neq \text{sgn}(\dot{y}_i)$, and opposes it otherwise.

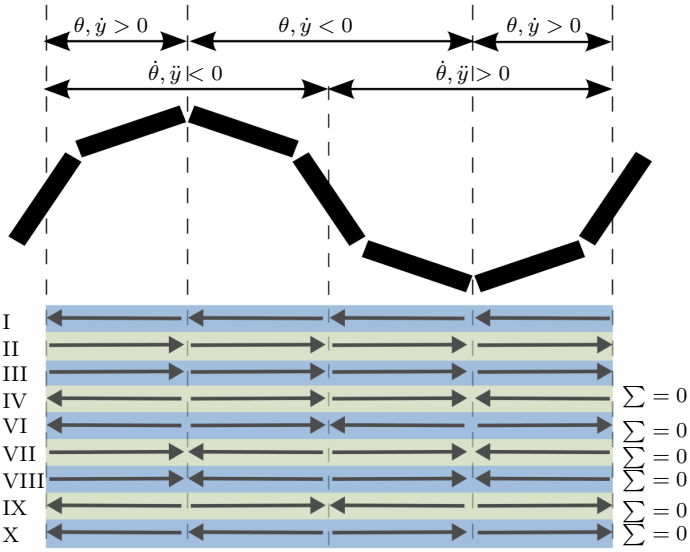


Figure 2.2: The effect of the single terms during lateral undulation.

- **Term IX:** Under the assumption that all link angles are $|\theta_i| \leq 90°$ this term has the same sign as V_x if $\text{sgn}(\theta_i) = \text{sgn}(\dot{\theta}_i)$.
- **Term X:** This term has the same sign as V_y if $\dot{\theta}_i > 0$ and the opposite sign if $\dot{\theta}_i < 0$.

So far, these properties are general and not dependent on the gait. It turns out that for lateral undulation according to (1.2), some simplifications are possible.

Assumption 4: An underwater snake robot is assumed to move forwards with lateral undulation according to (1.2). The phase offset between the links is chosen as $\delta = \frac{2\pi}{N-1}$, such that the period of the gait equals the body length of the robot.

It was pointed out in [97] that this choice of period is beneficial for the efficiency of the propulsion for robots moving in corridor-like environments.

The influence of the single terms on the forward propulsion of the robot under Assumption 4 is visualized in Figure 2.2 and summarized in the following. The arrows in Figure 2.2 indicate in which direction the resulting force points for each term of (2.19), under the assumption that both $V_x, V_y > 0$. Due to the symmetry of the sinusoidal shape, the effects of Terms IV and VI-X are cancelled when the sum $\sum_{i=1}^{N}$ is computed. This means that these terms do not have a considerable effect on forward locomotion. In the case that $V_x < 0$ (resp. $V_y < 0$), the arrows in Terms III and IX (resp. Terms IV and X) point in the opposite direction than in Figure 2.2. The resulting forces in Terms IV, IX, X still cancel each other when summed up, which means that the current component V_y does not have an effect on the propulsion in the x-direction. On the other hand, Term III, whose sign depends on the current component V_x, has an effect in the negative direction, i. e. the same direction as the current in x. The remaining terms in Figure 2.2 are Term I, which

represents the fluid drag force, and Term II. We see from Figure 2.2 that because of the geometry of lateral undulation, Term II constitutes the main propulsive force. Term V is not depicted in Figure 2.2 because its sign depends on the acceleration $\ddot{x}_i$. The sign of $\ddot{x}_i$ cannot be determined as straightforward from analysing the body shape during sinusoidal locomotion. However, during steady forward motion, $\ddot{x}_i$ will in general be small and vary about zero, because the acceleration in the x-direction is small and the robot neither accelerates or breaks down significantly. The result of the previous analysis is summarized in the following property.

Property 1 (Main propulsive forces)**:** For an underwater snake robot with the anisotropic drag property $c_n > c_t$, which moves with lateral undulation according to Assumption 4, Term II is the only term that causes forward propulsion independently of the current. Term I represents the fluid drag force on the robot, always opposing the forward motion, and the effect of the current is captured in Term III, acting in the same direction as V_x. Terms IV and VI-X are cancelled when summing up all forces along the body and Term V is small and varies about zero, which means that the effect of V_y and added mass effects can be disregarded.

Remark 2.5. When Assumption 4 is relaxed to allow other phase shifts δ and a more general sinusoidal gait that does not necessarily have a constant amplitude along the body, Property 1 will still hold. The effect of Terms IV and VI-X will not be cancelled completely, but the remaining effect can be interpreted as a small disturbance.

From [40, 65] we obtain a second and third important property:

Property 2 (Forward propulsion)**:** For a snake robot with $c_n > c_t$, forward propulsion is mainly achieved by the transversal motion of the link.

Property 3 (Link motion)**:** The link motion of a snake robot moving with a sinusoidal gait according to (1.3) consists mainly of a normal displacement of the CM of each link with respect to the direction of motion.

Property 2 was formulated for terrestrial snake robots in [65]. Property 2 also holds in the case of a slowly swimming snake robots since the model of the dominating linear hydrodynamic forces has the same structure as the ground friction model in [65] and added mass effects can be disregarded according to Property 1. Property 3 holds because it is an observation that depends only on the kinematics of the robot, which is identical for terrestrial and planar underwater snake robots. Finally, from [40], we get the following property regarding the turning locomotion of an underwater snake robot:

Property 4 (Turning locomotion)**:** During lateral undulation of a robot described by (2.18), the direction of motion is constant when the average joint angle is zero. The robot will turn (counter-)clockwise when the average joint angle is (positive) negative. The turning rate will increase with an increase of the average joint angle and/or the forward velocity.

The analysis in [40] disregards ocean currents. In the presence of such currents, the relative velocity, i. e. the velocity of the robot with respect to the surrounding fluid has to be taken into account.

2.4 A control-oriented model for sinusoidal gaits

In this section, we derive a *simplified* or *control-oriented* model for underwater snake robots using planar sinusoidal gaits with a small amplitude. The model is derived from the *complex* model that was presented in Section 2.2, and based on the simplifying assumption that the link motion can be approximated with prismatic joints. This idea has originally been employed for a terrestrial snake robot in [62], and later an underwater snake robot that is not exposed to currents in [40]. In this work, unlike in [40], we take ocean currents into account and we present an extensive simulation study for an accurate comparison of the control-oriented model with the original, complex model. The simulation study also adds new insights to the results for the model of terrestrial robots in [65], which falls out as a special case of the presented model when ocean current effects are disregarded.

At first, we will introduce some basic assumptions and the kinematics of the model. Secondly, the hydrodynamic model is presented, and the dynamic equations are derived. Afterwards, a method is proposed, which allows to approximate the sinusoidal gait with the control-oriented model. The section is concluded with a simulation study and a discussion of the model.

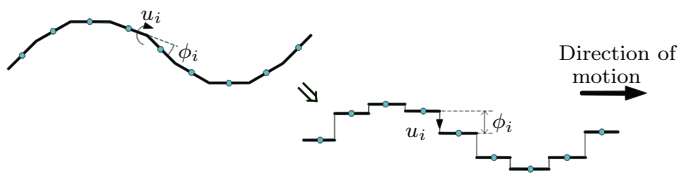
2.4.1 Modelling approach and kinematics

In order to derive the control-oriented model, the revolute joints of the robot are modelled as prismatic joints, with their degree of freedom normal to the direction of motion of the robot, as visualized in Figure 2.3(a). This is a strong simplification, but still a valid approximation for sinusoidal gaits, keeping in mind that according to Properties 2 and 3, the links move mainly in normal direction, which is responsible for obtaining forward propulsion. Just like the complex model presented in Section 2.2, it consists of N links of length $L = 2l$ and mass m, that are connected by $N-1$ joints. The joints are prismatic, with the joint coordinates $\phi_i, i = 1, \ldots, N-1$ that are assembled into the vector $\boldsymbol{\phi} = [\phi_1, \ldots, \phi_{N-1}]^T \in \mathbb{R}^{N-1}$. The joints are actuated with the control input $\mathbf{u} \in \mathbb{R}^{N-1}$. The position of the robot in the plane is defined by the position of its CM, (p_x, p_y). The robot thus has $N+2$ degrees of freedom, two corresponding to the position in the plane, $N-1$ corresponding to the joint coordinates ϕ_i, and one to the orientation θ. Since the single links do not rotate with respect to each other, they all have the same orientation θ, which also defines the orientation of the robot.

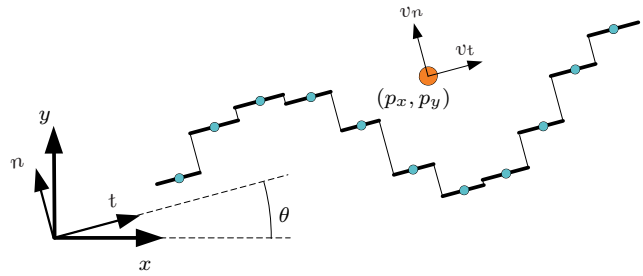
Definition 6: The orientation of a snake robot described by the control-oriented model is denoted by $\theta \in \mathbb{R}$.

For the description of the robot, two coordinate frames are introduced: the global x-y-frame, and the body-aligned t-n-frame. The origins of both frames coincide, as can be seen in Figure 2.3(b). Furthermore, the velocity of the robot is described by the forward velocity component, v_t, and the sideways velocity component, v_n.

Definition 7: The velocity components of the CM of a snake robot described by the control-oriented model that are tangential and normal in the body-aligned t-n-frame are denoted by v_t and v_n, respectively.



(a) The revolute joints are approximated by prismatic joints.



(b) The coordinate frames

Figure 2.3: The control-oriented model.

The external forces depend on the relative velocity, i. e. the velocity of the robot with respect to the surrounding fluid

$$\begin{bmatrix} v_{t,\mathrm{rel}} \\ v_{n,\mathrm{rel}} \end{bmatrix} = \begin{bmatrix} v_t \\ v_n \end{bmatrix} - \mathbf{R}_\theta^T \mathbf{v}_\mathrm{c}. \tag{2.20}$$

The following relationships are derived in [65] and are included here for completeness. The velocities in the x-y-frame can be obtained from the t-n-frame velocities by

$$\begin{aligned} \dot{p}_x &= v_t \cos\theta - v_n \sin\theta, \\ \dot{p}_y &= v_t \sin\theta + v_n \cos\theta. \end{aligned} \tag{2.21}$$

The positions of the CM of link i are given in the t-n-frame by (t_i, n_i). They can be obtained in vector form by

$$\begin{aligned} \mathbf{t} &= \begin{bmatrix} t_1 & \dots & t_N \end{bmatrix}^T &&= p_t \mathbf{e} - l\bar{\mathbf{D}}\bar{\mathbf{e}}, \\ \mathbf{n} &= \begin{bmatrix} n_1 & \dots & n_N \end{bmatrix}^T &&= p_n \mathbf{e} - \bar{\mathbf{D}}\boldsymbol{\phi}, \end{aligned} \tag{2.22}$$

where p_t and p_n are the position of the CM in the body-aligned frame. The link velocities are given by

$$\begin{aligned} \dot{\mathbf{t}} &= (v_t + p_n\dot{\theta})\mathbf{e}, \\ \dot{\mathbf{n}} &= (v_n - p_t\dot{\theta})\mathbf{e} - \bar{\mathbf{D}}\dot{\boldsymbol{\phi}}. \end{aligned} \tag{2.23}$$

2.4.2 Dynamics of the control-oriented model

In this section, the fluid forces that act on the control-oriented model are presented, Afterwards, the translational dynamic equations are derived. Furthermore, a simplified model for the rotational dynamics is proposed. Finally, the dynamic equations are combined with the kinematics and the control-oriented model is derived.

Hydrodynamic modelling

For the derivation of the external forces, we will once more, without loss of generality, assume that the orientation of the robot is aligned with the global x-axis. Two basic assumptions are made in [40] and [65]:

Assumption 5: The link angles θ_i are assumed to be small. Furthermore, the following approximations are made: $\sin\theta_i \cos\theta_i \approx \theta_i, \sin^2\theta_i \approx 0, \cos^2\theta_i \approx 1$.

It is shown in [65] that Assumption 5 is a very good approximation for $|\theta_i| < 20°$, which gradually becomes less accurate as $|\theta_i|$ approaches $30°$ and increases further up to $45°$.

Assumption 6: The effect of the angular velocity $\dot{\theta}$ on the link velocity is disregarded.

As pointed out in [65], this is a valid assumption because the dynamics of the angular rotation of the snake robot will generally be much slower than the body shape dynamics. Furthermore, since a sinusoidal gait with limited link angles according to Assumption 5 will result in slow motion, we disregard nonlinear drag effects.

Assumption 7: The effect of the fluid on a link is described by a linear drag force.

Equipped with Assumptions 5 and 7, the fluid forces (2.8) simplify to

$$\begin{bmatrix} f_{x,i} \\ f_{y,i} \end{bmatrix} = -\begin{bmatrix} c_t & (c_t - c_n)\theta_i \\ (c_t - c_n)\theta_i & c_n \end{bmatrix} \begin{bmatrix} \dot{x}_i - V_x \\ \dot{y}_i - V_y \end{bmatrix}. \tag{2.24}$$

Since the orientation of the single links θ_i is not captured by the control-oriented model, it has to be approximated. It is shown in [65] that the link angles can be estimated by

$$\theta_i \approx \frac{y_{i+1} - y_{i-1}}{2L} = \frac{\phi_{i-1} + \phi_i}{2L}. \tag{2.25}$$

Because the robot is assumed to be aligned with the global x-axis and in accordance with Assumption 6, (2.22) and (2.23) can be inserted into (2.24). Together with (2.25), the forces simplify to

$$\begin{bmatrix} f_{t,i} \\ f_{n,i} \end{bmatrix} = -\begin{bmatrix} c_t & -c_{\mathrm{p}}(\phi_{i-1} + \phi_i) \\ -c_{\mathrm{p}}(\phi_{i-1} + \phi_i) & c_n \end{bmatrix} \begin{bmatrix} \dot{t}_i - V_t \\ \dot{n}_i - V_n \end{bmatrix} \tag{2.26}$$

in the t-n frame. The new parameters in (2.26) are the propulsion coefficient

$$c_{\mathrm{p}} = \frac{c_n - c_t}{2L} \tag{2.27}$$

and $[V_t, V_n]^T = \mathbf{R}_\theta^T \mathbf{v}_\mathrm{c}$, the ocean current in the body-aligned frame. After assembling the equations for each of the links together in matrix form and inserting (2.23), the final form is

$$\begin{aligned} \mathbf{f}_t &= -c_t v_{t,\mathrm{rel}} \mathbf{e} + c_\mathrm{p} \mathrm{diag}(\mathbf{A}^T \boldsymbol{\phi})(v_{n,\mathrm{rel}} \mathbf{e} - \bar{\mathbf{D}} \dot{\boldsymbol{\phi}}), \\ \mathbf{f}_n &= -c_n (v_{n,\mathrm{rel}} \mathbf{e} - \bar{\mathbf{D}} \dot{\boldsymbol{\phi}}) + c_\mathrm{p} \mathrm{diag}(\mathbf{A}^T \boldsymbol{\phi}) v_{t,\mathrm{rel}} \mathbf{e}. \end{aligned} \tag{2.28}$$

Remark 2.6. In the above derivation, the effect of the resistive fluid torque has not been discussed. This results from disregarding the angular link velocity $\dot{\theta}_i$, which means that the fluid torque is assumed to be zero and that all links have the same orientation θ. The issue will be addressed in the following, where a simplified model for the angular dynamics of the robot will be derived.

Translational dynamics

Following the derivations in [65], the dynamic equations for the translational dynamics of the control-oriented model are given as

$$\begin{aligned} \ddot{\boldsymbol{\phi}} &= -\frac{1}{m} \mathbf{D} \mathbf{f}_n + \frac{1}{m} \mathbf{D} \mathbf{D}^T \mathbf{u}, \\ \dot{v}_t &= \frac{1}{Nm} \mathbf{e}^T \mathbf{f}_t, \\ \dot{v}_n &= \frac{1}{Nm} \mathbf{e}^T \mathbf{f}_n. \end{aligned} \tag{2.29}$$

In order to find the closed form, the fluid dynamical forces $\mathbf{f}_t$ and $\mathbf{f}_n$ in (2.28) are inserted into (2.29). The following relations from [65] are used to simplify the equations: $\mathbf{D}\mathbf{e} = \mathbf{0}$, $\mathbf{D}\bar{\mathbf{D}} = \mathbf{I}_{N-1}$, $\mathbf{D}\mathrm{diag}(\mathbf{A}^T \boldsymbol{\phi})\mathbf{e} = -\mathbf{A}\mathbf{D}^T \boldsymbol{\phi}$, $\mathbf{e}^T \mathrm{diag}(\mathbf{A}^T \boldsymbol{\phi})\mathbf{e} = 2\bar{\mathbf{e}}^T \boldsymbol{\phi}$, $\mathbf{e}^T \bar{\mathbf{D}} = \mathbf{0}$, and $\mathbf{e}^T \mathrm{diag}(\mathbf{A}^T \boldsymbol{\phi})\bar{\mathbf{D}} = \boldsymbol{\phi}^T \mathbf{A}\bar{\mathbf{D}}$. The equations of motion are then

$$\begin{aligned} \ddot{\boldsymbol{\phi}} &= -\frac{c_n}{m} \dot{\boldsymbol{\phi}} + \frac{c_\mathrm{p}}{m} v_{t,\mathrm{rel}} \mathbf{A}\mathbf{D}^T \boldsymbol{\phi} + \frac{1}{m} \mathbf{D}\mathbf{D}^T \mathbf{u}, \\ \dot{v}_t &= -\frac{c_t}{m} v_{t,\mathrm{rel}} + \frac{2c_\mathrm{p}}{Nm} \bar{e}^T \boldsymbol{\phi} v_{n,\mathrm{rel}} - \frac{c_\mathrm{p}}{Nm} \boldsymbol{\phi}^T \mathbf{A}\bar{\mathbf{D}} \dot{\boldsymbol{\phi}}, \\ \dot{v}_n &= -\frac{c_n}{m} v_{n,\mathrm{rel}} + \frac{2c_\mathrm{p}}{Nm} \bar{e}^T \boldsymbol{\phi} v_{t,\mathrm{rel}}. \end{aligned} \tag{2.30}$$

Rotational dynamics

The control-oriented model is based on the assumption that the overall model behaviour can be captured by approximating the motion of revolute joints by prismatic joints. Based on this assumption, the translational dynamics of the links and the robot were derived in the previous paragraph. A drawback of the control-oriented modelling approach becomes obvious when considering the rotational dynamics of the robot, i. e. finding an expression for $\ddot{\theta}$. In [65], it is recommended to develop a simplified model of the rotational dynamics based on understanding how the rotational dynamics qualitatively behave, instead of using first principles. Following this line of thoughts, we introduce a simplified model based on a resistive

fluid torque and, in accordance with Property 4, a term that induces turning based on the average of the joint coordinates, $\bar{\mathbf{e}}^T\boldsymbol{\phi}$, and the forward velocity $v_{t,\mathrm{rel}}$.

The rotational dynamics of an underwater snake robot modelled with prismatic joints has already been derived in [40] for the special case of zero current:

$$\ddot{\theta} = \tfrac{1}{1+\tilde{\lambda}_3}\Big(- \tilde{\lambda}_1\dot{\theta} + \tfrac{\tilde{\lambda}_2}{N-1} v_t \bar{\mathbf{e}}^T\boldsymbol{\phi}\Big) \tag{2.31}$$

In order to take into account the current, the absolute velocity v_t has to be replaced by the relative velocity $v_{t,\mathrm{rel}}$ [22]:

$$\ddot{\theta} = -\lambda_1\dot{\theta} + \tfrac{\lambda_2}{N-1} v_{t,\mathrm{rel}} \bar{\mathbf{e}}^T\boldsymbol{\phi} \tag{2.32}$$

In (2.32), the coefficients were redefined as $\lambda_1 := \tilde{\lambda}_1/(1+\tilde{\lambda}_3)$ and $\lambda_2 := \tilde{\lambda}_2/(1+\tilde{\lambda}_3)$ in order to simplify the expression. The equation now has the same structure as the formulation that is given for the control-oriented model of terrestrial snake robots in [65], which it reduces to for the particular case of terrestrial snake robots. In that case, the current velocities are set to zero and the drag forces are replaced by viscous ground friction forces.

The control-oriented model

We define the state vector $\mathbf{x} = [\boldsymbol{\phi}^T, \theta, p_x, p_y, \mathbf{v}_\phi, v_\theta, v_t, v_n]^T \in \mathbb{R}^{2N+4}$. The control-oriented model of the robot is then given by

$$\dot{\boldsymbol{\phi}} = \mathbf{v}_\phi, \tag{2.33a}$$
$$\dot{\theta} = v_\theta, \tag{2.33b}$$
$$\dot{p}_x = v_t\cos\theta - v_n\sin\theta, \tag{2.33c}$$
$$\dot{p}_y = v_t\sin\theta + v_n\cos\theta, \tag{2.33d}$$
$$\dot{\mathbf{v}}_\phi = -\tfrac{c_n}{m}\mathbf{v}_\phi + \tfrac{c_\mathrm{p}}{m} v_{t,\mathrm{rel}}\mathbf{A}\mathbf{D}^T\boldsymbol{\phi} + \tfrac{1}{m}\mathbf{D}\mathbf{D}^T\mathbf{u}, \tag{2.33e}$$
$$\dot{v}_\theta = -\lambda_1 v_\theta + \tfrac{\lambda_2}{N-1} v_{t,\mathrm{rel}}\bar{\mathbf{e}}^T\boldsymbol{\phi}, \tag{2.33f}$$
$$\dot{v}_t = -\tfrac{c_t}{m} v_{t,\mathrm{rel}} + \tfrac{2c_\mathrm{p}}{Nm}\bar{\mathbf{e}}^T\boldsymbol{\phi} v_{n,\mathrm{rel}} - \tfrac{c_\mathrm{p}}{Nm}\boldsymbol{\phi}^T\mathbf{A}\bar{\mathbf{D}}\mathbf{v}_\phi, \tag{2.33g}$$
$$\dot{v}_n = -\tfrac{c_n}{m} v_{n,\mathrm{rel}} + \tfrac{2c_\mathrm{p}}{Nm}\bar{\mathbf{e}}^T\boldsymbol{\phi} v_{t,\mathrm{rel}}. \tag{2.33h}$$

The model parameters are summarized in Table 2.2.

2.4.3 Approximating snake-like sinusoidal gaits

This section explains how sinusoidal gaits with rotational joints can be approximated by the control-oriented model derived in the previous section. In order to achieve a sinusoidal motion for the simplified model, a joint controller has to be designed, where the translational joints are controlled to oscillate with an amplitude a that is usually given in cm. In the complex model on the other hand, the joints are revolute and controlled to move with an amplitude α, which is an angle. In order for the control-oriented model to represent the behaviour of the complex model, a mapping $\alpha \mapsto a$ has to be found. In previous studies [40, 41, 62, 65],

Table 2.2: The parameters of the control-oriented model.

Symbol	Description
N	Number of links
L	Length of a link
m	Mass of a link
c_t	Tangential drag parameter
c_n	Normal drag parameter
c_p	Propulsion coefficient
λ_1	Rotational damping coefficient
λ_2	Rotational coupling coefficient

this mapping has been found by trial and error. This section presents a mathematical description of the mapping $\alpha \mapsto a$. This is achieved by analysing both $\alpha = \alpha(\theta_{i,\max})$ and $a = a(\theta_{i,\max})$, i.e. the geometric relations between the maximal link orientation angle, and the joint angle and the normal distance between the single links, respectively. The analysis is simplified by assuming that without loss of generality, the average orientation of the robot is zero, i. e. the orientation $\bar{\theta}$ of the robot is aligned with the global x-axis. At first the case of lateral undulation with the complex model is investigated and secondly the case of lateral undulation with the control-oriented model.

Lateral undulation with revolute joints

The gait lateral undulation is mathematically described by the serpenoid curve [31]. In [65], it is pointed out that this curve can be discretely approximated by

$$\theta_i(t) = \theta_\mathrm{max} \sin\left(\Omega t + (i-1)\Delta\right), \tag{2.34}$$

where each link angle, θ_i, oscillates with the amplitude θ_max, angular frequency Ω, and a constant offset Δ to the previous link. It is furthermore shown that this gait is achieved by controlling the joints to follow the sinusoidal part of the reference signal in (1.2):

$$\phi_{i,\mathrm{ref}} = \alpha \sin\left(\omega t + (i-1)\delta\right). \tag{2.35}$$

By inserting (2.34) and (2.35) into (2.1), the desired function $\alpha = \alpha(\theta_{i,\max})$ is obtained:

$$\begin{aligned}
\phi_i &= \theta_{i+1} - \theta_i \\
&= \theta_\mathrm{max} \sin\left(\Omega t + i\Delta\right) - \theta_\mathrm{max} \sin\left(\Omega t + (i-1)\Delta\right) \\
&= 2\theta_\mathrm{max} \cos\left(\tfrac{2\Omega t + i\Delta + (i-1)\Delta}{2}\right) \sin\left(\tfrac{\Omega t + i\Delta - \Omega t - (i-1)\Delta}{2}\right) \\
&= 2\theta_\mathrm{max} \cos\left(\Omega t + (i-1)\Delta + \tfrac{\Delta}{2}\right) \sin\left(\tfrac{\Delta}{2}\right) \\
&= \underbrace{2\theta_\mathrm{max} \sin\tfrac{\Delta}{2}}_{=\phi_\mathrm{max}} \sin\Bigg(\Omega t + (i-1)\Delta + \underbrace{\tfrac{\Delta}{2} + \tfrac{\pi}{2}}_{=\widehat{\delta}}\Bigg).
\end{aligned} \tag{2.36}$$

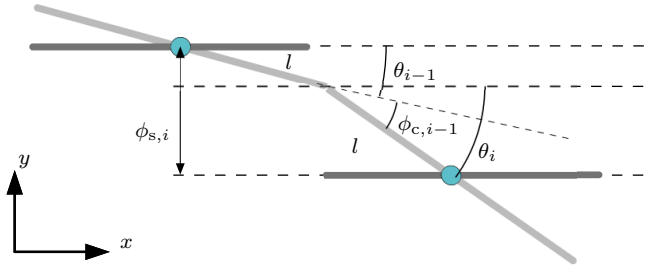


Figure 2.4: The joint coordinates of the simplified model. The notation ϕ_s and ϕ_c refers to the joint distance of the simplified model and the joint angle of the complex model, respectively

This shows that the joint angles oscillate with the amplitude $\phi_{\max}$ and the same angular frequency $\omega = \Omega$ and the same constant offset $\delta = \Delta$ between each other as the link orientation. There is just a constant shift $\widehat{\delta}$ between the maximal joint and link angle.

Proposition 2.1. *Let a snake robot described by the complex model (2.18) move with lateral undulation according to (2.34). Then, the amplitude of the joint angles is given by*

$$\alpha = \phi_{\max} = 2 \sin\left(\tfrac{\delta}{2}\right) \theta_{\max}. \tag{2.37}$$

Remark 2.7. The statement in Proposition 2.1 is not limited to underwater snake robots. It also holds true for terrestrial robots when their kinematics can be described as in Section 2.2.

Lateral undulation with translational joints

For the control-oriented model, the parameter that has to be found is the maximal normal distance between the single oscillating links, referred to as the joint distance ϕ_i. From the geometry of the robot, drawn in Figure 2.4, it can be seen that

$$\phi_i = l \sin\theta_{i+1} + l \sin\theta_i. \tag{2.38}$$

When (2.34) is inserted into (2.38), the amplitude for the oscillation of the joint coordinates, a, can be determined:

$$\begin{aligned} \phi_i = l \sin\Big(\theta_{\max} \sin\big(\Omega t + i\Delta\big)\Big) \\ + l \sin\Big(\theta_{\max} \sin\big(\Omega t + (i-1)\Delta\big)\Big). \end{aligned} \tag{2.39}$$

From (2.39) it is clear that the motion of the joint coordinates is not described by a simple sine function, but by the composition of two sine functions. For small angles, however, we can approximate $\sin\theta_i \approx \theta_i$ according to Assumption 5, and (2.39) simplifies to

$$\phi_i \approx l\theta_{\max}\Big(\sin(\Omega t + i\Delta) + \sin(\Omega t + (i-1)\Delta)\Big). \tag{2.40}$$

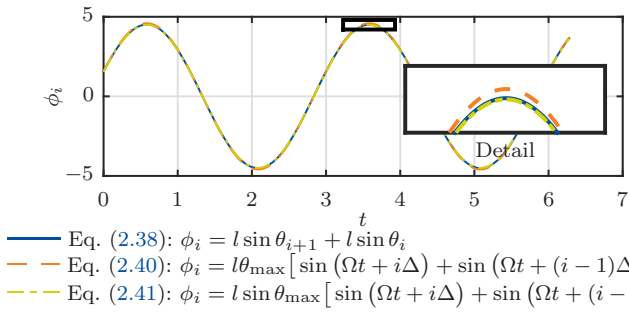


Figure 2.5: The joint coordinates of link i over time

Since the parameter we are looking to find is the amplitude of the oscillation, the approximation can be improved by taking into account the outer sine function in that amplitude:

$$\phi_i \approx l \sin\theta_{\max}\Big(\sin\big(\Omega t + i\Delta\big) + \sin\big(\Omega t + (i-1)\Delta\big)\Big). \tag{2.41}$$

This can be verified by simulations. The amplitude ϕ_i of the joint angle for link i over time is plotted in Figure 2.5 for a maximal joint angle of $\theta_{\max} = 20°$. The plot clearly shows that both (2.40) and (2.41) are good approximations of (2.39), but that (2.41) represents the amplitude more accurately. The bracket term in (2.41) can analogously to (2.36) be summarized as follows:

$$\begin{aligned}\phi_i &\approx 2l \sin\theta_{\max} \sin\Big(\tfrac{2\Omega t + (i-1)\Delta + i\Delta}{2}\Big) \cos\big(\tfrac{\Delta}{2}\big) \\ &= \underbrace{2l \sin\theta_{\max} \cos\big(\tfrac{\Delta}{2}\big)}_{=\phi_{\max}} \sin\Big(\Omega t + (i-1)\Delta + \underbrace{\tfrac{\Delta}{2}}_{=\widehat{\delta}}\Big).\end{aligned} \tag{2.42}$$

Just like for the motion of the complex model we see that the angular frequency $\Omega = \omega$ and the offset between subsequent links $\Delta = \delta$ are the same, and a constant shift $\widehat{\delta} = \frac{\delta}{2}$ remains.

Proposition 2.2. *Let a snake robot described by the simplified model (2.33) move with lateral undulation according to (2.34) and with a maximal link angle $|\theta_i| < 20°$. Then, the amplitude of the joint coordinates is given by*

$$a = \phi_{\max} \approx 2l \cos\big(\tfrac{\delta}{2}\big) \sin\theta_{\max}. \tag{2.43}$$

Remark 2.8. Note that the approximation of the joint amplitude in (2.43) will not become invalid for larger link angles, it will just become less accurate. It will still be a valid approximation for link angles up to $|\theta_i| = 45°$.

A generalized gait

In order to achieve forward propulsion, the joints of the robot are controlled such that they track a sinusoidal wave propagating through the body from head to tail.

For the derivation of the control-oriented model, the amplitude of that wave was so far assumed to be constant, resulting in the well-known gait lateral undulation. This assumption is restrictive, but it allowed to derive an analytical relation between the translational and the revolute amplitude by interpreting the translational link motion as a projection of the revolute link motion onto the subspace orthogonal to the orientation $\bar{\theta}$ of the robot. For eel-like motion on the other hand, an analogue relation cannot be found due to the lack of nose-tail symmetry in the gait. However, in [40] it was shown that eel-like motion according to (1.3) of a complex model can be approximated by the control-oriented model by simply mapping the amplitudes heuristically. We therefore conjecture that the control-oriented model is able to approximate the qualitative behaviour of the complex model also for the generalized gait

$$\phi_{i,\mathrm{ref}}(t) = \alpha g(i) \sin\left(\omega t + (i-1)\delta\right) + \phi_0 \tag{2.44}$$

by using the analytical mapping for the joint amplitudes. An extensive simulation study, where both models are compared, will be presented in Section 2.4.4.

2.4.4 Simulation study and discussion

In order to compare the two models used in this thesis, and to be able to discuss the limitations of the control-oriented model, an extensive simulation study has been carried out. To this end, both models were simulated conducting sinusoidal gaits in open loop in different scenarios, while varying the gait parameters. In the following, we will first introduce the feedback-linearizing controllers that were used in order to achieve locomotion with sinusoidal gaits for both models. Afterwards, the simulation results are presented, and the performance of the control-oriented model will be discussed.

The joint controllers for sinusoidal gaits

In order to stabilize the gait given in (1.3), feedback-linearizing controllers can be implemented. For the control-oriented model (2.33), it is straightforward to apply the following controller to the joint dynamics (2.33e):

$$\mathbf{u} = (\mathbf{D}\mathbf{D}^T)^{-1}\left[m\bar{\mathbf{u}} + c_n\dot{\boldsymbol{\phi}} - c_{\mathrm{p}}v_{t,\mathrm{rel}}\mathbf{A}\mathbf{D}^T\boldsymbol{\phi}\right] \tag{2.45}$$

It transforms the joint dynamics (2.33e) to $\dot{\mathbf{v}}_\phi = \bar{\mathbf{u}}$ such that the new input $\bar{\mathbf{u}} = [\bar{u}_1 \cdots \bar{u}_{N-1}]^T \in \mathbb{R}^{N-1}$ directly controls the joint coordinates. For the complex model (2.18), where the dynamics is given in terms of the link angles, not the joints, we propose the following controller that implicitly imposes the new control input $\bar{\mathbf{u}}$ on the joints (see similar controllers for terrestrial snake robots in [65] and [78]):

$$\begin{aligned}\mathbf{u} = (\mathbf{D}\mathbf{M}_{\boldsymbol{\theta}}^{-1}\mathbf{D}^T)^{-1}\big(&-\mathbf{D}\mathbf{M}_{\boldsymbol{\theta}}^{-1}\mathbf{W}_{\boldsymbol{\theta}}\dot{\boldsymbol{\theta}}^2 + \mathbf{D}\mathbf{M}_{\boldsymbol{\theta}}^{-1}\boldsymbol{\tau}_{\mathrm{R}} + l\mathbf{D}\mathbf{M}_{\boldsymbol{\theta}}^{-1}\mathbf{S}_{\boldsymbol{\theta}}\mathbf{K}\mathbf{f}_{\mathrm{D},x} \\ &- l\mathbf{D}\mathbf{M}_{\boldsymbol{\theta}}^{-1}\mathbf{C}_{\boldsymbol{\theta}}\mathbf{K}\mathbf{f}_{\mathrm{D},y} + \bar{\mathbf{u}}\big)\end{aligned} \tag{2.46}$$

The new control input $\bar{\mathbf{u}}$ is then chosen as

$$\bar{\mathbf{u}} = \ddot{\boldsymbol{\phi}}_{\mathrm{ref}} - k_d(\dot{\boldsymbol{\phi}} - \dot{\boldsymbol{\phi}}_{\mathrm{ref}}) - k_p(\boldsymbol{\phi} - \boldsymbol{\phi}_{\mathrm{ref}}), \tag{2.47}$$

Table 2.3: The simulation parameters.

Symbol	Description	Numerical values
N	Number of links	10
$L = 2l$	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
c_θ^{I}	Linear rotational damping parameter	0.0120
c_θ^{II}	Nonlinear rotational damping parameter	8.1160e-04
c_{p}	Propulsion coefficient control-oriented model	35.7
λ_1	Rotational damping control-oriented model	6
λ_2	Rotational coupling control-oriented model	120

where k_d and k_p are positive control gains, the single entries of $\boldsymbol{\phi}_{\mathrm{ref}}$ are given by

$$\phi_{i,\mathrm{ref}}(t) = \alpha g(i) \sin\left(\omega t - (i-1)\delta\right) + \phi_0 \tag{2.48}$$

and the entries of $\dot{\boldsymbol{\phi}}_{\mathrm{ref}}, \ddot{\boldsymbol{\phi}}_{\mathrm{ref}}$ are given by

$$\begin{aligned} \dot{\phi}_{i,\mathrm{ref}}(t) &= \alpha g(i)\omega \cos\left(\omega t + (i-1)\delta\right), \\ \ddot{\phi}_{i,\mathrm{ref}}(t) &= -\alpha g(i)\omega^2 \sin\left(\omega t + (i-1)\delta\right), \end{aligned} \tag{2.49}$$

under the assumption that ϕ_0 is constant. With the control law (2.45)/(2.46) and (2.47), the dynamics of the joint errors $\widetilde{\boldsymbol{\phi}} = \boldsymbol{\phi} - \boldsymbol{\phi}_{\mathrm{ref}}$ is uniformly globally exponentially stable.

Simulation set-up

Both the complex model (2.18) and the control-oriented (2.33) model were implemented in Matlab R2014b. The simulations were carried out using the `ode15s` solver with a relative and absolute error tolerance of 1e-5. The numerical values of the model parameters are given in Table 2.3. These values correspond to the parameters of the snake robot Mamba [66] and were experimentally validated in [45]. Note that the mass of a single link was assumed to exactly fulfil the neutral buoyancy assumption. The control laws (2.45)/(2.46) and (2.47) were applied to the models, and the control gains in (2.47) were chosen as $k_p = 20$ and $k_d = 5$. All initial conditions were set to zero.

In order to investigate the accuracy of the control-oriented model for sinusoidal gaits, several different combinations of gait parameters were considered, they are listed in Table 2.4. The amplitude α was derived from the maximal link angle $\theta_{\max}$ by the relations in Propositions 2.1 and 2.2. They are presented in Table 2.4 as α_{comp} for the complex model and α_{simp} for the control-oriented one. Each case was simulated both for lateral undulation and eel-like motion, for which the scaling function in (2.48) was set to $g(i) = 1$ and $g(i) = \frac{N-i}{N+1}$, respectively. The rotational model parameters of the control-oriented model, λ_1 and λ_2, were set to the values

Table 2.4: Combinations of gait parameters

Case #	1	2	3	4	5	6	7
$\theta_{\max}$ [deg]	10	20	30	20	20	20	20
α_{comp} [deg]	6.84	13.7	20.5	13.7	13.7	10.35	16.9
α_{simp} [cm]	2.94	5.79	8.46	5.79	5.79	5.95	5.58
ω [deg/s]	120	120	120	80	100	120	120
δ [deg]	40	40	40	40	40	30	50

Table 2.5: Simulation results for lateral undulation

Case #	1	2	3		2	6	7
$\theta_{\max}$ [deg]	10	20	30	δ [deg]	40	30	50
$\bar{v}_{\text{comp}}$ [cm/s]	1.84	6.25	10.87		6.25	5.36	5.56
$\bar{v}_{\text{simp}}$ [cm/s]	1.79	6.95	14.85		6.95	9.46	4.83

Table 2.6: Simulation results for eel-like motion

Case #	1	2	3		2	6	7
$\theta_{\max}$ [deg]	10	20	30	δ [deg]	40	30	50
$\bar{v}_{\text{comp}}$ [cm/s]	0.44	1.73	3.65		1.73	1.27	1.74
$\bar{v}_{\text{simp}}$ [cm/s]	0.44	1.72	3.69		1.72	2.10	1.28

in Table 2.3. These were tuned in simulations with lateral undulation according to Case 2 in Table 2.4 in a way that the turning behaviour of the control-oriented model would match the behaviour of the complex model.

In order to evaluate the model behaviour both for straight motion and turning motion, the following scenario was simulated for each case. The robot was moving according to (2.48) for 200 s, while the joint offset ϕ_0 was set to $\frac{\alpha}{6}$ in the time interval $t \in [40\,\text{s}, 70\,\text{s}]$, to $-\frac{\alpha}{6}$ in $t \in [130\,\text{s}, 160\,\text{s}]$, and to zero elsewhere. At first all simulations were carried out without any ocean currents. From these simulations, the average speed $\bar{v}$ of both models was calculated during the time interval $t \in [10\,\text{s}, 40\,\text{s}]$, i. e. after the joint controllers were converged but before turning motion was induced. In a second step, the simulations were repeated under the influence of an ocean current. The current magnitude was chosen as 0.7 times the speed of the complex model for each case in Table 2.4. The angle of attack was set to 30°. By this choice, the robot was experiencing a significant disturbance, but still able to compensate for it with the forward velocity.

Simulation results and discussion

From the simulation results it turned out that a variation of ω and if there was an ocean current did not influence the approximation quality, so we will disregard these aspects in the following discussion of the results. This was expected, since none of the simplifying assumptions of the control-oriented model are related to the oscillation frequency or the ocean current.

The average speed $\bar{v}$ of both models for the scenarios with different maximal

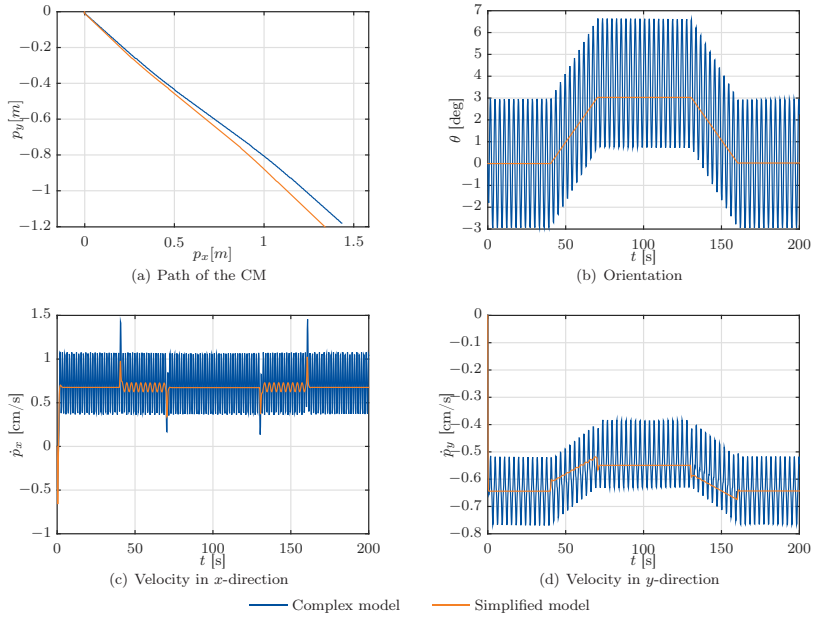


Figure 2.6: Simulation results for lateral undulation according to Case 1.

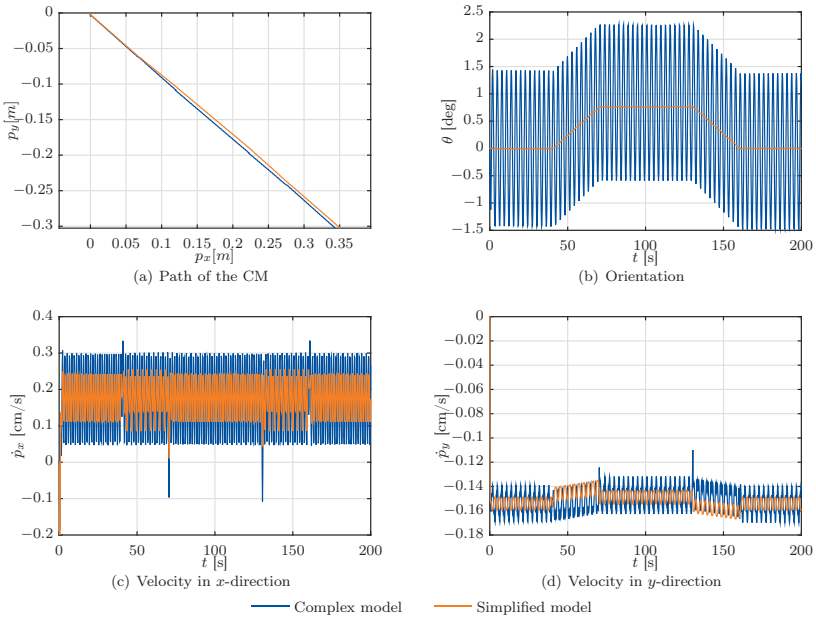


Figure 2.7: Simulation results for eel-like motion according to Case 1.

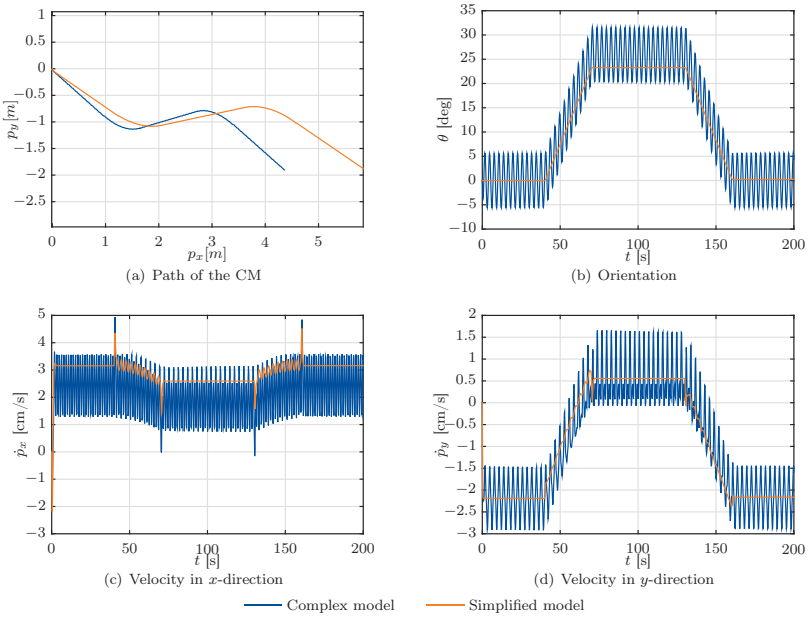


Figure 2.8: Simulation results for lateral undulation according to Case 2.

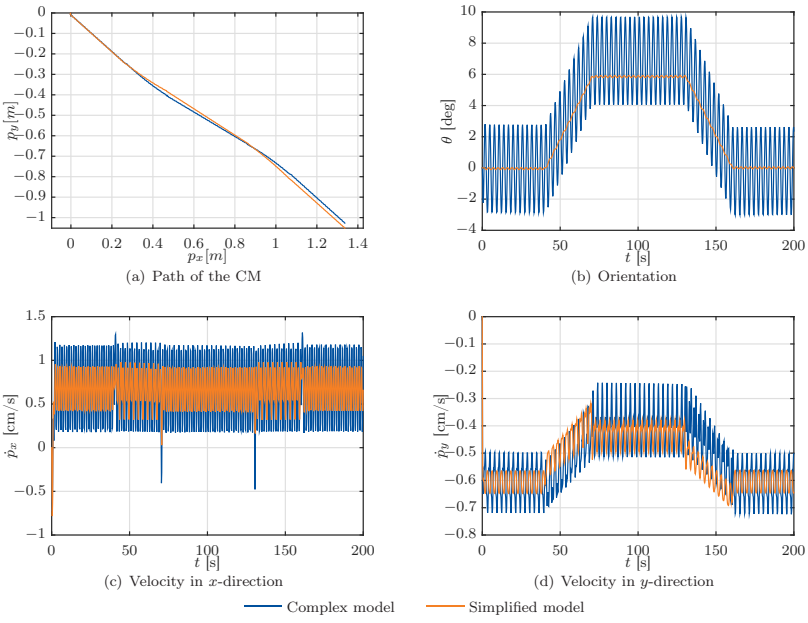


Figure 2.9: Simulation results for eel-like motion according to Case 2.

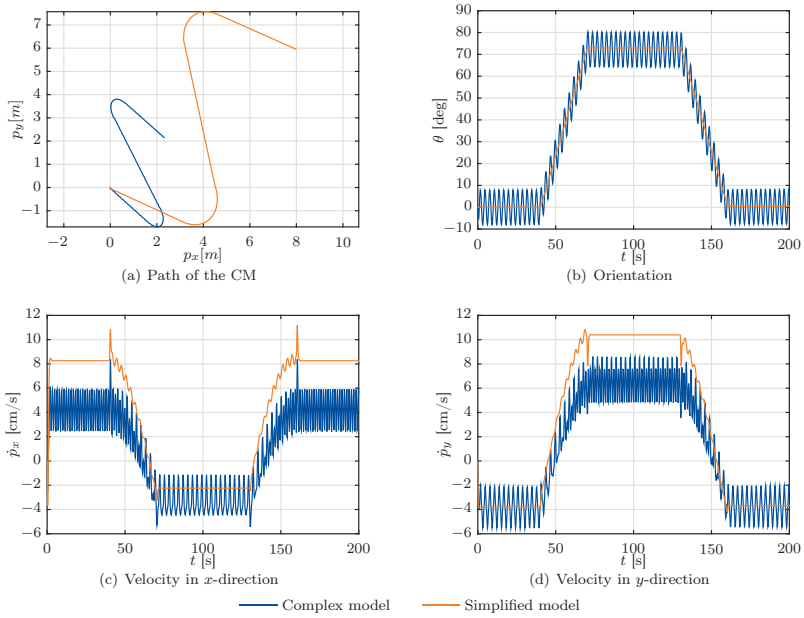


Figure 2.10: Simulation results for lateral undulation according to Case 3.

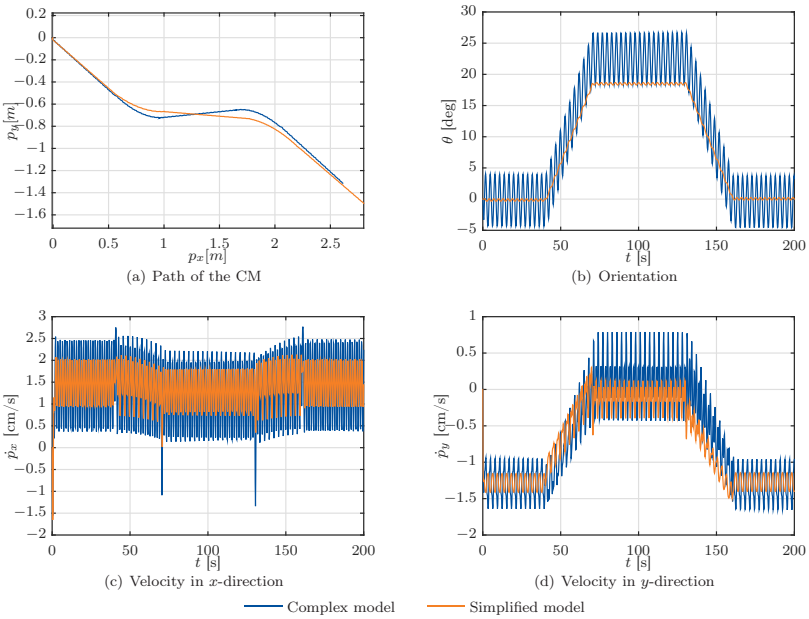


Figure 2.11: Simulation results for eel-like motion according to Case 3.

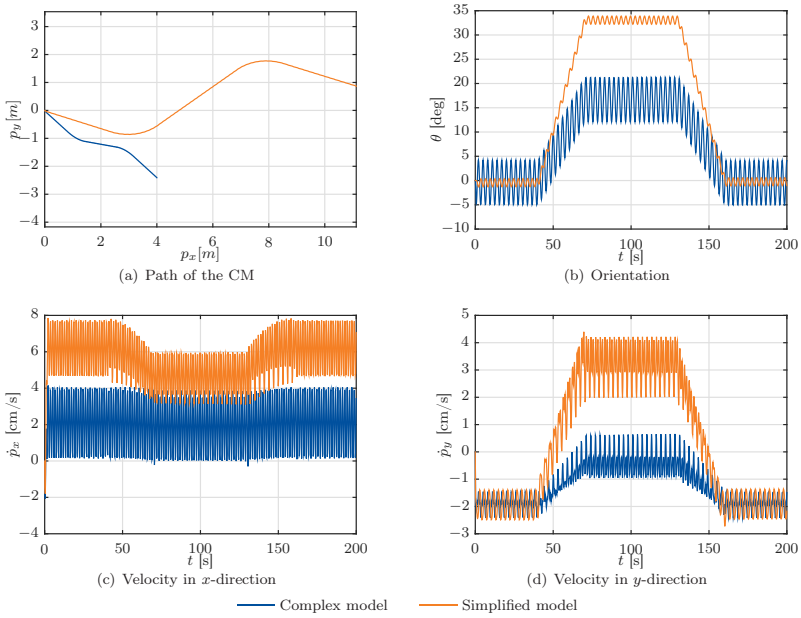


Figure 2.12: Simulation results for lateral undulation according to Case 6.

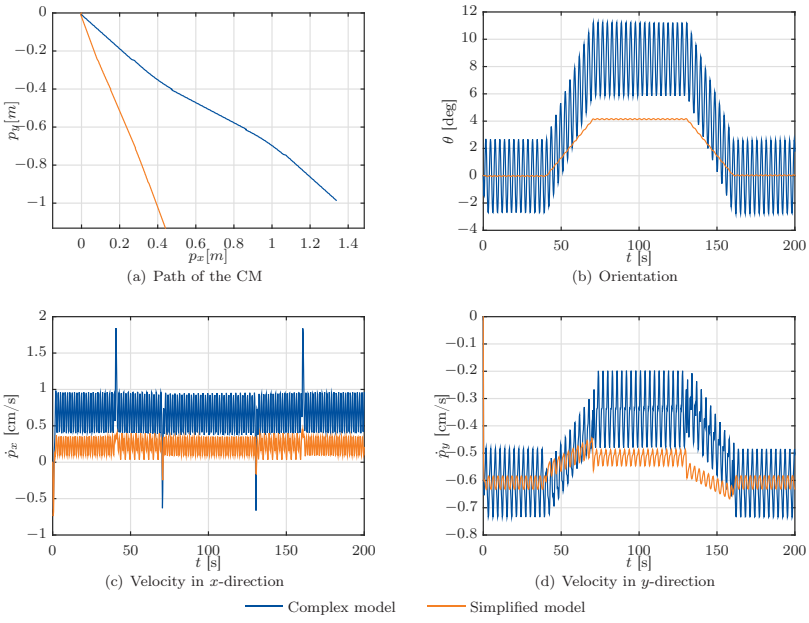


Figure 2.13: Simulation results for eel-like motion according to Case 7.

amplitudes $\theta_{\max}$ and phase shifts δ is summarized in Tables 2.5 and 2.6. The simulations show that the *qualitative* behaviour of the control-oriented model is in good accordance with the complex model in the sense that the control-oriented model moves faster when the complex model does, and vice versa. The *quantitative* similarity between the models, however, strongly depends on Assumption 5 concerning the link angles, $|\theta_i| < 20°$, and Assumption 4 concerning the phase shift, $\delta = \frac{2\pi}{N-1}$. For lateral undulation, the control-oriented model tends to overestimate the velocity for an increasing $\theta_{\max}$. This agrees with Assumption 5 of small link angles. For eel-like motion, this effect was not observed, as can be seen in Table 2.6., where a good *quantitative* similarity between the models is achieved even for a relatively large α. This can be explained by the fact that for eel-like motion, only the tail link is oscillating with the full amplitude, whereas for lateral undulation, every single link contributes with a higher amplitude than assumed. Changing the offset δ has the largest influence on the similarity between the models for both gaits. The tables show that the assumption $\delta = \frac{2\pi}{N-1}$ is important for a good *quantitative* approximation. More specifically, the control-oriented model overestimates the velocity when $\delta < \frac{2\pi}{N-1}$ and underestimates it when $\delta > \frac{2\pi}{N-1}$. This is a drawback of the control-oriented model that was not taken into account in previous studies [40, 65], where the dependency of the model quality on δ was not investigated. However, in [65] it was found that both the complex and the control-oriented model are in excellent accordance with regards to the δ that gives maximum forwards propulsion, which indicates strongly that a good *qualitative* similarity between both models is preserved for $\delta \neq \frac{2\pi}{N-1}$. Assumption 4 additionally restricted the gait to lateral undulation, but this is not a requirement for a good model approximation, as can be seen from the results in Table 2.6. Furthermore, the simulation results indicate that disregarding nonlinear effects according to Assumption 7 is a valid approximation.

The trajectories of the robot in some of the scenarios presented in Tables 2.5 and 2.6 are plotted in Figures 2.6 to 2.13. In these scenarios, the robot was exposed to an ocean current and the orientation of the complex model in these figures was obtained as the average of all link angles according to Definition 3. The figures confirm the discussion above. Figures 2.6 to 2.11 show the trajectories of Cases 1-3 for an increasing amplitude. It can be seen that the turning behaviour of the control-oriented model is in good accordance with that of the complex model in all cases, even though the rotational model parameters λ_1 and λ_2 were only tuned for lateral undulation according to Case 2. Furthermore, the velocities are captured well by the control-oriented model, except in Case 3 for lateral undulation, displayed in Figure 2.10, where the forward velocity is overestimated by the control-oriented model. A scenario according to Case 6 and 7 is presented in Figure 2.12 for lateral undulation and Figure 2.13 for eel-like motion, respectively. As discussed above, the control-oriented model did not quantitatively capture the velocity of the complex model. In addition, as can be seen from Figures 2.12(b) and 2.13(b), the turning rate of the robot was overestimated by the control-oriented model in Case 6, and underestimated in Case 7. This indicates that the rotational model parameters λ_1 and λ_2 should be tuned depending on the phase shift δ.

Remark 2.9. In [65], there was pointed out an additional issue with the control-oriented model. Because of the simplification in the fluid-dynamic forces (2.24),

where the term $c_n \sin\theta_i$ is disregarded, the accuracy of the control-oriented model will decrease not only for large link angles, but also for a small c_t/c_n ratio. As can be seen from the simulation results, this is not an issue in this work. For systems with a smaller c_t/c_n ratio, a strategy can be found in [49], which solves this problem by taking an additional term into account in the control-oriented model.

2.5 Analysis of the velocity during sinusoidal gaits

In this section, the velocity dynamics of the control-oriented model (2.33) is analysed using averaging theory. The joints of the robot are controlled to follow the reference signal (2.44) with the exponentially stabilizing controller used in Section 2.4.4. At first, the model is reduced to include only the relevant dynamics, and then transformed to an averaged model. Using the new model of the averaged velocities, the steady-state behaviour of the robot is analysed and relationships between the gait parameters and the relative forward velocity are derived.

2.5.1 A model of the averaged velocity dynamics

Similar to the analyses in [41, 65], the state vector for the velocity dynamics is defined as

$$\mathbf{v} = [v_t \; v_n \; v_\theta]^T \in \mathbb{R}^3. \tag{2.50}$$

Furthermore, for simplicity, we will assume that the joint offset ϕ_0 is constant and that the ocean current is constant in the body-aligned frame and given by the components V_t and V_n. The ocean current is in reality slowly varying in the body-aligned frame as the robot turns, but it will always be bounded and the variation is slow compared to the body oscillation, in particular for small ϕ_0. Disregarding this time variation is therefore a valid simplification. From (2.33f), (2.33g), and (2.33h) the velocity dynamics is

$$\dot{\mathbf{v}} = \begin{bmatrix} -\frac{c_t}{m} v_{t,\mathrm{rel}} + \frac{2c_\mathrm{p}}{Nm} v_{n,\mathrm{rel}} f_1(\omega t) - \frac{c_\mathrm{p}}{Nm} f_2(\omega t) \\ -\frac{c_n}{m} v_{n,\mathrm{rel}} + \frac{2c_\mathrm{p}}{Nm} v_{t,\mathrm{rel}} f_1(\omega t) \\ -\lambda_1 v_\theta + \frac{\lambda_2}{N-1} v_{t,\mathrm{rel}} f_1(\omega t) \end{bmatrix} = \mathbf{f}(t, \mathbf{v}), \tag{2.51}$$

with

$$\begin{aligned} f_1(\omega t) &= (N-1)\phi_0 + \sum_{i=1}^{N-1} \alpha g(i) \sin(\omega t + (i-1)\delta), \\ f_2(\omega t) &= \sum_{i=1}^{N-1} \sum_{j=1}^{N-1} a_{ij} \big[\phi_0 \alpha g(j) \omega \cos(\omega t + (j-1)\delta) \\ &\quad + \alpha^2 g(i) g(j) \omega \sin(\omega t + (i-1)\delta) \cos(\omega t + (j-1)\delta) \big], \end{aligned} \tag{2.52}$$

and a_{ij} denoting element (i, j) of the matrix $\mathbf{A}\bar{\mathbf{D}}$.

In order to apply averaging to the model of the velocity dynamics, (2.51) needs to be transformed to the standard form $\frac{\mathrm{d}\mathbf{v}}{\mathrm{d}\tau} = \epsilon \mathbf{f}(\tau, \mathbf{v})$ (cf. Chap. 10.4 in [48]). This is achieved by the choice $\tau = \omega t$, yielding $\frac{\mathrm{d}}{\mathrm{d}t} = \omega \frac{\mathrm{d}}{\mathrm{d}\tau}$ and $\epsilon = \frac{1}{\omega}$. The resulting

model is 2π-periodic in τ and the averaged system is

$$\frac{\mathrm{d}\mathbf{v}_{\mathrm{av}}}{\mathrm{d}\tau} = \epsilon \frac{1}{2\pi} \int_0^{2\pi} \mathbf{f}(\tau, \mathbf{v}) \mathrm{d}\tau. \tag{2.53}$$

Solving the integrals of $f_1(\tau)$ and $f_2(\tau)$ gives

$$\begin{aligned} \int_0^{2\pi} f_1(\tau)\mathrm{d}\tau &= 2\pi(N-1)\phi_0, \\ \int_0^{2\pi} f_2(\tau)\mathrm{d}\tau &= -\pi \underbrace{\alpha^2\omega}_{k_{\alpha\omega}} \underbrace{\sum_{i=1}^{N-1}\sum_{j=1}^{N-1} a_{ij} g(i) g(j) \sin((j-i)\delta)}_{k_\delta}, \end{aligned} \tag{2.54}$$

and the averaged model is obtained:

$$\begin{aligned} \frac{\mathrm{d}\mathbf{v}_{\mathrm{av}}}{\mathrm{d}\tau} &= \epsilon \begin{bmatrix} -\frac{c_t}{m} v_{t,\mathrm{rel}} + \frac{2c_{\mathrm{p}}(N-1)}{Nm}\phi_0 v_{n,\mathrm{rel}} + \frac{c_{\mathrm{p}}}{2Nm} k_{\alpha\omega} k_\delta \\ -\frac{c_n}{m} v_{n,\mathrm{rel}} + \frac{2c_{\mathrm{p}}(N-1)}{Nm}\phi_0 v_{t,\mathrm{rel}} \\ -\lambda_1 v_\theta + \lambda_2 \phi_0 v_{t,\mathrm{rel}} \end{bmatrix} \\ &= \epsilon\big(\boldsymbol{\mathcal{A}}\mathbf{v} + \boldsymbol{\mathcal{B}}\big), \end{aligned} \tag{2.55}$$

with

$$\begin{aligned} \boldsymbol{\mathcal{A}}(\phi_0) &= \begin{bmatrix} -\frac{c_t}{m} & \frac{2c_{\mathrm{p}}(N-1)}{Nm}\phi_0 & 0 \\ \frac{2c_{\mathrm{p}}(N-1)}{Nm}\phi_0 & -\frac{c_n}{m} & 0 \\ \lambda_2\phi_0 & 0 & -\lambda_1 \end{bmatrix}, \\ \boldsymbol{\mathcal{B}}(\phi_0) &= \begin{bmatrix} \frac{c_{\mathrm{p}}}{2Nm} k_{\alpha\omega} k_\delta \\ 0 \\ 0 \end{bmatrix} - \boldsymbol{\mathcal{A}}(\phi_0) \begin{bmatrix} V_t \\ V_n \\ 0 \end{bmatrix}. \end{aligned} \tag{2.56}$$

The final version of the averaged model of the velocity dynamics is obtained by changing the time-scale back to t:

$$\dot{\mathbf{v}}_{\mathrm{av}} = \frac{\mathrm{d}\mathbf{v}_{\mathrm{av}}}{\mathrm{d}t} = \frac{1}{\epsilon}\frac{\mathrm{d}\mathbf{v}_{\mathrm{av}}}{\mathrm{d}\tau} = \boldsymbol{\mathcal{A}}\mathbf{v} + \boldsymbol{\mathcal{B}}. \tag{2.57}$$

The averaged model is a linear system that is characterised by the gait parameters and the velocity of the ocean current.

Remark 2.10. Due to the similar structure of the systems, the dynamic matrix of the averaged model (2.57) has the same structure as the one of terrestrial snake robots in [65]. The constant offset $\boldsymbol{\mathcal{B}}$, however, now includes the velocity of the current.

2.5.2 The averaged velocity dynamics in steady state

Similar to the procedure in [61, 65] and [41], the offset $\boldsymbol{\mathcal{B}}$ is removed by the transformation $\mathbf{z} = \mathbf{v}_{\mathrm{av}} + \boldsymbol{\mathcal{A}}^{-1}\boldsymbol{\mathcal{B}}$ in order to analyse the stability of the averaged model:

$$\dot{\mathbf{z}} = \dot{\mathbf{v}}_{\mathrm{av}} = \boldsymbol{\mathcal{A}}(\mathbf{z} - \boldsymbol{\mathcal{A}}^{-1}\boldsymbol{\mathcal{B}}) + \boldsymbol{\mathcal{B}} = \boldsymbol{\mathcal{A}}\mathbf{z}. \tag{2.58}$$

In order to determine the stability properties of this linear system, the eigenvalues of $\boldsymbol{\mathcal{A}}$ are computed as

$$\text{eig}(\boldsymbol{\mathcal{A}}) = \begin{bmatrix} -\lambda_1 \\ -\frac{c_n+c_t}{2m} - \frac{\sqrt{(c_n N - c_t N)^2 + (4(N-1)c_{\text{p}}\phi_0)^2}}{2Nm} \\ -\frac{c_n+c_t}{2m} + \frac{\sqrt{(c_n N - c_t N)^2 + (4(N-1)c_{\text{p}}\phi_0)^2}}{2Nm} \end{bmatrix}. \tag{2.59}$$

Note that, even though the snake robot is now exposed to ocean currents, the eigenvalues are the same as for terrestrial snake robots (cf. Eq. (7.24) in [65]). It can easily be verified that all eigenvalues in (2.59) are negative if

$$|\phi_0| < \frac{N}{2(N-1)} \frac{\sqrt{c_n c_t}}{c_{\text{p}}}, \tag{2.60}$$

a condition which implies that the equilibrium $\mathbf{z} = 0$ is globally exponentially stable. The constraint (2.60) on the offset ϕ_0 indicates that modelling the joints as translational rather than revolute is restricted to a limited displacement.

Under the assumption that ϕ_0 is sufficiently small for (2.60) to hold, $\mathbf{v}_{\text{av}}$ will converge to the steady-state velocity

$$\begin{aligned} \mathbf{v}^* &= -\boldsymbol{\mathcal{A}}^{-1}\boldsymbol{\mathcal{B}} = -\boldsymbol{\mathcal{A}}^{-1} \begin{bmatrix} \frac{c_{\text{p}}}{2Nm} k_{\alpha\omega} k_\delta \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} V_t \\ V_n \\ 0 \end{bmatrix}, \\ &= k_{\alpha\omega} k_\delta \begin{bmatrix} \frac{N c_n c_{\text{p}}}{2(c_n c_t N^2 - 4(N-1)^2 c_{\text{p}}^2 \phi_0^2)} \\ \frac{c_{\text{p}}^2 \phi_0 (N-1)}{c_n c_t N^2 - 4(N-1)^2 c_{\text{p}}^2 \phi_0^2} \\ \frac{N c_n c_{\text{p}} \lambda_2 \phi_0}{2\lambda_1 (c_n c_t N^2 - 4(N-1)^2 c_{\text{p}}^2 \phi_0^2)} \end{bmatrix} + \begin{bmatrix} V_t \\ V_n \\ 0 \end{bmatrix}. \end{aligned} \tag{2.61}$$

From (2.61) we see that the steady-state velocity depends on the parameters of the gait, the drag parameters, the number of links, and the current velocity. The expression for the steady-state velocity (2.61) includes the same parameters as for terrestrial snake robots [65], to which the current velocities in the body frame are added. This was expected, since the averaged model was seen to have the same structure, with the additional capability of taking into account currents.

After establishing global exponential stability for the averaged velocity dynamics, the stability of the exact dynamics will be considered. According to Theorem 10.4 in [48], it follows from the stability of the averaged dynamics that, for sufficiently small ϵ, the average velocity given by (2.61) approximates the exact velocity (2.51) for all time and with an error that is bounded. This means that, if ω is sufficiently large, the periodic time-varying velocity will remain close to the exponentially stable averaged velocity for all time.

Theorem 2.3. *Consider an underwater snake robot described by* (2.33) *controlled in accordance with a planar sinusoidal gait according to equations* (2.44) *and* (2.49) *and the offset* ϕ_0 *satisfying* (2.60)*. Then there exist* $k > 0$, $\omega^* > 0$ *such that the following holds for all* $\omega > \omega^*$*:*

$$\|\mathbf{v}(t) - \mathbf{v}_{\text{av}}(t)\| \leq \tfrac{k}{\omega} \quad \forall t > 0. \tag{2.62}$$

In addition, the averaged velocity $\mathbf{v}_{\text{av}}(t)$ *converges exponentially to the steady-state velocity* $\mathbf{v}^*$ *in* (2.61).

Remark 2.11. Note that the presence of constant irrotational currents does not influence the stability properties of the snake robot, since the eigenvalues (2.59) do not depend on the current. The current only moves the equilibrium of the velocity dynamics. Moreover, by subtracting the current velocities from both sides of (2.61), it can be shown that the *relative* velocities converge to the same value as the velocities of a terrestrial snake robot.

With the averaged steady-state velocity for an underwater snake robot moving with a planar sinusoidal gait that was presented in the previous section, it becomes possible to analyse a scenario that is particularly interesting with respect to motion planning purposes: steady-state motion with zero offset $\phi_0 = 0$, which will be shown to be motion in a straight line.

By inserting $\phi_0 = 0$ into (2.61) and subtracting the current velocities from both sides, the expression

$$\begin{bmatrix} v^*_{t,\text{rel}} \\ v^*_{n,\text{rel}} \\ v^*_\theta \end{bmatrix} = \begin{bmatrix} k_{\alpha\omega} k_\delta \frac{c_{\text{p}}}{2c_t N} \\ 0 \\ 0 \end{bmatrix} \tag{2.63}$$

is obtained. It can easily be seen that the relative velocity normal to the robot's orientation is zero, as is the rotational velocity. This means that the robot moves in a straight line, with its absolute normal velocity equal to the normal current velocity. For the forward velocity, the following property can be derived from (2.63), keeping in mind that $k_{\alpha\omega} = \alpha^2\omega$:

Corollary 2.4. *Consider an underwater snake robot described by* (2.33) *controlled in accordance with a planar gait according to equations* (2.44) *and* (2.49). *For* $\omega > \omega^*$ *and* $\phi_0 = 0$, *the averaged relative forward velocity of the robot will converge exponentially to* $v^*_{t,\text{rel}}$, *which is proportional to*

- *the squared amplitude of the joints,* α^2
- *the frequency of the gait,* ω
- *a function of the phase shift* δ, *which is given by*

$$k_\delta = \sum_{i=1}^{N-1} \sum_{j=1}^{N-1} a_{ij} g(i) g(j) \sin((j-i)\delta). \tag{2.64}$$

This result extends the findings of previous studies: In [70], it was shown that the averaged forward dynamics of a three- and a five-link eel-like robot are captured by a function proportional to the squared amplitude, the frequency, and a sum of sinusoidal functions. In [65], the special case of lateral undulation, yielding both $g(i) = g(j) = 1$, was investigated. As pointed out in [65], Corollary 2.4 provides a powerful tool for motion planning: an increase/decrease of the relative forward velocity can be invoked by using α or ω as a control input. Furthermore, the controller can be optimized by finding the optimal phase shift δ that maximizes k_δ for the given number of links and choice of gait.

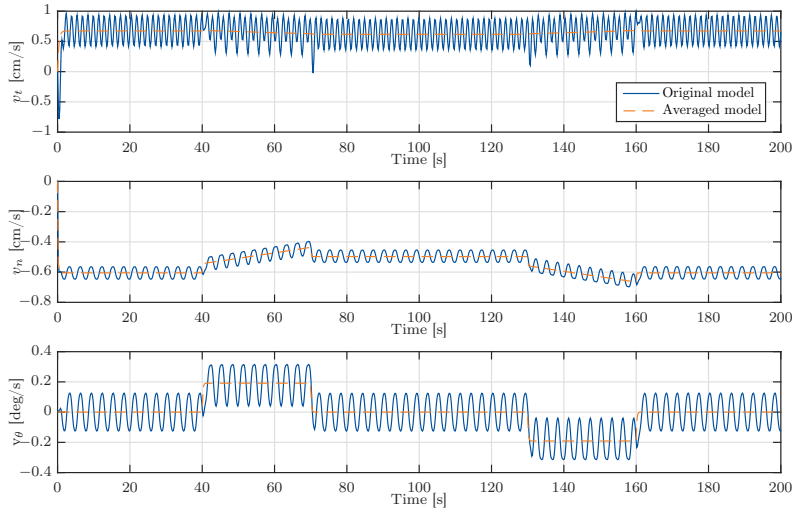


Figure 2.14: The averaged and the exact velocity dynamics for eel-like motion.

2.5.3 Simulation study

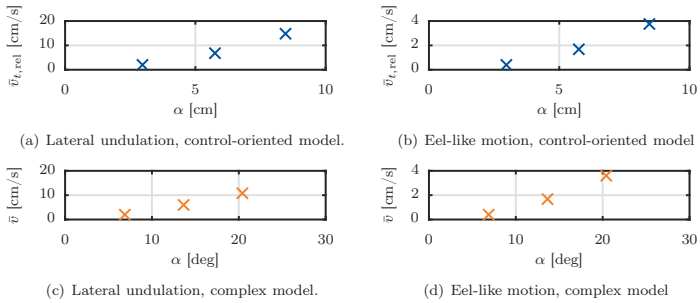
The results of the averaging analysis in this section are confirmed in simulations. At first, the original and the averaged velocity dynamics are compared, and afterwards the relationship between the gait parameters and the relative forward velocity is investigated.

Comparison of the original and the averaged velocity

In order to investigate the performance of the averaged velocity dynamics, the scenarios with ocean currents that are summarized in Table 2.4 were additionally simulated with the averaged model. The simulation results with the control-oriented model (2.33) and with the averaged model (2.57) showed excellent accordance. An example is presented in Figure 2.14, where the velocities of both the control-oriented model and the averaged model are plotted for eel-like motion according to Case 2 in Table 2.4.

Relationship between the gait parameters and $v_{t,\mathrm{rel}}$

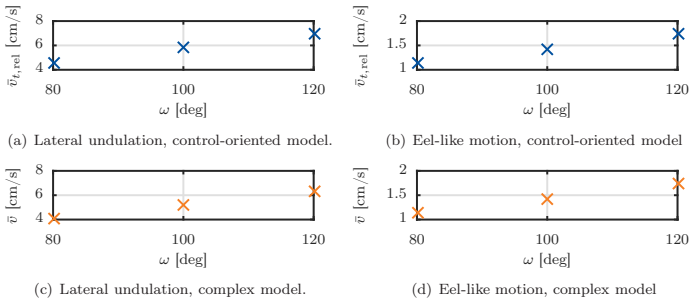
The simulation results of an underwater snake robot moving with a sinusoidal gait in the presence of ocean currents in Section 2.4.4 were evaluated for several combinations of gait parameters in order to validate the relationships between the relative forward velocity and the gait parameters α and ω, that were derived in Section 2.5.2. The scenarios are Cases 1-3 in Table 2.4 in order to check the dependency on α, and Cases 4,5, and 2 for ω.



(a) Lateral undulation, control-oriented model. (b) Eel-like motion, control-oriented model

(c) Lateral undulation, complex model. (d) Eel-like motion, complex model

Figure 2.15: The relationship between $v_{t,\mathrm{rel}}$ and α.



(a) Lateral undulation, control-oriented model. (b) Eel-like motion, control-oriented model

(c) Lateral undulation, complex model. (d) Eel-like motion, complex model

Figure 2.16: The relationship between $v_{t,\mathrm{rel}}$ and ω.

Figure 2.15 shows the average of the relative forward velocity that was obtained for the different choices of α in the time interval $t \in [10\,\mathrm{s}, 40\,\mathrm{s}]$, i. e. for straight motion (with $\phi_0 \equiv 0$). In the same manner, the average of the relative forward velocities for the different choices of ω are plotted in Figure 2.16. For the control-oriented model, the average of the relative forward velocity, $\bar{v}_{t,\mathrm{rel}}$, was obtained as the mean of $v_{t,\mathrm{rel}}$ in the time interval $t \in [10\,\mathrm{s}, 40\,\mathrm{s}]$. For the complex model, the definition of $v_{t,\mathrm{rel}}$ is not the same as for the control-oriented model. The average relative forward velocity of the complex model in Figures 2.15 and 2.16 is therefore approximated by average of the relative speed $\bar{v} = \sqrt{(\dot{p}_x - V_x)^2 + (\dot{p}_y - V_y)^2}$ in the time interval $t \in [10\,\mathrm{s}, 40\,\mathrm{s}]$. It can be seen that there is clearly a quadratic relationship in Figure 2.15 and a linear relationship in Figure 2.16

2.6 Chapter summary

This chapter presented two models that will be used for control design in the remainder of this thesis as well as some fundamental properties regarding sinusoidal locomotion. The complex model, which was derived from first principles and analytical fluid-dynamics, will be the basis for the control design in Chapter 4. Based on this model, fundamental properties of locomotion with planar sinusoidal gaits were derived in this chapter. These properties were used to motivate some simplifying assumptions, based on which the second, control-oriented model used in this

thesis was derived. The control-oriented model will be used for control design in Chapters 3 and 5. The two models were compared in an extensive simulation study, which showed that the control-oriented model qualitatively captures the behaviour of the complex model, and even quantitatively approximates the complex model well for certain combinations of gait parameters. Based on the second model, the velocity dynamics of snake robot locomotion during planar sinusoidal gaits were analysed using averaging theory, revealing relationships between the forward velocity and the gait parameters.

Chapter 3

Integral Line-of-Sight Guidance and Model-Based Control

This chapter presents a model-based control system for straight line path-following of neutrally buoyant underwater snake robots that move with a planar sinusoidal gait in the presence of an unknown, constant, and irrotational ocean current. The control system is based on a cascaded design, where a LOS guidance law is employed in the outer control loop in order to provide an orientation reference for the robot. Ocean currents are accounted for in the guidance scheme by augmenting it with integral action in order to compensate for the steady state error. The closed-loop system is formally analysed using cascaded systems theory. A simulation example and experimental test results with a swimming snake robot demonstrate the concept of the control system and validate the theoretical analysis.

Contributions of this chapter The first contribution of this chapter is the development of a control system that enables the robot to converge to and follow a straight path in a virtual plane in the presence of constant irrotational currents. An orientation controller is designed that exponentially stabilizes the orientation of the robot towards the reference angle given by an integral LOS guidance law. Unlike previous approaches for terrestrial snake robots [63, 65], this control system is able to handle the disturbance produced by an ocean current. The second and main contribution of this chapter compared to previous work [43] is the stability analysis. Using cascaded systems theory, it is formally proved that under the assumption of a constant positive forward velocity, the cross-track error between the robot and the desired path is guaranteed to converge to zero. The control system will thus fulfil the control objective, which is confirmed by an experimental study, the third contribution of this chapter.

Organization of this chapter This chapter is structured as follows. Some basic assumptions are made in Section 3.1. In Section 3.2, a model transformation facilitating the control design is proposed. The control objectives are formulated in Section 3.3. The path-following control system and the stability analysis are pre-

sented in Section 3.4. Finally, a simulation example is provided in Section 3.5, and an experimental validation in Section 3.6. The chapter is summarized in Section 3.7.

Publications This chapter is based on [52, 53].

3.1 Assumptions

In this chapter, we consider a robot that is described by the control-oriented model in 2.4. The robot is neutrally buoyant, exposed to an unknown ocean current $\mathbf{v}_\mathrm{c}$ and conducts a planar sinusoidal gait with small joint angles, which results in a positive relative forward velocity $v_{t,\mathrm{rel}}$. The following assumption holds regarding the forward velocity:

Assumption 8: The robot is moving with some constant relative forward velocity $v_{t,\mathrm{rel}} \in [V_\mathrm{min}, V_\mathrm{max}]\ \forall t \geq 0$ with the bounds $V_\mathrm{max} \geq V_\mathrm{min} > 0$.

Remark 3.1. Controlling the exact value of $v_{t,\mathrm{rel}}$ is not within the scope of this chapter. However, it has been shown in Section 2.5 that for a sinusoidal gait, the averaged $v_{t,\mathrm{rel}}$ converges to a constant value given by the design parameters of the gait. If Assumption 8 would be relaxed to allow for a time-varying velocity $v_{t,\mathrm{rel}}$, its dynamics would have to be taken into account in the analysis, which will not be considered at this point of the thesis.

Assumption 9: The forward velocity is large enough to compensate for the ocean current, i.e. $v_{t,\mathrm{rel}} > V_\mathrm{min} > V_{\mathrm{c,max}}$.

3.2 Model transformation

In order to make the control-oriented model (2.33) more suitable for developing the guidance system, a two-step model transformation is employed. In the first step, the point that defines the position of the robot is moved in order to provide a simpler reference. In the second step, the absolute velocities are eliminated from the model equations such that only the relative velocities are considered. These transformation steps simplify the controller design and are the basis for achieving a closed-loop system with a cascaded structure.

In the dynamical equations (2.33f) and (2.33h) it can be seen that the joint coordinates $\boldsymbol{\phi}$, and thus ϕ_0, enter the dynamics of both v_n and v_θ:

$$\dot{v}_\theta = -\lambda_1 v_\theta + \tfrac{\lambda_2}{N-1} v_{t,\mathrm{rel}} \bar{\mathbf{e}}^T \boldsymbol{\phi}, \tag{3.1a}$$

$$\dot{v}_n = -\tfrac{c_n}{m} v_{n,\mathrm{rel}} + \tfrac{2c_\mathrm{p}}{Nm} v_{t,\mathrm{rel}}, \bar{\mathbf{e}}^T \boldsymbol{\phi}. \tag{3.1b}$$

As pointed out in [65], this complicates the controller design and analysis because ϕ_0 is used to control the orientation of the robot. Motivated by [19, 25], it is suggested in [65] to solve the problem by moving the point that defines the position of the snake robot by a distance ϵ in the tangential direction, from the CM to the point where joint offset ϕ_0 induces a purely rotational motion. The coordinate

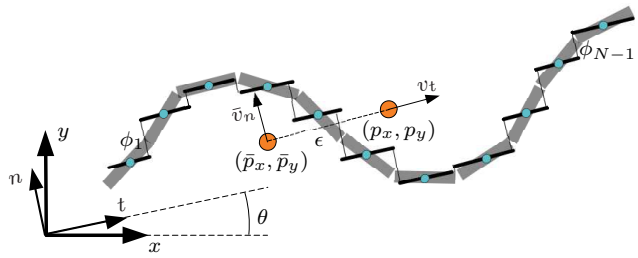


Figure 3.1: The model transformation.

transformation is visualized in Figure 3.1. Like in [65], the new coordinates are then defined as

$$\bar{p}_x = p_x + \epsilon \cos\theta, \tag{3.2a}$$
$$\bar{p}_y = p_y + \epsilon \sin\theta, \tag{3.2b}$$
$$\bar{v}_n = v_n + \epsilon v_\theta. \tag{3.2c}$$

The coefficient ϵ is obtained by taking the time derivative of (3.2c), inserting (3.1a) and (3.1b), and choosing ϵ such that the joint coordinates $\boldsymbol{\phi}$ are eliminated from the equation:

$$\epsilon = -\tfrac{2(N-1)c_{\mathrm{p}}}{Nm\lambda_2}. \tag{3.3}$$

In addition, the absolute velocities are eliminated from the model by inserting the relations

$$\begin{aligned} v_t &= v_{t,\mathrm{rel}} + V_x \cos\theta + V_y \sin\theta, \\ \bar{v}_n &= \bar{v}_{n,\mathrm{rel}} - V_x \sin\theta + V_y \cos\theta. \end{aligned} \tag{3.4}$$

Using the transformation equations (3.2b) and (3.2c) and the relations in (3.4), the model written in the new coordinates is

$$\dot{\boldsymbol{\phi}} = \mathbf{v}_\phi, \tag{3.5a}$$
$$\dot{\theta} = v_\theta, \tag{3.5b}$$
$$\dot{\bar{p}}_y = v_{t,\mathrm{rel}} \sin\theta + \bar{v}_{n,\mathrm{rel}} \cos\theta + V_y, \tag{3.5c}$$
$$\dot{\mathbf{v}}_\phi = -\tfrac{c_n}{m}\mathbf{v}_\phi + \tfrac{c_{\mathrm{p}}}{m} v_{t,\mathrm{rel}} \mathbf{A}\mathbf{D}^T\boldsymbol{\phi} + \tfrac{1}{m}\mathbf{D}\mathbf{D}^T\mathbf{u}, \tag{3.5d}$$
$$\dot{v}_\theta = -\lambda_1 v_\theta + \tfrac{\lambda_2}{N-1} v_{t,\mathrm{rel}} \bar{\mathbf{e}}^T\boldsymbol{\phi}, \tag{3.5e}$$
$$\dot{\bar{v}}_{n,\mathrm{rel}} = (X + V_x \cos\theta + V_y \sin\theta) v_\theta + Y \bar{v}_{n,\mathrm{rel}}, \tag{3.5f}$$

where X and Y are defined as $X = \epsilon(\frac{c_n}{m} - \lambda_1)$ and $Y = -\frac{c_n}{m}$.

Remark 3.2. This model does not include the dynamics of the relative forward velocity $v_{t,\mathrm{rel}}$. This is because the purpose of the model is to design a path-following control system where the forward velocity is not feedback controlled. Instead, the robot propels itself forward by using a biologically inspired gait. In the control design process, $v_{t,\mathrm{rel}}$ is therefore treated as a positive model parameter in accordance with Assumption 8 and the procedure in [65].

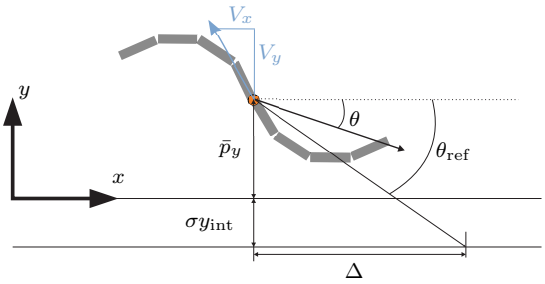


Figure 3.2: The integral LOS guidance law.

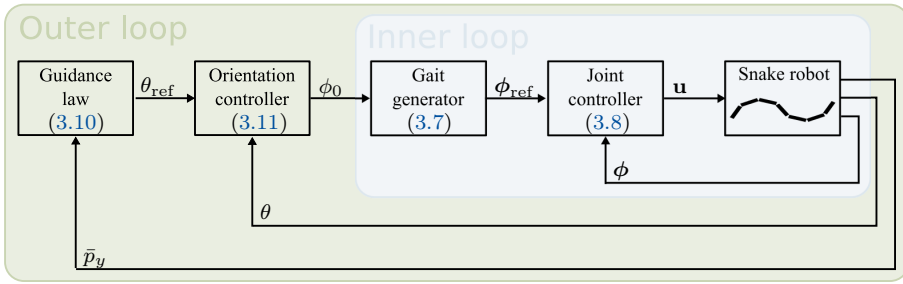


Figure 3.3: The structure of the control system.

Remark 3.3. The dynamics of $\bar{v}_{n,\text{rel}}$ depends on the orientation θ. Compared to the surface vessel model from [22], which the controllers in [8, 12] are based on, an additional term has to be considered in the stability analysis.

3.3 Control objectives

In this section, the control problem that is solved in this chapter is formulated. The control system is supposed to make the robot converge to and subsequently follow a desired straight path $\mathcal{P}$ with some sufficiently large velocity $v_{t,\text{rel}} > 0$. For simplicity and without loss of generality, the inertial coordinate frame is defined such that the desired path $\mathcal{P}$ is aligned with the global x-axis. The cross-track error is then defined by the robot's position $\bar{p}_y$, and the control objectives are

$$\lim_{t\to\infty} \bar{p}_y(t) = 0, \tag{3.6a}$$

$$\lim_{t\to\infty} \theta(t) = \theta^{\text{eq}}. \tag{3.6b}$$

The desired orientation θ^{eq} is constant and $\theta^{\text{eq}} \in (-\frac{\pi}{2}, \frac{\pi}{2})$. The equilibrium orientation θ^{eq} is in general non-zero [43], thus providing a crab angle that allows the robot to compensate for the transversal current component, as can be seen in Figure 3.2. Its value will be defined later.

3.4 The path-following control system

This section introduces the design of the control system. Subsequently, the stability of the closed-loop system is analysed.

3.4.1 Control system design

The control law that is designed to meet the control objectives in (3.6) will be presented in the following. Motivated by [65], it consists of two control loops, the inner loop contains a gait controller for propulsion, and the outer loop an orientation controller for path-following. The structure of the control system can be seen in Figure 3.3.

Gait controller

As shown in Section 2.5, forward motion by a sinusoidal gait is achieved by controlling each joint ϕ_i to track the reference

$$\phi_{i,\text{ref}} = \alpha g(i) \sin\left(\omega t + (i-1)\delta\right) + \phi_0, \qquad i = 1, \ldots, N-1, \tag{3.7}$$

where α is the maximum amplitude, ω is the frequency, δ is the phase shift, and ϕ_0 is a constant offset that induces turning motion. The function $g : \mathbb{Z} \mapsto [0, 1]$ scales the amplitude of the wave along the body. The reference signals for the single joints are assembled into the vector $\boldsymbol{\phi}_{\text{ref}} \in \mathbb{R}^{N-1}$. The controller that enforces the reference (3.7) closes the inner control loop of the cascaded control system and is given by the feedback-linearizing control law

$$\mathbf{u} = m(\mathbf{D}\mathbf{D}^T)^{-1}\left[\bar{\mathbf{u}} + \tfrac{c_n}{m}\dot{\boldsymbol{\phi}} - \tfrac{c_p}{m} v_{t,\text{rel}} \mathbf{A}\mathbf{D}^T \boldsymbol{\phi}\right], \tag{3.8a}$$

$$\bar{\mathbf{u}} = \ddot{\boldsymbol{\phi}}_{\text{ref}} - k_{v_\phi}(\dot{\boldsymbol{\phi}} - \dot{\boldsymbol{\phi}}_{\text{ref}}) - k_\phi(\boldsymbol{\phi} - \boldsymbol{\phi}_{\text{ref}}) \tag{3.8b}$$

with the scalar control gains $k_{v_\phi}, k_\phi > 0$. With the control law (3.8) and the model equations (3.5a), (3.5d), the closed-loop dynamics of the joint error $\tilde{\boldsymbol{\phi}} = \boldsymbol{\phi} - \boldsymbol{\phi}_{\text{ref}}$ can be written as

$$\ddot{\tilde{\boldsymbol{\phi}}} + k_{v_\phi}\dot{\tilde{\boldsymbol{\phi}}} + k_\phi \tilde{\boldsymbol{\phi}} = 0. \tag{3.9}$$

After closing the inner control loop according to the above equations, ϕ_0 can be interpreted as a new control input that induces turning motion to the inner cascade.

Orientation controller

The integral LOS guidance method has first been proposed for marine surface vessels in [11], where integral action was added to the traditional LOS guidance law in order to compensate for the disturbance by the current. Based on [11], the reference orientation for the robot is defined as

$$\theta_{\text{ref}} = -\arctan\left(\tfrac{\bar{p}_y + \sigma y_{\text{int}}}{\Delta}\right), \tag{3.10a}$$

$$\dot{y}_{\text{int}} = \tfrac{\Delta \bar{p}_y}{(\bar{p}_y + \sigma y_{\text{int}})^2 + \Delta^2}, \tag{3.10b}$$

with the look-ahead distance $\Delta > 0$ and the integral gain $\sigma > 0$. Note that (3.10b) includes an anti-wind-up effect as $\dot{y}_{\text{int}}$ converges to zero when the cross track error $\bar{p}_y$ is large.

As proposed for terrestrial snake robots in [63], the joint offset ϕ_0 in (3.7) will be used to ensure that the orientation θ tracks the desired angle θ_{ref} in (3.10a), and the error $\widetilde{\theta} = \theta - \theta_{\text{ref}}$ thus goes to zero. Along the lines of the derivation for terrestrial snake robots in [65], from expanding (3.5e) and inserting the relation $\boldsymbol{\phi} = \widetilde{\boldsymbol{\phi}} + \boldsymbol{\phi}_{\text{ref}}$, we see that choosing

$$\phi_0 = \frac{1}{\lambda_2 v_{t,\text{rel}}} \left[\ddot{\theta}_{\text{ref}} + \lambda_1 \dot{\theta}_{\text{ref}} - k_\theta (\theta - \theta_{\text{ref}}) \right] - \frac{1}{N-1} \sum_{i=1}^{N-1} \alpha g(i) \sin\left(\omega t + (i-1)\delta\right) \tag{3.11}$$

yields the following error dynamics of the orientation angle:

$$\dddot{\widetilde{\theta}} + \lambda_1 \ddot{\widetilde{\theta}} + k_\theta \dot{\widetilde{\theta}} = \frac{\lambda_2}{N-1} v_{t,\text{rel}} \bar{\mathbf{e}}^T \dot{\widetilde{\boldsymbol{\phi}}}. \tag{3.12}$$

Remark 3.4. In (3.11), a singularity will occur when $v_{t,\text{rel}} = 0$. Note, however, that by Assumption 8, $v_{t,\text{rel}} > 0$. When implementing the control system, the singularity problem can be circumvented by only starting the orientation controller after the robot has gained a sufficiently large forward velocity.

Remark 3.5. Note that, just like in [65], the first and second time derivatives of ϕ_0 and θ_{ref} are required for the implementation of the gait controller (3.8) and the orientation controller (3.11), respectively. In [65], the analytical expressions for $\dot{\phi}_0, \ddot{\phi}_0$ and $\dot{\theta}_{\text{ref}}, \ddot{\theta}_{\text{ref}}$ are omitted and instead it is proposed to obtain the required time derivatives using a 3rd-order low-pass filtering reference model which satisfies $\lim_{t\to\infty} x_{\text{ref}} = r$, with the commanded state reference $r \in \mathbb{R}$:

$$\frac{\mathrm{d}}{\mathrm{d}t} \begin{bmatrix} x_{\text{ref}} \\ \dot{x}_{\text{ref}} \\ \ddot{x}_{\text{ref}} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -\omega_{\text{f}}^3 & -(2\zeta_{\text{f}}+1)\omega_{\text{f}}^2 & -(2\zeta_{\text{f}}+1)\omega_{\text{f}} \end{bmatrix} \begin{bmatrix} x_{\text{ref}} \\ \dot{x}_{\text{ref}} \\ \ddot{x}_{\text{ref}} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \omega_{\text{f}}^3 \end{bmatrix} r. \tag{3.13}$$

Also in this thesis, the 3rd order low-pass filtering reference model is employed for the required time derivatives of both ϕ_0 and θ_{ref}, analogously to the approach in [65]. Incorporating these models into the analysis remains a theoretical gap. However, in this thesis, we made the additional assumption (Assumption 8) that the robot moves with a constant forward velocity. This assumption makes sure that analytical expressions can indeed be found for the signals $\dot{\phi}_0, \ddot{\phi}_0$ and $\dot{\theta}_{\text{ref}}, \ddot{\theta}_{\text{ref}}$ that are required for the terrestrial robot in [65], thus completing the theoretical analysis for terrestrial snake robots and underwater snake robots that are not exposed to ocean currents.

The reason why the low-pass filtering reference models are still included in this thesis has been pointed out in [7] for a similar control system for surface vessels: In the presence of an ocean current and under the assumption that only relative velocity measurements are available for feedback, the unknown ocean current will enter the analytical expression for $\dot{\theta}_{\text{ref}}$. This can be circumvented by using the

low-pass filtering reference model (3.13) instead of the analytical expressions for the implementation of $\dot{\phi}_0, \ddot{\phi}_0$ and $\dot{\theta}_{\text{ref}}, \ddot{\theta}_{\text{ref}}$. This implementation method is also beneficial from a practical perspective, since it makes sure that the signals sent to the controller are sufficiently smooth. Another alternative is the use of an ocean current observer to obtain the required derivatives of θ_{ref}, which is the approach taken in Chapter 4 of this thesis.

3.4.2 Analysis of the closed-loop system

This section presents the conditions under which the control system proposed in Section 3.4.1 is guaranteed to achieve the control objectives formulated in Section 3.3.

Theorem 3.1. *Consider a fully submerged, neutrally buoyant underwater snake robot described by (3.5) that moves with a planar sinusoidal gait, exposed to a constant irrotational current. Suppose that Assumptions 2, 8 and 9 are fulfilled. If the look-ahead distance Δ and the integral gain σ are chosen such that*

$$\Delta > \frac{|X|+2V_{\text{c,max}}}{|Y|}\left[\frac{5}{4}\frac{V_{\text{max}}+V_{\text{c,max}}+\sigma}{V_{\text{min}}-V_{\text{c,max}}-\sigma}+1\right], \tag{3.14a}$$

$$0 < \sigma < V_{\text{min}} - V_{\text{c,max}}, \tag{3.14b}$$

then the control system described in Section 3.4.1 guarantees that the control objectives (3.6a) and (3.6b) are achieved. Control objective (3.6b) is met with

$$\theta^{\text{eq}} = -\arctan\left(\frac{V_y}{\sqrt{v_{t,\text{rel}}^2 - V_y^2}}\right). \tag{3.15}$$

Proof. The proof will be given in Section 3.4.3. □

3.4.3 Proof of Theorem 3.1

The main theorem will be proved in three steps, applying cascaded systems theory that has been presented in [67, 85]. For more details on the theorems that will be used in the following proof, the reader is referred to the summary in Appendix A and the theoretical background in [67, 85]. A definition of the stability notions can be found in Definitions 12 and 13.

The first step of the proof is to transform the complete system to a cascaded system. In the second step we consider the stability of the perturbing system. In the third step the stability of the nominal perturbed system will be analysed. Finally, a bound on the interconnection term will be derived, which concludes the stability proof.

The dynamics of the cross-track error $\bar{p}_y$ and the relative normal velocity $v_{n,\text{rel}}$ are obtained from (3.10b), (3.5c), and (3.5f):

$$\begin{aligned}
\dot{y}_{\text{int}} &= \frac{\Delta \bar{p}_y}{(\bar{p}_y+\sigma y_{\text{int}})^2+\Delta^2}, \\
\dot{\bar{p}}_y &= v_{t,\text{rel}} \sin\theta + \bar{v}_{n,\text{rel}} \cos\theta + V_y, \\
\dot{\bar{v}}_{n,\text{rel}} &= (X + V_x \cos\theta + V_y \sin\theta) v_\theta + Y \bar{v}_{n,\text{rel}}.
\end{aligned} \tag{3.16}$$

The equilibrium of this system is

$$y_{\text{int}}^{\text{eq}} = \frac{\Delta}{\sigma}\frac{V_y}{\sqrt{v_{t,\text{rel}}^2 - V_y^2}},\ \bar{p}_y^{\text{eq}} = 0,\ \bar{v}_{n,\text{rel}}^{\text{eq}} = 0. \tag{3.17}$$

With $\theta = \theta_{\text{ref}} + \widetilde{\theta}$ and $v_\theta = \dot{\theta}_{\text{ref}} + \dot{\widetilde{\theta}}$, the relations

$$\begin{aligned}
\sin\theta &= \frac{\sin\widetilde{\theta}\Delta}{\sqrt{(\bar{p}_y+\sigma y_{\text{int}})^2+\Delta^2}} - \frac{\cos\widetilde{\theta}(\bar{p}_y+\sigma y_{\text{int}})}{\sqrt{(\bar{p}_y+\sigma y_{\text{int}})^2+\Delta^2}},\\
\cos\theta &= \frac{\cos\widetilde{\theta}\Delta}{\sqrt{(\bar{p}_y+\sigma y_{\text{int}})^2+\Delta^2}} + \frac{\sin\widetilde{\theta}(\bar{p}_y+\sigma y_{\text{int}})}{\sqrt{(\bar{p}_y+\sigma y_{\text{int}})^2+\Delta^2}},\\
v_\theta &= -\frac{\Delta}{(\bar{p}_y+\sigma y_{\text{int}})^2+\Delta^2}(\dot{\bar{p}}_y + \sigma\dot{y}_{\text{int}}) + \dot{\widetilde{\theta}}
\end{aligned} \tag{3.18}$$

can be derived. With (3.18) and the new set of variables $e_1 = y_{\text{int}} - y_{\text{int}}^{\text{eq}}$, $e_2 = \bar{p}_y + \sigma e_1$, $e_3 = \bar{v}_{n,\text{rel}}$, $\boldsymbol{\eta} = [\widetilde{\boldsymbol{\phi}}^{\,T},\ \dot{\widetilde{\boldsymbol{\phi}}}^{\,T}]^T$, and $\boldsymbol{\xi} = [\widetilde{\theta},\ \dot{\widetilde{\theta}}]^T$, the whole system can be re-written as

$$\begin{bmatrix}\dot{e}_1 & \dot{e}_2 & \dot{e}_3\end{bmatrix}^T = \mathbf{A}_e(e_2)\begin{bmatrix}e_1 & e_2 & e_3\end{bmatrix}^T + \mathbf{B}(e_2)f(e_2) + \mathbf{H}_\xi(\boldsymbol{e},\boldsymbol{\xi})\boldsymbol{\xi}, \tag{3.19a}$$

$$\dot{\boldsymbol{\xi}} = \begin{bmatrix}0 & 1\\ -k_\theta & -\lambda_1\end{bmatrix}\boldsymbol{\xi} + \begin{bmatrix}0_{1\times(N-1)} & 0_{1\times(N-1)}\\ \frac{\lambda_2}{N-1}v_{t,\text{rel}}\bar{\mathbf{e}}^T & 0_{1\times(N-1)}\end{bmatrix}\boldsymbol{\eta}, \tag{3.19b}$$

$$\dot{\boldsymbol{\eta}} = \begin{bmatrix}0_{(N-1)\times(N-1)} & \mathbf{I}_{(N-1)}\\ -k_\phi\mathbf{I}_{(N-1)} & -k_{v_\phi}\mathbf{I}_{(N-1)}\end{bmatrix}\boldsymbol{\eta} \tag{3.19c}$$

where $\mathbf{H}_\xi(\boldsymbol{e},\boldsymbol{\xi})$ contains all terms that vanish at $\boldsymbol{\xi} = 0$. For better readability, the arguments of $\mathbf{A}_e, \mathbf{B}$, and $\mathbf{H}_\xi$ will be omitted in the following. The expressions for $\mathbf{A}_e, \mathbf{B}$, $f(e_2)$, and $\mathbf{H}_\xi$ are given in Section 3.A. The closed-loop system (3.19) is a cascaded system with (3.19a) as the perturbed system and equations (3.19b) and (3.19c) as the perturbing system. Furthermore, the perturbing system (3.19b),(3.19c) is a cascaded system by itself. Note that it has the same structure like the perturbing system for terrestrial snake robots in Chapter 8 of [65].

Lemma 3.2. *The origin of the system equations* (3.19b) *and* (3.19c) *is UGES.*

Proof. Both system matrices in (3.19b), (3.19c) are Hurwitz, and the interconnection matrix is bounded. By Proposition A.2, the perturbing system is therefore UGES. □

Next we consider the unperturbed nominal system for $(\boldsymbol{\eta},\boldsymbol{\xi}) = 0$,

$$\begin{bmatrix}\dot{e}_1 & \dot{e}_2 & \dot{e}_3\end{bmatrix}^T = \mathbf{A}_e\begin{bmatrix}e_1 & e_2 & e_3\end{bmatrix}^T + \mathbf{B}f(e_2). \tag{3.20}$$

The system (3.20) has a similar structure like the nominal system in [12].

Lemma 3.3. *The nominal system* (3.20) *is USGES with a quadratic Lyapunov function* $V = \frac{1}{2}\sigma^2 e_1^2 + \frac{1}{2}e_2^2 + \frac{1}{2}\mu e_3^2,\ \mu > 0.$

Proof. The proof is given in Section 3.B. □

According to Theorem A.1, the cascaded system (3.19) is UGAS, when Lemmata 3.2 and 3.3 hold and the interconnection term $\mathbf{H}_\xi$ is bounded by $\|\mathbf{H}_\xi\| \leq \mathcal{F}_1(\boldsymbol{\xi}) + \mathcal{F}_2(\boldsymbol{\xi})\|\boldsymbol{e}\|$.

Lemma 3.4. *The induced 2-norm of the interconnecting matrix $\mathbf{H}_\xi$ in (3.19a) is trivially bounded by $\|\mathbf{H}_\xi\|_2 \leq \mathcal{F}_1 + \mathcal{F}_2\|\boldsymbol{e}\|_2$, where $\mathcal{F}_1$ and $\mathcal{F}_2$ are strictly positive constants.*

Proof. The proof is given in Section 3.C. □

With these three lemmata we can now conclude that the complete system (3.19) is UGAS. Since both nominal systems are in addition USGES, the system (3.19) is by Proposition A.2 also USGES. Hence, the control objectives are achieved with θ^{eq} defined in (3.15). ■

Remark 3.6. Note that the exponential stability property of the control system provides some robustness to disturbances and modelling errors, cf. Lemmata 9.1-9.2 in [48].

3.5 Simulation example

This section presents simulation results that demonstrate the performance of the control system proposed in Section 3.4.1.

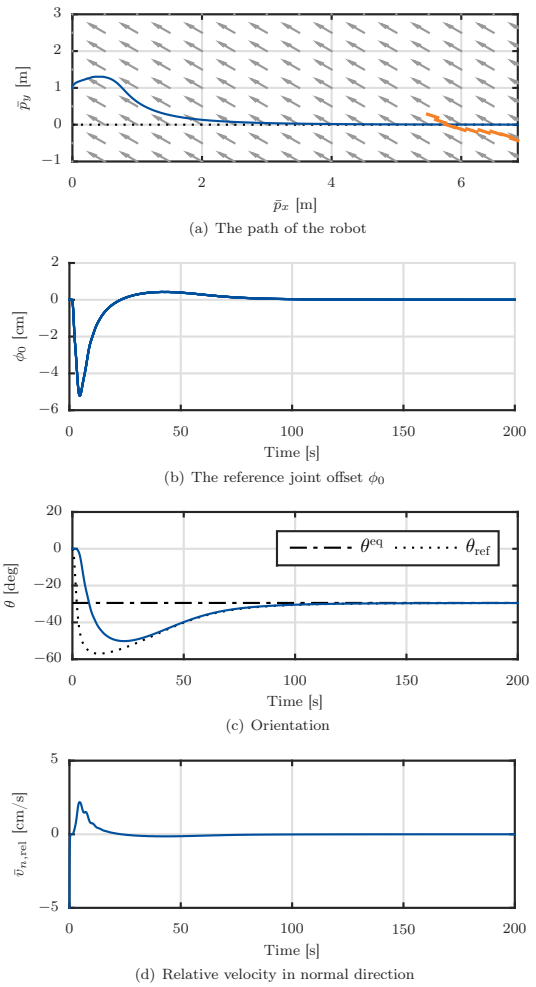
3.5.1 Simulation set-up

The model of the robot and the path-following control system were implemented and simulated in Matlab R2014b. The dynamics was computed using the `ode45` solver with both the relative and absolute error tolerance set to 10^{-4}.

A snake robot with $N = 10$ links was considered. The simulation parameters were chosen in accordance with the parameters of the physical snake robot Mamba [45] and can be found in Table 2.3. From these values, the distance ϵ was computed by (3.3) as $\epsilon = -34.3$ cm. The robot was exposed to a constant irrotational ocean current $\mathbf{v}_{\text{c}} = [-5,\ 5]^T$ cm/s. The parameters for the gait reference signal (3.7) were set to $\alpha = 7$ cm, $\omega = 120°/\text{s}, \delta = 40°$ and the scaling function to $g(i) = 1$. The gains for the control system were chosen as follows: $k_\phi = 20, k_{v_\phi} = 5, k_\theta = 0.5$. The look-ahead distance for the guidance law was chosen as $\Delta = 90$ cm and the integral gain as $\sigma = 2\ \frac{\text{cm}}{\text{s}}$.

For the time derivatives of ϕ_0 and θ_{ref} that are required for the controller, third-order low-pass filter reference models according to (3.13) were implemented. More details on these reference models can be found in Appendix C.2 in [65]. The parameters of the reference models were chosen as $\omega_{\text{f}} = \frac{\pi}{2}, \zeta_{\text{f}} = 1$.

The initial position of the robot was set to $\bar{p}_x = 0, \bar{p}_y = 1$ m, the initial orientation was $\theta = 0°$, i.e. aligned with the desired path, and the initial joint coordinates were $\boldsymbol{\phi} = 0$. All initial velocities were set to zero.



(a) The path of the robot

(b) The reference joint offset ϕ_0

(c) Orientation

(d) Relative velocity in normal direction

Figure 3.4: Simulation results: Straight line path-following for an underwater snake robot with $N = 10$ links initially headed along the desired path and with an initial distance to the path of $\bar{p}_y = 1$ m.

3.5.2 Simulation results

The results of the simulation are visualized in Figure 3.4. The position of the CM can be seen in Figure 3.4(a). After being dragged away by the ocean current in the beginning, the robot turns and converges nicely to the path. The control input ϕ_0 is visualized in Figure 3.4(b). The orientation of the robot over time can be seen in Figure 3.4(c). It converges fast towards θ_{ref} provided by the integral LOS guidance law, and subsequently towards the constant θ^{eq}, which was calculated from (3.15). Figure 3.4(d) shows the relative velocity in the normal direction. It can be seen that $\bar{v}_{n,\text{rel}}$ converges to zero.

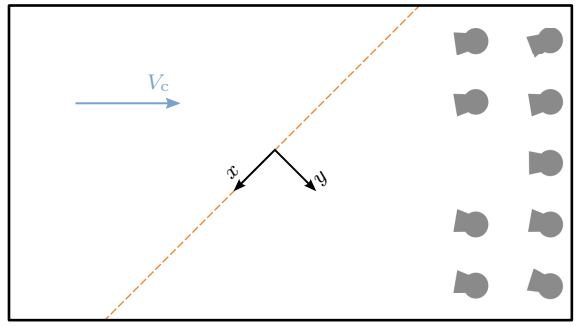


Figure 3.5: The coordinate transformation: the x-y-frame is rotated by 45° with respect to the flow, such that the current has a negative x-component and a positive y-component. The cameras of the motion capture system (displayed in grey) are mounted on one end of the tank. The path is indicated in orange.

3.6 Experimental validation

In this section we present experimental results that demonstrate the performance of the control system proposed in Section 3.4.1 and validate the theoretical analysis in Section 3.4.2.

3.6.1 Experimental set-up

Experimental tests of the integral LOS control system proposed in Section 3.4.1 were performed in the North Sea Centre Flume Tank[1]. The flume tank is 30 m long, 8 m wide, and 6 m deep, and is equipped with four propellers that can generate a circulating flow of up to 1 m/s. For our tests, nine cameras of the Qualisys motion capture system were mounted on one end of the tank. During the experiments, the global coordinate frame was rotated by 45° with respect to the basin, such that the generated current, which is aligned with the long side of the tank, had both an x and a y component. The coordinate transformation is sketched in Figure 3.5. The snake robot Mamba [66] served as the test platform for the experiments. In order to obtain the necessary measurements, reflective markers were attached to the robot tail, and the angle and position measurements of these markers were obtained from the external motion capture system. More details on the snake robot Mamba can be found in Appendix B.

3.6.2 Implementation of the control system

The control system presented in Section 3.4.1 was implemented in LabVIEW 2013. The built-in proportional controllers of the single servo motors in the robot replaced the low level control law (3.8), because the theoretical feedback linearizing control law (3.8a) requires torque control while the joints of Mamba are position controlled.

[1]The North Sea Centre Flume Tank – Managed and operated by SINTEF Fisheries and Aquaculture in Hirtshals, Denmark.

Table 3.1: The control gains of the integral LOS path-following control system.

	α	$g(i)$	ω	δ	k_θ	λ_1	λ_2	Δ	σ
Lateral undulation	30°	1	110°	40°	0.8	0.5	0.2	2 m	0.5
Eel-like motion	50°	$\frac{N-i}{N+1}$	130°	40°	0.9	0.5	0.2	2 m	0.3

This does not invalidate the theoretical control structure, because the cascaded analysis just requires that the joint error dynamics are exponentially stabilized, regardless which controller is used.

The control input ϕ_0, which is used to induce turning motion, is modelled as a linear displacement in the control-oriented model. In accordance with Section 2.4, the control-oriented model that the control system is based on captures the qualitative behaviour of a robot with revolute joints. Therefore, (3.11) was implemented as the orientation controller. The model parameters of the control-oriented model, λ_1 and λ_2, that show up in the orientation controller (3.11) were treated as control gains analogously to the implementation in [65], where a similar control system was tested with a terrestrial snake robot. In order to implement the orientation controller (3.11), the relative velocity $v_{t,\text{rel}}$ needed to be approximated from the data of the external motion capture system. The absolute velocity v_t was estimated as the displacement of the CM divided by a sampling interval of 2 s. The relative velocity was then approximated by subtracting the current speed from the absolute velocity v_t. This way of approximating $v_{t,\text{rel}}$ is only accurate when the robot is heading against the current and will become less accurate when the robot turns. However, this was a reasonable approximation since the current was mainly opposing the forward motion of the robot, and the calculations were significantly simplified. Note that the estimate of the relative velocity $v_{t,\text{rel}}$ is the only velocity measurement that enters the control system, the knowledge of the current and the absolute velocity were merely used to approximate the relative velocity. In order to obtain smooth time derivatives of the orientation reference θ_{ref}, the commanded angle θ_{ref} was passed through a 3rd-order low-pass filtering reference model. The parameters of the reference model were $T = \frac{1}{2\pi}$ and $\zeta = 1$. Finally, in order to avoid self-collision and provide a smooth reference to the physical robot, the orientation control input ϕ_0 was saturated at $\phi_{0,\max} = \pm 20°$, and filtered with a first-order low-pass filter with a cut-off frequency of 1.25 Hz. In order to avoid singularities in the orientation control input ϕ_0, the orientation controller was started after 2 s.

The guidance law (3.10) was included by numerically integrating (3.10b) in LabVIEW. Since the exact model parameters that are required for calculating the offset ϵ are unknown, it was assumed that the robot turns about its CM, i. e. the parameter was set to $\epsilon = 0$, and $\bar{p}_y$ was approximated by the CM position p_y. In order to close the feedback control loop, angle and position measurements of the markers attached to the robot tail were obtained from the external motion capture system. The angles of each single link and the position of the CM, p_x, p_y, were obtained from the angle and position measurements of the markers in combination with the single joint angles. The orientation of the robot was calculated according to Definition 3, by the average of the link angles, $\bar{\theta}$.

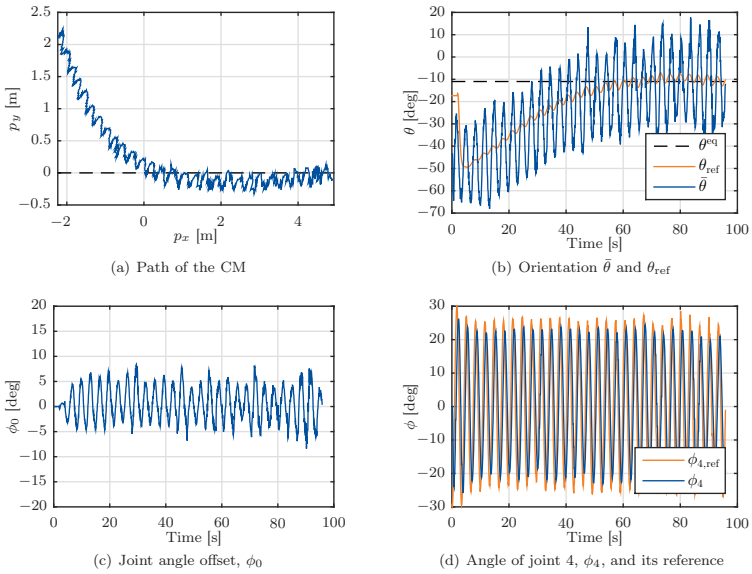
The control gains of the system can be found in Table 3.1. Since a lower bound

on the velocity of the robot was not known a priori, it was not feasible to use the theoretical condition (3.14b) to determine a value for σ, and consequently (3.14a) could not be employed to find a bound on Δ. In this light, we made sure to choose a sufficiently large Δ, and a sufficiently small σ in order to converge to the path.

3.6.3 Experimental results

The control system was tested experimentally in four different scenarios. In the first two scenarios, the robot was moving with the gait lateral undulation and was exposed to a constant current of $V_c = 7$ cm/s. In the first case, it was initially headed towards the path, in the second scenario it was initially headed approximately parallel to the path. Similarly, in the third and fourth scenario, the robot was initially oriented towards the path and approximately parallel to the path respectively, and propelled with eel-like motion against a current of $V_c = 5$ cm/s. In all cases, the robot was initially straightened and kept in a fixed position.

The results of the four different test scenarios are presented in Figures 3.6, 3.7, 3.8, and 3.9. The path of the CM of the robot is presented in Figures 3.6(a), 3.7(a), 3.8(a), and 3.9(a), respectively. In the first and fourth scenario, there was a small overshoot, but nevertheless, the robot approached the path and stayed on it in all four scenarios. The CM did not stay constantly on the path after convergence, but oscillated about it. This was expected, since the oscillations of the CM are a consequence of the sinusoidal gait and merely not captured by the simplifications due to the model assumptions. The same effect can also be observed for the orientation of the robot, $\bar{\theta}$, that is plotted in Figures 3.6(b), 3.7(b), 3.8(b), and 3.9(b). The small overshoots in the first and fourth scenario can also be seen in these plots. After convergence, the reference signal θ_{ref} oscillated about the steady-state crab angle θ^{eq}. This allowed the robot to side-slip along the path and thus compensate for the sideways component of the current. The oscillating nature of the reference signal θ_{ref} is a consequence of the deviations of the CM from the path. The steady-state crab angle θ^{eq} that is depicted in Figures 3.6(b), 3.7(b), 3.8(b), and 3.9(b) was determined from (3.15) a posteriori. To this end, we needed the current component V_y and the relative velocity $v_{t,\text{rel}}$. The current component V_y was directly calculated as $V_y = V_c \cos(45°)$ since both the direction and magnitude V_c of the current are known in the lab, and the relative velocity $v_{t,\text{rel}}$ was approximated by the average speed $\bar{v}_{\text{rel}}$. The average speed $\bar{v}_{\text{rel}}$ was calculated a posteriori from the norm of the relative velocity, and the relative velocity components in x and y were obtained by subtracting the current components V_x and V_y from the velocity components of the CM, $\dot{p}_x$ and $\dot{p}_y$. These had been extracted from the position measurements that were obtained during the experiments by using finite differences with a time step of 0.2 s. In order to calculate the average speed for lateral undulation, the measurements of the first scenario were evaluated, and for the average speed for eel-like motion, those of the third scenario were used. It can be seen from the figures that the theoretical result (3.15) predicted θ^{eq} correctly. The control input for the turning motion, ϕ_0, is shown in Figures 3.6(c), 3.7(c), 3.8(c), and 3.9(c), respectively. After convergence of the robot, ϕ_0 was oscillating about zero, because the robot stayed approximately at the constant orientation θ^{eq}. The reference signal and the measured signal of joint number four are displayed



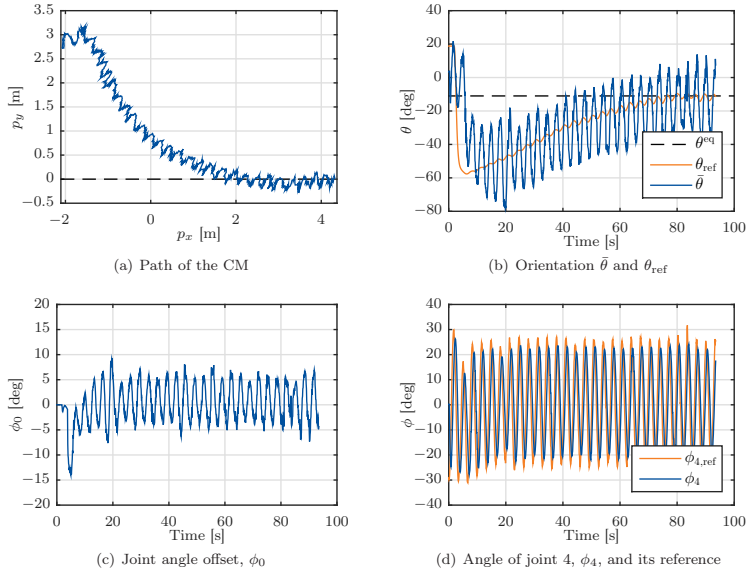
(a) Path of the CM

(b) Orientation $\bar{\theta}$ and θ_{ref}

(c) Joint angle offset, ϕ_0

(d) Angle of joint 4, ϕ_4, and its reference

Figure 3.6: Experimental results of the first scenario: Model-based integral LOS path-following with a flow speed $V_{\text{c}} = 7$ cm/s, the gait lateral undulation, and the robot initially headed towards the path.



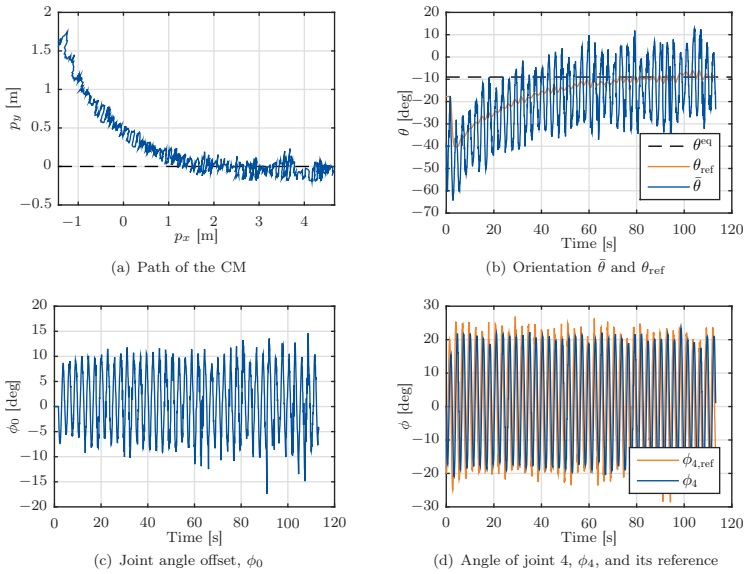
(a) Path of the CM

(b) Orientation $\bar{\theta}$ and θ_{ref}

(c) Joint angle offset, ϕ_0

(d) Angle of joint 4, ϕ_4, and its reference

Figure 3.7: Experimental results of the second scenario: Model-based integral LOS path-following with a flow speed $V_{\text{c}} = 7$ cm/s, the gait lateral undulation, and the robot initially headed parallel to the path.



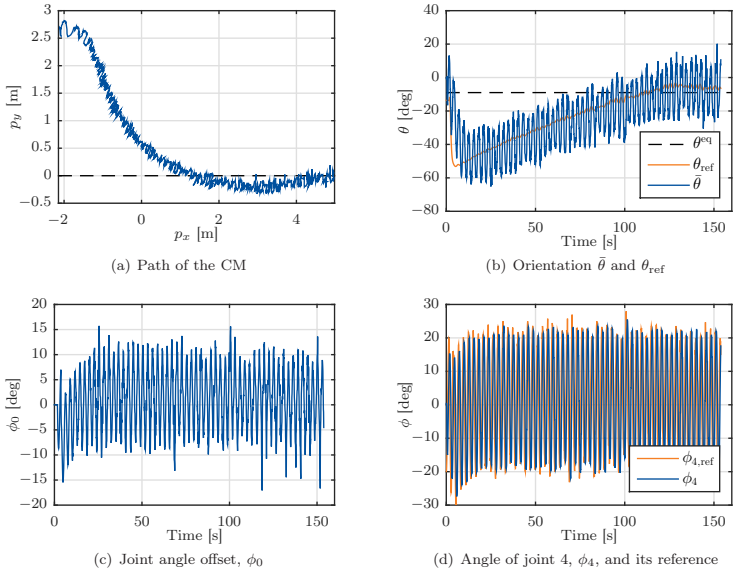
(a) Path of the CM

(b) Orientation $\bar{\theta}$ and θ_{ref}

(c) Joint angle offset, ϕ_0

(d) Angle of joint 4, ϕ_4, and its reference

Figure 3.8: Experimental results of the third scenario: Model-based integral LOS path-following with a flow speed $V_{\text{c}} = 5$ cm/s, eel-like motion, and the robot initially headed towards the path.



(a) Path of the CM

(b) Orientation $\bar{\theta}$ and θ_{ref}

(c) Joint angle offset, ϕ_0

(d) Angle of joint 4, ϕ_4, and its reference

Figure 3.9: Experimental results of the fourth scenario: Model-based integral LOS path-following with a flow speed $V_{\text{c}} = 5$ cm/s, eel-like motion, and the robot initially headed parallel to the path.

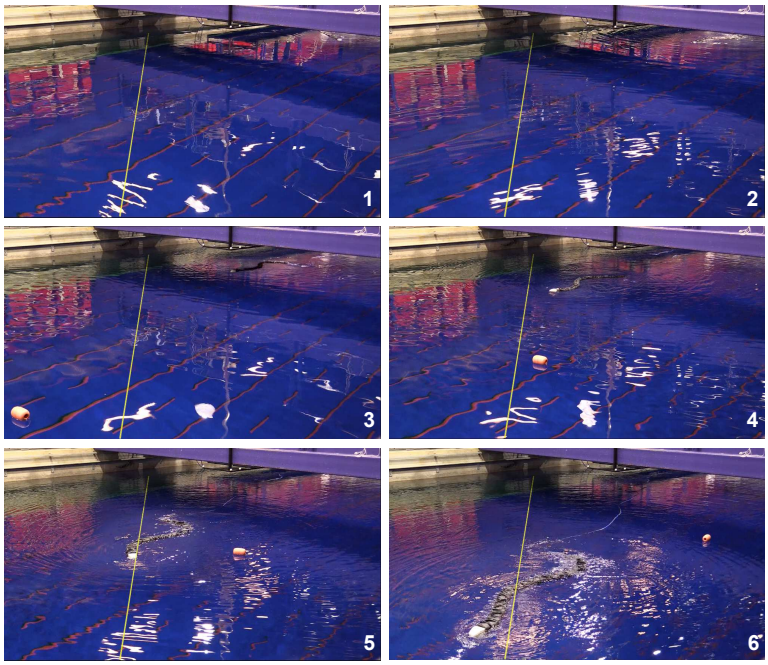


Figure 3.10: The snake robot Mamba during a test run of scenario two. The yellow line indicates the path and the buoy visualizes the current effect. The robot is initially approximately parallel to the path (1), turns towards it (2), approaches the path (3) and (4) and subsequently follows it (5) while side-slipping (6).

in Figures 3.6(d), 3.7(d), 3.8(d), and 3.9(d). The measured joint angle clearly also tracked its reference. Pictures of the physical robot during the second scenario are presented in Figure 3.10.

The experimental results validate the model-based integral LOS path-following controller in the presence of constant irrotational currents. The steady-state crab angle θ^{eq} was predicted correctly by the analytical relation in (3.15). The control objectives in (3.6) were satisfied in the sense that the robot oscillates about the desired values after convergence. These oscillations are a result of the sinusoidal motion that the robot conducts and that are not captured by the simplifying model assumptions that were used for the control design. It was therefore not expected from the theory that these oscillations would be suppressed by the model-based path-following controller.

3.7 Chapter summary

This chapter has presented a control system that enables an underwater snake robot to converge towards and follow a straight path in the presence of constant irrotational ocean currents. The control design was based on the assumption that the robot is fully submerged, neutrally buoyant, and moving with a planar si-

nusoidal gait. The guidance method that was used in the path-following control system was a LOS guidance scheme, which was augmented with integral action in order to eliminate the steady-state error that the original LOS guidance would give in the presence of currents. This was achieved by allowing the robot to head towards a look-ahead point upstream of the path and thus travel at a non-zero crab angle. The proposed control approach applied an exponentially stabilizing orientation controller in order to steer the robot towards the desired orientation obtained by the integral LOS guidance law. Using cascaded systems theory, the system has been formally shown to be UGAS and USGES. Experimental results were presented that verify the concept of the control system. Furthermore, the experimental results showed that the crab angle was correctly predicted by the theoretical analysis.

3.A Function definitions

The matrix $\mathbf{A}_e(e_2)$ is defined as

$$\mathbf{A}_e(e_2) = \begin{bmatrix} -\frac{\sigma\Delta}{\nu_e(e_2)} & \frac{\Delta}{\nu_e(e_2)} & 0 \\ -\frac{\sigma^2\Delta}{\nu_e(e_2)} & \frac{\sigma\Delta}{\nu_e(e_2)} - \frac{v_{t,\mathrm{rel}}}{\sqrt{\nu_e(e_2)}} & \frac{\Delta}{\sqrt{\nu_e(e_2)}} \\ \frac{\sigma^2\Delta^2\bar{X}(e_2)}{\nu_e(e_2)^2} & \frac{\Delta\bar{X}(e_2)v_{t,\mathrm{rel}}}{\nu_e(e_2)^{3/2}} - \frac{\sigma\Delta^2\bar{X}(e_2)}{\nu_e(e_2)^2} & Y - \frac{\Delta^2\bar{X}(e_2)}{\nu_e(e_2)^{3/2}} \end{bmatrix} \tag{3.21}$$

with $\nu_e(e_2) = (e_2 + \sigma y_{\mathrm{int}}^{\mathrm{eq}})^2 + \Delta^2$ and $\bar{X}(e_2) = X + \frac{V_x\Delta - V_y(e_2+\sigma y_{\mathrm{int}}^{\mathrm{eq}})}{\sqrt{(e_2+\sigma y_{\mathrm{int}}^{\mathrm{eq}})^2+\Delta^2}}$. Furthermore,

$$\mathbf{B}(e_2) = \begin{bmatrix} 0 & V_y & -\frac{\Delta\bar{X}(e_2)V_y}{(e_2+\sigma y_{\mathrm{int}}^{\mathrm{eq}})^2+\Delta^2} \end{bmatrix}^T, \tag{3.22}$$

$$f(e_2) = 1 - \frac{\sqrt{(\sigma y_{\mathrm{int}}^{\mathrm{eq}})^2+\Delta^2}}{\sqrt{(e_2+\sigma y_{\mathrm{int}}^{\mathrm{eq}})^2+\Delta^2}}, \tag{3.23}$$

and

$$\mathbf{H}_\xi(\boldsymbol{e},\boldsymbol{\xi}) = \begin{bmatrix} 0 & 0 \\ \frac{\gamma_1}{\widetilde{\theta}} & 0 \\ -\frac{\Delta\bar{X}(e_2)-\Delta\gamma_2}{(e_2+\sigma y_{\mathrm{int}}^{\mathrm{eq}})^2+\Delta^2}\frac{\gamma_1}{\widetilde{\theta}} - \frac{\Delta f_2(\boldsymbol{e})+\Delta V_y f(e_2)}{(e_2+\sigma y_{\mathrm{int}}^{\mathrm{eq}})^2+\Delta^2}\frac{\gamma_2}{\widetilde{\theta}} & \gamma_2 + \bar{X}(e_2) \end{bmatrix}, \tag{3.24}$$

with

$$\gamma_1(e_2,e_3,\widetilde{\theta}) = \sin\widetilde{\theta}\frac{\Delta v_{t,\mathrm{rel}}+(e_2+\sigma y_{\mathrm{int}}^{\mathrm{eq}})e_3}{\sqrt{(e_2+\sigma y_{\mathrm{int}}^{\mathrm{eq}})^2+\Delta^2}} + (1-\cos\widetilde{\theta})\frac{(e_2+\sigma y_{\mathrm{int}}^{\mathrm{eq}})v_{t,\mathrm{rel}}-\Delta e_3}{\sqrt{(e_2+\sigma y_{\mathrm{int}}^{\mathrm{eq}})^2+\Delta^2}}, \tag{3.25}$$

$$\gamma_2(e_2,\widetilde{\theta}) = \sin\widetilde{\theta}\frac{\Delta V_y+(e_2+\sigma y_{\mathrm{int}}^{\mathrm{eq}})V_x}{\sqrt{(e_2+\sigma y_{\mathrm{int}}^{\mathrm{eq}})^2+\Delta^2}} + (1-\cos\widetilde{\theta})\frac{(e_2+\sigma y_{\mathrm{int}}^{\mathrm{eq}})V_y-\Delta V_x}{\sqrt{(e_2+\sigma y_{\mathrm{int}}^{\mathrm{eq}})^2+\Delta^2}}, \tag{3.26}$$

$$\begin{aligned} f_2(\boldsymbol{e}) = &-\frac{\sigma^2\Delta}{(e_2+\sigma y_{\mathrm{int}}^{\mathrm{eq}})^2+\Delta^2}e_1 + \Big(\frac{\sigma\Delta}{(e_2+\sigma y_{\mathrm{int}}^{\mathrm{eq}})^2+\Delta^2} - \frac{v_{t,\mathrm{rel}}}{\sqrt{(e_2+\sigma y_{\mathrm{int}}^{\mathrm{eq}})^2+\Delta^2}}\Big)e_2 \\ &+ \frac{\Delta}{\sqrt{(e_2+\sigma y_{\mathrm{int}}^{\mathrm{eq}})^2+\Delta^2}}e_3. \end{aligned} \tag{3.27}$$

3.B Proof of Lemma 3.3

The structure of the nominal system (3.20) is identical to the system that is presented in [12, 117], where the stability of an integral LOS guidance system for a surface vessel is analysed. The main difference in this chapter is an additional dependence of $\bar{X}$ on e_2. Because of the similar system structure, the same Lyapunov function candidate as in [12, 117] can be used.

With the quadratic Lyapunov function

$$V = \tfrac{1}{2}\sigma^2 e_1^2 + \tfrac{1}{2}e_2^2 + \tfrac{1}{2}\mu e_3^2 = \tfrac{1}{2}\boldsymbol{e}^T\mathbf{P}\boldsymbol{e}, \tag{3.28}$$

the notation

$$\bar{e}_i = \frac{e_i}{\sqrt{(e_2+\sigma y_{\text{int}}^{\text{eq}})^2+\Delta^2}}, \quad i = 1, 2, \tag{3.29}$$

Assumption 2, the bound $|f(e_2)| \leq \frac{|e_2|}{\sqrt{(e_2+\sigma y_{\text{int}}^{\text{eq}})^2+\Delta^2}}$ [12], and the easily verifiable bound $|\bar{X}(e_2)| \leq |X| + 2V_{\text{c,max}}$, the following bound on $\dot{V}$ can be found:

$$\dot{V} \leq -W_1(|\bar{e}_1|, |e_3|) - W_2(|\bar{e}_2|, |e_3|), \tag{3.30a}$$

$$W_1 = \sigma^3\Delta|\bar{e}_1|^2 - \mu\sigma^2\tfrac{|X|+2V_{\text{c,max}}}{\Delta}|\bar{e}_1||e_3| + \eta\mu(|Y| - \tfrac{|X|+2V_{\text{c,max}}}{\Delta})|e_3|^2, \tag{3.30b}$$

$$W_2 = \Delta\begin{bmatrix}|\bar{e}_2| & |e_3|\end{bmatrix}\begin{bmatrix}\chi_1 & -\chi_2 \\ -\chi_2 & \frac{\chi_2(2\chi_2-1)}{\chi_1}\end{bmatrix}\begin{bmatrix}|\bar{e}_2| \\ |e_3|\end{bmatrix}, \tag{3.30c}$$

where $0 < \eta < 1$, $\chi_1 = V_{\min} - V_{\text{c,max}} - \sigma$, and

$$\chi_2 = (1-\eta)\chi_1\tfrac{\Delta|Y|-(|X|+2V_{\text{c,max}})}{(|X|+2V_{\text{c,max}})(V_{\max}+V_{\text{c,max}}+\sigma)}. \tag{3.31}$$

For (3.30c) and (3.31) to hold, μ is chosen as

$$\mu = \tfrac{\Delta^2(2\chi_2-1)}{(|X|+2V_{\text{c,max}})(V_{\max}+V_{\text{c,max}}+\sigma)}. \tag{3.32}$$

Following standard Lyapunov theory, the nominal system (3.20) is UGAS if V is positive definite and $\dot{V}$ is negative definite, which is equivalent to both W_1, W_2 being positive definite. It is straightforward to verify that the conditions

$$0 < \mu < \tfrac{4\eta\Delta^2(\Delta|Y|-(|X|+2V_{\text{c,max}}))}{\sigma(|X|+2V_{\text{c,max}})^2}, \tag{3.33}$$

$$\Delta > \tfrac{|X|+2V_{\text{c,max}}}{|Y|} \tag{3.34}$$

ensure positive definiteness of V and W_1. Inequality (3.33) can be guaranteed with a proper choice of η, which will be defined in the next paragraph, while (3.34) is implied by condition (3.14a). In order to achieve positive definiteness of W_2, χ_1 and χ_2 have to fulfil $\chi_1 > 0$ and $\chi_2 > 1$. The latter condition also ensures that $\mu > 0$ holds. The first condition, $\chi_1 > 0$, is guaranteed by Assumption 9 and (3.14b), whereas the second one, $\chi_2 > 1$, is implied by (3.14) and the choice $\eta = \frac{1}{5}$, which

also ensures that (3.33) holds. We can therefore conclude that both V, W_1 and W_2 are positive definite, and the equilibrium of the nominal system (3.20) is UGAS.

In addition, the single terms of W_1, W_2 can be assembled into a matrix, which leads to the expression

$$\dot{V} \leq -\left[|\bar{e}_1|,\ |\bar{e}_2|,\ |e_3|\right] \mathbf{Q} \begin{bmatrix} |\bar{e}_1| \\ |\bar{e}_2| \\ |e_3| \end{bmatrix}, \tag{3.35}$$

where

$$\mathbf{Q} = \begin{bmatrix} \sigma^3 \Delta & 0 & -\mu\sigma^2 \frac{|X|+2V_{\mathrm{c,max}}}{2\Delta} \\ 0 & \Delta\chi_1 & -\Delta\chi_2 \\ -\mu\sigma^2 \frac{|X|+2V_{\mathrm{c,max}}}{2\Delta} & -\Delta\chi_2 & \mu\frac{\Delta|Y|-|X|-2V_{\mathrm{c,max}}}{\Delta} \end{bmatrix} \tag{3.36}$$

is a time-invariant, positive definite matrix. From [48] it follows that $\dot{V}$ is bounded by

$$\dot{V} \leq -\lambda_{\min}[|\bar{e}_1|,\ |\bar{e}_2|,\ |e_3|] \begin{bmatrix} |\bar{e}_1| \\ |\bar{e}_2| \\ |e_3| \end{bmatrix}, \tag{3.37}$$

with $\lambda_{\min}$ being the smallest eigenvalue of $\mathbf{Q}$. It is pointed out in [23] that UGES cannot be achieved for LOS guidance law error dynamics, because the system gain of the cross-track error e_2 decreases with the magnitude of the cross-track error. The same holds for the dynamics of the integral state e_1 [117]. By combining (3.29) and (3.37), this behaviour can also be observed in the structure of $\dot{V}$:

$$\dot{V} \leq -\frac{\lambda_{\min}}{(e_2+\sigma y_{\mathrm{int}}^{\mathrm{eq}})^2+\Delta^2}(|e_1|^2 + |e_2|^2) - \lambda_{\min}|e_3|^2 \tag{3.38}$$

where the denominator leads to a slow convergence rate in e_1 and e_2 for large e_2. In order to prove that the system is still USGES, the function

$$\phi(e_2) = \min\{\lambda_{\min}, \frac{\lambda_{\min}}{(e_2+\sigma y_{\mathrm{int}}^{\mathrm{eq}})^2+\Delta^2}\} \tag{3.39}$$

is defined. It can be shown that for any ball $\mathcal{B}_r = \{|e_2| \leq r\}$,

$$\phi(e_2) \geq \min\{\lambda_{\min}, \frac{\lambda_{\min}}{(r+\kappa)^2+\Delta^2}\} = c(r), \tag{3.40}$$

with the bound on $|\sigma y_{\mathrm{int}}^{\mathrm{eq}}|$ similar to the one in [117]:

$$|\sigma y_{\mathrm{int}}^{\mathrm{eq}}| \leq \frac{\Delta V_{\mathrm{c,max}}}{\sqrt{V_{\min}^2 - V_{\mathrm{c,max}}^2}} = \kappa. \tag{3.41}$$

With (3.40) and (3.38), the following expression holds on any ball $\mathcal{B}_r = \{|e_2| \leq r\}$:

$$\dot{V} \leq -c(r)\|\boldsymbol{e}\|^2. \tag{3.42}$$

From (3.28) follows that

$$\tfrac{1}{2}p_{\min}\|\boldsymbol{e}\|^2 \leq V \leq \tfrac{1}{2}p_{\max}\|\boldsymbol{e}\|^2, \tag{3.43}$$

where $p_{\min} = \min\{\sigma^2, 1, \mu\}$ and $p_{\max} = \max\{\sigma^2, 1, \mu\}$. With (3.43) and (3.42) , it can be seen that

$$\dot{V} \leq -2\tfrac{c(r)}{p_{\max}} V. \tag{3.44}$$

Since (3.44) holds on any ball $\mathcal{B}_r = \{|e_2| \leq r\}$ and the equilibrium of the nominal system (3.20) is UGAS, we can now invoke the Comparison Lemma (Lemma A.3), which leads to the following relation:

$$V(t,x) \leq V(t_0, x(t_0)) e^{-2\frac{c(r)}{p_{\max}}(t-t_0)}. \tag{3.45}$$

With (3.43) it can be concluded that

$$\begin{aligned} \|\boldsymbol{e}(t)\| &\leq \sqrt{2\tfrac{V(t,x)}{p_{\min}}} \leq \sqrt{2\tfrac{V(t_0,x(t_0))e^{-2\frac{c(r)}{p_{\max}}(t-t_0)}}{p_{\min}}} \\ &\leq \sqrt{\tfrac{p_{\max}\|\boldsymbol{e}(t_0)\|^2}{p_{\min}}} e^{-\frac{c(r)}{p_{\max}}(t-t_0)} \\ &\leq \sqrt{\tfrac{p_{\max}}{p_{\min}}} e^{-\frac{c(r)}{p_{\max}}(t-t_0)} \|\boldsymbol{e}(t_0)\| \end{aligned} \tag{3.46}$$

for all $t \geq t_0$ and any $r > 0$. We can thus conclude that the equilibrium of system (3.20) is USGES (Definition 13). □

3.C Proof of Lemma 3.4

According to Appendix A in [48], the induced 2-norm of the matrix $\mathbf{H}_\xi$ satisfies

$$\begin{aligned} \|\mathbf{H}_\xi\|_2 &\leq \sqrt{2} \max_j \sum_{i=1}^{3} |\{\mathbf{H}_\xi\}_{ij}| \\ &\leq \sqrt{2}\Big(|\tfrac{\gamma_1}{\widetilde{\theta}}| + |-\tfrac{\Delta\bar{X}(e_2) - \Delta\gamma_2}{(e_2+\sigma y_{\text{int}}^{\text{eq}})^2+\Delta^2} \tfrac{\gamma_1}{\widetilde{\theta}}| \\ &\quad + |\tfrac{\Delta f_2(\boldsymbol{e}) + \Delta V_y f(e_2)}{(e_2+\sigma y_{\text{int}}^{\text{eq}})^2+\Delta^2} \tfrac{\gamma_2}{\widetilde{\theta}}| + |\gamma_2| + |\bar{X}(e_2)| \Big). \end{aligned} \tag{3.47}$$

The final expression

$$\begin{aligned} &\|\mathbf{H}_\xi\|_2 \leq \mathcal{F}_1 + \mathcal{F}_2 \|\boldsymbol{e}\|_2, \\ &\mathcal{F}_1 = \sqrt{2}\left(2V_{\max} + \tfrac{(|X|+8V_{\text{c,max}})(\Delta+2V_{\max})}{\Delta} \right), \\ &\mathcal{F}_2 = \sqrt{2}\left(\tfrac{4V_{\text{c,max}}(\sigma^2+\sigma+V_{\max}+V_{\text{c,max}}+\Delta)}{\Delta^2} + \tfrac{2(|X|+8V_{\text{c,max}})}{\Delta} + 2 \right) \end{aligned} \tag{3.48}$$

follows from (3.47) with Assumption 8, the bound on $\bar{X}(e_2)$ in Section 3.B, and the following bounds on the single parts of $\mathbf{H}_\xi$:

$$\begin{aligned} &|\tfrac{\gamma_1}{\widetilde{\theta}}| \leq 2V_{\max} + 2|e_3|, && |\tfrac{\gamma_2}{\widetilde{\theta}}| \leq 4V_{\text{c,max}}, \\ &|\gamma_2| \leq 6V_{\text{c,max}}, && |f(e_2)| \leq \tfrac{|e_2|}{\Delta}, \\ &|f_2(\boldsymbol{e})| \leq \tfrac{\sigma^2}{\Delta}|e_1| + \tfrac{\sigma+V_{\max}}{\Delta}|e_2| + |e_3|. \end{aligned} \tag{3.49}$$

□

Chapter 4

Observer-Based Guidance and Control Using Virtual Holonomic Constraints

This chapter investigates the problem of planar manoeuvring control for biologically inspired underwater snake robots that are exposed to unknown ocean currents. The control objective is to make a snake robot that 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 VHCs which encode biologically inspired gaits on the snake robot configuration. The virtual constraints, parametrized by the states of dynamic compensators, are used to regulate the heading and forward velocity 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.

Contributions of this chapter The first contribution of this chapter is extending the control framework that was developed for terrestrial snake robots in [77, 78] to underwater snake robots under the influence of hydrodynamic forces and ocean currents. The analysis shows that the 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 enabling the path-following controller to maintain the performance in the presence of currents. In addition to the lateral undulatory gait that was investigated in [77, 78], it is shown in this chapter that the control framework can also be used along with a more general class of biologically inspired gaits. Finally, the control framework is experimentally validated. In particular, the control system was successfully tested for straight line path-following for different gaits.

Organization of this chapter In Section 4.1, the VHCs are introduced that encode the biologically inspired gait used to propel the robot. The control strategy is presented in Section 4.2. The proposed control system is validated by extensive simulations in Section 4.3 and an experimental study in Section 4.4. The chapter ends with a discussion of the results in Section 4.5 and a summary of the results in Section 4.6.

Publications This chapter is based on [51].

4.1 Forward propulsion using virtual holonomic constraints

As pointed out in Section 1.2.1, the sinusoidal gait that propels a snake robot can be achieved by enforcing VHCs on the robot configuration. The VHCs encode the gait that propels the robot forward, as has been used for terrestrial snake robots in [77, 78]. For underwater snake robots, we generalize the VHC (1.4) from [77, 78] with the scaling function $g(i)$ similar to the time-dependent lateral undulatory gait (1.3) in order to achieve a more general class of gaits:

$$\phi_{i,\mathrm{ref}}(\lambda, \phi_0) = \alpha g(i) \sin\big(\lambda + (i-1)\delta\big) + \phi_0. \tag{4.1}$$

In (4.1), λ and ϕ_0 are states of the controllers

$$\ddot{\lambda} = u_\lambda, \quad \ddot{\phi}_0 = u_{\phi_0}, \tag{4.2}$$

with the new 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 heading, while the state $\dot{\lambda}$ governs the speed of the snake robot. 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 [77, 78].

4.2 Manoeuvring control for underwater snake robots

In this section the control system for manoeuvring control of underwater snake robots in the presence of ocean currents is presented. The control framework for terrestrial robots from [78], 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 manoeuvring control of underwater snake robots.

In [78], a formal stability proof for VHC-based manoeuvring control of terrestrial snake robots was presented. In this chapter, the proof details that are analogous to those developed for terrestrial snake robots in [78] 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.

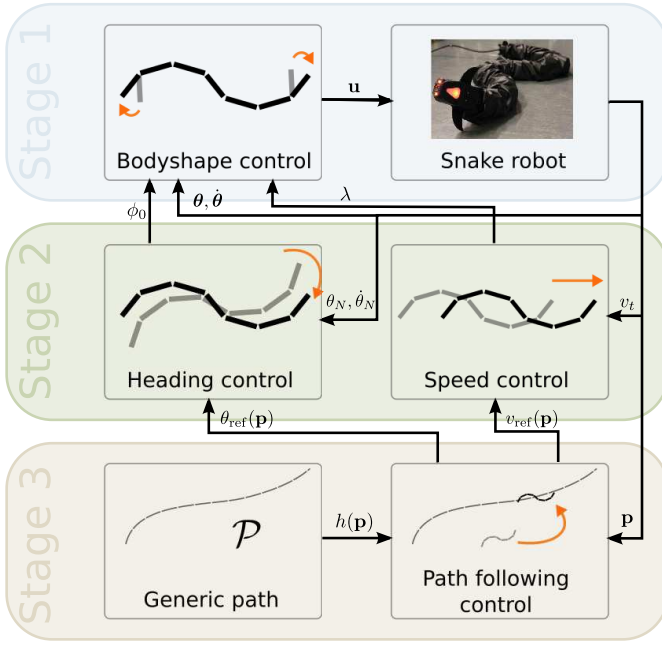


Figure 4.1: The structure of the control system in [78].

4.2.1 A hierarchical 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 terrestrial snake robots in [78]. 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 [78].

The control design approach in [78] is hierarchical in the sense that the design has three main stages corresponding to three prioritized control specifications. Figure 4.1 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 $\mathbf{u}$ of the snake robot is used to stabilize the VHCs (4.1). In [78], 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 (4.1), 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 [78] in order to take into account the perturbing effect of ocean currents.

Remark 4.1. 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 [107]. 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.

4.2.2 Control system design

We now describe an enhancement of the control methodology outlined in Section 4.2.1 to handle underwater snake robots and compensate for the presence of constant unknown ocean currents.

Stage 1: Body shape controller

In order to enforce the biologically inspired VHCs (4.1) on the snake robot configuration, the PD feedback law

$$\mathbf{u} = k_p\big(\mathbf{D}\boldsymbol{\theta} + \boldsymbol{\Phi}(\lambda) + \mathbf{b}\phi_0\big) + k_d\big(\mathbf{D}\dot{\boldsymbol{\theta}} + \boldsymbol{\Phi}'(\lambda)\dot{\lambda} + \mathbf{b}\dot{\phi}_0\big) + k_{u_\lambda}\boldsymbol{\Phi}'(\lambda)u_\lambda + k_{u_\phi}\mathbf{b}u_{\phi_0} \tag{4.3}$$

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 $\boldsymbol{\Phi}(\lambda) = [\alpha\sin(\lambda), \ldots, \alpha\sin(\lambda + (N-2)\delta)]^T \in \mathbb{R}^{N-1}$ contains the sinusoidal part of (4.1). We can see that the new control inputs u_λ, u_{ϕ_0} appear in (4.3). The control law (4.3) 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 (4.3) 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 4.2. In [78], 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 work, 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.

Stage 2: Velocity controller

In the second stage of the control system, feedback laws for the new control inputs u_{ϕ_0}, u_λ are 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 [78] 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

$$\mathbf{v}_{\text{rel}} = \begin{bmatrix} v_{t,\text{rel}} \\ v_{n,\text{rel}} \end{bmatrix} = \mathbf{R}_{\theta_N}^T \big(\dot{\mathbf{p}} - \mathbf{v}_{\text{c}}\big). \tag{4.4}$$

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 (4.4) 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 Stage 3 below. In light of (4.4), 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 $\dot{\mathbf{p}} - \mathbf{v}_{\text{c}}$ to $\mathbf{R}_{\theta_N}[v_{\text{ref}}(\cdot), 0]^T$. This fact is exploited in the design of Stage 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 [78] that this can be achieved by the control law

$$u_{\phi_0} = -\frac{1}{\epsilon}\big(\dot{\tilde{\theta}}_N + k_n \tilde{\theta}_N\big) - k_1 \phi_0 - k_2 \dot{\phi}_0. \tag{4.5}$$

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 $v_{t,\text{rel}}$ and $v_{n,\text{rel}}$ defined in (4.4) converge to the reference speed $v_{\text{ref}}(\cdot)$ and a small neighbourhood of the origin, respectively. The feedback law

$$u_\lambda = -k_z\big(\dot{\lambda} + k_\lambda \Delta v_{t,\text{rel}}\big) \tag{4.6}$$

achieves this control objective. In (4.6), the control gains k_z and k_λ are positive constants, and $\Delta v_{t,\text{rel}} = v_{t,\text{rel}} - v_{\text{ref}}(\cdot)$ 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 [3] or from DVLs without bottom lock [16].

Remark 4.3. The speed controller presented in this chapter is a simplified version of the speed controller in [78]. It can be shown that the feedback law (4.6) makes the steady-state velocity error arbitrarily small if the control gain k_λ is chosen sufficiently large.

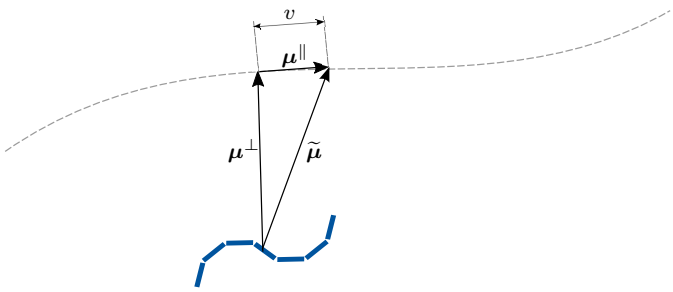


Figure 4.2: The path-following controller from [78]. The velocity reference vector $\widetilde{\boldsymbol{\mu}}$ is composed of the normal component $\boldsymbol{\mu}^{\perp}$, used to steer the robot towards the path, and the parallel component $\boldsymbol{\mu}^{\parallel}$, used to regulate the speed of the robot along the path.

Stage 3: Path-following controller and ocean current observer

For underwater snake robots, the third control stage needs to be adapted compared to [78] 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 [78], and an ocean current observer.

We start with reviewing the path-following controller from [78], 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 [78] relies on the observation that once the speed and heading controllers have converged, in the absence of ocean current one has $\dot{\mathbf{p}} \approx \mathbf{R}_{\theta_{\text{ref}}}[v_{\text{ref}}(\mathbf{p}), 0]^T$. Letting $\widetilde{\boldsymbol{\mu}}(\mathbf{p}) := \mathbf{R}_{\theta_{\text{ref}}}[v_{\text{ref}}(\mathbf{p}), 0]^T$, $\widetilde{\boldsymbol{\mu}}(\mathbf{p})$ is viewed as a control input and designed to make $\mathbf{p}$ converge to the path $\mathcal{P}$. The resulting control law is

$$\widetilde{\boldsymbol{\mu}}(\mathbf{p}) = \underbrace{-\frac{dh_p^T}{\|dh_p\|^2} k_{\text{tran}} h(\mathbf{p})}_{\boldsymbol{\mu}^{\perp}(\mathbf{p})} + \underbrace{\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} dh_p^T \frac{v}{\|dh_p\|}}_{\boldsymbol{\mu}^{\parallel}(\mathbf{p})}, \tag{4.7}$$

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 (4.7) 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 $\boldsymbol{\mu}^{\parallel}(\mathbf{p})$ that regulates the robot speed along the path $\mathcal{P}$ (see also Figure 4.2).

We now enhance the ideas presented above to handle the presence of the unknown ocean current. To begin with, the identity in (4.4) may be re-written as

$$\dot{\mathbf{p}} = \mathbf{R}_{\theta_N} \begin{bmatrix} v_{t,\text{rel}} \\ v_{n,\text{rel}} \end{bmatrix} + \mathbf{v}_c. \tag{4.8}$$

An ocean current observer for marine vehicles based on this equation was first presented in [1], where a Luenberger observer was implemented. It was later used

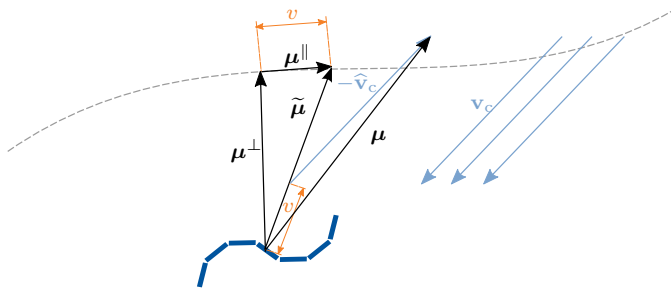


Figure 4.3: The path-following controller with current compensation. In order to avoid a very large reference velocity far away from the path, the original reference $\widetilde{\boldsymbol{\mu}}$ is scaled to have the length v that was originally assigned to the along path component $\boldsymbol{\mu}^{\parallel}$ (cmp. Figure 4.2 and the derivation in [78]). 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 $\boldsymbol{\mu}$.

for path-following in [74, 83]. In this chapter, we propose a reduced order observer based on (4.8), which provides an exponentially stable estimate of the ocean current velocities. Letting $\mathbf{x} = [\mathbf{p}, \mathbf{v}_{\mathrm{c}}]^T$, we may re-write (4.8) as

$$\begin{aligned} \dot{\mathbf{x}} &= \begin{bmatrix} \mathbf{0} & \mathbf{I}_2 \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \mathbf{x} + \begin{bmatrix} \mathbf{R}_{\theta_N} \\ \mathbf{0} \end{bmatrix} \mathbf{v}_{\mathrm{rel}}, \\ \mathbf{y} &= \begin{bmatrix} \mathbf{I}_2 & \mathbf{0} \end{bmatrix} \mathbf{x}. \end{aligned} \tag{4.9}$$

Assuming that $\mathbf{p}$ and $\mathbf{v}_{\mathrm{rel}}$ are available for measurement, the reduced-order observer (see e. g. [14]) for $\mathbf{v}_{\mathrm{c}}$ is given by

$$\dot{\mathbf{z}} = -k_o \mathbf{z} - k_o^2 \mathbf{p} - k_o \mathbf{R}_{\theta_N} \mathbf{v}_{\mathrm{rel}}, \tag{4.10}$$

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

$$\widehat{\mathbf{v}}_{\mathrm{c}} = \mathbf{z} + k_o \mathbf{p}. \tag{4.11}$$

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 (4.8) 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 (4.8) as $\mathbf{p} = \boldsymbol{\mu} + \mathbf{v}_{\mathrm{c}} + \Delta$, where

$$\boldsymbol{\mu} = \mathbf{R}_{\theta_{\mathrm{ref}}} \begin{bmatrix} v_{\mathrm{ref}} \\ 0 \end{bmatrix}, \tag{4.12}$$

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

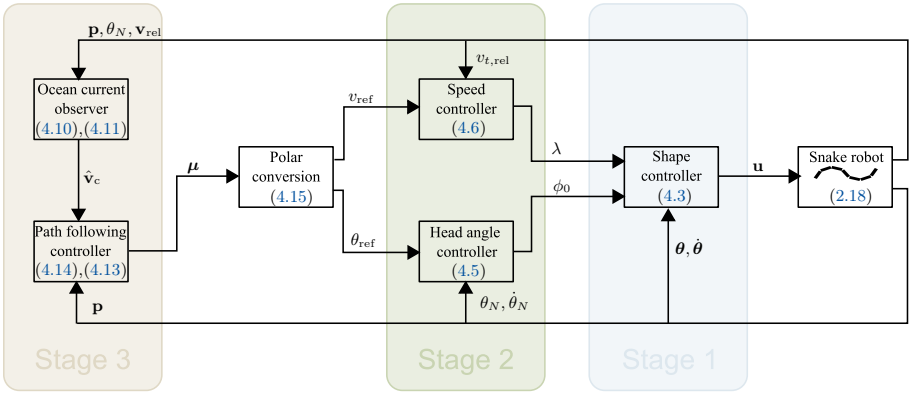


Figure 4.4: Block diagram of the control system.

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 (4.7) and is given by

$$\boldsymbol{\mu}(\mathbf{p}, \xi, \mathbf{z}) = v \frac{\boldsymbol{\mu}_0(\mathbf{p}, \xi)}{\|\boldsymbol{\mu}_0(\mathbf{p}, \xi)\|} - \widehat{\mathbf{v}}_c(\mathbf{p}, \mathbf{z}). \tag{4.13}$$

In (4.13), $\widehat{\mathbf{v}}_c(\mathbf{p}, \mathbf{z})$ is the estimation of the ocean current and $\boldsymbol{\mu}_0(\mathbf{p}, \xi)$ is an alteration of $\widetilde{\boldsymbol{\mu}}(\mathbf{p})$:

$$\begin{aligned} \boldsymbol{\mu}_0(\mathbf{p}, \xi) &= -\frac{dh_p^T}{\|dh_p\|^2}\left(k_{\text{tran}} h(\mathbf{p}) + k_{\text{int}} \xi\right) + \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} dh_p^T \frac{v}{\|dh_p\|}, \\ \dot{\xi} &= h(\mathbf{p}). \end{aligned} \tag{4.14}$$

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 (4.13) 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 (4.12) by the following polar conversion:

$$\begin{aligned} \theta_{\text{ref}}(\mathbf{p}, \xi, \mathbf{z}) &= \arctan\left(\frac{\boldsymbol{\mu}_y}{\boldsymbol{\mu}_x}\right), \\ v_{\text{ref}}(\mathbf{p}, \xi, \mathbf{z}) &= \|\boldsymbol{\mu}(\cdot)\|_2. \end{aligned} \tag{4.15}$$

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 4.3, and a block diagram can be found in Figure 4.4.

Table 4.1: The parameters of the control system.

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

Remark 4.4. Note that $\dot{\theta}_{\text{ref}}$ is required for the implementation of the heading controller (4.5). Similar to the approach in Chapter 3, $\dot{\theta}_{\text{ref}}$ depends on the unknown ocean current velocity. With the ocean current observer proposed in this chapter, $\dot{\theta}_{\text{ref}}$ can be obtained using the estimate of the ocean current velocity.

4.3 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 manoeuvring scenarios based on the theoretical model in Section 2.2 are presented.

4.3.1 Simulation set-up

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 2.3, and the current components were chosen as V_x = -0.01 m/s and V_y = 0.01 m/s. The gains of the control system were chosen according to Table 4.1.

4.3.2 Simulation results

Three different scenarios were tested for the theoretical model in Section 2.2 and the control system (4.3),(4.5),(4.6),(4.13). The first one is straight line path-following

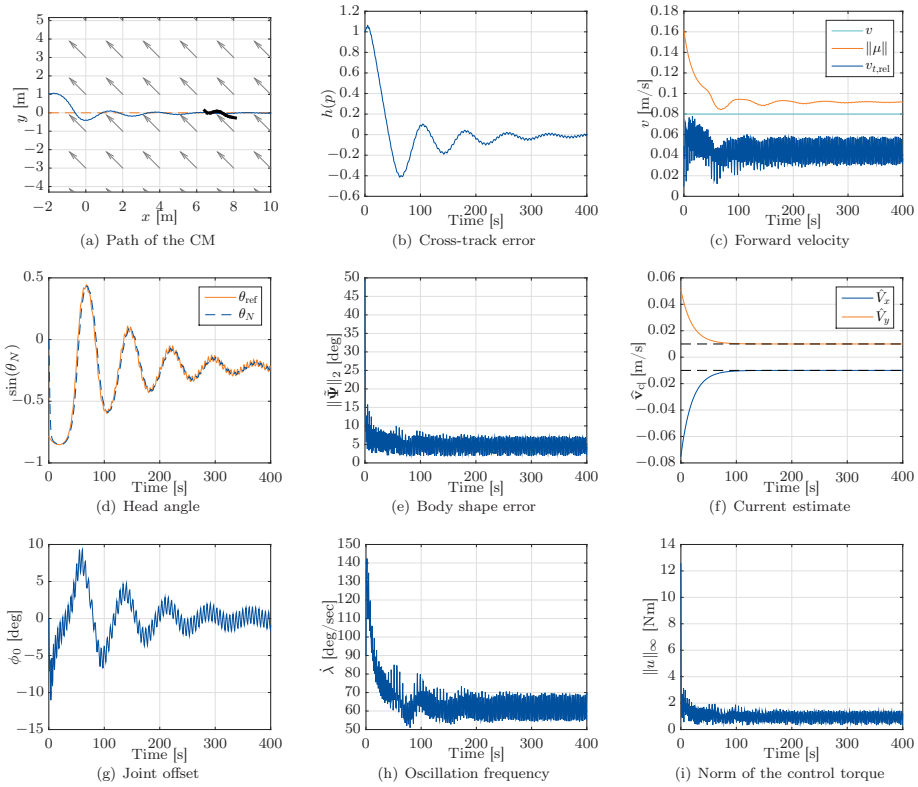


Figure 4.5: Simulation results for straight line path-following with eel-like motion: (a) The path of the underwater snake robot (b) The path-following error (c) The forward velocity $v_{t,\mathrm{rel}}$ and the reference velocity v_{ref} (d) The head link angle θ_N and the reference heading θ_{ref} (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 $\mathbf{u}$.

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.

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. 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 [44], the scaling function in the gait (4.1) 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 4.5. 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. Oscillations 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 4.5(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.

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 4.3.2, which means that the robot was initially headed towards the path.

The results of the simulation are depicted in Figure 4.6. 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 4.3.2, the reference velocity $v_{\text{ref}} = \|\boldsymbol{\mu}\|$ now oscillates about the predefined value v, as shown in Figure 4.6(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.

Following a sinusoidal path with eel-like motion

The sinusoidal reference path for the robot was $h(\mathbf{p}) = p_y - 2\sin(2\pi/12 p_x)$. The variation of eel-like motion that was tested in this case study uses the following scaling function defined in [116]: $g(i) = e^{-0.1li}$. The maximum joint amplitude was set to $\alpha = 40°$. The initial conditions were chosen the same as in Section 4.3.2.

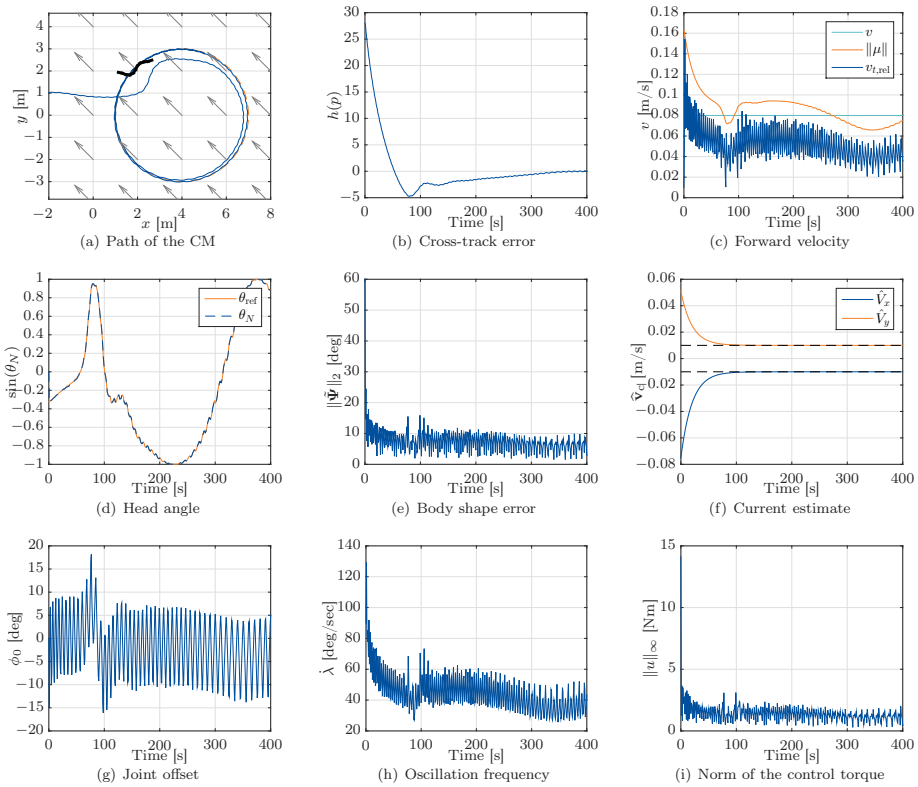


Figure 4.6: Simulation results for tracking a circular reference path with lateral undulation: (a) The path of the underwater snake robot (b) The path-following error (c) The forward velocity $v_{t,\mathrm{rel}}$ and the reference velocity v_{ref} (d) The head link angle θ_N and the reference heading 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 $\mathbf{u}$.

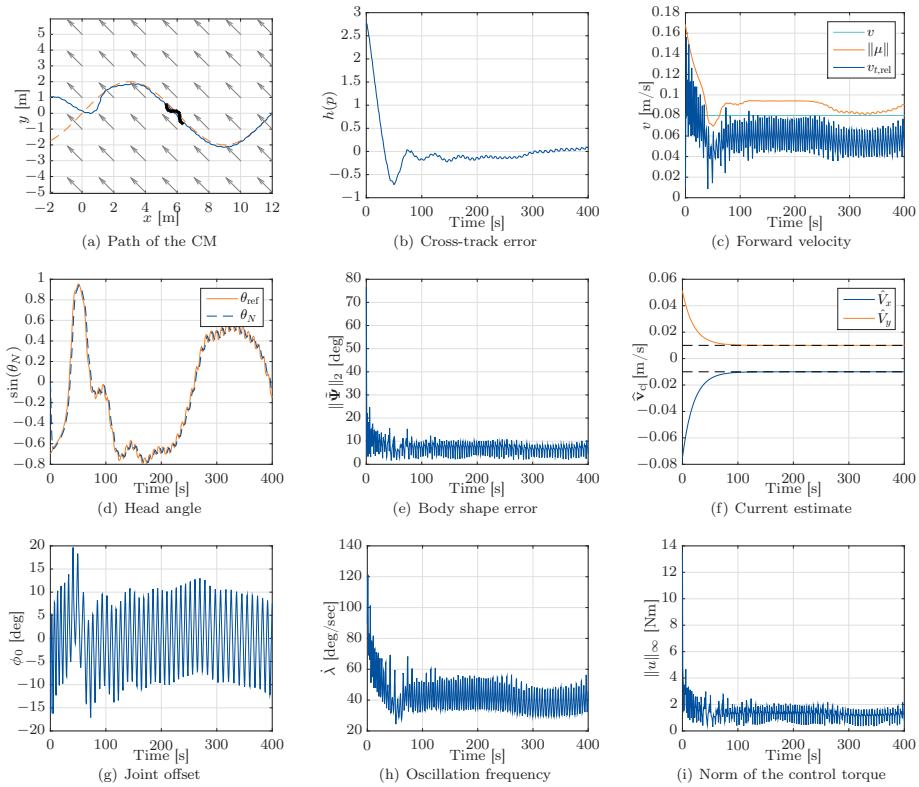


Figure 4.7: Simulation results for following a sinusoid path with eel-like motion: (a) The path of the underwater snake robot (b) The path-following error (c) The forward velocity $v_{t,\mathrm{rel}}$ and the reference velocity v_{ref} (d) The head link angle θ_N and the reference heading θ_{ref} (e) 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 $\mathbf{u}$.

The results of the simulation are depicted in Figure 4.7. 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 $v_{t,\mathrm{rel}}$ and the reference $v_{\mathrm{ref}} = \|\boldsymbol{\mu}\|$.

4.4 Experimental study

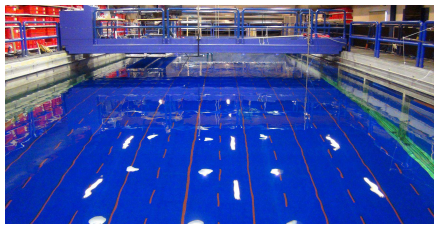
In this section we present experimental results for the control system proposed in Section 4.2. At first, the heading and speed controllers presented in (4.5) and (4.6) were implemented and tested separately. Subsequently, the manoeuvring controller displayed in Figure 4.4 was tested for straight line path-following first without, then with a known ocean current.

4.4.1 The set-up of the experiments

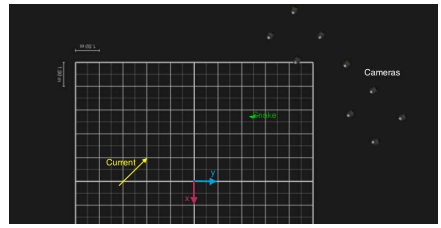
The experiments were performed in the North Sea Centre Flume Tank[1]. The tank can be seen in Figure 4.8(a). It is 30 m long, 8 m wide, and 6 m deep and is equipped with four propellers in order to generate a circulating flow of up to 1 m/s. On the back side, the tank is furthermore equipped with 9 cameras of the Qualisys underwater motion capture system in order to accurately measure positions within the basin. Figure 4.8(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 [66] 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. For the tests in this chapter, the reflective markers for the Qualisys motion capture system were attached to the head link of the robot. The heading θ_N of the head link was therefore 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. More details on the physical robot can be found in Appendix B. The robot is depicted in Figure 4.8(c). The control system (4.5),(4.6),(4.13),(4.14) was implemented in LabVIEW. This internal joint position control of Mamba replaced (4.3) in the practical implementation of the manoeuvring control system. Details about the implementation and the choice of control parameters will be explained in the following for the respective experimental tests.

[1]The North Sea Centre Flume Tank – Managed and operated by SINTEF Fisheries and Aquaculture in Hirtshals, Denmark.



(a) The flume tank



(b) The camera positioning system

(c) The underwater snake robot

Figure 4.8: The experimental set-up: (a) The North Sea Centre Flume Tank. (b) A screen-shot of QTM: the coordinate frame has been rotated with respect to the basin and the cameras. The yellow arrow indicates the flow direction. (c) The snake robot Mamba with an attachment for the reflective markers on the head link.

4.4.2 Heading controller

As a first step, the heading controller (4.5) 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 (4.1), 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 $\phi_{0,\max} = \pm 20°$. The system was tuned according to Table 4.2 and the behaviour of the robot was studied for different reference headings θ_{ref}. The results for two cases can be seen in Figure 4.9 and Figure 4.10. In the first presented case, the robot was initially straight and heading towards $\theta_N = -13.2°$. The reference heading 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 heading of the straightened robot was $\theta_N = -33.5°$, and

Table 4.2: 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	Controller gains for the heading controller
k_n	20	
k_1	1	
k_2	1	

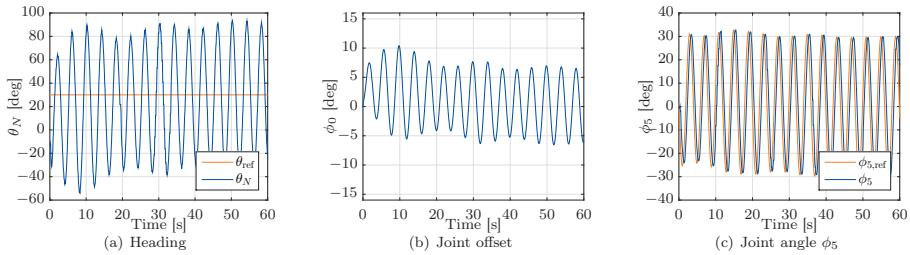


Figure 4.9: Experimental results for the heading controller with $\theta_{\text{ref}} = 30°$: (a)The head link angle θ_N and the reference heading θ_{ref} (b) The joint offset ϕ_0 which is used to control the turning rate (c) The joint angle ϕ_5 and its reference.

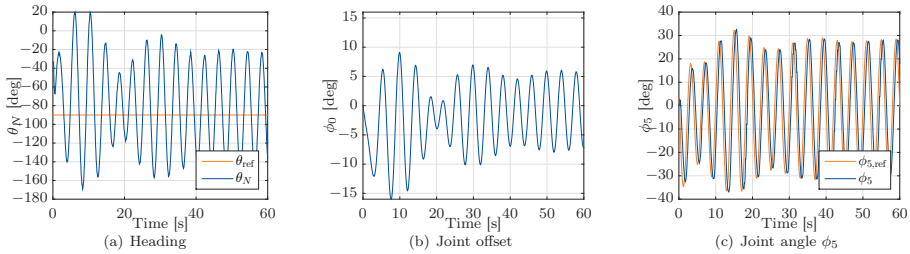


Figure 4.10: Experimental results for the heading controller with $\theta_{\text{ref}} = -90°$: (a)The head link angle θ_N and the reference heading θ_{ref} (b) The joint offset ϕ_0 which is used to control the turning rate (c) The joint angle ϕ_5 and its reference.

the set point was $\theta_{\text{ref}} = -90°$, 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.

4.4.3 Speed controller

In order to implement the speed controller (4.6), velocity measurements were required. Since the snake robot Mamba is not equipped with velocity sensors, the

Table 4.3: 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	

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 4.2 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

$$v_t \approx u = \sqrt{\dot{p}_x^2 + \dot{p}_y^2}. \tag{4.16}$$

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 [78]. 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 (4.1) with $g(i) = 1$ and the control gains in Table 4.3. 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 4.11, Figure 4.12, and Figure 4.13, respectively. Consistently with the simulation results presented in Section 4.3, the mean of the velocity reached a constant offset to the reference velocity. A discussion of these findings will be presented in Section 4.5. Another observation can be made when looking at Figure 4.13(b) and Figure 4.13(c). When the speed controller requires a large oscillation frequency at around 150°/sec (Figure 4.13(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 plotted in Figure 4.13(c). The same effect can be observed when increasing the control gain k_λ, as displayed in Figure 4.14. Here, the test for $v_{\text{ref}} = 5$ 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 4.12 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 oscil-

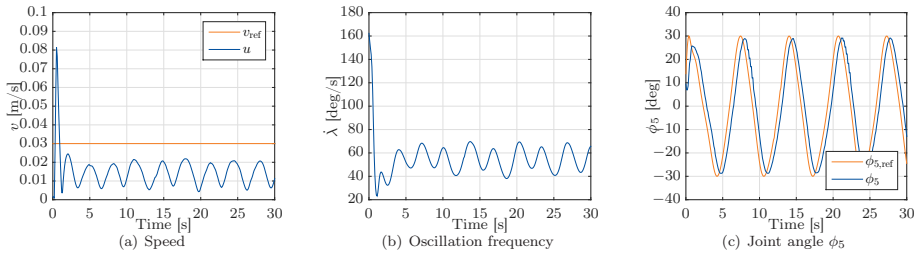


Figure 4.11: Experimental results for the speed controller with $v_{\text{ref}} = 3$ cm/s and $k_\lambda = 0.6$: (a) The velocity u and the reference velocity v_{ref} (b) The frequency of the undulatory gait is governed by $\dot{\lambda}$ (c) The joint angle ϕ_5 and its reference.

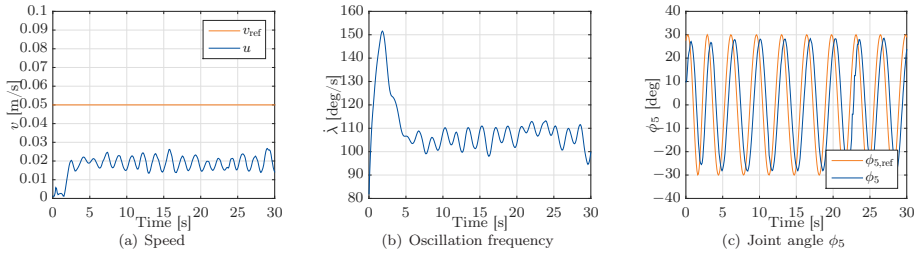


Figure 4.12: Experimental results for the speed controller with $v_{\text{ref}} = 5$ cm/s and $k_\lambda = 0.6$: (a) The velocity u and the reference velocity v_{ref} (b) The frequency of the undulatory gait is governed by $\dot{\lambda}$ (c) The joint angle ϕ_5 and its reference.

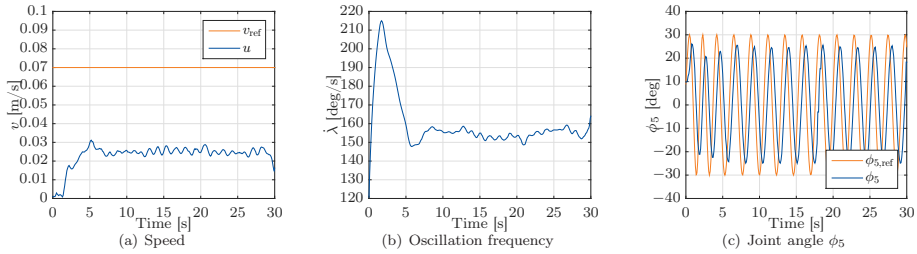
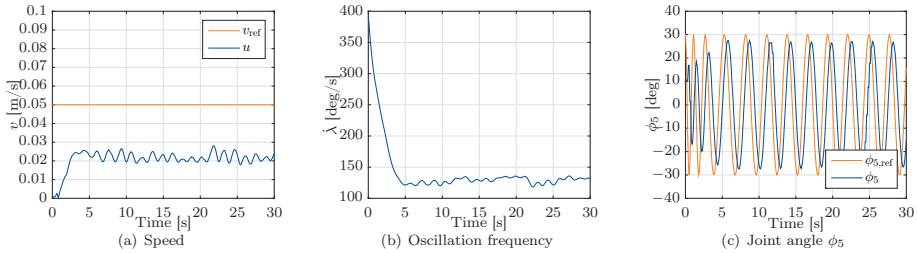


Figure 4.13: Experimental results for the speed controller with $v_{\text{ref}} = 7$ cm/s and $k_\lambda = 0.6$: (a) The velocity u and the reference velocity v_{ref} (b) The frequency of the undulatory gait is governed by $\dot{\lambda}$ (c) The joint angle ϕ_5 and its reference.



(a) Speed (b) Oscillation frequency (c) Joint angle ϕ_5

Figure 4.14: Experimental results for the speed controller with $v_{\text{ref}} = 5$ 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.

lation frequency. It can be seen from Figure 4.14(c) that the single joints did not quite reach the maximal reference amplitude already, an effect that would become even stronger with a further increase of k_λ.

4.4.4 Straight line path-following without current

After establishing the heading and speed controller, the two were combined with the path-following controller (4.13),(4.14) according to the structure in Figure 4.4. 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 set-up in the Flume Tank, which are all mounted on the same wall. 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 manoeuvring control problem of terrestrial snake robots, which has been solved theoretically in [78], 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 4.4 in order to ensure smooth convergence for the straight line path. As explained in Section 4.4.3, 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 4.5. 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

Table 4.4: The parameters of the control system.

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

experimental effects such as the power cable and 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 4.4. In particular, for the case of eel-like motion, the scaling function in the gait (4.1) 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 $k_{\text{trans}} = 0.025$. The results for eel-like motion are presented in Figure 4.15. 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 4.4.3. 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.

4.4.5 Straight line path-following with known current

In contrast to the previously presented tests of the speed controller in Section 4.4.3 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 approximate the relative velocity by

$$v_{t,\text{rel}} \approx u_{\text{rel}} = \sqrt{(\dot{p}_x - V_x)^2 + (\dot{p}_y - V_y)^2}, \tag{4.17}$$

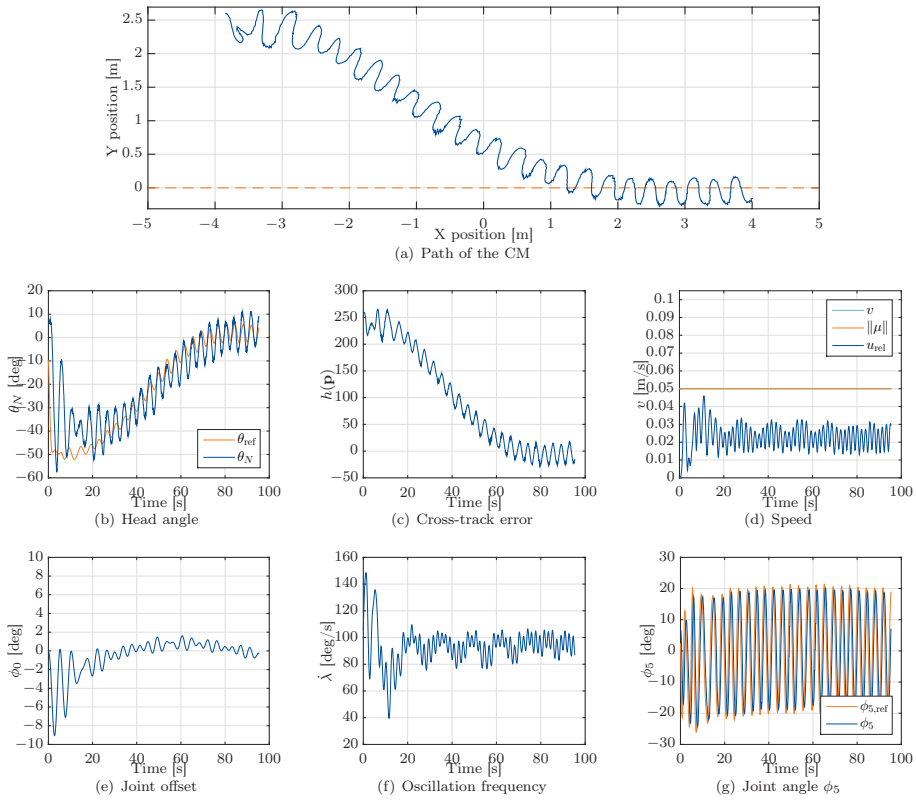


Figure 4.15: Experimental results for straight line path-following with eel-like motion and zero ocean current: (a) The path of the underwater snake robot (b) The head link angle θ_N and the reference heading θ_{ref} (c) The path-following error (d) The velocity u and the reference velocity v_{ref} (e) 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.

analogously to the absolute velocity in Section 4.4.3. The current V_x and V_y were obtained from the set flow speed V_c by $V_x = -V_c \cos(45°)$ and $V_y = V_c \sin(45°)$, according to the rotation of the coordinate frame in Figure 4.8(b).

Remark 4.6. 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 [104] and DVLs without bottom lock for relative velocity measurements [16]. The components of the ocean current entering the control system in (4.14) are then extracted from the position and relative velocity measurements

by the observer (4.10),(4.11). Implementing the observer (4.10),(4.11) 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 (4.14), and the observer (4.10),(4.11) was omitted.

In addition, for the practical implementation of the control system, anti-wind-up was added to the integral action in the path-following control stage by replacing the integrator update law in (4.14) by

$$\dot{\xi} = \frac{\frac{h(\mathbf{p})}{k_{\text{tran}}}}{(h(\mathbf{p})+k_{\text{int}}\xi)^2+k_{\text{tran}}^{-2}}. \tag{4.18}$$

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 $V_c = 0.05$ m/s. For lateral undulation, the joint amplitude was set to $\alpha = 30°$, and the scaling function of the gait (4.1) 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 $k_{\text{trans}} = 0.035$. All other gains were chosen in accordance with Table 4.4. Figure 4.16 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 4.4.4 it can be seen in Figure 4.16(b) that θ_{ref} now oscillates about a non-zero value in steady state, i. e. it automatically finds the steady-state crab 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 $v_{\text{ref}} = \|\boldsymbol{\mu}\|$ now changed, as the heading of the robot changed. This can be explained by the fact that the robot was started with a heading 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. This drawback is compensated for by the integral action in the path-following control stage (4.14), 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 (4.1) was set to $g(i) = \frac{N-i}{N+1}$ just like in the current-free scenario in Section 4.4.4. Just like for lateral undulation, the flow speed was $V_c = 0.05$ m/s, and the integral gain was

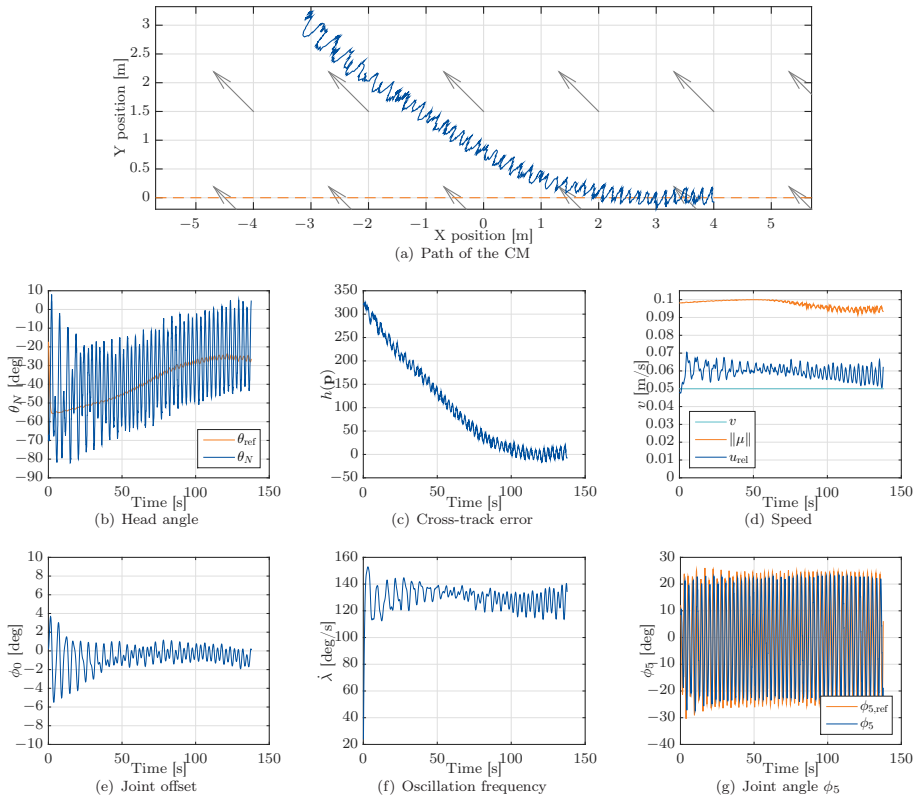


Figure 4.16: Experimental results for straight line path-following with lateral undulation and a known ocean current: (a) The path of the underwater snake robot (b) The head link angle θ_N and the reference heading θ_{ref} (c) The path-following error (d) The velocity u and the reference velocity v_{ref} (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.

chosen according to Table 4.4. The other control gains were chosen analogously to the current-free case in Section 4.4.4. The results of the test run are displayed in Figure 4.17. It can be seen that the robot approached and followed the reference path, and the cross-track error 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 4.18, where it can be seen how it approaches and follows the path.

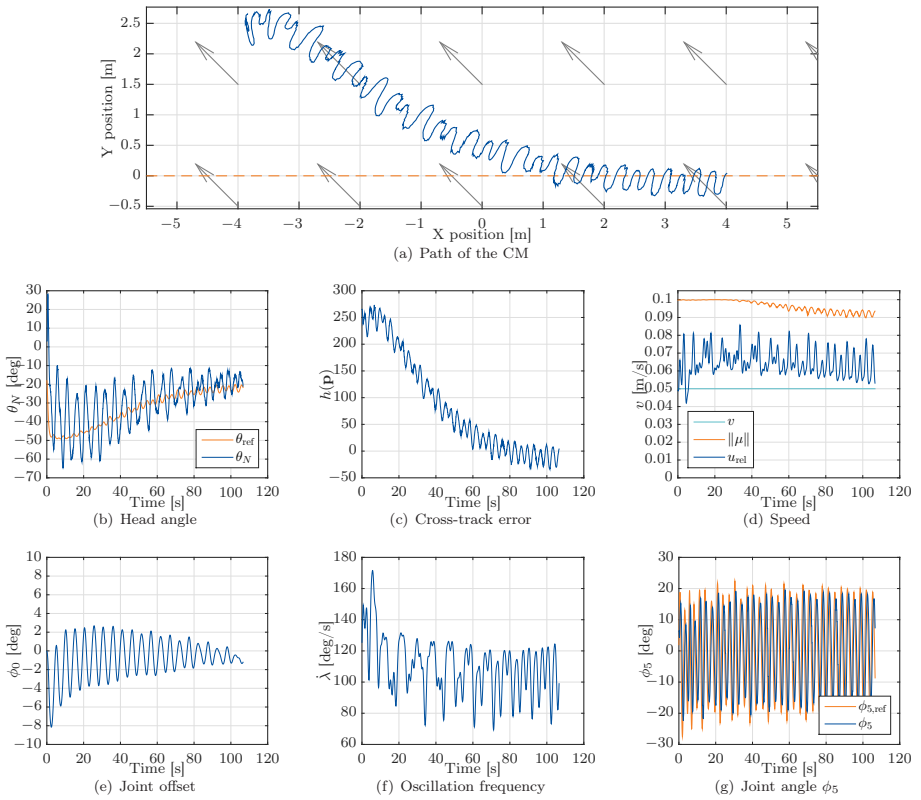


Figure 4.17: Experimental results for straight line path-following with eel-like motion and a known ocean current: (a) The path of the underwater snake robot (b) The head link angle θ_N and the reference heading θ_{ref} (c) The path-following error (d) The velocity u and the reference velocity v_{ref} (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.

4.5 Discussion

The control system presented in this chapter 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 terrestrial snake robots presented in [78].

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 [78]. 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

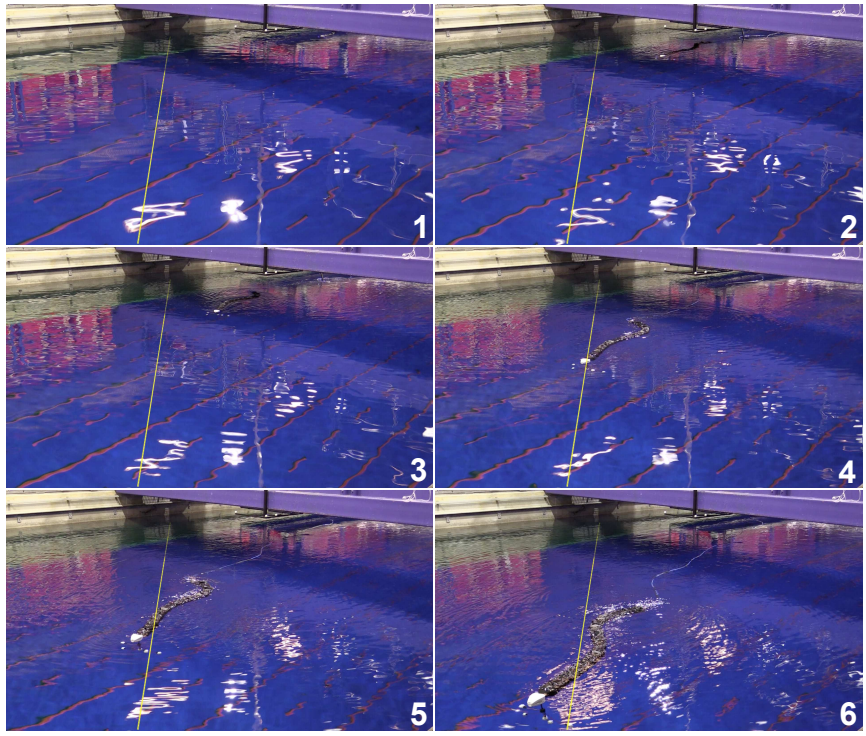


Figure 4.18: The snake robot Mamba: 1) The robot is launched with an offset to the path. 2) The robot turns towards the path. 3),4) The robot approaches the path smoothly. 5),6) The robot stays on the path.

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 micro-controllers, 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 (4.15) 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.

4.6 Chapter summary

This chapter studied the problem of planar manoeuvring 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 heading and forward speed of the snake robot. The desired heading 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 terrestrial 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.

Chapter 5

Adaptive Control Using Absolute Velocity Measurements

In this chapter, a first step is taken towards the control of underwater snake robots using absolute velocity measurements for feedback. A control system for velocity and orientation control of underwater snake robots is proposed, motivated by the direction following results using VHCs for terrestrial snake robots in [91]. The proposed control system is structured in a hierarchical way, in order to fulfil the single control objectives with different priorities simultaneously. We will therefore base the control design on the hierarchical methodology from [20], that was used for control design for ships in [9] and terrestrial snake robots in [78]. In order to make the feedback independent of the unknown relative velocities that enter the dynamic equations via the fluid drag forces, we make use of adaptive backstepping control [56].

Contributions of this chapter An adaptive joint controller is proposed, which asymptotically stabilizes the VHCs that encode a sinusoidal gait with absolute velocity measurements available for feedback, as opposed to the design based on relative velocity measurements in Chapters 2 and 3. The gait that is encoded in the VHCs is modified in this chapter to take into account actuator constraints. The velocity controller and the orientation controller are designed subsequently. In doing so, we prioritize the velocity control higher than in previous approaches for terrestrial snake robots, a change of paradigm that removes a singularity from the orientation controller that was an issue in [90, 91]. The velocity controller is designed along the lines of adaptive backstepping. The design is not a straightforward application of the method, because the system is non-autonomous and unknown terms enter the equations at every stage of the backstepping procedure. By postponing the design of the adaptive update law until the last step, we are able to compensate for the effect of these signals in the closed-loop system, and show asymptotic stability of the origin of the velocity error dynamics. In the last step, the orientation controller is designed as a feedback-linearizing controller utilizing the adaptive term of the velocity controller as a current estimate. We show that the zero dynamics of the closed-loop system is bounded for the proposed controller.

Simulation results of a snake robot that is exposed to an unknown ocean current are presented. The results validate the theoretical analysis.

Organization of this chapter This chapter is structured as follows. The control objective is formulated in Section 5.1. The control system design is presented in Section 5.2, as well as a stability analysis of each of the single controllers. Section 5.3 gives a stability analysis of the zero dynamics and a sketch of how to formally establish stability of the closed-loop system. Simulation results that validate the theoretical findings can be found in Section 5.4. The chapter is summarized in Section 5.5.

Publications This chapter is based on [55].

5.1 Control objectives

The control objectives can be categorized into three stabilization tasks that are prioritized in a hierarchical manner. The control-oriented model (2.33) is used as the model to base the control design on, i. e. the stabilization tasks will be formulated in terms of the dynamics in (2.33).

The highest priority is to stabilize VHCs that encode a sinusoidal gait, thus propelling the robot forward. We will design a feedback $\mathbf{u}$ for the joints, described by (2.33a),(2.33e), that controls each joint i, for $i = 1, \ldots, N-1$, to follow the reference signal

$$\phi_{\mathrm{ref},i}(\lambda, \phi_0) = \alpha g_1(i) \sin(\lambda + (i-1)\delta) + g_2(\phi_0), \tag{5.1}$$

where α is the maximal amplitude and δ is the phase shift between adjacent joints. The VHC in (5.1) is an adaptation of (4.1), where $g_1 : \mathbb{Z} \mapsto [0, 1]$ is a scaling function that varies the amplitude along the body and a smooth, strictly increasing, and the twice differentiable saturation function $g_2 : \mathbb{R} \mapsto [-\phi_{0,\max}, \phi_{0,\max}]$ has been added. The additional function g_2 ensures that the theoretical joint reference always respects the physical constraints of the robot and that the joint reference remains uniformly bounded by $|\phi_{\mathrm{ref},i}| \leq \frac{\epsilon_\phi}{N-1}$, with $\epsilon_\phi = (N-1)(\alpha + \phi_{0,\max})$. The two states λ and ϕ_0 are generated by dynamic compensators that will be developed later in order to control the velocity and the orientation of the robot, respectively.

The second priority of the control system is to control the forward velocity v_t, the dynamics of which are given in (2.33g), to a constant reference v_{ref} using the dynamic compensator $\ddot{\lambda} = u_\lambda$. We employ the second derivative of λ as the control input, because in order to stabilize the constraints in (5.1), the time derivative of $\phi_{\mathrm{ref},i}$ up to $\ddot{\phi}_{\mathrm{ref},i}$ are required, as we will see in Section 5.2.1. Thus $\ddot{\lambda} = u_\lambda$ is a natural choice. In order to make sure that the robot does not just float with the ocean current, we will furthermore choose the reference v_{ref} larger than the current magnitude: $v_{\mathrm{ref}} > V_{\mathrm{c},\max}$.

Remark 5.1. As opposed to similar control approaches for terrestrial snake robots in the literature [90, 91], in this work, we consider the regulation of the forward velocity as a higher priority than the orientation control. Several motion control

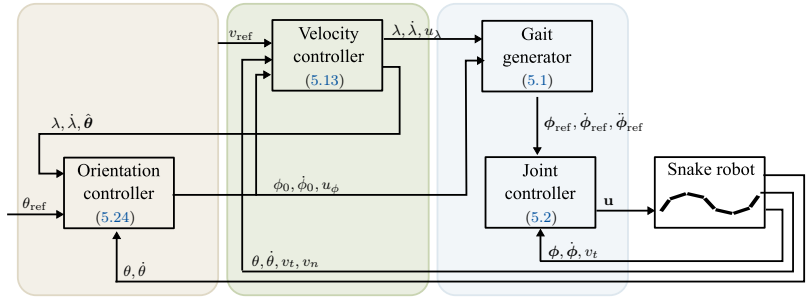


Figure 5.1: A block diagram of the proposed control system with absolute velocity measurements for feedback.

systems for snake robots in the literature (e. g. [45, 65]) and in Chapter 3 of this thesis do not control the velocity of the robot, but propose to choose a constant undulation frequency to achieve some forward velocity, while controlling the orientation of the robot to follow the reference of a guidance system. We conjecture that making velocity control a higher priority than orientation control will provide a velocity controller that is a natural complement of motion control approaches such as in Chapter 3 or [45, 65].

Controlling the orientation of the robot, θ, to a constant reference θ_{ref} is the third and lowest priority of the proposed control system. Based on (2.33b),(2.33f), we will design a feedback for the control input u_ϕ to the dynamic compensator $\ddot{\phi}_0 = u_\phi$ in order to achieve this task. The structure of the proposed control system is visualized in Figure 5.1.

Remark 5.2. The robot described by (2.33) is an underactuated system. In particular, we cannot directly control the sideways velocity v_n. However, we will design the control system such that v_n remains bounded. It was shown in Section 2.5 that the average of v_n converges to the sideways component of the current when the robot conducts steady forward motion. In the presence of a current, we will therefore expect the robot to drift sideways, as it travels in the direction θ_{ref} with the forward velocity v_{ref}. If this is not the desired behaviour of the robot, one typically designs a guidance system that provides the references θ_{ref} and v_{ref} in a way that implicitly controls the position and sideways velocity of the robot.

5.2 Control design

In this section, we present the proposed control system for velocity and orientation control of underwater snake robots. The control system is obtained using the hierarchical control design for set stabilization proposed in [20].

5.2.1 Joint control

The aim of the joint controller is to stabilize the joint coordinates to obey the VHCs given by (5.1), which is analogous to stabilizing the system to a constraint manifold.

Details on set stability can be found in Definition 14 and [20]. The dynamics of the joints are described by (2.33a),(2.33e) and the unknown relative velocity $v_{t,\mathrm{rel}}$ enters the system in (2.33e). To this end, the error coordinates $\tilde{\phi}_i = \phi_i - \phi_{\mathrm{ref},i}$ and $\dot{\tilde{\phi}}_i = \dot{\phi}_i - \dot{\phi}_{\mathrm{ref},i}$, $i = 1, \ldots, N-1$, will be stabilized to zero with the following adaptive backstepping controller:

$$\begin{aligned} \mathbf{u} &= m(\mathbf{D}\mathbf{D}^T)^{-1}\big[\tfrac{c_n}{m}\mathbf{v}_\phi - \mathbf{X}_t(\boldsymbol{\phi})v_t + \boldsymbol{\beta}^T(\boldsymbol{\phi},\theta)\hat{\boldsymbol{\theta}} \\ &\quad + \ddot{\boldsymbol{\phi}}_{\mathrm{ref}} - (k_1+k_2)\dot{\tilde{\boldsymbol{\phi}}} - (1+k_1k_2)\tilde{\boldsymbol{\phi}}\big], \\ \dot{\hat{\boldsymbol{\theta}}} &= -\gamma\boldsymbol{\beta}(\boldsymbol{\phi},\theta)(\dot{\tilde{\boldsymbol{\phi}}} + k_1\tilde{\boldsymbol{\phi}}), \end{aligned} \tag{5.2}$$

with the control gains $k_1, k_2 > 0$ and the adaptive gain $\gamma > 0$. In (5.2), the joint errors and references are assembled in the vectors $\boldsymbol{\phi}, \boldsymbol{\phi}_{\mathrm{ref}} \in \mathbb{R}^{N-1}$. Furthermore, $\mathbf{X}_t(\boldsymbol{\phi}) = \frac{c_n}{m}\mathbf{A}\mathbf{D}^T\boldsymbol{\phi}$ and $\boldsymbol{\beta}^T(\boldsymbol{\phi},\theta) = \mathbf{X}_t(\boldsymbol{\phi})\mathbf{R}_\theta^T$.

Proposition 5.1. *Consider a robot described by* (2.33). *The joint control law* (5.2) *globally asymptotically stabilizes the set*

$$\begin{aligned} \Gamma_3 = \{(\mathbf{x}, \dot{\mathbf{x}}, \lambda, \dot{\lambda}, \phi_0, \dot{\phi}_0) \in \mathbb{R}^{2N+8} : \\ \boldsymbol{\phi} = \boldsymbol{\phi}_{\mathrm{ref}}, \mathbf{v}_\phi = \dot{\lambda}\tfrac{\partial \boldsymbol{\phi}_{\mathrm{ref}}}{\partial \lambda} + \dot{\phi}_0 \tfrac{\partial \boldsymbol{\phi}_{\mathrm{ref}}}{\partial \phi_0}\}, \end{aligned} \tag{5.3}$$

i. e. $(\tilde{\boldsymbol{\phi}}, \dot{\tilde{\boldsymbol{\phi}}}) \to (\mathbf{0}, \mathbf{0})$.

Proof. We define the new error coordinates $\boldsymbol{\xi}_1 = \tilde{\boldsymbol{\phi}}$, $\boldsymbol{\xi}_2 = \dot{\tilde{\boldsymbol{\phi}}} + k_1\tilde{\boldsymbol{\phi}}$, and $\tilde{\boldsymbol{\theta}} = \mathbf{v}_{\mathrm{c}} - \hat{\boldsymbol{\theta}}$. In the new coordinates, the closed-loop system (2.33a),(2.33e) with (5.2) transforms to

$$\begin{aligned} \begin{bmatrix} \dot{\boldsymbol{\xi}}_1 \\ \dot{\boldsymbol{\xi}}_2 \end{bmatrix} &= \begin{bmatrix} -k_1\mathbf{I}_{N-1} & \mathbf{I}_{N-1} \\ -\mathbf{I}_{N-1} & -k_2\mathbf{I}_{N-1} \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi}_1 \\ \boldsymbol{\xi}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\beta}^T(\boldsymbol{\phi},\theta) \end{bmatrix} \tilde{\boldsymbol{\theta}}, \\ \dot{\tilde{\boldsymbol{\theta}}} &= -\gamma\boldsymbol{\beta}(\boldsymbol{\phi},\theta)\boldsymbol{\xi}_2. \end{aligned} \tag{5.4}$$

With the Lyapunov function $V = \frac{1}{2}\boldsymbol{\xi}_1^T\boldsymbol{\xi}_1 + \frac{1}{2}\boldsymbol{\xi}_2^T\boldsymbol{\xi}_2 + \frac{1}{2\gamma}\tilde{\boldsymbol{\theta}}^T\tilde{\boldsymbol{\theta}}$, we have

$$\dot{V} = -k_1\boldsymbol{\xi}_1^T\boldsymbol{\xi}_1 - k_2\boldsymbol{\xi}_2^T\boldsymbol{\xi}_2 \leq 0. \tag{5.5}$$

Therefore, $\dot{V}$ is negative semi-definite, and thus the origin $(\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \tilde{\boldsymbol{\theta}}) = (\mathbf{0}, \mathbf{0}, \mathbf{0})$ is uniformly globally stable and $\boldsymbol{\xi}_1, \boldsymbol{\xi}_2$, and $\tilde{\boldsymbol{\theta}}$ are bounded. The error system (5.4) is a non-autonomous system because it depends on time-varying parameters. Furthermore, V is lower bounded by zero and $\ddot{V}$ is finite because $\boldsymbol{\beta}(\boldsymbol{\phi},\theta)$ in (5.4) is bounded, since $\boldsymbol{\beta}(\boldsymbol{\phi},\theta) = \boldsymbol{\beta}(\boldsymbol{\phi},\mathbf{R}_\theta)$, $\boldsymbol{\xi}_1 = \tilde{\boldsymbol{\phi}}$ is bounded, and $\boldsymbol{\phi}_{\mathrm{ref}}$ is bounded by design. With these conditions, it follows from Barbalat's Lemma [48] that $(\boldsymbol{\xi}_1, \boldsymbol{\xi}_2) \to (\mathbf{0}, \mathbf{0})$ asymptotically as $t \to \infty$. This implies that also $\tilde{\boldsymbol{\phi}}$ and $\dot{\tilde{\boldsymbol{\phi}}}$ converge to zero asymptotically. □

Remark 5.3. Since the regressor $\boldsymbol{\beta}(\boldsymbol{\phi},\theta)$ is not persistently exciting (PE), we cannot expect the adaptive estimate, $\hat{\boldsymbol{\theta}}$ to converge to the true ocean current $\mathbf{v}_{\mathrm{c}}$.

5.2.2 Velocity control

For the design of the velocity controller, the reduced system dynamics on the invariant manifold Γ_3 is considered. On Γ_3, the dynamics of the forward velocity (2.33g) reduces to

$$\dot{v}_t = -\tfrac{c_t}{m} v_{t,\mathrm{rel}} + X_n(\lambda,\phi_0) v_{n,\mathrm{rel}} - \mathbf{X}_\phi(\lambda,\phi_0)\mathbf{v}_\phi, \tag{5.6}$$

with $X_n(\lambda,\phi_0) = \frac{2c_\mathrm{p}}{Nm}\bar{\mathbf{e}}^T\boldsymbol{\phi}_\mathrm{ref}(\lambda,\phi_0)$ and $\mathbf{X}_\phi(\lambda,\phi_0) = \frac{c_\mathrm{p}}{Nm}\boldsymbol{\phi}_\mathrm{ref}^T(\lambda,\phi_0)\mathbf{A}\bar{\mathbf{D}}$. For the feedback control of the forward velocity, we propose an adaptive backstepping controller in order to stabilize the error coordinates $z_1 = \tilde{v}_t = v_t - v_\mathrm{ref}$ and $z_2 = \dot{\lambda} - \zeta$, where ζ is a virtual control input. While the adaptive backstepping controller in Section 5.2.1 was derived using standard techniques (e. g. [56]), the adaptive velocity controller design is not as straightforward. Both the dynamics in Section 5.2.1 and in this section are non-autonomous, but for the joint controller in Section 5.2.1, this did not affect the control design, it only complicated the analysis. For the velocity controller, however, the system structure is such that parameter-varying terms have to be compensated in the first design step. The dynamics of these terms depend on the unknown ocean current as well, which makes the control design different from the method for non-autonomous systems in [24]. The derivation of the controller is therefore presented step by step in the following.

To begin with, (5.6) is re-written and expanded to

$$\dot{v}_t = -\tfrac{c_t}{m} v_t + X_n(\lambda,\phi_0) v_n - \mathbf{X}_\phi(\lambda,\phi_0)\bar{\mathbf{e}}\tfrac{\partial g_2}{\partial \phi_0}\dot{\phi}_0 - \boldsymbol{\beta}_{\lambda,1}^T(\lambda,\phi_0,\theta)\mathbf{v}_\mathrm{c} - X_\lambda(\lambda,\phi_0)\dot{\lambda}, \tag{5.7a}$$

$$\ddot{\lambda} = u_\lambda, \tag{5.7b}$$

with $X_\lambda(\lambda,\phi_0) = \mathbf{X}_\phi(\lambda,\phi_0)[\alpha g_1(1)\cos(\lambda),\ldots,\alpha g_1(N-1)\cos(\lambda+(N-2)\delta)]^T$ and $\boldsymbol{\beta}_{\lambda,1}^T(\lambda,\phi_0,\theta) = [-\frac{c_t}{m}, X_n(\lambda,\phi_0)]\mathbf{R}_\theta^T$. The first design step is then to use the virtual control input in (5.7a), $\zeta = \dot{\lambda}$, and the current estimate $\hat{\boldsymbol{\theta}}_\lambda$ to stabilize $z_1 = \tilde{v}_t$ to zero. With the Lyapunov function $V_1 = \frac{1}{2}z_1^2$ and $\dot{z}_1 = \dot{v}_t$, we choose

$$\zeta = \tfrac{1}{X_\lambda(\lambda,\phi_0)}\zeta^*(\lambda,\phi_0,\dot{\phi}_0,v_n,\theta,z_1,\hat{\boldsymbol{\theta}}_\lambda), \tag{5.8a}$$

$$\zeta^*(\cdot) = -\tfrac{c_t}{m} v_\mathrm{ref} + X_n(\lambda,\phi_0)v_n - \mathbf{X}_\phi(\lambda,\phi_0)\bar{\mathbf{e}}\tfrac{\partial g_2}{\partial\phi_0}\dot{\phi}_0 - \boldsymbol{\beta}_{\lambda,1}^T(\lambda,\phi_0,\theta)\hat{\boldsymbol{\theta}}_\lambda + k_{\lambda,1}z_1 \tag{5.8b}$$

with the control gain $k_{\lambda,1} > 0$. Note that $X_\lambda(\lambda,\phi_0) > 0$ is bounded away from zero because of the phase shift δ [90]. However, when analysing $\dot{V}_1$, we need to keep in mind that we do not directly control $\dot{\lambda}$, and that the current estimate $\hat{\boldsymbol{\theta}}_\lambda$ is not necessarily exact, i. e. take into account the errors $z_2 = \dot{\lambda} - \zeta$ and $\tilde{\boldsymbol{\theta}}_\lambda = \mathbf{v}_\mathrm{c} - \hat{\boldsymbol{\theta}}_\lambda$ in

$$\dot{z}_1 = -(\tfrac{c_t}{m} + k_{\lambda,1})z_1 - X_\lambda(\lambda,\phi_0)z_2 - \tilde{\boldsymbol{\theta}}_\lambda^T\boldsymbol{\beta}_1(\lambda,\phi_0,\theta), \tag{5.9}$$

yielding

$$\dot{V}_1 = -(\tfrac{c_t}{m} + k_{\lambda,1})z_1^2 - X_\lambda(\lambda,\phi_0)z_1z_2 - \tilde{\boldsymbol{\theta}}_\lambda^T\boldsymbol{\beta}_1(\lambda,\phi_0,\theta)z_1. \tag{5.10}$$

In order to stabilize z_2 to zero, we design the control law u_λ in the second step, using the Lyapunov function $V_2 = V_1 + \frac{1}{2}z_2^2 + \frac{1}{2\gamma_\lambda}\tilde{\boldsymbol{\theta}}_\lambda^T\tilde{\boldsymbol{\theta}}_\lambda$. For doing so, the time

derivative $\dot{\zeta}$ is required. This complicates the control design significantly, because the dynamics of the uncontrolled state v_n enters the controller and it depends on the relative velocities, i. e. the unknown term. The time derivative of ζ is given by

$$\dot{\zeta} = \frac{\dot{\zeta}^*(\lambda,\dot{\lambda},\phi_0,\dot{\phi}_0,\ddot{\phi}_0,v_n,\dot{v}_n,\theta,v_\theta,z_1,\dot{z}_1,\hat{\boldsymbol{\theta}}_\lambda,\dot{\hat{\boldsymbol{\theta}}}_\lambda)}{X_\lambda(\lambda,\phi_0)} - \frac{\zeta^*(\lambda,\phi_0,\dot{\phi}_0,v_n,\theta,z_1,\hat{\boldsymbol{\theta}}_\lambda)\dot{X}_\lambda(\lambda,\dot{\lambda},\phi_0,\dot{\phi}_0)}{X_\lambda^2(\lambda,\phi_0)}. \tag{5.11}$$

The second term in (5.11) only contains known signals and is not written down explicitly for compactness. However, $\dot{\zeta}^*$ in the first term contains unknown signals that have to be compensated for by the adaptive controller. It is obtained by taking the time derivative of ζ^* in (5.8b) and inserting (2.33h), (5.9) and $\tilde{\boldsymbol{\theta}}_\lambda = \mathbf{v}_\mathrm{c} - \hat{\boldsymbol{\theta}}_\lambda$:

$$\begin{aligned}
\dot{\zeta}^* = &\Big(\tfrac{\partial X_n}{\partial\lambda}\dot{\lambda} + \tfrac{\partial X_n}{\partial\phi_0}\dot{\phi}_0 - X_n\tfrac{c_n}{m}\Big)v_n + X_n^2 v_t + \begin{bmatrix}-X_n^2 & X_n\frac{c_n}{m}\end{bmatrix}\mathbf{R}_\theta^T\mathbf{v}_\mathrm{c} \\
&- \tfrac{\partial \mathbf{X}_\phi}{\partial\lambda}\bar{\mathbf{e}}\dot{\lambda}\tfrac{\partial g_2}{\partial\phi_0}\dot{\phi}_0 - \tfrac{\partial \mathbf{X}_\phi}{\partial\phi_0}\bar{\mathbf{e}}\tfrac{\partial g_2}{\partial\phi_0}\dot{\phi}_0^2 - \mathbf{X}_\phi\bar{\mathbf{e}}\tfrac{\partial^2 g_2}{\partial\phi_0^2}\dot{\phi}_0^2 - \mathbf{X}_\phi\bar{\mathbf{e}}\tfrac{\partial g_2}{\partial\phi_0}\ddot{\phi}_0 \\
&- \Big(\tfrac{\partial\boldsymbol{\beta}_{\lambda,1}^T}{\partial\lambda}\dot{\lambda} + \tfrac{\partial\boldsymbol{\beta}_{\lambda,1}^T}{\partial\phi_0}\dot{\phi}_0 + \tfrac{\partial\boldsymbol{\beta}_{\lambda,1}^T}{\partial\theta}v_\theta\Big)\hat{\boldsymbol{\theta}}_\lambda - \boldsymbol{\beta}_{\lambda,1}^T\dot{\hat{\boldsymbol{\theta}}}_\lambda \\
&+ k_{\lambda,1}\Big(-(\tfrac{c_t}{m}+k_{\lambda,1})z_1 - X_\lambda z_2 + \boldsymbol{\beta}_{\lambda,1}^T\hat{\boldsymbol{\theta}}_\lambda - \boldsymbol{\beta}_{\lambda,1}^T\mathbf{v}_\mathrm{c}\Big).
\end{aligned} \tag{5.12}$$

In (5.12), the function arguments are omitted for better readability and the time derivative of the current estimate $\dot{\hat{\boldsymbol{\theta}}}_\lambda$ has not been inserted yet, because it will be designed in the next step. From (5.12) we can see why standard backstepping methods from the literature [24, 56] cannot be applied: the unknown term, $\mathbf{v}_\mathrm{c}$, enters the time derivative of the virtual control input ζ through $\dot{\zeta}^*$. With the Lyapunov function V_2 and $\dot{z}_2 = u_\lambda - \dot{\zeta}$, the control input is chosen as

$$\begin{aligned}
u_\lambda = &-(k_{\lambda,2}+k_{\lambda,1})z_2 + \Big(X_\lambda + \tfrac{X_n^2}{X_\lambda} - \tfrac{k_{\lambda,1}}{X_\lambda}(\tfrac{c_t}{m}+k_{\lambda,1})\Big)z_1 + \tfrac{X_n^2}{X_\lambda}v_\mathrm{ref} \\
&- \tfrac{\zeta^*\dot{X}_\lambda}{X_\lambda^2} + \tfrac{1}{X_\lambda}\Big(\tfrac{\partial X_n}{\partial\lambda}\dot{\lambda} + \tfrac{\partial X_n}{\partial\phi_0}\dot{\phi}_0 - X_n\tfrac{c_n}{m}\Big)v_n \\
&- \tfrac{1}{X_\lambda}\Big(\tfrac{\partial \mathbf{X}_\phi}{\partial\lambda}\bar{\mathbf{e}}\dot{\lambda}\tfrac{\partial g_2}{\partial\phi_0}\dot{\phi}_0 + \tfrac{\partial \mathbf{X}_\phi}{\partial\phi_0}\bar{\mathbf{e}}\tfrac{\partial g_2}{\partial\phi_0}\dot{\phi}_0^2 + \mathbf{X}_\phi\bar{\mathbf{e}}\tfrac{\partial^2 g_2}{\partial\phi_0^2}\dot{\phi}_0^2 + \mathbf{X}_\phi\bar{\mathbf{e}}\tfrac{\partial g_2}{\partial\phi_0}\ddot{\phi}_0\Big) \\
&- \tfrac{1}{X_\lambda}\Big(\tfrac{\partial\boldsymbol{\beta}_{\lambda,1}^T}{\partial\lambda}\dot{\lambda} + \tfrac{\partial\boldsymbol{\beta}_{\lambda,1}^T}{\partial\phi_0}\dot{\phi}_0 + \tfrac{\partial\boldsymbol{\beta}_{\lambda,1}^T}{\partial\theta}v_\theta - k_{\lambda,1}\boldsymbol{\beta}_{\lambda,1}^T\Big)\hat{\boldsymbol{\theta}}_\lambda - \tfrac{1}{X_\lambda}\boldsymbol{\beta}_{\lambda,1}^T\dot{\hat{\boldsymbol{\theta}}}_\lambda + \boldsymbol{\beta}_{\lambda,2}^T\hat{\boldsymbol{\theta}}_\lambda
\end{aligned} \tag{5.13}$$

with the control gain $k_{\lambda,2} > 0$ and

$$\boldsymbol{\beta}_{\lambda,2}^T = \tfrac{1}{X_\lambda}\begin{bmatrix}k_{\lambda,1}\frac{c_t}{m} - X_n^2 & -k_{\lambda,1}X_n + X_n\frac{c_n}{m}\end{bmatrix}\mathbf{R}_\theta^T. \tag{5.14}$$

Again, the function arguments in (5.13) and (5.14) have been omitted for compactness. The control law (5.13) yields the following dynamics of the error z_2:

$$\dot{z}_2 = X_\lambda(\lambda,\phi_0)z_1 - k_{\lambda,2}z_2 - \tilde{\boldsymbol{\theta}}_\lambda^T\boldsymbol{\beta}_2(\lambda,\phi_0,\theta) \tag{5.15}$$

and thus

$$\begin{aligned}
\dot{V}_2 = &-(\tfrac{c_t}{m}+k_{\lambda,1})z_1^2 - k_{\lambda,2}z_2^2 \\
&- \tilde{\boldsymbol{\theta}}_\lambda^T\boldsymbol{\beta}_{\lambda,1}(\lambda,\phi_0,\theta)z_1 - \tilde{\boldsymbol{\theta}}_\lambda^T\boldsymbol{\beta}_{\lambda,2}(\lambda,\phi_0,\theta)z_2 - \tfrac{1}{\gamma_\lambda}\tilde{\boldsymbol{\theta}}_\lambda^T\dot{\hat{\boldsymbol{\theta}}}_\lambda.
\end{aligned} \tag{5.16}$$

Finally, the adaptive update law is designed as

$$\dot{\hat{\boldsymbol{\theta}}}_\lambda = -\gamma_\lambda(\boldsymbol{\beta}_{\lambda,1} z_1 + \boldsymbol{\beta}_{\lambda,2} z_2), \tag{5.17}$$

thus cancelling the remaining indefinite terms in $\dot{V}_2$ and making

$$\dot{V}_2 = -(\tfrac{c_t}{m} + k_{\lambda,1}) z_1^2 - k_{\lambda,2} z_2^2 \tag{5.18}$$

negative semi-definite. Postponing the design of the adaptive update law until the last step is inspired by the tuning function procedure in [56]. Note, however, that the system structure in this article is fundamentally different from the autonomous system in [56].

Proposition 5.2. *Suppose that the robot described by* (2.33) *moves according to* (5.2) *on the manifold* Γ_3*. The adaptive control law* (5.13),(5.17) *globally asymptotically stabilizes the set*

$$\Gamma_2 = \{(\mathbf{x}, \dot{\mathbf{x}}, \lambda, \dot{\lambda}, \phi_0, \dot{\phi}_0) \in \Gamma_3 : v_t = v_{\text{ref}}\} \tag{5.19}$$

relative to Γ_3*, i. e.* $\tilde{v}_t \to 0$.

Proof. The closed-loop system with the error coordinates z_1 and z_2 reads as given by (5.9) and (5.15). With the Lyapunov function $V_2 = \frac{1}{2} z_1^2 + \frac{1}{2} z_2^2 + \frac{1}{2\gamma_\lambda} \tilde{\boldsymbol{\theta}}_\lambda^T \tilde{\boldsymbol{\theta}}_\lambda$ we get the time derivative $\dot{V}_2$ in (5.18), which is negative semi-definite. We can conclude that the origin $(z_1, z_2, \tilde{\boldsymbol{\theta}}_\lambda) = (0, 0, \mathbf{0})$ is uniformly globally stable and z_1, z_2, and $\tilde{\boldsymbol{\theta}}_\lambda$ are bounded. Again, the analysed error system is a non-autonomous system because it depends on the time-varying parameters ϕ_0, θ. However, V_2 is lower bounded by zero and $\dot{V}_2$ is finite because all time-varying parameters enter the dynamics of z_1, z_2 within a sine, cosine, or saturation function. With these conditions, we can employ Barbalat's Lemma to conclude that $(z_1, z_2) \to (0, 0)$ as $t \to \infty$. This implies that $v_t \to v_{\text{ref}}$ and $\dot{\lambda} \to \zeta$. □

Remark 5.4. Note that the result in Proposition 5.2 implies that $\tilde{\boldsymbol{\theta}}_\lambda$ is bounded. However, in order to show convergence of the estimate $\hat{\boldsymbol{\theta}}_\lambda \to \mathbf{v}_c$, one has to check if the regressor defined by $\boldsymbol{\beta}_{\lambda,1}, \boldsymbol{\beta}_{\lambda,2}$ is PE. In the particular case of snake robots, this is complicated by the fact that the regressor and thereby the PE property depend on the gait parameters, as will be discussed in Section 5.4. A formal analysis of the regressor and rigorous conditions on the gait for PE remain a topic for future work.

5.2.3 Orientation control

In the last step of the hierarchical control design, we propose an orientation controller based on the reduced system dynamics on Γ_2. The reduced system is obtained by evaluating (2.33b),(2.33f) on the invariant manifold Γ_2:

$$\dot{\theta} = v_\theta, \tag{5.20a}$$

$$\dot{v}_\theta = -\lambda_1 v_\theta + \tfrac{\psi_1(\theta)}{N-1} \bar{\mathbf{e}}^T \boldsymbol{\phi}_{\text{ref}}(\lambda, \phi_0), \tag{5.20b}$$

where $\psi_1(\theta) = \lambda_2(v_{\text{ref}} - [\cos\theta, \sin\theta]\mathbf{v}_\text{c})$. The objective of the orientation controller is to use the control input $u_\phi = \ddot{\phi}_0$ to stabilize the error coordinate $\tilde{\theta} = \theta - \theta_{\text{ref}}$ to zero. Inspired by the control design in [91], this is achieved by taking the time derivatives of (5.20b) until the input u_ϕ shows up:

$$v_\theta^{(3)} = -\lambda_1 \ddot{v}_\theta + \psi_2(\phi_0, \dot{\phi}_0, \lambda, \dot{\lambda}, \theta, v_\theta, \dot{v}_\theta) + \psi_1(\theta)\frac{\partial g_2}{\partial \phi_0} u_\phi + \psi_3(\theta, \lambda)\ddot{\lambda} \tag{5.21}$$

with

$$\begin{aligned}\psi_2(\cdot) = &- \tfrac{2\lambda_2}{N-1}\begin{bmatrix}-\sin\theta & \cos\theta\end{bmatrix} v_\theta \mathbf{v}_\text{c}\bar{\mathbf{e}}^T \dot{\boldsymbol{\phi}}_{\text{ref}}(\lambda, \dot{\lambda}, \phi_0, \dot{\phi}_0) \\ &- \tfrac{\lambda_2}{N-1}\Big(\begin{bmatrix}-\sin\theta & \cos\theta\end{bmatrix} \dot{v}_\theta \mathbf{v}_\text{c} + \begin{bmatrix}-\cos\theta & -\sin\theta\end{bmatrix} v_\theta^2 V \mathbf{v}_\text{c}\Big)\bar{\mathbf{e}}^T \boldsymbol{\phi}_{\text{ref}}(\lambda, \phi_0) \\ &- \tfrac{\lambda_2}{N-1}\Big(v_{\text{ref}} - \begin{bmatrix}\cos\theta & \sin\theta\end{bmatrix}\mathbf{v}_\text{c}\Big)\bar{\mathbf{e}}^T \begin{bmatrix}\alpha\dot{\lambda}^2 g_1(1)\sin\lambda \\ \cdots \\ \alpha\dot{\lambda}^2 g_1(N-1)\sin(\lambda - (N-2)\delta)\end{bmatrix} \\ &+ \lambda_2\Big(v_{\text{ref}} - \begin{bmatrix}\cos\theta & \sin\theta\end{bmatrix}\mathbf{v}_\text{c}\Big)\tfrac{\partial^2 g_2}{\partial \phi_0^2}\dot{\phi}_0^2\end{aligned} \tag{5.22}$$

and

$$\psi_3(\theta, \lambda) = \tfrac{\lambda_2}{N-1}\Big(v_{\text{ref}} - \begin{bmatrix}\cos\theta & \sin\theta\end{bmatrix}\mathbf{v}_\text{c}\Big)\bar{\mathbf{e}}^T \begin{bmatrix}\alpha g_1(1)\cos\lambda \\ \cdots \\ \alpha g_1(N-1)\cos(\lambda - (N-2)\delta)\end{bmatrix}. \tag{5.23}$$

All of the equations above depend on the unknown ocean current, which complicates the control design. In particular, the function ψ_1 that is multiplied with the control input contains the unknown signal, which is why we cannot design an adaptive controller analogously to the joint and velocity controllers. We will therefore make the following assumption.

Assumption 10: For the design of the orientation controller, an exact current estimate $\hat{\boldsymbol{\theta}} = \mathbf{v}_\text{c}$ is available.

Remark 5.5. The adaptive term in the velocity controller (5.13) provides an estimate of the unknown current. More specifically, the analysis in Section 5.2.2 shows that $\hat{\boldsymbol{\theta}}_\lambda$ converges and the estimation error $\tilde{\boldsymbol{\theta}}_\lambda$ is bounded. If the regressor in (5.17) is PE, the remaining offset will converge to zero.

Based on Assumption 10 and (5.21), the control law

$$u_\phi = \frac{1}{\hat{\psi}_1(\theta)\frac{\partial g_2}{\partial \phi_0}}\left[\lambda_1\tilde{\theta}^{(3)} - \hat{\psi}_2(\cdot) - k_{\phi,3}\tilde{\theta}^{(3)} - k_{\phi,2}\ddot{\tilde{\theta}} - k_{\phi,1}\dot{\tilde{\theta}} - k_{\phi,0}\tilde{\theta}\right] \tag{5.24}$$

is proposed. In (5.24), the control gains $k_{\phi,3}, k_{\phi,2}, k_{\phi,1}, k_{\phi,0} > 0$ are introduced and the superscript $\hat{\cdot}$ indicates that instead of the unknown current, the estimate $\hat{\boldsymbol{\theta}}_\lambda$ is used to compute $\psi_1(\cdot)$ and $\psi_2(\cdot)$. The same is done for the computation of the time derivatives of θ. Note that the last term on the right hand side of (5.21) cannot be compensated by the feedback-linearizing controller (5.24). This is due to the design of u_λ in (5.13), which contains a term depending on $\ddot{\phi}_0$.

Remark 5.6. Because of prioritizing the velocity control higher than the orientation control, the choice $v_{\text{ref}} > V_{\text{c,max}}$, and Assumption 10, ψ_1 is bounded away

from zero. Furthermore, $\frac{\partial g_2}{\partial \phi_0} \neq 0$, because g_2 is strictly increasing. The controller in (5.24) therefore improves previous results for terrestrial snake robots [90, 91], where singularities were an issue in the orientation controller. However, care needs to be taken when implementing (5.24). When tuning the control system it needs to be made sure that the current estimate converges sufficiently fast in order to avoid a singularity. Similarly, the control gains in (5.24) should be tuned such that ϕ_0 remains sufficiently small to not drive g_2 into saturation. Even though $\frac{\partial g_2}{\partial \phi_0} \neq 0$, numerical problems can occur for large ϕ_0.

Under Assumption 10, the following result holds for the orientation controller in (5.24):

Proposition 5.3. *Suppose that the robot described by (2.33) moves according to (5.2) and (5.13) on the manifold Γ_2 and that measurements of the ocean current velocities are available. If the control input of the velocity controller has no finite escape times and goes to zero, $u_\lambda \to 0$, the control law (5.24) globally asymptotically stabilizes the manifold*

$$\Gamma_1 = \{(\mathbf{x}, \dot{\mathbf{x}}, \lambda, \dot{\lambda}, \phi_0, \dot{\phi}_0) \in \Gamma_2 : \|\theta - \theta_{\text{ref}}\| = 0\} \tag{5.25}$$

relative to Γ_2, i. e. $\theta \to \theta_{\text{ref}}$.

Proof. The closed-loop system (5.21) with (5.24) is a linear system with the parameter-varying input matrix

$$\mathbf{b}(\lambda, \theta) = \begin{bmatrix} 0 & 0 & 0 & \frac{\lambda_2}{N-1}(v_{\text{ref}} - [\cos\theta, \sin\theta]\mathbf{v}_c) \sum_{i=0}^{N-1} \alpha g_1(i) \cos(\lambda + (i-1)\delta) \end{bmatrix}^T \tag{5.26}$$

and input $\ddot{\lambda}$:

$$\begin{bmatrix} \dot{\tilde{\theta}} \\ \ddot{\tilde{\theta}} \\ \tilde{\theta}^{(3)} \\ \tilde{\theta}^{(4)} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \\ -k_{\phi,0} & -k_{\phi,1} & -k_{\phi,2} & -k_{\phi,3} \end{bmatrix}}_{\mathbf{H}} \underbrace{\begin{bmatrix} \tilde{\theta} \\ \dot{\tilde{\theta}} \\ \ddot{\tilde{\theta}} \\ \tilde{\theta}^{(3)} \end{bmatrix}}_{\mathbf{\Theta}} + \mathbf{b}(\lambda, \theta)\ddot{\lambda}. \tag{5.27}$$

The matrix $\mathbf{H}$ is Hurwitz by design and it can be verified with the matrix exponential of $\mathbf{H}$ and the bound $\|\mathbf{b}(\lambda, \theta)\| \leq B$ that (5.27) is input-to-state stable and disturbed by the input $\ddot{\lambda}$:

$$\|\mathbf{\Theta}(t)\| \leq ce^{-\tilde{\lambda}(t-t_0)}\|\mathbf{\Theta}(t_0)\| + \frac{cB}{\tilde{\lambda}}\ddot{\lambda}. \tag{5.28}$$

If the finite bound $\|\ddot{\lambda}\| \leq \epsilon_\lambda$ exists and $\ddot{\lambda} \to 0$ as $t \to \infty$, the orientation will therefore converge to $\theta \to \theta_{\text{ref}}$ as $t \to \infty$. □

Note that even if $\ddot{\lambda}$ does not converge to zero, the disturbance by $\ddot{\lambda}$ can be made small by making $\tilde{\lambda}$ in (5.28) large, i. e. placing the eigenvalues of $\mathbf{H}$ such that they are far in the left hand plane by choosing the gains $k_{\phi,3}, k_{\phi,2}, k_{\phi,1}, k_{\phi,0}$ accordingly. In this case, $\tilde{\theta}$ is practically stabilized to zero.

5.3 Stability analysis

In Section 5.4, simulation results will demonstrate that the proposed control system (5.2),(5.13),(5.24) can stabilize all error coordinates to zero for an underwater snake robot that is exposed to an unknown, constant ocean current. In this section we will present an analysis that shows boundedness of the zero dynamics. Furthermore, we sketch how to show uniform global asymptotic stability of the closed-loop system under conditions that guarantee PE in the velocity controller.

As was pointed out in Remark 5.2, the sideways velocity v_n of the robot cannot be controlled directly. However, the following property holds for the closed-loop system.

Proposition 5.4. *Suppose that the body shape, forward velocity, and orientation of an underwater snake robot described by (2.33) are controlled according to (5.2),(5.13), and (5.24). Then, the sideways velocity, v_n, is uniformly bounded.*

Proof. With the Lyapunov function $V_n = \frac{1}{2}v_n^2$ and (2.33h) the time derivative $\dot{V}_n$ is given by

$$\dot{V}_n = -\tfrac{c_n}{m}v_n^2 + \tfrac{2c_p}{Nm}\bar{\mathbf{e}}^T\boldsymbol{\phi}v_t v_n + (\tfrac{c_n}{m}V_n - \tfrac{2c_p}{Nm}\bar{\mathbf{e}}^T\boldsymbol{\phi}V_t)v_n, \tag{5.29}$$

where V_n and V_t are the components of the current velocity in the body-aligned frame. The second and third term on the right hand side of (5.29) are indefinite. However, we know that v_t and $\bar{\mathbf{e}}^T\boldsymbol{\phi}$ are bounded since $v_t \to v_{\text{ref}}$ and $\boldsymbol{\phi} \to \boldsymbol{\phi}_{\text{ref}}$, and $\|\bar{\mathbf{e}}^T\boldsymbol{\phi}_{\text{ref}}\| \leq \epsilon_\phi$. We denote these bounds by $\|v_t\| \leq \bar{v}_t$ and $\|\bar{\mathbf{e}}^T\boldsymbol{\phi}\| \leq \bar{\epsilon}_\phi$. Furthermore we conclude from the bound on the current magnitude in Assumption 2, $V_{\text{c,max}}$, that $\|V_n\| \leq V_{\text{c,max}}$ and $\|V_t\| \leq V_{\text{c,max}}$.

Therefore we have that

$$\dot{V}_n \leq -\tfrac{c_n}{m}v_n^2 + v_n k. \tag{5.30}$$

We can now use Young's inequality [6]

$$ab \leq \tfrac{\xi a^2}{2} + \tfrac{b^2}{2\xi}, \quad \xi > 0 \tag{5.31}$$

to reformulate (5.30) to

$$\dot{V}_n \leq (-\tfrac{c_n}{m} + \tfrac{\xi}{2})v_n^2 + \tfrac{k^2}{2\xi}. \tag{5.32}$$

By choosing $\xi < \frac{2c_n}{m}$ we can make sure that the coefficient of v_n^2 is negative and conclude from the Comparison Lemma (Lemma A.3) that

$$V_n(t) \leq e^{-c_1 t}V_n(0) + c_2, \tag{5.33}$$

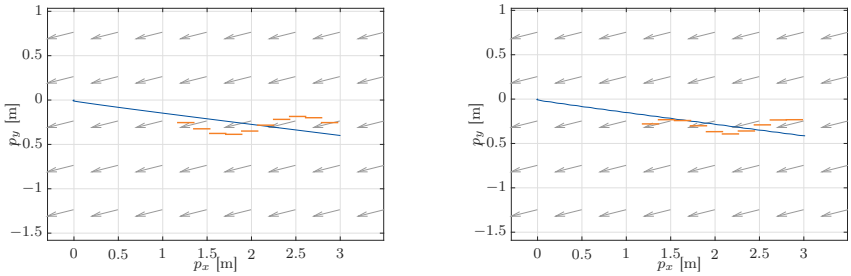
which implies that v_n is bounded. □

Remark 5.7. Note that the result in Proposition 5.4 does not depend on Assumption 10.

Future work will establish conditions on the gait in (5.1) under which the regressor in (5.17) is PE, and the convergence of the current estimate to the true

Table 5.1: Control gains of the adaptive velocity and orientation control system.

k_1	k_2	γ_1	$k_{\lambda,1}$	$k_{\lambda,2}$	γ_2	$k_{\phi,0}$	$k_{\phi,1}$	$k_{\phi,2}$	$k_{\phi,3}$
3	6	1	0.1	10	0.04	2	21	60	20



(a) The robot for $\delta = 40°$

(b) The robot for $\delta = 50°$

Figure 5.2: Simulation results of the velocity controller: The path of the robot.

value can be shown. With these conditions, the orientation controller can be shown to asymptotically stabilize the set

$$\Gamma_1 = \{(\mathbf{x}, \dot{\mathbf{x}}, \lambda, \dot{\lambda}, \phi_0, \dot{\phi}_0) \in \Gamma_2 : \|\theta - \theta_{\text{ref}}\| \leq \epsilon\} \tag{5.34}$$

relative to Γ_2, if all states are bounded.

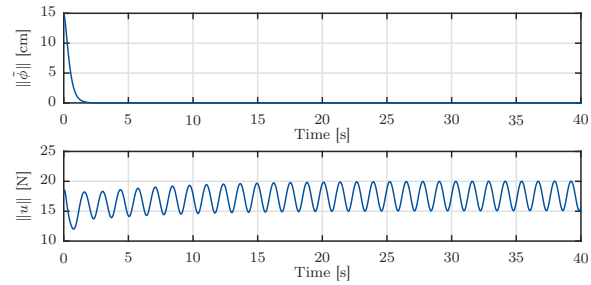
In previous studies for terrestrial snake robots [90, 91], numerical simulations were used to show that the states generated by u_λ and u_ϕ remain bounded. Similarly, in the next section of this article, we will see simulation results that indicate that all states remain bounded, and the error coordinates converge to zero.

5.4 Simulation study

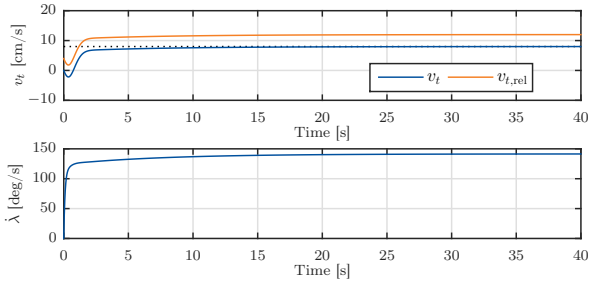
This section presents simulation results that illustrate the performance of the control system that was developed in this chapter.

5.4.1 Simulation set-up

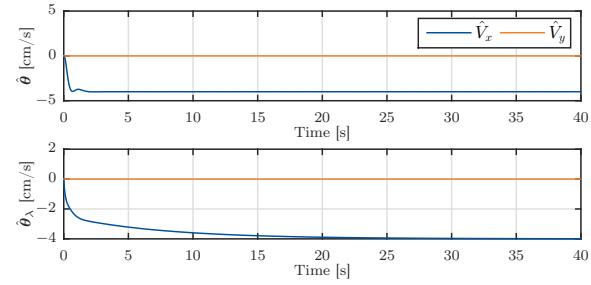
The model (2.33) and the control system (5.2),(5.13),(5.24) were implemented in Matlab and simulated using the `ode23t` solver with an absolute and relative error tolerance of 10e-6. The model parameters were chosen according to Table 2.3 and the snake robot was exposed to an ocean current with the components $V_x = -4$ cm/s and $V_y = -1$ cm/s. The gains of the control system are displayed in Table 5.1. They were obtained by tuning the control system stage by stage. The gait parameters of (5.1) were chosen as $\alpha = 7$ cm and $\delta = 40°$, and the gait functions were chosen as $g(i) = 1$ and $g_2(\phi_0) = \phi_{0,\max} \tanh(\frac{\phi_0}{\phi_{0,\max}})$ with $\phi_{0,\max} = \alpha$. The references for the velocity and orientation controller were set to $v_{\text{ref}} = 8$ cm/s and



(a) Joint errors and control force



(b) Forward velocity and $\dot{\lambda}$



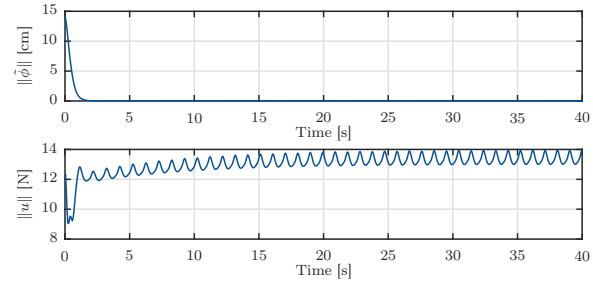
(c) The current compensation terms of the adaptive controllers

Figure 5.3: Simulation results of the velocity controller for $\delta = 40°$: All controlled states converge to their reference values. All control inputs remain bounded.

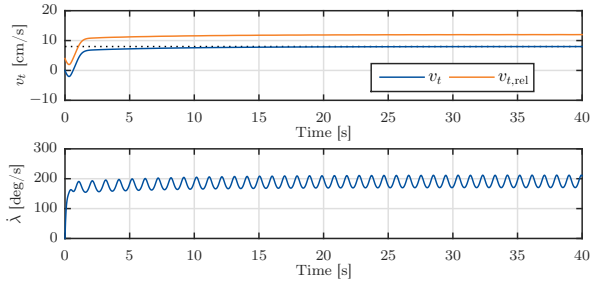
$\theta_{\mathrm{ref}} = 45°$. All states were initialized at zero, i. e. the robot was initially straight, aligned with the x-axis, and fixed in the origin. As a first step, the control system was simulated without the restrictive Assumption 10, i. e. only the joint and velocity controllers were included, and the orientation control input u_ϕ was set to zero. In a second step, the entire control system was simulated.

5.4.2 Simulation results

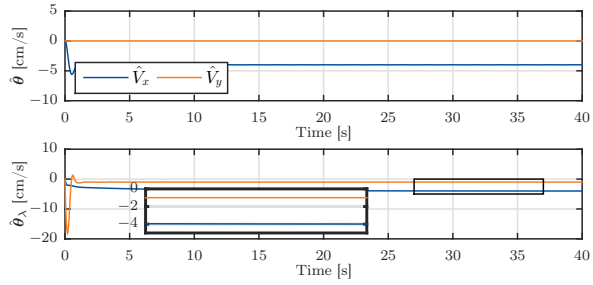
Figure 5.2(a) shows the path of the robot during the first simulation. The error signal of the joint controller and the control torque are displayed in Figure 5.3(a), the forward velocity and the state related to the body frequency, $\dot{\lambda}$, in Figure 5.3(b),



(a) Joint errors and control force



(b) Forward velocity and $\dot{\lambda}$

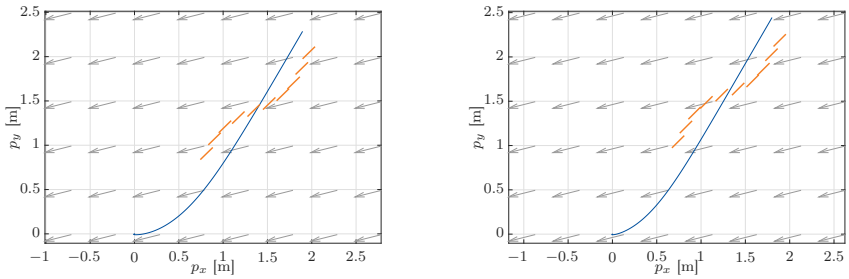


(c) The current compensation terms of the adaptive controllers

Figure 5.4: Simulation results of the velocity controller for $\delta = 50°$: All controlled states converge to their reference values. All control inputs remain bounded.

and the adaptive term of both the joint and the velocity controller in Figure 5.3(c). It becomes clear from the figures that all error coordinates converge to zero and all signals remain bounded. However, for the current estimate of both control stages, only one component converges to the correct value. Clearly, the regressors used in the adaptive control laws are not PE for the particular choice of gait parameters.

The simulation was therefore re-run with a phase shift of $\delta = 50°$. The path of the robot during this simulation is displayed in Figure 5.2(b). Figure 5.4 shows the analogous signals of Figure 5.3 for the second simulation. All controlled states still converge, all signals remain bounded, and the current term of the joint controller still does not provide an exact estimate. However, it can now be seen from Figure 5.4(c) that the current estimate of the velocity controller, $\hat{\boldsymbol{\theta}}_\lambda$, converges



(a) The robot for $\delta = 40°$

(b) The robot for $\delta = 50°$

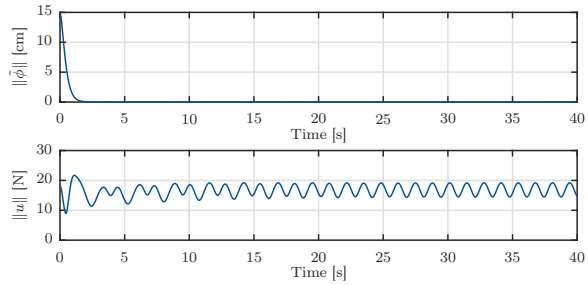
Figure 5.5: Simulation results of the complete controller: The path of the robot.

to the correct values after an initial overshoot. This overshoot is a result of not tuning the control gains for the new gait parameters. The parameter convergence in Figure 5.4(c) indicates that Assumption 10 is a valid assumption if the gait parameters are chosen accordingly.

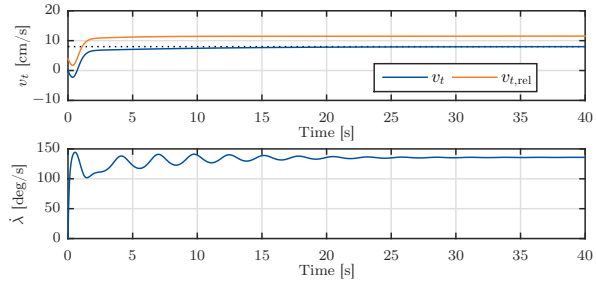
Both simulations were repeated with the orientation controller activated. The path of the robot during both scenarios is shown in Figure 5.5, where we see that the robot now turns towards its reference orientation. The signals of the single control stages are displayed in Figures 5.6 and 5.7, which show that all error signals converge to zero while all signals remain bounded. In addition to the signals presented in the above discussion, the orientation of the robot, and the offset ϕ_0 that induces turning motion are now displayed in Figures 5.6(c) and 5.7(c). It can be concluded from Figure 5.6(c) that the orientation of the robot converges towards its reference, even for $\delta = 40°$, where Assumption 10 is violated in the first scenario. This is a result of the turning of the robot, which provides additional information to the adaptive controller. The same effect can be observed in Figure 5.6(d), where the current estimate of the velocity controller now converges to the correct value.

5.5 Chapter summary

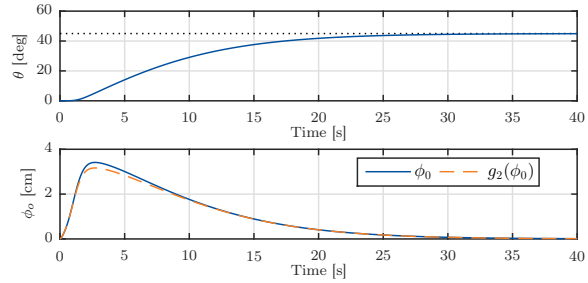
We have presented a control system for velocity and orientation control of underwater snake robots using absolute velocity feedback. The control system has a hierarchical structure, where the highest priority is to stabilize virtual joint constraints that encode a planar gait. In order to do so, we designed an adaptive joint controller and showed that the controller asymptotically stabilizes the constraint manifold. The second and third priority of the control system were to control the velocity and the orientation of the robot using dynamic compensators whose states parametrize the virtual joint constraints. We proposed an adaptive controller that asymptotically stabilizes the forward velocity error to zero and an orientation controller that utilizes the current estimate of the velocity controller. It was shown that the zero dynamics of the closed-loop system remains bounded, and simulation results demonstrated the performance of the controller.



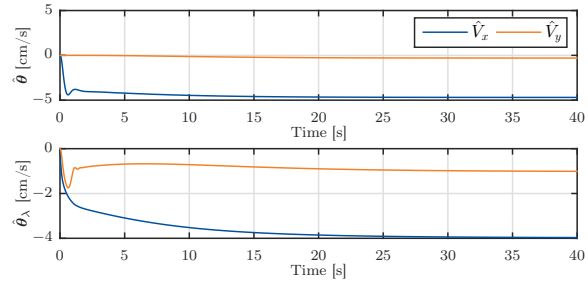
(a) Joint errors and control force



(b) Forward velocity and $\dot{\lambda}$

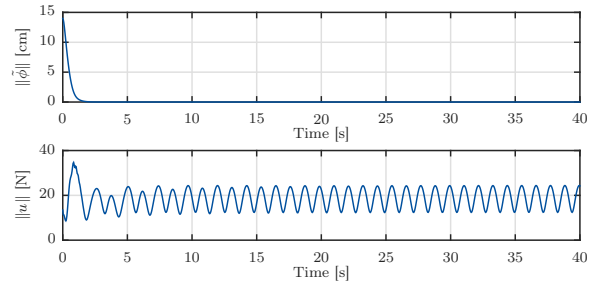


(c) Orientation angle θ and ϕ_0

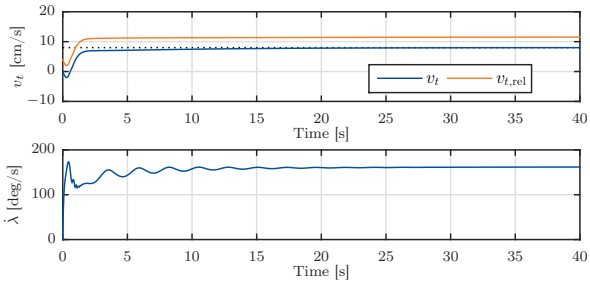


(d) The current compensation terms of the adaptive controllers

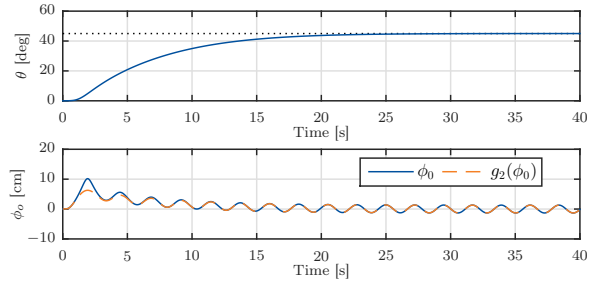
Figure 5.6: Simulation results of the complete controller for $\delta = 40°$: All controlled states converge to their reference values. All control inputs remain bounded.



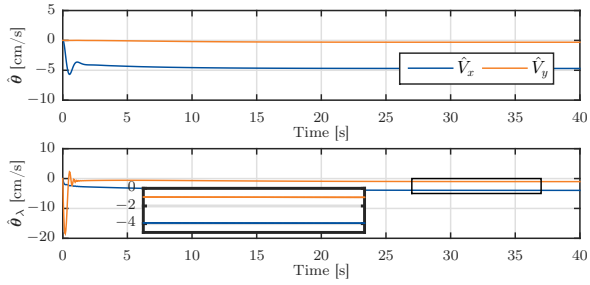
(a) Joint errors and control force



(b) Forward velocity and $\dot{\lambda}$



(c) Orientation angle θ and ϕ_0



(d) The current compensation terms of the adaptive controllers

Figure 5.7: Simulation results of the complete controller for $\delta = 50°$: All controlled states converge to their reference values. All control inputs remain bounded.

Chapter 6

Set-Based Path-Following and Obstacle Avoidance Guidance

This chapter presents a set-based guidance strategy for path-following with obstacle avoidance for underwater snake robots conducting planar sinusoidal motion. The guidance scheme is designed such that the robot follows a straight path and avoids obstacles on the way by following a circle around them. In order to enable a switching between these two tasks, we generalize a strategy that was introduced for surface vessels, by making the switching condition independent of the dynamics and thus applicable for a larger class of systems. The guidance system is shown to fulfil the control objectives at a kinematic level. We present new test results that validate the set-based guidance scheme for the first time experimentally.

Contributions of this chapter The main idea of the proposed guidance strategy is to make the robot converge to and follow a straight path and only leave it in order to circumvent an obstacle that is in the way. In that case, the guidance switches into obstacle avoidance mode, and the robot follows a circle around the obstacle. As soon as the obstacle is passed, the robot converges to the original straight path again. To this end, we generalize the set-based guidance strategy for obstacle avoidance of autonomous surface vessels in [73] in order to make it suitable for snake robots. More specifically, the switching conditions are reformulated in a more general, purely kinematic manner. This makes the switching strategy independent of the dynamic model, and thus applicable to a more general class of systems, including snake robots. For a snake robot, considering the dynamics would introduce oscillations to the guidance due to the oscillating nature of snake robot locomotion, and thus make the set-based system vulnerable to chattering. Furthermore, the new switching strategy is combined with a modified guidance law for snake robots from [78]. This guidance controller is suitable for generic paths, and can thus be applied both for the straight line path-following mode, and the obstacle avoidance mode, which requires a circular reference path. In this chapter, the guidance law is modified in order to allow circular path-following in both directions, thus enabling the robot to choose the shortest way around the obstacles blocking its path. We show that the modifications to the guidance law preserve the stability

properties. Finally, the obstacle avoidance strategy is experimentally tested with a swimming snake robot for different stationary obstacles. The experimental results validate the proposed set-based path-following and obstacle avoidance guidance scheme, and are the first test results of the set-based control strategy that have been obtained with a floating-base robot.

Organization of this chapter In Section 6.1, some basic assumptions are made. The control objectives are formulated in Section 6.2. In Section 6.3, the set-based path-following and obstacle avoidance guidance strategy for underwater snake robots is proposed and analysed. An experimental study, which validates the approach, is presented in Section 6.4. Concluding remarks are given in Section 6.5.

Publications This chapter is based on [54].

6.1 Basic assumptions

This chapter focuses on a snake robot that employs lateral undulation according to (1.2):

$$\phi_i = \alpha \sin(\omega t + (i-1)\delta) + \phi_0, \quad i = 1, \ldots, N-1. \tag{6.1}$$

Assumption 11: The underwater snake robot is neutrally buoyant and moves in a plane according to (6.1), with a forward velocity $v_t > 0$.

Remark 6.1. It has been shown that lateral undulation according to (6.1) results in a forward velocity in Section 2.5.

Assumption 12: The robot is not exposed to ocean currents.

Assumption 13: The robot moves within an environment containing k stationary obstacles. The single obstacles are located sufficiently far away from each other so that the robot can travel safely between them.

Remark 6.2. Assumption 13 implies that only one obstacle at a time has to be accounted for by the guidance system.

6.2 Control objectives

The problem of following a path whilst avoiding obstacles along the way can be considered as a multiple-task control problem, where the path-following objective defines one task, and the obstacle avoidance objective defines another, higher prioritized task. The snake robot is supposed to follow a straight path while avoiding k stationary obstacles on the way. This is achieved by ensuring that the robot always stays outside a circle with obstacle j located in the circle center, and a safe radius $r_{\mathrm{s},j}$. The tasks of path-following and simultaneous obstacle avoidance can be in conflict with each other if an obstacle is on or close to the path. In this case, we will prioritize the obstacle avoidance task over the path-following task in order to ensure a safe operation. Thus, we can formalize the above considerations in the following prioritized control objectives:

Objective 1: The distance between the CM of the snake robot (p_x, p_y) and the center of every obstacle j, $(p_{\mathrm{o},jx}, p_{\mathrm{o},jy})$, should always be greater than or equal to some safe radius $r_{\mathrm{s},j}$: $\sqrt{(p_x - p_{\mathrm{o},jx})^2 + (p_y - p_{\mathrm{o},jy})^2} \geq r_{\mathrm{s},j}$, $j = 1, \ldots, k$.

Remark 6.3. As opposed to conventionally propelled marine vehicles, an underwater snake robot changes its configuration continuously. It can therefore not be guaranteed that keeping the CM outside of the obstacle will at all times prevent collisions. We therefore propose to define the safe radius r_{s} as the maximal extent of the obstacle r_{o} plus half the snake length $r_{\mathrm{s}} = r_{\mathrm{o}} + \frac{NL}{2}$.

Objective 2: The robot should, without loss of generality, converge to and follow the global y-axis: $\lim_{t \to \infty} \|p_y(t)\| = 0$.

The difference between the tasks is that the path-following task constitutes an equality task, whereas the obstacle avoidance task is set-based, which can be seen from the inequality in Objective 1.

6.3 Set-based guidance for path-following and obstacle avoidance

In [75], a singularity-robust multiple task-priority inverse kinematics framework was proposed, which can handle both equality and set-based tasks. The framework from [75] was later used for collision avoidance of underactuated surface vessels in [73]. In this section, we will generalize the framework from [73], by making the conditions that switch between the single tasks independent of the dynamic model and thus applicable for other systems as well. In order to make it suitable for snake robots, the new set-based switching strategy is combined with a general path-following guidance for snake robots. The guidance method provides both the option to follow a straight reference path, which is well-suited to for the path-following task, and a circular reference path, which can be used for the obstacle avoidance task. In the following, we will derive the strategy for one obstacle for simplicity. Note that it is straightforward to extend it to k obstacles by repeating the strategy k times, once for each obstacle, as long as the obstacles are not overlapping.

6.3.1 Definitions

As a preparation for the set-based guidance scheme that will be introduced in Sections 6.3.2 and 6.3.3, we make the following definitions:

Definition 8: The path-following mode is defined as Mode 1, and the obstacle avoidance mode as Mode 2.

Definition 9: The set-based obstacle avoidance task σ_{o} is defined as the distance between the robot and the obstacle center $\sigma_{\mathrm{o}} = \sqrt{(p_x - p_{\mathrm{o},x})^2 + (p_y - p_{\mathrm{o},y})^2}$, and the path-following equality task σ_{p} as the distance between the robot and the path $\sigma_{\mathrm{p}} = p_y$.

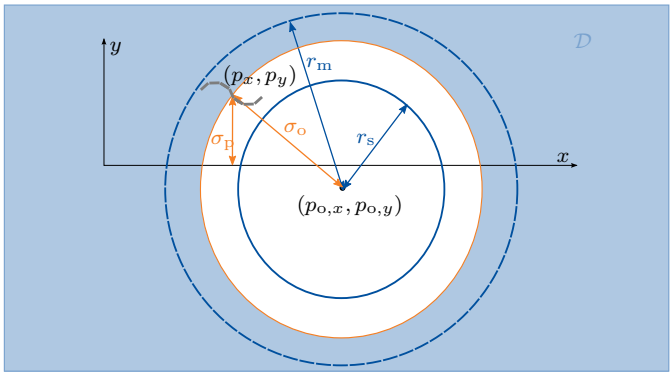


Figure 6.1: Definitions of the set-based framework.

The key idea of the framework [75], that we will base the guidance strategy on, is to include set-based tasks into the multiple task control framework by freezing the set-based task, thus turning it into an equality task, if it is about to leave its valid set $\mathcal{D}$.

Definition 10: The valid set for σ_o is $\mathcal{D} = [\sigma_\mathrm{min}, \infty)$ with the boundary $\sigma_\mathrm{min} = \min(r_\mathrm{m}, \max(\sigma_\mathrm{o}, r_\mathrm{s}))$, and the tangent cone to $\mathcal{D}$ is

$$T_\mathcal{D}(\sigma_\mathrm{o}) = \begin{cases} \mathbb{R}_o^+, & \sigma_\mathrm{o} = \sigma_\mathrm{min} \\ \mathbb{R}, & \sigma_\mathrm{o} > \sigma_\mathrm{min} \end{cases}. \tag{6.2}$$

The tangent cone $T_\mathcal{D}(\sigma_\mathrm{o})$ is used as an indicator of whether the task is about to leave the set $\mathcal{D}$: this is the case if the task derivative $\dot{\sigma}_\mathrm{o} \notin T_\mathcal{D}(\sigma_\mathrm{o})$.

It was pointed out in [73] that the framework from [75] has to be adapted when used with underactuated dynamic systems, which can be achieved by introducing a mode change radius $r_\mathrm{m} > r_\mathrm{s}$. Within the circle with radius r_m, switching from path-following to obstacle avoidance mode can be activated if certain conditions are satisfied. These conditions will be discussed in Section 6.3.3.

Definition 11: The radius within which obstacle avoidance can be active is called the mode change radius r_m.

As soon as these conditions are no longer valid, the control system switches back to the default path-following mode. The above definitions are illustrated in Figure 6.1. In order to ensure that both control objectives are fulfilled, we make an additional assumption:

Assumption 14: The mode change radius r_m is chosen sufficiently large for the robot to converge to steady state in obstacle avoidance mode without overshooting.

6.3.2 The guidance law

The general formulation of a reference velocity vector $\boldsymbol{\mu} = [\mu_x, \mu_y]^T$ for the robot was proposed for terrestrial snake robots in [78] as

$$\boldsymbol{\mu}(\mathbf{p}) = -\frac{dh_p^T}{\|dh_p\|^2}\big(k_{\text{tran}} h(\mathbf{p})\big) + \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} dh_p^T \frac{v}{\|dh_p\|}, \tag{6.3}$$

where $h(\mathbf{p})$ is an error function implicitly defining the desired path, $dh_p^T = \nabla h(\mathbf{p})$ is a vector that is normal to the level sets of h, $k_{\text{tran}} > 0$ is a transversal gain, and $v > 0$ is the desired along-path velocity. It was shown in Chapter 4 that (6.3) can also be used for underwater snake robots, as long as there are no currents, which is in accordance with Assumption 12.

In this chapter, the guidance law (6.3) is modified in order to enable a choice of direction:

$$\begin{aligned} \boldsymbol{\mu}(\mathbf{p}) &= -\frac{dh_p^T}{\|dh_p\|^2}\big(k_{\text{tran}} h(\mathbf{p})\big) + \nu \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} dh_p^T \frac{v}{\|dh_p\|}, \\ \nu &= \begin{cases} -1, & \text{Mode} = 2 \ \wedge \ p_{\text{o},y} > 0 \\ +1, & \text{else.} \end{cases} \end{aligned} \tag{6.4}$$

If $p_{\text{o},y} > 0$, the sign of the second term is changed from positive to negative in obstacle avoidance mode, which makes the robot follow the circle counter-clockwise instead of clockwise, thus always choosing the shortest way around the obstacle. The reference heading θ_{ref} for the robot is then obtained from (6.4) by the relation

$$\theta_{\text{ref}} = \arctan(\tfrac{\mu_y}{\mu_x}). \tag{6.5}$$

For Mode 1, the error function $h(\mathbf{p})$ in (6.4) is simply given by $h_1(\mathbf{p}) = p_y$. In the second case, Mode 2, $h(\mathbf{p})$ is defined as $h_2(\mathbf{p}) = (p_x - p_{\text{o},x})^2 + (p_y - p_{\text{o},y})^2 - r_{\text{s}}^2$, describing a circle with radius r_{s} around the obstacle.

Proposition 6.1. *If the heading reference (6.5) obtained from the guidance law (6.4) is tracked by an underwater snake robot in a sufficiently accurate manner, the robot is practically stabilized to the desired path. In particular, for any $\epsilon > 0$, there exists a k_{tran} such that the set $\{h(\mathbf{p}) \leq \epsilon\}$ is asymptotically stable, i. e. the path-following error function $h(\mathbf{p})$ stays close to zero.*

Proof. The result follows from the proof of Proposition 17 in [78]. It is straightforward to extend the Lyapunov analysis in [78] for the adapted guidance law in (6.4), because the second part on the RHS of (6.3) leads to an indefinite term that can be cancelled out by a sufficiently large k_{tran}. The analysis therefore does not depend on the sign of ν. □

Remark 6.4. In this chapter we focus only on the heading control of the robot, which means that we do not actively control the velocity. This is in accordance with Assumption 11, which states that the sinusoidal gait (6.1) ensures some forward velocity v_t. The desired velocity v in (6.4) will therefore be treated as a positive control constant.

6.3.3 Switching conditions

In the following we propose a strategy for switching between the two modes of the system that were presented in the previous paragraphs. In contrast to the condition in [73], the switching condition in this work is independent of the vehicle dynamics. This is necessary because snake robots moving according to (6.1) display an oscillating behaviour. Using the dynamical model of such a robot in the switching conditions would therefore result in a control system that is susceptible to chattering.

The general idea of the strategy is that it is safe to do path-following as long as the robot is outside of the mode change radius of the obstacle, $\sigma_o \geq r_\mathrm{m}$, and thus the default mode of the system is chosen to be Mode 1. In this case, the equality based task is active, and the set-based task is implicitly satisfied and thus inactive. As soon as the vehicle is inside the mode change radius, i. e. $\sigma_\mathrm{o} < r_\mathrm{m}$, Mode 1 might no longer be safe. Whether Mode 1 is safe or not can be determined by checking if the reference velocity according to Mode 1 will drive the robot closer towards the obstacle. In this case, the system needs to switch into Mode 2, which means that the set-based task is activated by changing the control objective to $\sigma_o = r_\mathrm{s}$ and thus making the robot follow a circle with the safe radius r_s around the obstacle, and hence achieving obstacle avoidance. As soon as the guidance according to Mode 1 will take the robot further away from the obstacle, it is safe to switch back to Mode 1, move back to the set-based objective $\sigma_o \geq r_\mathrm{s}$, and thus fulfil both control objectives. This strategy can be formalized using the task derivative $\dot{\sigma}_\mathrm{o}$ and the set $\mathcal{D}$ with its tangent cone $T_\mathcal{D}(\sigma_\mathrm{o})$ from Definition 10. The set $\mathcal{D}$ corresponds to the area that is safe for the robot to move towards. If $\dot{\sigma}_\mathrm{o} \in T_\mathcal{D}(\sigma_\mathrm{o})$, the time evolution of the task σ_o remains in the set $\mathcal{D}$. If $\dot{\sigma}_\mathrm{o} \notin T_\mathcal{D}(\sigma_\mathrm{o})$, the time evolution of the task σ_o moves out of the set $\mathcal{D}$, i. e. the vehicle comes closer to the obstacle, and the guidance switches into Mode 2.

Whether the task derivative $\dot{\sigma}_\mathrm{o}$ is in the tangent cone $T_\mathcal{D}(\sigma_\mathrm{o})$ can be checked by looking at the orientation of the reference velocity vector $\boldsymbol{\mu}$. This relies on the assumption that the controllers will ensure tracking of the heading reference orientation θ_ref. How to check if the task derivative is in the tangent cone $\dot{\sigma}_\mathrm{o} \in T_\mathcal{D}(\sigma_\mathrm{o})$ by analysing $\boldsymbol{\mu}$ is visualized in Fig. 6.2. It is done by comparing the reference angle θ_ref to the angle θ_o, which is defined as

$$\theta_\mathrm{o} = \arctan\left(\frac{p_y - p_{\mathrm{o},y}}{p_x - p_{\mathrm{o},x}}\right). \tag{6.6}$$

The angle θ_o is the angle between the global x-axis and the virtual line between the robot and the obstacle. If their absolute values add up to less than 90°, the vector $\boldsymbol{\mu}$ points away from the obstacle and is safe to track.

Proposition 6.2. *Given the guidance system (6.4), (6.5). If the condition* $|\theta_\mathrm{ref} - \theta_\mathrm{o}| \leq \frac{\pi}{2}$ *holds for the reference heading* θ_ref*, Mode 1 of the guidance provides a reference that fulfils both Objectives 1 and 2.*

Proof. A reference velocity vector with an orientation that fulfils the above condition will increase the distance between the robot and the obstacle, and thus not violate Objective 1. It will furthermore, according to Proposition 6.1, make the

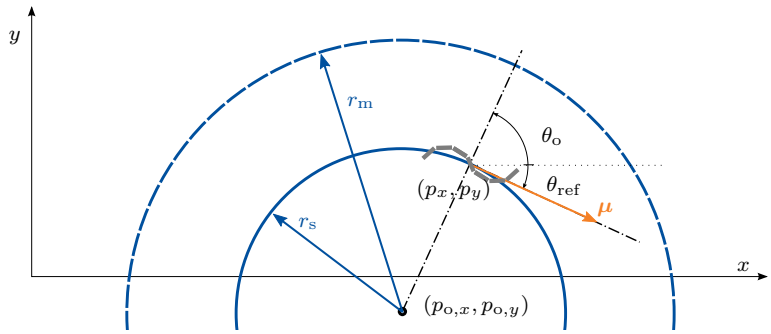


Figure 6.2: Checking if the conditions for switching back to Mode 1 are met: if $|\theta_{\mathrm{ref}} - \theta_{\mathrm{o}}| \leq \frac{\pi}{2}$, the vector $\boldsymbol{\mu}$ is safe to track.

robot converge to the set $\{p_y \leq \epsilon\}$, which can be made arbitrarily small, and thus fulfil also Objective 2. □

If the reference of the guidance law is tracked exactly by the low level controllers, the switching condition is equivalent to the one in [73].

Remark 6.5. The task derivative $\dot{\sigma}_{\mathrm{o}}$ depends on the dynamical model of the robot. In this work, we disregard the dynamics of the system in the design of the switching conditions, and base these solely on the kinematics. This choice was made because of the oscillating nature of snake robot locomotion according to (6.1), which inevitably implies oscillations of the center of mass. Therefore, basing the switching condition on the actual velocity of the robot predicted by a dynamic model would result in chattering between the two modes, as the robot oscillates. The reference velocity, on the other hand, is only based on the kinematics, and is therefore much less exposed to oscillations, and thus a better fit for designing the switching conditions. With the geometric considerations illustrated in Figure 6.2, it is possible to determine if $\dot{\sigma}_{\mathrm{o}} \in T_{\mathcal{D}}(\sigma)$, provided that the control system tracks the reference, without explicitly computing $\dot{\sigma}_{\mathrm{o}}$. Note that this approach makes the guidance strategy more general in the sense that it can be used for any type of vehicle, because it is independent of the dynamic model.

6.3.4 The switching algorithm

The implementation of the switching strategy from the previous paragraphs is summarized in Algorithm 1. As an extension to k obstacles, the algorithm is run in a for-loop for $i = 1, \ldots, k$, with each iteration checking one obstacle.

Remark 6.6. The proposed approach is general in the sense that it is independent of the dynamics of the robot. This implies that the proposed guidance for path-following with obstacle avoidance can also be applied to terrestrial snake robots without any modifications.

For the proposed guidance algorithm, the following stability result holds.

Initialize:
last_mode = path_following;
while *True* **do**
  if $\sigma_o \geq r_{\mathrm{m}}$ **then**
    $h(\mathbf{p}) = h_1(\mathbf{p})$;
    mode = path_following;
  else
    if $|\theta_{\mathrm{ref}} - \theta_{\mathrm{o}}| \leq \frac{\pi}{2}$ **then**
      $h(\mathbf{p}) = h_1(\mathbf{p})$;
      mode = path_following;
    else
      $h(\mathbf{p}) = h_2(\mathbf{p})$;
      mode = obstacle_avoidance;
    end
  end
  last_mode = mode
end

Algorithm 1: The set-based path-following and obstacle avoidance guidance scheme for underwater snake robots.

Proposition 6.3. *Consider a neutrally buoyant underwater snake robot conducting planar motion according to (6.1). Provided Assumptions 11 to 14 hold and the reference heading given by the set-based guidance algorithm in Algorithm 1 is tracked, Objective 1 is fulfilled. Furthermore, as long as the system is in Mode 1, Objective 2 is satisfied in the sense that* $\lim_{t\to\infty} \|p_y(t)\|$ *can be made arbitrarily small.*

Proof. According to Proposition 6.1, the guidance law (6.4) makes sure that the robot converges to the set $\mathcal{H} = \{h_2(\mathbf{p}) \leq \epsilon\}$ in Mode 2. Because the reference $\boldsymbol{\mu}$ is aligned with the path tangential when the robot is on the path, the robot will converge to a circle with a radius larger than r_{s}, the offset can be made small by making ϵ small, i. e. choosing a large k_{tran}. Given Assumption 14, we can use the result from [75] regarding satisfaction of set-based tasks with the valid set $\mathcal{D}$ from Definition 10 to show that the robot converges to $\mathcal{H}$ without violating Objective 1. It follows directly from Proposition 6.1, that Objective 2 is satisfied in the sense that $\lim_{t\to\infty} \|p_y(t)\|$ can be made arbitrarily small if the system is in Mode 1. □

6.4 Experimental study

In this section we present an experimental study that validates the proposed set-based guidance scheme. First, the set-up of the experimental tests is described, before the experimental results are demonstrated.

6.4.1 Experimental set-up

The set-based path-following and obstacle avoidance guidance control scheme in Algorithm 1 was tested in the Marine Cybernetics lab (MC-lab)[1]. The MC-lab

[1]The Marine Cybernetics lab (MC-lab) – operated by the Department of Marine Technology at the Norwegian University of Science and Technology.

Figure 6.3: The snake robot Mamba in the MC-lab. The basin is equipped with six motion capture cameras, three mounted on each side.

comprises a 1.5 m deep basin of 40×6.45 m^2 and is equipped with six cameras of the underwater motion capture system Qualisys, that are used to measure the position of reflective markers inside the basin. The camera set-up is shown in Figure 6.3. The snake robot Mamba served as the test platform for the guidance strategy. The robot can also be seen in Figure 6.3. Details about the robot can be found on Appendix B. During the tests, a construction with five reflective markers was attached to the head of the robot in order to provide a reference for the camera positioning system. The geometry of the attachment makes sure that its three-dimensional pose can be determined by the positioning system as soon as it is within the range of at least two cameras. The position of the robot according to the definition in Chapter 2 was computed from the position of the markers by using the kinematic relations from Chapter 2. The orientation of the robot was defined by the orientation of the head link, and could therefore be obtained directly from the motion capture system.

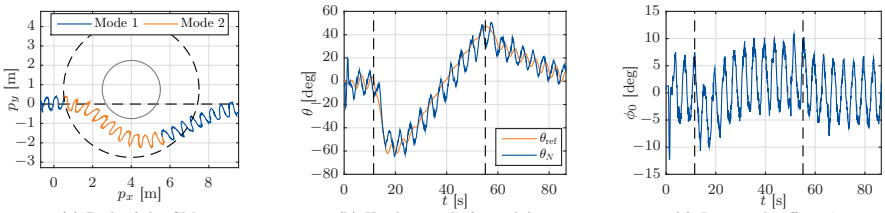
The guidance algorithm, Algorithm 1, was implemented in LabVIEW. The parameters in (6.4) were chosen as $k_{\text{tran}} = 0.02$ and $v = 0.05$ for Mode 1, and $k_{\text{tran}} = 0.012$ and $v = 0.05$ for Mode 2. The required position measurements were sent from the Qualisys Motion Tracker (QTM) software to LabVIEW at a frequency of 10 Hz in order to be available for the guidance. In order to make the switching feasible for the robot, the smoothing function $f_{\text{s}}(t, t_{\text{switch}})$ was used to interpolate between the heading reference of the active mode and the last heading reference before the previous switch:

$$f_{\text{s}}(t, t_{\text{switch}}) = \tfrac{1}{2} \tanh(0.8(t - t_{\text{switch}} - 2)) + \tfrac{1}{2}. \tag{6.7}$$

The time constant of the smoothing function was chosen such that the jump in the reference was smoothed within one oscillation of the snake robot. The resulting reference signal θ_{ref} was then sent to the heading controller. For the heading control input ϕ_0 in (6.1), a PD-controller was implemented:

$$\phi_o = -k_p(\theta_N - \theta_{\text{ref}}) - k_d(\dot{\theta}_N - \dot{\theta}_{\text{ref}}). \tag{6.8}$$

The gains of the controller were tuned as $k_p = 0.56$ and $k_d = 0.03$. The time derivative $\dot{\theta}_N$ was obtained by numerically differentiating the angular measurement



(a) Path of the CM (b) Heading angle θ_N and θ_{ref} (c) Joint angle offset, ϕ_0

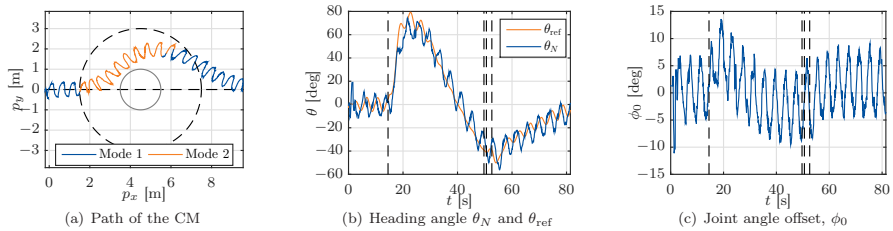
Figure 6.4: Experimental results of the first scenario: An obstacle with safety radius $r_{\text{s}} = 1.5$ m was placed 0.75 m to the left side of the path. In (a), the safe radius r_{s} is displayed as a grey, and the mode change radius r_{m} as a dashed circle. The dashed lines in (b) and (c) indicate the switching times.

from the positioning system in LabVIEW and the time derivative of the reference heading, $\dot{\theta}_{\text{ref}}$, was obtained from (6.5) through a third order low-pass filtering reference model. The frequency of the reference model was chosen as $\frac{\pi}{2}$, and the damping was set to one. The resulting signal ϕ_0 was saturated at $\phi_{0,\max} = \pm 20°$ to respect the joint limitations, and sent to the low level joint controllers that enforce the reference (6.1) on the single joints. These are P-controllers that are included in the servo motors inside each joint. The parameters in (6.1) were chosen as $\alpha = 30°$, $\omega = 90°$, and $\delta = 50°$. In our tests the position of the obstacle was assumed to be known and directly included in the implementation. Combining the set-based guidance system with robot vision in order to detect the obstacles autonomously will be subject of future work.

6.4.2 Experimental results

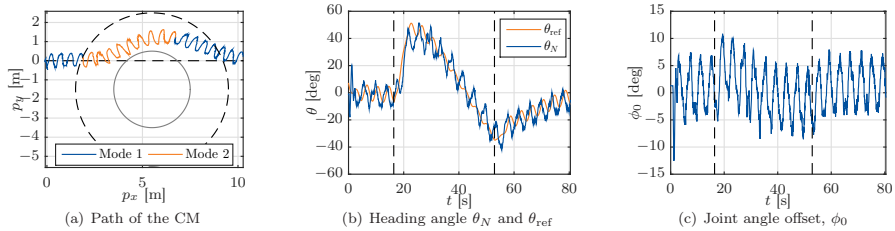
The set-based path-following and obstacle avoidance guidance was tested in three different scenarios with obstacles of different sizes and location. Because the range of the motion capture system was limited to a length of ca. 10 m, only one obstacle at a time was considered. Results of one case of each scenario are illustrated in Figures 6.4 to 6.6. Photos of the robot Mamba during a test run are presented in Figure 6.7.

Because of short available range of the camera system, the initial conditions were chosen such that the robot started on and parallel to the path: $p_y(0) \approx 0, \theta_N(0) \approx 0$. The path of the robot is plotted in Figures 6.4(a), 6.5(a) and 6.6(a). It can be seen that in every scenario, the robot left the path right after entering the mode change circle. The robot then followed a circular path without violating the safety radius. The offset of the robot to the grey circle was intentionally achieved by the choice of the transversal gain k_{tran} for Mode 2: in accordance with Proposition 6.1, the offset to the circle can be made small by a higher choice of k_{tran}. A better tracking of the circle would however lead to an intersection of some of the robot's oscillations with the safety radius, so in accordance with Objective 1, we chose a small gain k_{tran}. In all three scenarios, the robot managed to converge back to the path just before leaving the range of the camera system. The reference θ_{ref} that was provided by the guidance law is plotted in Figures 6.4(b), 6.5(b) and 6.6(b).



(a) Path of the CM (b) Heading angle θ_N and θ_{ref} (c) Joint angle offset, ϕ_0

Figure 6.5: Experimental results of the second scenario: An obstacle with safety radius $r_{\text{s}} = 1$ m was placed on the path. In (a), the safe radius r_{s} is displayed as a grey, and the mode change radius r_{m} as a dashed circle. The dashed lines in (b) and (c) indicate the switching times.



(a) Path of the CM (b) Heading angle θ_N and θ_{ref} (c) Joint angle offset, ϕ_0

Figure 6.6: Experimental results of the third scenario: An obstacle with safety radius $r_{\text{s}} = 2$ m was placed 1.5 m to the right side of the path. In (a), the safe radius r_{s} is displayed as a grey, and the mode change radius r_{m} as a dashed circle. The dashed lines in (b) and (c) indicate the switching times.

It can be seen in Figures 6.4(b), 6.5(b) and 6.6(b) that the smoothing function (6.7) smooths the reference signal. The heading controller turned out to track the reference nicely, and the control input ϕ_0 remained within reasonable values, as displayed in Figures 6.4(c), 6.5(c) and 6.6(c). The oscillations in the control input ϕ_0 are a result of the oscillations of the CM, that enter ϕ_0 via the heading reference θ_{ref}. Such oscillations are inherent to snake locomotion according to (6.1) and we do not attempt to suppress them in the control design. In some cases, the guidance system switched back and forth between the two modes when exiting obstacle avoidance mode. An example is provided in Figure 6.5. The heading reference θ_{ref} in Figure 6.5(b) and the control input ϕ_0 in Figure 6.5(c) do, however, not demonstrate chattering behaviour. This is owed to the smoothing function (6.7), which prevents the chattering from entering the controllers.

6.5 Chapter summary

This chapter proposed a set-based guidance strategy for path-following with obstacle avoidance for planar underwater snake robots. The guidance system utilizes a switching strategy originally developed for autonomous surface vessels, which we generalized in this work by introducing a new switching condition It is used in

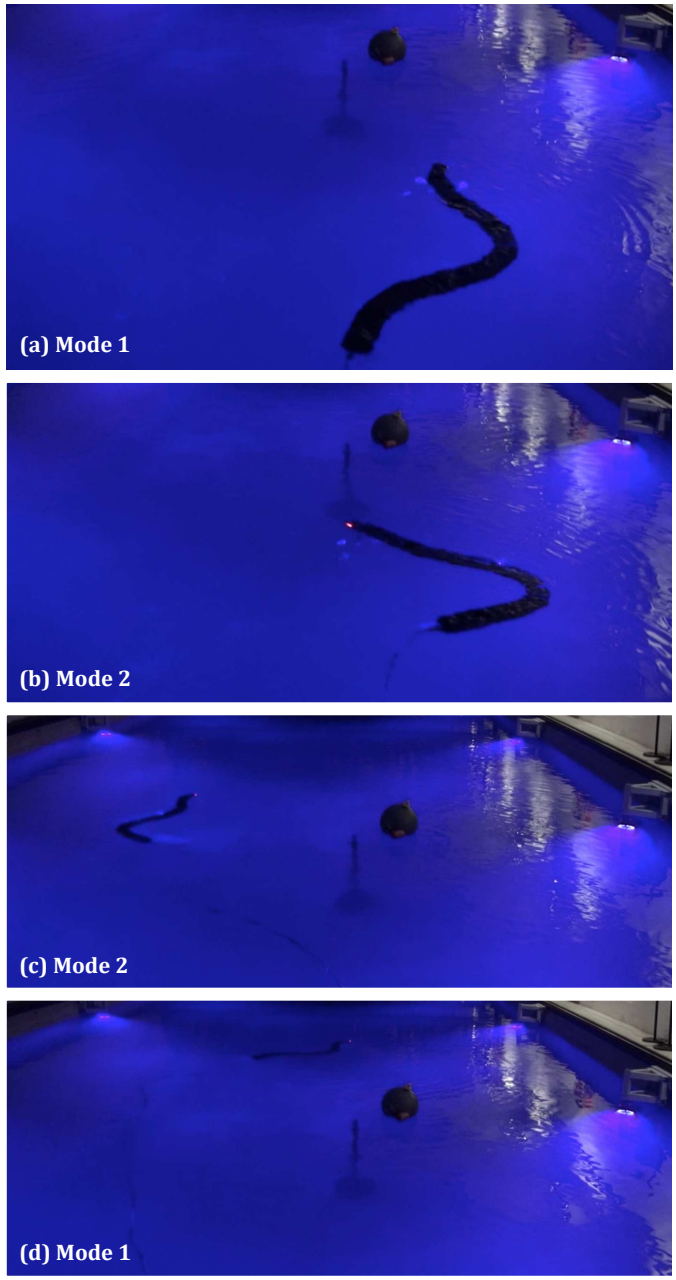


Figure 6.7: The snake robot Mamba during a test run. (a) The robot is started on the path and towards the obstacle. (b) The set-based guidance strategy switches into obstacle avoidance mode and the robot starts turning. (c) The robot circumvents the obstacle at a safe distance. (d) After passing the obstacle, the robot safely converges back to the path.

combination with a guidance scheme for underwater snake robots, which has been extended to enable a choice of direction. The new switching condition makes the set-based switching independent from the underlying dynamic model by considering only the kinematics, and thus generalizes the approach in [73] to a larger class of systems, including terrestrial and underwater snake robots. This approach was seen to be more suitable for snake robot locomotion, where the dynamic model will induce oscillations due to the sinusoidal snake locomotion, and thus increase the risk of chattering behaviour. The guidance system was shown to fulfil the control objectives on a kinematic level. The proposed guidance strategy was combined with a PD-heading controller and experimentally tested with a swimming snake robot for several stationary obstacles of different sizes and locations with respect to the path. These were the first tests of the set-based theory for a floating-base robot, and the results validate the proposed guidance scheme.

Chapter 7

Conclusions and Future Work

This thesis has considered guidance and control approaches for underwater snake robots that move slowly with a planar, biologically inspired sinusoidal gait. The control approaches developed in this thesis are model-based, which enabled formal stability analyses. The guidance systems were designed such that they can reject environmental disturbances or avoid collision with stationary obstacles.

In the first part of the thesis, two models for planar underwater snake robots were presented. Both rely on the assumption that the robot is neutrally buoyant, which allows for a two-dimensional model. The first model was adapted from previous literature and is based on first principles. Based on this model, an analysis of planar sinusoidal locomotion was conducted, revealing several fundamental properties of planar underwater snake robot locomotion. These properties justified a simplification in the first model and were used to motivate some further simplifying assumptions, based on which the second model used in this thesis was derived. This control-oriented model is an extension of previous models for terrestrial and underwater snake robot, now taking into account ocean currents. The two models were compared in an extensive simulation study, which showed that the control-oriented model qualitatively captures the behaviour of the complex model, and even quantitatively approximates the complex model well for certain restrictions on the gait. The chapter was concluded by an analysis of the velocity dynamics during sinusoidal gaits, which revealed relationships between the forward velocity and certain gait parameters based on averaging theory. Interesting future work in the area of modelling and locomotion analysis would be an extension of the models to be valid for robots that move with a more general gait and including more unmodelled effects. Furthermore, a relevant step would be to relax the assumption of neutral buoyancy, include hydrostatic forces into the models, and extend the equations of motion to a fully three dimensional model. This would allow the use of other propulsion methods than planar sinusoidal gaits and pave the way for extending the following motion control approaches to three dimensions.

Based on the two models, guidance and motion control approaches for underwater snake robots were developed in this thesis. The first one was based on the control-oriented model and used a sinusoidal gait in open loop to achieve a positive velocity in the inner control loop. Using cascaded systems theory, a model-based ori-

entation controller was designed as an outer loop, which steered the robot towards and along a straight path while compensating for an unknown ocean current. The reference for the orientation controller was obtained from an integral line-of-sight guidance law. The control system was experimentally validated and formally shown to be uniformly semi-globally exponentially stable. Future work on the control approach would include adding a velocity feedback controller in order to stabilize the forward velocity to a given reference. The second method for guidance and motion control was based on the complex model and is essentially a manoeuvring controller that controls both the heading and the velocity of the robot, such that it follows a generic path. It was an adaptation of a similar approach for terrestrial snake robots, that was formally shown to practically stabilize the control objectives. The proposed feedback control strategy utilized virtual constraints encoding biologically inspired gaits on the snake robot configuration that are parametrized by states of dynamic compensators and used to regulate the heading and forward velocity of the snake robot. In order to adapt the control system for underwater robots, a two-state ocean current observer based on relative velocity measurements was proposed. The performance of the proposed control algorithm for several biologically inspired gaits was demonstrated both in simulations for different path geometries and in experiments. It turned out that the proposed velocity controller resulted in a steady-state offset of the velocity, a drawback that should be addressed in future work, for instance by adding integral action to the controller.

As a complement to the motion control approaches described above, which use relative velocity measurements for feedback, one chapter of this thesis has considered the development of a method for robots that rely on absolute velocity measurements. The control approach is based on enforcing virtual constraints that impose a sinusoidal gait on the body shape using an adaptive backstepping controller. The virtual constraints are parametrized by states of dynamic compensators that control both the velocity and the orientation of the robot. For the velocity control, a second adaptive controller was designed using backstepping techniques and providing a current estimate for the orientation controller. A preliminary stability analysis showed that both the errors of the joint coordinates and the forward velocity are asymptotically stabilized to zero, while stability of the orientation controller relies on some assumptions. Simulation results illustrated the theoretical discussion. Future work on this topic would include formulating conditions for the gait under which we can show persistence of excitation in the velocity controller. This will enable a more systematic design of the orientation controller that allows for a formal stability analysis of the entire closed-loop system and an improvement of the control method by including a guidance law.

Finally, a guidance strategy for path-following with obstacle avoidance for underwater snake robots was developed in this thesis. The guidance strategy was obtained by adapting a guidance system for surface vessels, that consists of two modes, a path-following mode and an obstacle avoidance mode, for snake robot locomotion. The guidance strategy was formally shown to guarantee obstacle avoidance on a kinematic level and validated experimentally. In future work, the method can be combined with an autonomous obstacle detection and extended to consider moving obstacles and environmental disturbances additionally.

Appendix A

Stability Definitions and Theorems

Stability definitions

Consider the non-autonomous system

$$\dot{\mathbf{x}} = \mathbf{f}(t, \mathbf{x}), \tag{A.1}$$

where $\mathbf{f} : [0, \infty) \times D \to \mathbb{R}^n$ is piecewise continuous in t and locally Lipschitz in $\mathbf{x}$ on $[0, \infty) \times D$, and $D \subset \mathbb{R}^n$ is a domain that contains the origin $\mathbf{x} = \mathbf{0}$. Suppose that the origin $\mathbf{x} = \mathbf{0}$ is an equilibrium of the system (A.1). We define the following.

Definition 12 (see Definition 4.4 in [48])**:** The equilibrium $\mathbf{x} = \mathbf{0}$ of (A.1) is

- **stable** if, for each $\epsilon > 0$, there is $\delta = \delta(\epsilon, t_0) > 0$ such that

$$\|\mathbf{x}(t_0)\| < \delta \Rightarrow \|\mathbf{x}(t)\| < \epsilon, \quad \forall t \geq t_0 \geq 0. \tag{A.2}$$

- **uniformly stable (US)** if, for each $\epsilon > 0$, there is $\delta = \delta(\epsilon) > 0$, independent of t_0, such that (A.2) is satisfied.
- **unstable** if it is not stable
- **asymptotically stable (AS)** if it is stable and there is a positive constant $c = c(t_0)$ such that $\mathbf{x}(t) \to \mathbf{0}$ as $t \to \infty$, for all $\|\mathbf{x}(t_0)\| < c$.
- **uniformly asymptotically stable (UAS)** if it is uniformly stable and there is a positive constant c, independent of t_0, such that for all $\|\mathbf{x}(t_0)\| < c, \mathbf{x}(t) \to \mathbf{0}$ as $t \to \infty$, uniformly in t_0; that is, for each $\eta > 0$, there is $T = T(\eta)$ such that

$$\|\mathbf{x}(t)\| < \eta, \quad \forall t \geq t_0 + T(\eta), \quad \forall \|\mathbf{x}(t_0)\| < c. \tag{A.3}$$

- **uniformly globally asymptotically stable (UGAS)** if it is uniformly stable, $\delta(\epsilon)$ can be chosen to satisfy $\lim_{\epsilon \to \infty} \delta(\epsilon) = \infty$, and, for each pair of positive numbers η and c, there is $T = T(\eta, c) > 0$ such that

$$\|\mathbf{x}(t)\| < \eta, \quad \forall t \geq t_0 + T(\eta, c), \quad \forall \|\mathbf{x}(t_0)\| < c. \tag{A.4}$$

Definition 13 (see Definition 2.7 in [67])**:** The origin of the system (A.1) is said to be **uniformly locally exponentially stable (ULES)** if there exist constants $\gamma_1, \gamma_2, r > 0$ such that for all $(t_0, \mathbf{x}(t_0)) \in \mathbb{R}_{\geq 0} \times \mathcal{B}_r$

$$\|\mathbf{x}(t, t_0, \mathbf{x}(t_0))\| \leq \gamma_1 \|\mathbf{x}(t_0)\| e^{-\gamma_2 (t - t_0)} \quad \forall t \geq t_0. \tag{A.5}$$

If for each $r > 0$ there exist γ_1, γ_2 such that (A.5) holds for all $(t_0, \mathbf{x}(t_0)) \in \mathbb{R}_{\geq 0} \times \mathcal{B}_r$ then, the system is said to be **uniformly semi-globally exponentially stable (USGES)**. Finally, the origin of system (A.1) is said to be **uniformly globally exponentially stable (UGES)** if there exist $\gamma_1, \gamma_2 > 0$ such that (A.5) holds for all $(t_0, \mathbf{x}(t_0)) \in \mathbb{R}_{\geq 0} \times \mathbb{R}^n$.

Cascaded systems

One system that is studied in this thesis is a cascaded nonlinear time-varying system of the structure that was defined in [85]:

$$\begin{aligned} \Sigma_1 : \ \dot{\mathbf{x}}_1 &= \mathbf{f}_1(t, \mathbf{x}_1) + \mathbf{G}(t, \mathbf{x}) \mathbf{x}_2, \\ \Sigma_2 : \ \dot{\mathbf{x}}_2 &= \mathbf{f}_2(t, \mathbf{x}_2), \end{aligned} \tag{A.6}$$

where $\mathbf{x}_1 \in \mathbb{R}^l$, $\mathbf{x}_2 \in \mathbb{R}^m$, and $\mathbf{x} = [\mathbf{x}_1^T, \mathbf{x}_2^T]^T$. We assume that the functions $\mathbf{f}_1(\cdot)$, $\mathbf{f}_2(\cdot)$, and $\mathbf{G}(\cdot)$ are continuous in their arguments, locally Lipschitz in $\mathbf{x}$, uniformly in t, and $\mathbf{f}_1(\cdot)$ is continuously differentiable in both arguments. For the stability analysis of such systems, the following theory can be applied.

Theorem A.1 (Theorem 2 in [85])**.** *If the following assumptions are satisfied, the cascaded system (A.6) is UGAS.*

1. $\dot{\mathbf{x}}_1 = \mathbf{f}_1(t, \mathbf{x}_1)$ *is UGAS with a radially unbounded Lyapunov function satisfying*

$$\begin{aligned} \left\| \frac{\partial V}{\partial \mathbf{x}_1} \right\| \|\mathbf{x}_1\| &\leq c_1 V(t, \mathbf{x}_1) \quad && \forall \|\mathbf{x}_1\| \geq \eta, \\ \left\| \frac{\partial V}{\partial \mathbf{x}_1} \right\| &\leq c_2, && \forall \|\mathbf{x}_1\| \leq \eta, \end{aligned} \tag{A.7}$$

   with $c_1, c_2, \eta > 0$.

2. $\|\mathbf{G}(t, \mathbf{x}))\| \leq \theta_1(\|\mathbf{x}_2\|) + \theta_2(\|\mathbf{x}_2\|)\|\mathbf{x}_1\|$, *where* $\theta_1, \theta_2 : \mathbb{R}_{\geq 0} \mapsto \mathbb{R}_{\geq 0}$ *are continuous.*

3. Σ_2 *is UGAS and*

$$\int_{t_0}^{\infty} \|\mathbf{x}_2(t)\| \mathrm{d}t \leq \phi(\|\mathbf{x}_2(t_0)\|), \tag{A.8}$$

   $\phi(\cdot)$ *is a class* $\mathcal{K}$ *function.*

In particular, Ass. 1) always holds for a quadratic Lyapunov function V [85], and Ass. 3) is always fulfilled if Σ_2 is κ-exponentially stable [65], i.e. UGAS and ULES [101].

Proposition A.2 (Proposition 2.3 in [67])**.** *If in addition to the assumptions in Theorem A.1 the systems* Σ_2 *and* $\dot{\mathbf{x}}_1 = \mathbf{f}_1(t, \mathbf{x}_1)$ *are USGES then the cascaded system (A.6) is USGES and UGAS. If the subsystems are UGES the cascade is UGES.*

Comparison Lemma

Lemma A.3 (Lemma 3.4 in [48]). *Consider the scalar differential equation*

$$\dot{u} = f(t,u), \quad u(t_0) = u_0 \tag{A.9}$$

where $f(t,u)$ is continuous in t and locally Lipschitz in u, for all $t \geq 0$ and all $u \in J \subset \mathbb{R}$. Let $[t_0, T)$ (T could be infinity) be the maximal interval of existence of the solution $u(t)$, and suppose $u(t) \in J$ for all $t \in [t_0, T)$. Let $v(t)$ be a continuous function whose upper right-hand derivative $D^+v(t)$ satisfies the differential inequality

$$D^+v(t) \leq f(t, v(t)), \quad v(t_0) \leq u_0 \tag{A.10}$$

with $v(t) \in J$ for all $t \in [t_0, T)$. Then, $v(t) \leq u(t)$ for all $t \in [t_0, T)$.

Stability of nested closed sets

Consider a control system Σ of the form

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \tag{A.11}$$

with the state space $\mathcal{X} \subset \mathbb{R}^n$. Let $\mathbf{f}$ be locally Lipschitz on $\mathcal{X}$.

Given an interval I of the real line and a set $S \in \mathcal{X}$, $\phi(I, S)$ is the set $\phi(I, S) = \{\phi(t, \mathbf{x}(t_0)) : t \in I, \mathbf{x}(t_0) \in \mathcal{X}\}$. Solving a control problem where the control objectives can be formulated hierarchically, can be approached as the task of simultaneous asymptotic stabilization of a chain of nested closed sets $\Gamma_1 \subset \Gamma_2 \subset \ldots \Gamma_l$ [20]. Let $\mathbf{u}$ be a locally Lipschitz feedback that makes the sets $\Gamma_1 \subset \Gamma_2 \subset \ldots \subset \Gamma_l$ positively invariant for the closed-loop system.

Let Γ be a closed positively invariant set for Σ. With the point-to-set distance $\|\phi(t, \mathbf{x}(t_0))\|_\Gamma$ and the open set $\mathcal{B}_\epsilon(\Gamma) = \{\mathbf{y} \in \mathcal{X} : \|\mathbf{y}\|_\Gamma < \epsilon\}$, we obtain the following definitions:

Definition 14 (**Set stability and attractivity**, see Definition 1 in [20]): The set Γ is

- **stable** for Σ if for all $\epsilon > 0$ there exists a neighbourhood $\mathcal{N}(\Gamma)$ such that $\phi(\mathbb{R}^+, \mathcal{N}(\Gamma)) \subset \mathcal{B}_\epsilon(\Gamma)$.
- an **attractor** for Σ if there exists a neighbourhood $\mathcal{N}(\Gamma)$ such that $\lim_{t\to\infty} \|\phi(t, \mathbf{x}(t_0))\|_\Gamma = 0$ for all $\mathbf{x}(t_0) \in \mathcal{N}(\Gamma)$.
- a **global attractor** for Σ if it is an attractor with $\mathcal{N} = \mathcal{X}$.
- **(globally) asymptotically stable** for Σ if it is stable and (globally) attractive for Σ.

Definition 15 (**Relative set stability and attractivity**, see Definition 4 in [20]): Let Γ_1 and Γ_2, $\Gamma_1 \subset \Gamma_2 \subset \mathcal{X}$, be closed positively invariant sets. We say that Γ_1 is **stable relative to** Γ_2 for Σ if, for any $\epsilon > 0$, there exists a neighbourhood $\mathcal{N}(\Gamma_1)$ such that $\phi(\mathbb{R}^+, \mathcal{N}(\Gamma_1) \cap \Gamma_2) \subset \mathcal{B}_\epsilon(\Gamma_1)$. Similarly, one modifies all other notions above by restricting initial conditions to lie in Γ_2.

Appendix B

The Snake Robot Mamba

The snake robot Mamba was developed at NTNU to support the ongoing research on snake robot locomotion, including underwater locomotion. The robot has served as an experimental platform for the work that was carried out for this thesis. This appendix gives a brief description of the robot, which can be seen in Figure B.1. More details can be found in [66].

Mamba is a modular snake robot that can be operated both on land and in water. It consists of nine horizontal joints and nine vertical joints and is connected to a power source and communication unit with a slender, positively buoyant cable. During the tests for this thesis, only planar sinusoidal gaits and two-dimensional control schemes were considered. Therefore, the vertical joints were stiffened by setting their reference to zero during the experiments. The single joints of Mamba are waterproof down to 5 m, and equipped with a servo motor, various sensors, and a micro-controller card that communicates with the adjacent joints over a CAN bus. The servo motors have internal proportional controllers which drive the servo output shaft to the reference angle that is requested by the micro-controller. This internal joint position control replaced the theoretical torque controllers in the

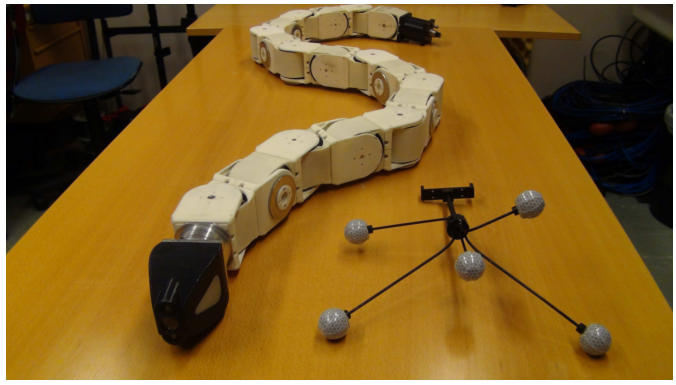


Figure B.1: The snake robot Mamba and the marker attachment for the motion capture system.

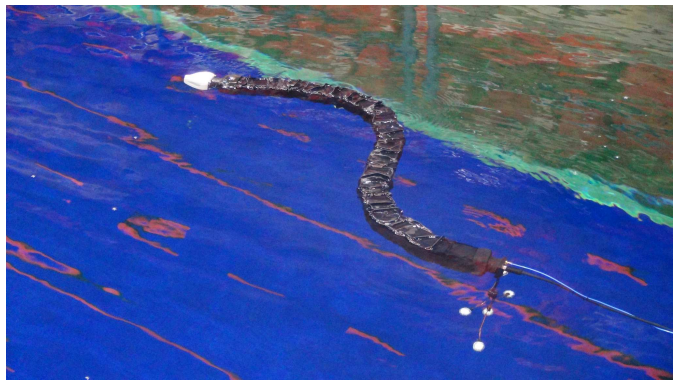


Figure B.2: The snake robot Mamba. In the basin, the reflective markers are attached to the robot, which is additionally waterproofed with a synthetic skin.

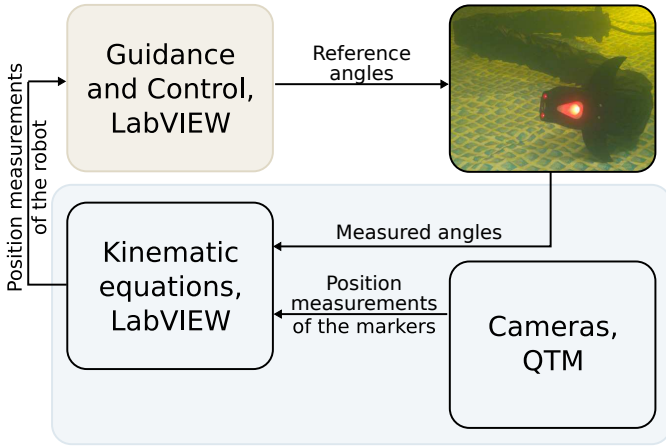


Figure B.3: The experimental set-up: the position of the robot is obtained from an external positioning measurement system.

practical implementation of the control systems proposed in this thesis. Despite the waterproof design of the single joints, the robot was additionally equipped with a synthetic, waterproof skin for the swimming tests that are part of this thesis. The deployed robot in the basin is shown in Figure B.2. More details on the skin can be found in [45]. An additional advantage of the skin is that the amount of air inside of it can be varied with a pneumatic valve, which influences the buoyancy of the robot. During the tests of the two-dimensional control schemes considered in this thesis, enough air was left inside the skin to provide a slightly positive buoyancy in order to keep the robot close to the surface and thus not require depth control. In addition, washers where attached to the bottom of each horizontal joint. Thereby, the robot was slightly heavier on the bottom and thus self-stabilizing in roll.

The robot is controlled from a laptop that runs LabVIEW 2013. All control

schemes that were tested in this thesis were therefore implemented in LabVIEW 2013. In order to obtain the required position and orientation feedback for the controllers, reflective markers were attached to the robot and the motion capture system Qualisys was used to track them. The marker attachment can be seen in Figure B.1. The angle and position measurements of the marker attachment were obtained from a second laptop, on which both Qualisys Track Manager (QTM) and LabVIEW 2013 are installed. The data was then 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 of the single robot links and the CM position of the robot were calculated from the QTM data and the joint angles from the internal sensors of the robot by using the kinematic equations of the robot. A schematic sketch of the experimental set-up is presented in Figure B.3.

References

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

[2] J. Aguilar, T. Zhang, F. Qian, M. Kingsbury, B. McInroe, N. Mazouchova, C. Li, R. Maladen, C. Gong, M. Travers, R. L. Hatton, H. Choset, P. B. Umbanhowar, and D. I. Goldman. A review on locomotion robophysics: the study of movement at the intersection of robotics, soft matter and dynamical systems. *Reports on Progress in Physics*, 79:110001, 2016.

[3] O. Akanyeti, J. C. Brown, L. D. Chambers, H. el Daou, M.-C. Fiazza, P. Fiorini, J. Ježov, D. S. Jung, M. Kruusmaa, M. Listak, A. Liszewski, J. L. Maud, W. M. Megill, L. Rossi, A. Qualtieri, F. Rizzi, T. Salumäe, G. Toming, R. Venturelli, F. Visentin, and M. D. Vittorio. FILOSE for Svenning: A Flow Sensing Bioinspired Robot. *IEEE Robotics & Automation Magazine*, 21(3):51–62, 2014.

[4] B. M. Albaker and N. A. Rahim. A survey of collision avoidance approaches for unmanned aerial vehicles. In *Int. Conf. Technical Postgraduates*, Kuala Lumpur, Malaysia, Dec. 2009.

[5] G. Antonelli. *Underwater Robots – Motion and Force Control of Vehicle-Manipulator Systems*. Springer Tracts in Advanced Robotics. Springer, 2nd edition, 2006.

[6] V. I. Arnold. *Mathematical Methods of Classical Mechanics*, volume 60 of *Graduate Texts in Mathematics*. Springer, 1989.

[7] D. J. W. Belleter. *Control of Underactuated Marine Vehicles in the Presence of Environmetal Disturbances*. PhD thesis, Norwegian University of Science and Technology (NTNU), 2016.

[8] D. J. W. Belleter and K. Y. Pettersen. Path following for Formations of Underactuated Marine Vessels under Influence of Constant Ocean Currents. In *Proc. 53rd IEEE Conf. Decision and Control*, Los Angeles, California USA, Dec. 2014.

[9] D. J. W. Belleter, C. Paliotta, M. Maggiore, and K. Y. Pettersen. Path Following for Underactuated Marine Vessels. *IFAC-PapersOnLine*, 49(18): 588–593, 2016. 10th IFAC Symposium on Nonlinear Control Systems.

[10] J. Blair and T. Iwasaki. Optimal Gaits for Mechanical Rectifier Systems. *IEEE Transactions on Automatic Control*, 56(1):59–71, 2011.

[11] E. Børhaug, A. Pavlov, and K. Y. Pettersen. Integral LOS Control for Path Following of Underactuated Marine Surface Vessels in the Presence of Constant Ocean Currents. In *Proc. 47th IEEE Conf. Decision and Control*, Cancun, Mexico, Dec. 2008.

[12] W. Caharija, M. Candeloro, K. Y. Pettersen, and A. J. Sørensen. Relative Velocity Control and Integral LOS for Path Following of Underactuated Surface Vessels. *IFAC Proceedings Volumes*, 45(27):380–385, 2012. 9th IFAC Conference on Manoeuvring and Control of Marine Craft.

[13] 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.

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

[15] 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, 2009.

[16] R. Christ and R. Wernli. *The ROV Manual: A User Guide for Remotely Operated Vehicles*. Butterworth-Heinemann, Oxford, 2014.

[17] J. E. Colgate and K. M. Lynch. Mechanics and Control of Swimming: A Review. *IEEE Journal of Oceanic Engineering*, 29(3):660–673, 2004.

[18] 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.

[19] K. D. Do and J. Pan. Global tracking control of underactuated ships with off-diagonal terms. In *Proc. 42nd IEEE Conf. Decision and Control*, Maui, Hawaii USA, Dec. 2003.

[20] M. I. El-Hawwary and M. Maggiore. Reduction Theorems for Stability of Closed Sets with Application to Backstepping Control Design. *Automatica*, 49(1):214–222, 2013.

[21] F. Enner, D. Rollinson, and H. Choset. Simplified Motion Modeling for Snake Robots. In *Proc. 2012 IEEE Int. Conf. Robotics and Automation*, Saint Paul, Minnesota USA, May 2012.

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

[23] T. I. Fossen and K. Y. Pettersen. On uniform semiglobal exponential stability (USGES) of proportional line-of-sight guidance laws. *Automatica*, 50(11): 2912–2917, 2014.

[24] T. I. Fossen, A. Loría, and A. Teel. A theorem for UGAS and ULES of (passive) nonautonomous systems: robust control of mechanical systems and ships. *International Journal of Robust and Nonlinear Control*, 11:95–108, 2001.

[25] E. Fredriksen and K. Y. Pettersen. Global κ-exponential way-point maneuvering of ships: Theory and experiments. *Automatica*, 42(4):677–687, 2006.

[26] G. B. Gillis. Environmental effects on undulatory locomotion in the American eel *Anguilla rostrata*: kinematics in water and on land. *The Journal of Experimental Biology*, 201:949–961, 1998.

[27] J. Gray. Studies in Animal Locomotion I. The Movement of Fish with Special Reference to the Eel. *Journal of Experimental Biology*, 10(1):88–104, 1933.

[28] J. Gray. The mechanism of locomotion in snakes. *Journal of Experimental Biology*, 23(2):101–120, 1946.

[29] 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(8):1955–1988, 2014.

[30] J. Guo. A waypoint-tracking controller for a biomimetic autonomous underwater vehicle. *Ocean Engineering*, 33(17-18):2369–2380, 2006.

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

[32] Y. Hitaka, T. Yoshitake, and M. Yokomichi. Obstacle avoidance of snake robot by switching control constraint. *Artificial Life and Robotics*, 17(2): 180–185, 2012.

[33] J. K. Hopkins, B. W. Spranklin, and S. K. Gupta. A survey of snake-inspired robot designs. *Bioinspiration & Biomimetics*, 4:021001, 2009.

[34] A. J. Ijspeert. Central pattern generators for locomotion control in animals and robots: A review. *Neural Networks*, 21(4):642–653, 2008.

[35] A. J. Ijspeert. Biorobotics: Using robots to emulate and investigate agile locomotion. *Science*, 346(6206):196–203, 2014.

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

[37] F. Kamil, S. H. Tang, W. Khaksar, N. Zulkifli, and S. A. Ahmad. A Review on Motion Planning and Obstacle Avoidance Approaches in Dynamic Environments. *Advances in Robotics & Automation*, 4(2):1000134, 2015.

[38] E. Kelasidi. *Modeling, Control and Energy Efficiency of Underwater Snake Robots*. PhD thesis, Norwegian University of Science and Technology (NTNU), 2015.

[39] E. Kelasidi, K. Y. Pettersen, and J. T. Gravdahl. A waypoint guidance strategy for underwater snake robots. In *Proc. 22nd Mediterranean Conf. on Control and Automation*, Palermo, Italy, Jun. 2014.

[40] E. Kelasidi, K. Y. Pettersen, and J. T. Gravdahl. A Control-Oriented Model of Underwater Snake Robots. In *Proc. 2014 IEEE Int. Conf. Robotics and Biomimetics*, Bali, Indonesia, Dec. 2014.

[41] E. Kelasidi, K. Y. Pettersen, and J. T. Gravdahl. Stability Analysis of Underwater Snake Robot Locomotion Based on Averaging Theory. In *Proc. 2014 IEEE Int. Conf. Robotics and Biomimetics*, Bali, Indonesia, Dec. 2014.

[42] E. Kelasidi, K. Y. Pettersen, J. T. Gravdahl, and P. Liljebäck. Modeling of underwater snake robots. In *Proc. IEEE Int. Conf. Robotics and Automation*, Hong Kong, China, May, Jun. 2014.

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

[44] 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), 2015.

[45] 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.

[46] E. Kelasidi, P. Liljebäck, K. Y. Pettersen, and J. T. Gravdahl. Integral Line-of-Sight Guidance for Path Following Control of Underwater Snake Robots: Theory and Experiments. *IEEE Transactions on Robotics*, 33(3):610–628, 2017.

[47] E. Kelasidi, K. Y. Pettersen, J. T. Gravdahl, S. Strømsøyen, and A. J. Sørensen. Modeling and Propulsion Methods of Underwater Snake Robots. In *Proc. IEEE Conf. Control Technology and Applications*, Kohala Coast, Hawaii USA, Aug. 2017.

[48] H. Khalil. *Nonlinear Systems*. Prentice Hall, 3rd edition, 2002.

[49] A. M. Kohl, E. Kelasidi, K. Y. Pettersen, and J. T. Gravdahl. A control-oriented model of underwater snake robots exposed to currents. In *Proc. 2015 IEEE Conf. Control Applications*, Sydney, Australia, Sep. 2015.

[50] A. M. Kohl, K. Y. Pettersen, E. Kelasidi, and J. T. Gravdahl. Analysis of underwater snake robot locomotion based on a control-oriented model. In *Proc. 2015 IEEE Int. Conf. Robotics and Biomimetics*, Zhuhai, China, Dec. 2015.

[51] A. M. Kohl, E. Kelasidi, A. Mohammadi, M. Maggiore, and K. Pettersen. Planar maneuvering control of underwater snake robots using virtual holonomic constraints. *Bioinspiration & Biomimetics*, 11(6):065005, 2016.

[52] 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.

[53] A. M. Kohl, E. Kelasidi, K. Y. Pettersen, and J. T. Gravdahl. *Sensing and Control for Autonomous Vehicles – Applications to Land, Water and Air Vehicles*, volume 474 of *Lecture Notes in Control and Information Sciences*, chapter Model-Based LOS Path-Following Control of Planar Underwater Snake Robots, pages 343–363. Springer International Publishing, 2017.

[54] A. M. Kohl, S. Moe, E. Kelasidi, K. Y. Pettersen, and J. T. Gravdahl. Set-based path following and obstacle avoidance for underwater snake robots. In *IEEE Int. Conf. Robotics and Biomimetics*, 2017. Accepted.

[55] A. M. Kohl, K. Y. Pettersen, and J. T. Gravdahl. Velocity and orientation control of underwater snake robots using absolute velocity feedback. In *Proc. IEEE Conf. Control Technology and Applications*, Kohala Coast, Hawaii USA, Aug. 2017.

[56] M. Krstić, I. Kanellakopoulos, and P. V. Kokotović. *Nonlinear and Adaptive Control Design*. Wiley, 1995.

[57] L. Labastida-Valdés, L. A. Torres-Méndez, and S. A. Hutchinson. Using the Motion Perceptibility Measure to Classify Points of Interest for Visual-based AUV Guidance in a Reef Ecosystem. In *Proc. MTS/IEEE OCEANS 2015*, Washington, DC, USA, Oct. 2015.

[58] L. Lapierre and B. Jouvencel. Path Following Control for an Eel-like Robot. In *Proc. Oceans 2005 - Europe*, Brest, France, Jun. 2005.

[59] M. J. Lighthill. Large-Amplitude Elongated-Body Theory of Fish Locomotion. *Proceedings of the Royal Society of London*, 179(1055):125–138, 1971.

[60] P. Liljebäck, Ø. Stavdahl, and A. Beitnes. SnakeFighter - Development of a Water Hydraulic Fire Fighting Snake Robot. In *Proc. 9th Int. Conf. Control, Automation, Robotics and Vision*, Singapore, Dec. 2006.

[61] P. Liljebäck, K. Y. Pettersen, Ø. Stavdahl, and J. T. Gravdahl. Stability analysis of snake robot locomotion based on averaging theory. In *Proc. 49th IEEE Conf. Decision and Control*, Atlanta, Georgia USA, Dec. 2010.

[62] P. Liljebäck, K. Y. Pettersen, Ø. Stavdahl, and J. T. Gravdahl. A simplified model of planar snake robot locomotion. In *Proc. 2010 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Taipei, Taiwan, Oct. 2010.

[63] P. Liljebäck, I. U. Haugstuen, and K. Y. Pettersen. Path Following Control of Planar Snake Robots Using a Cascaded Approach. *IEEE Transactions on Control Systems Technology*, 20(1):111–126, 2012.

[64] 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(1):29–40, 2012.

[65] P. Liljebäck, K. Y. Pettersen, Ø. Stavdahl, and J. T. Gravdahl. *Snake Robots: Modelling, Mechatronics, and Control*. Advances in Industrial Control. Springer London, 2013.

[66] P. Liljebäck, Ø. Stavdahl, K. Pettersen, and J. Gravdahl. Mamba -A Waterproof Snake Robot with Tactile Sensing. In *Proc. 2014 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Chicago, Illinois USA, Sep. 2014.

[67] A. Loría and E. Panteley. Cascaded Nonlinear Time-Varying Systems: Analysis and Design. In F. Lamnabhi-Lagarrigue, A. Loría, and E. Panteley, editors, *Advanced Topics in Control Systems Theory: Lecture Notes from FAP 2004*, chapter 2, pages 23–64. Springer London Ltd., 2005.

[68] S. Ma. Analysis of creeping locomotion of a snake-like robot. *Advanced Robotics*, 15(2):205–224, 2001.

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

[70] K. A. McIsaac and J. P. Ostrowski. Motion Planning for Anguilliform Locomotion. *IEEE Transactions on Robotics and Automation*, 19(4):637–652, 2003.

[71] K. A. Mclsaac and J. P. Ostrowski. A Geometric Approach to Anguilliform Locomotion: Modelling of an Underwater Eel Robot. In *Proc. 1999 IEEE Int. Conf. Robotics and Automation*, Detroit, Michigan USA, May 1999.

[72] M. S. Menon, V. C. Ravi, and A. Ghosal. Trajectory Planning and Obstacle Avoidance for Hyper-Redundant Serial Robots. *Journal of Mechanisms and Robotics*, 9(1):041010, 2017.

[73] S. Moe and K. Y. Pettersen. Set-Based Line-of-Sight (LOS) Path Following with Collision Avoidance for Underactuated Unmanned Surface Vessel. In *Proc. 24th Mediterranean Conf. Control and Automation*, Athens, Greece, Jun. 2016.

[74] 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, Oregon USA, Jun. 2014.

[75] S. Moe, G. Antonelli, A. R. Teel, K. Y. Pettersen, and J. Schrimpf. Set-Based Tasks within the Singularity-Robust Multiple Task-Priority Inverse Kinematics Framework: General Formulation, Stability Analysis, and Experimental Results. *Frontiers in Robotics and AI*, 3:16, 2016.

[76] S. Moe, K. Y. Pettersen, T. I. Fossen, and J. T. Gravdahl. Line-of-Sight Curved Path Following for Underactuated USVs and AUVs in the Horizontal Plane under the influence of Ocean Currents. In *Proc. 24th Mediterranean Conference Control and Automation*, Athens, Greece, June 2016.

[77] A. Mohammadi, E. Rezapour, M. Maggiore, and K. Y. Pettersen. Direction Following Control of Planar Snake Robots Using Virtual Holonomic Constraints. In *Proc. 53rd IEEE Conf. Decision and Control*, Los Angeles, California USA, Dec. 2014.

[78] A. Mohammadi, E. Rezapour, M. Maggiore, and K. Y. Pettersen. Maneuvering Control of Planar Snake Robots Using Virtual Holonomic Constraints. *IEEE Transactions on Control Systems Technology*, 24(3):884 – 899, 2016.

[79] K. A. Morgansen, P. A. Vela, and J. W. Burdick. Trajectory Stabilization for a Planar Carangiform Robot Fish. In *Proc. 2002 IEEE Int. Conf. Robotics and Automation*, Washington, DC, USA, May 2002.

[80] K. A. Morgansen, B. I. Triplett, and D. J. Klein. Geometric Methods for Modeling and Control of Free-Swimming Fin-Actuated Underwater Vehicles. *IEEE Transactions on Robotics*, 23(6):1184–1199, 2007.

[81] S. Y. Na, D. Shin, J. Y. Kim, S.-J. Baek, and S. H. Min. *Obstacle Recognition and Collision Avoidance of a Fish Robot Based on Fuzzy Neural Networks*, pages 337–344. Springer Berlin Heidelberg, 2007.

[82] C. D. Onal and D. Rus. Autonomous undulatory serpentine locomotion utilizing body dynamics of a fluidic sort robot. *Bioinspiration & Biomimetics*, 8(2):026003, 2013.

[83] C. Paliotta and K. Y. Pettersen. Geometric Path Following with Ocean Current Estimation for ASVs and AUVs. In *Proc. 2016 American Control Conference*, Boston, Massachusetts USA, Jul. 2016.

[84] E. Paljug, T. Ohm, and S. Hayati. The JPL Serpentine Robot: a 12 DOF System for Inspection. In *Proc. IEEE Int. Conf. Robotics and Automation*, Nagoya, Japan, May 1995.

[85] E. Panteley and A. Loría. On global uniform asymptotic stability of nonlinear time-varying systems in cascade. *Systems & Control Letters*, 33(2):131–138, 1998.

[86] M. Porez, V. Lebastard, A. J. Ijspeert, and F. Boyer. Multi-physics model of an electric fish-like robot : numerical aspects and application to obstacle avoidance. In *Proc. 2011 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, San Francisco, California USA, Sep. 2011.

[87] M. Porez, F. Boyer, and A. J. Ijspeert. Improved Lighthill fish swimming model for bio-inspired robots: Modeling, computational aspects and experimental comparisons. *The International Journal of Robotics Research*, 33(10): 1322–1341, 2014.

[88] A. Raj and A. Thakur. Fish-inspired robots: design, sensing, actuation, and autonomy – a review of research. *Bioinspiration & Biomimetics*, 11 (3):031001, 2016.

[89] F. Reyes, W. Tang, and S. Ma. Using a Planar Snake Robot as a Robotic Arm Taking into Account the Lack of a Fixed Base: Feasible Region. In *Proc. 2015 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Hamburg, Germany, Sep. 2015.

[90] E. Rezapour, A. Hofmann, and K. Y. Pettersen. Maneuvering control of planar snake robots based on a simplified model. In *Proc. IEEE Int. Conf. Robotics and Biomimetics*, Bali, Indonesia, Dec. 2014.

[91] E. Rezapour, A. Hofmann, 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.

[92] 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), 2014.

[93] D. Rollinson and H. Choset. Pipe Network Locomotion with a Snake Robot. *Journal of Field Robotics*, 33(3):322–336, 2014.

[94] D. Rollinson, A. Buchan, and H. Choset. Virtual Chassis for Snake Robots: Definition and Applications. *Advanced Robotics*, 26(17):2043–2064, 2012.

[95] M. Saito, M. Fukaya, and T. Iwasaki. Serpentine Locomotion with Robotic Snakes. *IEEE Control Systems Magazine*, 22(1):64–81, 2002.

[96] D. Scaradozzi, G. Palmieri, D. Costa, and A. Pinelli. BCF swimming locomotion for autonomous underwater robots: a review and a novel solution to improve control and efficiency. *Ocean Engineering*, 130:437–453, 2017.

[97] M. Sfakiotakis and D. P. Tsakiris. Biomimetic Centering for Undulatory Robots. *The International Journal of Robotics Research*, 26(11-12):1267–1282, 2007.

[98] M. Sfakiotakis, D. M. Lane, and J. B. C. Davies. Review of Fish Swimming Modes for Aquatic Locomotion. *IEEE Journal of Oceanic Engineering*, 24 (2):237–252, 1999.

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

[100] R. Skjetne, T. I. Fossen, and P. V. Kokotović. Robust output maneuvering for a class of nonlinear systems. *Automatica*, 40(3):373–383, 2004.

[101] O. J. Sørdalen and O. Egeland. Exponential Stabilization of Nonholonomic Chained Systems. *IEEE Transactions on Automatic Control*, 40(1):35–49, 1995.

[102] T. Statheros, G. Howells, and K. McDonald-Maier. Autonomous Ship Collision Avoidance Navigation Concepts, Technologies and Techniques. *The Journal of Navigation*, 61(1):129–142, 2008.

[103] J. Sverdrup-Thygeson, E. Kelasidi, K. Y. Pettersen, and J. T. Gravdahl. The Underwater Swimming Manipulator - A Bio-Inspired AUV. In *Proc. IEEE / OES Autonomous Underwater Vehicles*, Tokyo, Japan, Nov. 2016.

[104] 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.

[105] M. Tanaka and F. Matsuno. Control of snake robots with switching constraints: trajectory tracking with moving obstacle. *Advanced Robotics*, 28(6): 415–429, 2014.

[106] M. Tanaka and K. Tanaka. Control of a Snake Robot for Ascending and Descending Steps. *IEEE Transactions on Robotics*, 31(2):511–520, 2015.

[107] A. Teel and L. Praly. Tools for Semiglobal Stabilization by Partial State and Output Feedback. *SIAM Journal on Control and Optimization*, 33(5): 1443–1488, 1995.

[108] A. A. Transeth, R. I. Leine, C. Glocker, K. Y. Pettersen, and P. Liljebäck. Snake Robot Obstacle-Aided Locomotion: Modeling, Simulations, and Experiments. *IEEE Transactions on Robotics*, 24(1):88–104, 2008.

[109] A. A. Transeth, K. Y. Pettersen, and P. Liljebäck. A survey on snake robot modeling and locomotion. *Robotica*, 27(7):999–1015, 2009.

[110] D. Trivedi, C. D. Rahn, W. M. Kier, and I. D. Walker. Soft robotics: Biological inspiration, state of the art, and future research. *Applied Bionics and Biomechanics*, 5(3):99–117, 2008.

[111] S. Tully, G. Kantor, M. A. Zenati, and H. Choset. Shape Estimation for Image-Guided Surgery with a Highly Articulated Snake Robot. In *Proc. 2011 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, San Francisco, California USA, Sep. 2011.

[112] J. Wang. *Robotic Fish: Development, Modeling, and Application to Mobile Sensing*. PhD thesis, Michigan State University, 2014.

[113] J. Wang, S. Chen, and X. Tan. Control-oriented Averaging of Tail-actuated Robotic Fish Dynamics. In *Proc. 2013 American Control Conference*, Washington, DC, USA, Jun. 2013.

[114] R. J. Webster and B. A. Jones. Design and Kinematic Modeling of Constant Curvature Continuum Robots: A Review. *The International Journal of Robotics Research*, 29(13):1661–1683, 2010.

[115] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. H. Choi, and B. Morris. *Feedback Control of Dynamic Bipedal Robot Locomotion*, volume 28 of *Automation and Control Engineering*. CRC Press, 2007.

[116] A. J. Wiens and M. Nahon. Optimally efficient swimming in hyper-redundant mechanisms: control, design, and energy recovery. *Bioinspiration & Biomimetics*, 7(4):046016, 2012.

[117] M. S. Wiig, K. Y. Pettersen, and T. R. Krogstad. Uniform Semiglobal Exponential Stability of Integral Line-of-Sight Guidance Laws. *IFAC-PapersOnLine*, 48(16):61–68, 2015. 10th IFAC Conference on Manoeuvring and Control of Marine Craft.

[118] E. Wolbrecht, J. Osborn, S. Qualls, R. Ross, J. Canning, M. Anderson, and D. Edwards. Estimating and Compensating for Water Currents: Field Testing. In *Proc. MTS/IEEE OCEANS 2016*, Monterey, California USA, Sep. 2016.

[119] A. Wolf, H. B. Brown, R. Casciola, A. Costa, M. Schwerin, E. Shamas, and H. Choset. A Mobile Hyper Redundant Mechanism for Search and Rescue Tasks. In *Proc. 2003 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Las Vegas, Nevada USA, Oct. 2003.

[120] A. Wolf, H. M. Choset, H. B. Brown, and R. Casciola. Design and control of a mobile hyper-redundant urban search and rescue robot. *Advanced Robotics*, 19(8):221–248, 2005.

[121] C. Wright, A. Buchan, B. Brown, J. Geist, M. Schwerin, D. Rollinson, M. Tesch, and H. Choset. Design and Architecture of the Unified Modular Snake Robot. In *Proc. 2012 IEEE Int. Conf. Robotics and Automation*, Saint Paul, Minnesota USA, May 2012.

[122] X. Wu and S. Ma. CPG-based control of serpentine locomotion of a snake-like robot. *Mechatronics*, 20(2):326–334, 2010.

[123] H. Yamada, S. Chigisaki, M. Mori, K. Takita, K. Ogami, and S. Hirose. Development of Amphibious Snake-like Robot ACM-R5. In *Proc. 36th Int. Symp. Robotics*, Tokyo, Japan, 2005.

[124] A. Zhang, S. Ma, B. Li, M. Wang, X. Guo, and Y. Wang. Adaptive controller design for underwater snake robot with unmatched uncertainties. *Science China: Information Sciences*, 59(5):052205, 2016.