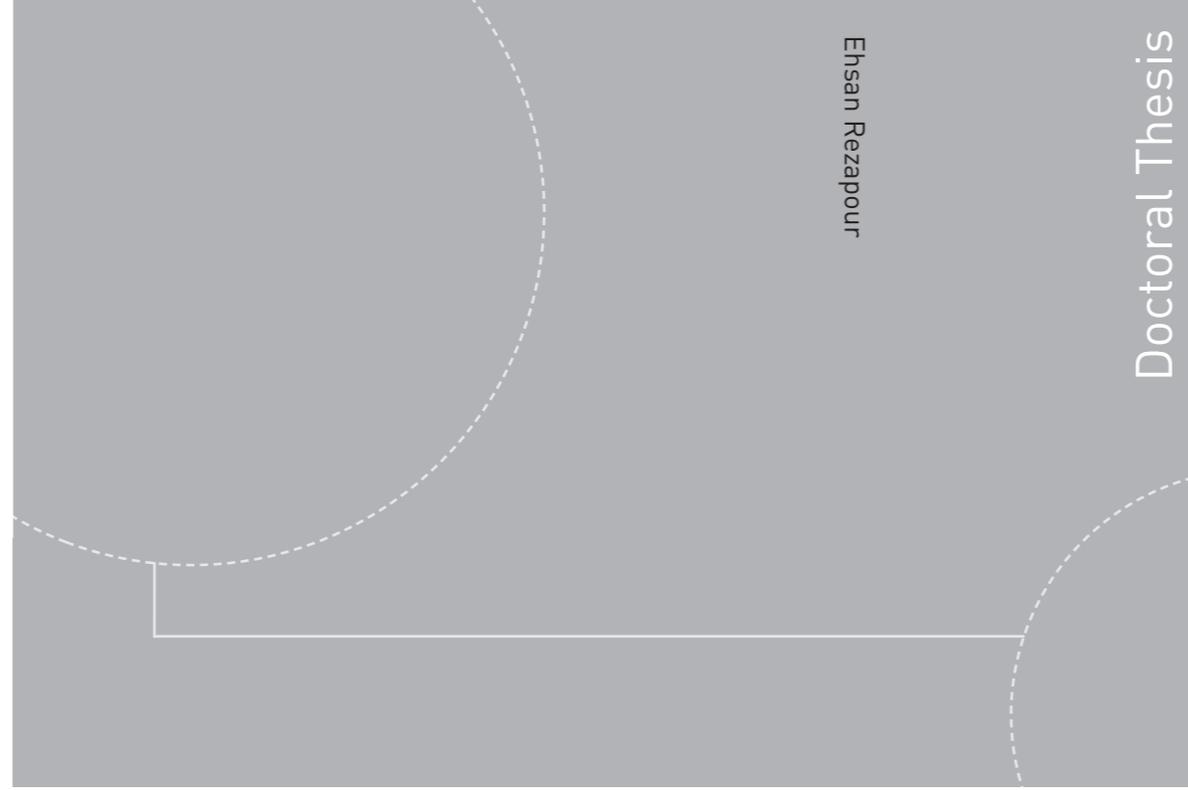


ISBN 978-82-326-0754-9 (printed version)
ISBN 978-82-326-0755-6 (electronic version)
ISSN 1503-8181



NTNU – Trondheim
Norwegian University of
Science and Technology



Doctoral theses at NTNU, 2015:46

Ehsan Rezapour

Doctoral Thesis

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



NTNU – Trondheim
Norwegian University of
Science and Technology

Doctoral theses at NTNU, 2015:46

Ehsan Rezapour

Model-based Locomotion Control of Underactuated Snake Robots

Ehsan Rezapour

Model-based Locomotion Control of Underactuated Snake Robots

Thesis for the degree of Philosophiae Doctor

Trondheim, March 2015

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



NTNU – Trondheim
Norwegian University of
Science and Technology

NTNU

Norwegian University of Science and Technology

Thesis for the degree of Philosophiae Doctor

Faculty of Information Technology,
Mathematics and Electrical Engineering
Department of Engineering Cybernetics

© Ehsan Rezapour

ISBN 978-82-326-0754-9 (printed version)

ISBN 978-82-326-0755-6 (electronic version)

ISSN 1503-8181

Doctoral theses at NTNU, 2015:46



Printed by Skipnes Kommunikasjon as

Summary

Snake robots are a class of biologically inspired robots which are built to emulate the features of biological snakes. These robots are underactuated, i.e. they have fewer control inputs than degrees of freedom, and are hyper redundant, i.e. they have many degrees of freedom. Furthermore, snake robots utilize complex motion patterns and possess a complicated but highly flexible physical structure. These properties make locomotion control of snake robots a complicated and challenging control problem.

This thesis considers model-based locomotion control of planar snake robots. In particular, based on kinematic and dynamic models of the snake robot locomotion, using different control approaches we derive feedback control laws in order to solve various control problems. Moreover, through rigorous mathematical stability analysis, we prove the stability of the controlled system. It is noteworthy to mention that due to the complicated dynamical behavior of snake robots which gives rise to a complex dynamic model, and also the underactuation which is characterized by the lack of direct and independent control inputs for at least three degrees of freedom of the snake robot, the vast majority of the previous works on snake robots and similar multi-link robotic structures use numerical simulations and experimental results which are obtained using different robotic snakes, as the main tools to show the performance of the proposed controllers. In contrast, however, in this work based on nonlinear control theory, we take a model-based control design approach and we present formal stability proofs for the closed-loop systems along with numerical simulations and experimental results. The simulations and experiments are performed for a snake robot which is composed of N similar links which are serially connected through $N - 1$ joints. The first $N - 1$ links are independently actuated using electric motors, however, the N -th link which we refer to as the head link of the snake robot is passive. This makes the orientation and position of the center of mass of the robot underactuated.

The contributions of the thesis are presented in six chapters, and can be categorized in two types; contributions to modelling and contributions to control design for snake robots. The contributions and contents of each chapter are as follows.

In Chapter 1, we discuss the fundamental properties of the snake robot locomotion, and we investigate the most common types of gait patterns used by biological snakes. Furthermore, in this chapter we review the relevant previous works on snake robots and we present the abstracts of

the academic papers which form the basis of the thesis.

In Chapter 2, we present three different modelling techniques for the snake robot locomotion on horizontal and flat surfaces. The first dynamic model is derived based on the Lagrangian approach to modelling mechanical systems, and the equations of motion are written in the standard second-order form. The second dynamic model is derived using the techniques of differential geometry, and this model contains the effects of parametric modelling uncertainties on the locomotion of the robot. The first and the second models which are referred to as the complex model of the snake robot, are among the contributions of the thesis and to our best knowledge have not been presented in any previous works. The third model that we present in Chapter 2 is a simplified model of the snake robot locomotion which is previously presented in [11]. In this simplified model, the rotational motion of the joints is mapped to translational link displacements. Through this mapping, which is shown to be valid for small joint angles, many of the strong nonlinear terms which are present in the dynamics of the system are approximated by simpler linear terms, and these approximations make the resulting simplified dynamic model more amenable to model-based control design.

In Chapter 3, we consider body shape and orientation control for locomotion of snake robots. In particular, in this chapter we aim to control the body shape of the robot to a desired gait pattern, and the orientation of the robot to a reference angle defined by a path following guidance law. To this end, using the joint torques we stabilize a desired gait pattern among the directly actuated body shape variables which define the internal configuration of the robot. Furthermore, we use a gait parameter in the form of a dynamic compensator which controls the orientation of the robot to a reference angle defined by the path following guidance law. Through numerical simulations and experiments which are performed using a robotic snake, we show that this control approach makes the robot converge to and follow a desired geometric path. Moreover, using an input-output stability analysis we show that the solutions of the controlled system remain uniformly bounded. Furthermore, in this chapter using sliding mode techniques, we design a body shape and orientation feedback controller which successfully makes the robot follow a desired path even in the presence of strong nonlinear terms in the dynamics of the robot arising due to parametric modelling uncertainties.

In Chapter 4, we utilize the simplified model of the snake robot locomotion to carry out the model-based feedback control design for the robot. In particular, we use the method of virtual holonomic constraints (VHC) to address direction following and maneuvering control of the snake robot.

In the direction following problem, the control objective is to regulate the linear velocity vector of the snake robot to a constant reference while guaranteeing boundedness of the system states. Furthermore, in the maneuvering problem, the control objective is to make the robot converge to a geometric path, and to move along the path according to a desired velocity profile. Using the VHC method, we stabilize the solutions of the dynamics of the robot to a constraint manifold. The constraint manifold is defined based on VHC which encode a lateral undulatory gait pattern. Moreover, this gait pattern is parameterized by the states of two dynamic compensators which are used to control the forward velocity and orientation of the robot.

In Chapter 5, we utilize the complex model of the snake robot locomotion in order to address the direction following and maneuvering control problems. In particular, first we stabilize VHC for the body shape variables of the system which encode a lateral undulatory gait pattern. The VHC are composed of a sinusoidal part and an offset term. The sinusoidal part is employed to induce the lateral undulatory motion and the offset term is used to reorient the robot in the plane. Furthermore, the VHC are dynamic in that they depend on the states of two dynamic compensators which are used in order to control the forward velocity and orientation of the robot. In particular, using a high-gain feedback on the offset term, we turn the controlled orientation dynamics of the robot into a singularly perturbed form, for which we show that the orientation error can be made arbitrarily small. In addition, we use the frequency of the oscillations of the snake body, i.e. the frequency of the desired gait pattern, as a virtual control input which is used to control the forward velocity of the robot. Using backstepping techniques, we make the forward velocity error arbitrarily small and make the normal velocity converge to a small neighborhood of zero. This solves the direction following problem. In order to address the maneuvering problem using VHC, we use a hierarchical control approach based on a reduction theorem for asymptotic stability of dynamical systems presented in [98]. In particular, first we stabilize a constraint manifold for the robot and then we control the reduced dynamics of the robot on the constraint manifold using two dynamic compensators. These dynamic compensators control the forward velocity and the head angle of the robot to given references. Furthermore, we define the reference head angle and the reference velocity of the robot such that the convergence of the path following error to an arbitrarily small neighborhood of the origin is guaranteed. Extensive numerical simulations are presented which validate the performance of the proposed control strategies.

Finally, in Chapter 6, we summarize the contributions of the chapters and present some concluding remarks. Furthermore, we present topics for possible future works on locomotion control of snake robots.

Acknowledgments

During the almost last four years, snake robots, these fascinating robotic structures, have not only offered me great opportunities to learn more and more about complex motion control methods, but also have provided me with huge chances to learn from some of the excellent scientists who are among the leading persons of their fields. The results of my findings are collected inside this book, which has my name written on its cover alone, however, without the help of many individuals I would have never been able to finish this work.

Above all, I owe my deepest gratitude to my supervisor, Professor Kristin Ytterstad Pettersen, for her excellent support throughout this way. Kristin's always positive and inspiring attitude, her excellent feedback on my publications, and her profound knowledge of control theory are only some of the reasons of my infinite respect for her. From Kristin I have learned that one can be brilliant in mathematics, and still be an amazingly sociable person.

I would like to thank my co-supervisors, Professor Jan Tommy Gravdahl and Associate Professor Øyvind Stavdahl for the fruitful discussions that we had during our meetings, and for their constructive feedback on my ideas and works. I am grateful to my (un-official) co-supervisor, Dr. Pål Liljebäck, for his always friendly and great support. Many of the ideas presented in this work are inspired by Pål's excellent book and publications on snake robots.

I express my sincere gratitude to Professor Manfredi Maggiore and Alireza Mohammadi at the Systems Control Group at the University of Toronto, for their kind hospitality during my one semester visit in the department of Electrical and Computer Engineering at U of T. I truly appreciate all our meetings and discussions which led to the most remarkable parts of the results presented here. Representing his genius, Manfredi's magical fountain pen can amazingly solve any challenging mathematical problem on a piece of paper.

I am thankful to all my friends and colleagues at the department of Engineering Cybernetics at NTNU, specially my best friend Amirhossein Nikoofard, and my officemate Mark Haring. I sincerely thank Eleni Kela-sidi for her great assistance during the experiments in Snake Robots Lab at NTNU. Outside the department, I am thankful to my friends, Behrooz Moghimian Hoosh and Ali Mohammad Darvish for their always reliable friendship.

Finally, this thesis never would have taken form without the loving support of my family. I am infinitely indebted to, especially, my father Ali Rezapour for his great support throughout my whole life, and my brother Mohsen and my sisters Minoos and Maryam for encouraging and helping me pursue my dreams. I also would like to write in commemoration of my mother, Nahid, whom I carry with me the lovely (though short) memories of in every single second of my life, and whom making her happy is a main source of inspiration for me. Last but not least, my most profound appreciation goes to my wife, Niloufar Farazani, for her love, support and patience during the long time I spent for writing this thesis. To me, you are the special one.

Trondheim, February 2015

Ehsan Rezapour

Contents

1	INTRODUCTION	1
1.1	Characteristics of Snake Robots	2
1.2	Biological Snakes	4
1.2.1	Anatomy of Snakes	4
1.2.2	Gait Patterns for Biological Snakes	6
1.3	Preliminary Remarks and Literature Review	8
1.3.1	Holonomic versus Nonholonomic Constraints	9
1.3.2	Underactuated Mechanical Systems	10
1.3.3	Snake Robots with Nonholonomic Velocity Constraints	11
1.3.4	Snake Robots without Nonholonomic Velocity Constraints	12
1.3.5	Snake Robots in Cluttered Environments	13
1.3.6	Real-Time Applications of Snake Robots	18
1.4	Scope of the Thesis	19
1.4.1	Motion on a Planar and Flat Surface	19
1.4.2	Locomotion without Nonholonomic Velocity Constraints	19
1.4.3	Model-based Locomotion Feedback Control	20
1.4.4	Locomotion based on Lateral Undulation	20
1.5	Contributions of the Thesis	20
2	Modelling Methods for Snake Robot Locomotion on Planar Surfaces	29
2.1	Kinematics of the Snake Robot	33
2.1.1	The Geometry of the Problem	34
2.1.2	The Forward Kinematic Map of the Snake Robot	34
2.2	Complex Model of the Snake Robot Locomotion: The Lagrangian Approach	36
2.2.1	Euler-Lagrange Equations of Motion of the Snake Robot	36
2.2.2	The Ground Friction Model	38
2.2.3	Partial Feedback Linearization of the Dynamic Model	39
2.2.4	Dynamic Model via Absolute Link Angles	41
2.3	Complex Model of the Snake Robot Locomotion: Differential Geometric Approach	42

2.3.1	Geometry and Kinematics of the Snake Robot	43
2.3.2	Equations of Motion	43
2.3.3	Partial Feedback Linearization of the Geometric Model	45
2.4	Simplified Modelling Approach	48
2.4.1	An Overview of the Simplified Modelling Approach	49
2.4.2	Simplified Kinematics and Dynamics of the Snake Robot	49
2.4.3	Model Transformation	52

3 Body Shape and Orientation Control for Locomotion of Snake Robots 55

3.1	Control Design Objectives	58
3.2	Virtual Holonomic Constraints Based Body Shape and Orientation Control of Snake Robots	59
3.2.1	The idea of Virtual Holonomic Constraints	59
3.2.2	Path Following Control with Virtual Holonomic Constraints	60
3.3	Robust Locomotion Control of Snake Robots Using Sliding Mode Techniques	67
3.3.1	Sliding Mode Tracking Control of the Joint Angles	67
3.3.2	Underactuated Tracking Control via Sliding Mode Design	70
3.4	Stability Analysis of the Controlled Systems	73
3.4.1	Guidance-based Control Strategy for the Snake Robot	75
3.5	Simulation and Experimental Results	84
3.5.1	Simulation and Experimental Results for the VHC based Body Shape and Orientation Controller	84
3.5.2	Simulation Results for the Sliding Mode Controller	93
3.5.3	Simulation Results for the Dynamic Feedback Controller with Analysis of the Controlled System	96

4 Model-based Locomotion Control Approaches for Snake Robots

Part I: The Simplified Model		101
4.1	Direction Following Control of Snake Robots based on a Simplified Model	106
4.1.1	Control Design Objectives	106
4.1.2	Body Shape Control	107
4.1.3	Enforcing the VHC for the Shape Variables of the Robot	108
4.1.4	Orientation Control	108
4.1.5	Velocity Regulation	110
4.2	Maneuvering Control of Snake Robots along Straight Paths	116
4.2.1	Control Design Objectives	117
4.2.2	Body Shape and Orientation Control	118
4.2.3	Maneuvering Control	121

4.3	Maneuvering Control of Snake Robots along Curved Paths	126
4.3.1	Control Design Objectives	127
4.3.2	Body Shape and Orientation Control	128
4.3.3	Maneuvering Control along Curved Paths: The Dynamic Task	131
4.3.4	Maneuvering Control along Curved Paths: The Geometric Task	135
4.4	Simulation Results	140
5	Model-based Locomotion Control Approaches for Snake Robots	
	Part II: The Complex Model	149
5.1	Direction Following Control of Snake Robots	152
5.1.1	Control Design Objectives	152
5.1.2	Body Shape Control	154
5.1.3	Head Angle Control	157
5.1.4	Velocity Regulation	159
5.2	Maneuvering Control of Snake Robots	164
5.2.1	Control Design Objectives	164
5.2.2	Body Shape Control	166
5.2.3	Velocity Control	167
5.2.4	Path Following Control of Snake Robots	174
5.3	Simulation Results	178
6	Conclusions and Future Challenges	189
6.1	Summary of the Chapters	191
6.2	Future Challenges in Snake Robot Control	195
	References	198

INTRODUCTION

Snake robots are a class of biologically-inspired robots which are built to emulate the structural characteristics of biological snakes. In general, the physical structure of snake robots consists of serially connected joint modules which have the capability of motion in one or more planes, see Figure 1.1. Robust crawling locomotion in cluttered environments, capability of traversing narrow terrains due to the slenderness of the body, and the absence of dedicated locomotion organs such as legs and wings, are among the interesting characteristics of biological snakes. Inspired by these properties, during the last four decades many robotic researchers have attempted to build snake robots which can emulate the features of biological snakes. The resulting biologically-inspired robotic systems, however, pose many interesting challenges both in terms of theoretical developments and real-time applications. Proposing various methods for overcoming these challenges has been the subject of many scientific publications since the 70's when the first snake robot was built [1], see Figure 1.1. However, due to the intricacies in the mechanical structure and dynamical behaviour of snake robots, locomotion control of these robots is still an active and open area of research. Furthermore, the complexity in the dynamical behaviour of these systems has enabled this type of robots to serve as a valuable benchmark example for validating the effectiveness of many nonlinear control approaches. Although control problems for snake robots are challenging, the attractive idea of using these robots for operations where human presence is unsafe or impossible, or where the traditional types of locomotion tools such as wheels and legs are not effective, has made the snake robot community dynamic and progressive.

In this thesis, motivated by the lack of model-based feedback control strategies for snake robots, we try to bridge the gap between the nonlinear control theory and applications for these robots. In particular, based on kinematic and dynamic models of snake robot locomotion, we will employ various nonlinear

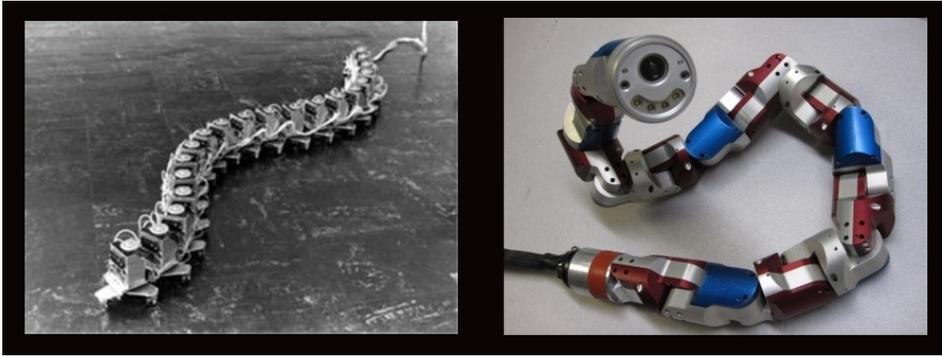


Figure 1.1: The snake robot ACM III (left), which was the world's first snake robot developed by Prof. Shigeo Hirose in 1972. **Image source:** [1]. The snake robot Uncle Sam (right) developed at Carnegie Mellon University. **Image source:** [2]

control design tools, in order to design effective locomotion control strategies which are verified by formal stability proofs for the corresponding closed-loop systems. Furthermore, we will validate the proposed control strategies through extensive numerical simulations and experimental results which are obtained using a robotic snake.

1.1 Characteristics of Snake Robots

In this section, we investigate some structural characteristics of snake robots, which are important both for theoretical developments and real-time applications. Some of the following characteristics have made snake robots an interesting alternative for many applications. However, some of these properties complicate the locomotion control problems and turn these robots into an interesting benchmark example for theoretical developments on robotic systems.

Small Cross-Section: Snake robots are characterized by a slender body which enables them to perform tasks in narrow environments. This makes them an interesting alternative for many practical situations where human presence is impossible or undesired. Pipeline inspection [3]–[5] and various medical applications [6]–[9] are examples of areas in which the slender body of snake robots has made them a potentially useful tool.

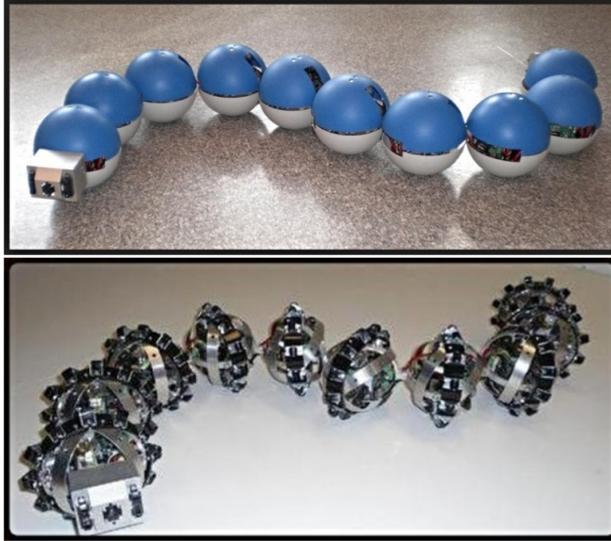


Figure 1.2: The snake robot Kulko (above), developed at the NTNU Snake Robot Lab for motion in cluttered environments.. The snake robot Wheeko (below), developed at the NTNU Snake Robot Lab for motion on horizontal and flat surfaces. **Image source:** [11]

Hyper-Redundancy: Snake robots are a class of hyper-redundant robots which are characterized by many degrees of freedom, see [90]. As a result, snake robots are capable of performing many complicated tasks due to the presence of many degrees of freedom. Moreover, this hyper-redundancy enables them to keep mechanical stability even during the failure of some their actuators. However, the presence of a large number of degrees of freedom complicates the dynamic behaviour of the system and enlarges the resulting state-space for the dynamical system. This makes the control problem challenging.

Underactuation: Snake robots are a class of underactuated mechanical systems which are characterized by fewer control inputs than degrees of freedom [15]. In particular, a planar snake robot is underactuated with respect to the orientation and planar position. The underactuated variables of the system can only be indirectly controlled, for example by exploiting the coupling terms with the dynamics of the fully-actuated degrees of freedom of the system. Furthermore,

motion control of underactuated mechanical systems is an active and open area of research where there are very few general approaches. As a result, underactuated snake robots can be considered as a valuable benchmark example for theoretical developments on underactuated mechanical systems.

Robust Locomotion: Massiveness in the variety of biological snakes on the planet arises from the fact that snakes can robustly traverse various terrains. Mimicking the features of biological snakes, snake robots can robustly locomote on unstructured surfaces and cluttered environments. This superior property of snake robots has made them an interesting alternative to traditional wheeled and tracked locomotion systems, which might get tangled up in irregularities in the terrain.

Vehicle–Manipulator Properties: Snake robots have the capability of being used both as a vehicle, i.e. they can transport objects from one point to another point, and as a robotic manipulator. The problem of mobile manipulation can be broken down into three sub-problems [10]: 1) the robot moves from its initial configuration to a configuration close to the target object, 2) the robot grasps the object, 3) the robot moves (holding the object) to another configuration which places the object into its goal configuration. Due to mobility and manipulability, snake robots are capable of effectively performing all these steps. This simultaneous vehicle-manipulator property has made them an interesting potential agent for many real-time applications.

1.2 Biological Snakes

Snake robots are a class of biologically-inspired robots, i.e. they are built to emulate the features of biological snakes. In particular, snake robots are inspired by structural characteristics of biological snakes, which enable them to robustly traverse various challenging terrains which might be inaccessible for many other types of traditional wheeled and legged based locomotion systems.

Throughout this thesis, we derive many parts of our design approaches for locomotion control of snake robots based on biological observations. Consequently, in this section we briefly review the unique characteristics of biological snakes. The materials presented here are based on [1], [11], and [12].

1.2.1 Anatomy of Snakes

The skeletal structure of snakes is in general composed of vertebrae, ribs, and a skull, see Figure 1.3. Snakes have between 130 to 500 vertebrae, and the ribs are

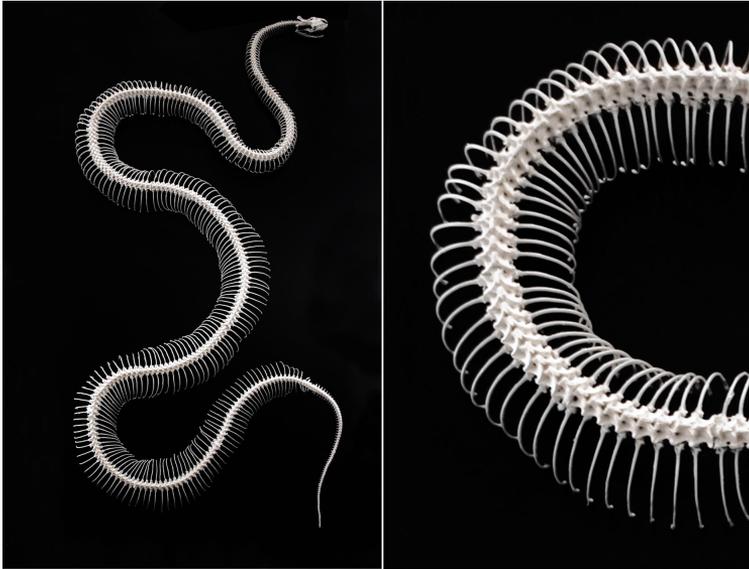


Figure 1.3: The skeleton of biological snakes consists of vertebrae, ribs, and a skull.
Image source: www.shutterstock.com

attached to each side of every vertebra in order to protect the internal organs of the animal. The vertebrae constitute a column along the body of snakes, and the spinal cord runs through a channel along this column. The primary function of the spinal cord is to transmit neural signals between the brain and the rest of body. Vertebral column protects the spinal cord.

The neighbouring vertebrae are connected to each other and can perform quite limited relative rotation (about 10° to 20° about the vertical axes and a few degrees about the horizontal axes). However, the sum of all these small rotations along the snake body gives a superior flexibility to the animal motion.

Snakes use their muscles for periodic body shape changes which lead to locomotion. In particular, the muscles are diagonally arranged along each side of the snake's body, and their terminals are attached to the ribs. These muscles often connect distant ribs, and sometimes adjacent ribs. Such as other animals, snakes use contraction and relaxation of these muscles in order to gain different motion patterns. We discuss these patterns in the next subsection.

The snake body is covered by a skin that is completely coated by scales which are actually the thickened parts of the skin, see Figure 1.4. When the



Figure 1.4: The skin of biological snakes is covered by scales. **Image source:** www.photography.nationalgeographic.com

snake moves on rough terrains, the scales protect the skin from cutting and tearing. Furthermore, the scales on the skin play a fundamental role in snake locomotion by providing **anisotropic** friction properties, i.e. the friction in the tangential direction of the body is smaller than the friction in normal direction. Studies of biological snakes and simulation studies have shown the importance of this anisotropy in the locomotion of the snakes. In particular, in [28] experimental studies on frictional characteristics of biological snakes skin shows that the friction coefficient in the tangential direction of the body is smaller than the friction coefficient in the normal direction. In case of snake robots, in [11] it is shown that a snake robot which is subject to **isotropic** friction forces, i.e. equal friction forces in the tangential and the normal direction of the links of the robot, is not controllable. Consequently, we consider this anisotropic friction property as a fundamental assumption for the control designs which are presented throughout this thesis.

1.2.2 Gait Patterns for Biological Snakes

Such as other animals, snakes use periodic movement of the limbs in order to achieve locomotion in the plane. These periodic body motions are known as the "**gait pattern**", and are among the diverse and elegant characteristics of each animal, e.g. walking, running, and galloping. Similar to other animals, snakes employ different types of locomotion patterns according to the characteristics of the environments which they are moving in, e.g. a narrow environment or a sandy surface, or based on the situation that the animal is experiencing, e.g. preying, searching. In particular, snake robots use the following gait patterns:

- **Lateral Undulation:** This is the most common locomotion pattern among biological snakes, see Figure 1.5. During lateral undulation, which is also

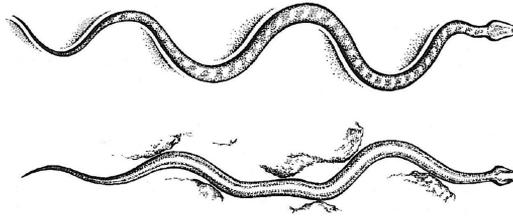


Figure 1.5: Lateral undulation. **Image source: The Encyclopaedia of Snakes [12]**

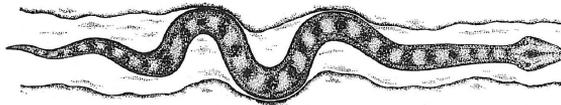


Figure 1.6: Concertina locomotion. **Image source: The Encyclopaedia of Snakes [12]**

known as serpentine crawling, continuous waves are propagated backward along the snake body from head to tail [1]. While these waves are travelling along the snake body, the animal pushes the sides of its body against the irregularities of the surface, and consequently moves forward. Lateral undulation provides the fastest type of locomotion among biological snakes. As noted in e.g. [1], each part of a snake body which is moving based on lateral undulation follows the path traced out by the head.

- **Concertina Locomotion:** This type of locomotion pattern is often used for narrow spaces where the possible range of motion is small. The motion is carried out by first extending the front part of the body forward while the back part is curved several times to provide an anchor against the narrow environment [11]. Once the front part of the snake body is completely extended, it is then similarly employed to provide an anchor so that the back part of the snake body can be drawn up. The same pattern gets repeated, see Figure 1.6.
- **Sidewinding:** This type of locomotion pattern is often used by the snakes which live on sandy terrains, e.g. desert snakes. Roughly speaking, the sidewinding motion consists of two parts. In the first part, the snake raises and moves the front part of its body sideways, while the back part provides an anchor against the ground, until the front part is again on the ground. In the second part, the front part provides an anchor against the ground and the back part is raised and moves sideways. During sidewinding, the snake moves at about 45° with respect to its heading and leaves a trail of characteristic marking in the sand [11], see Figure 1.7.

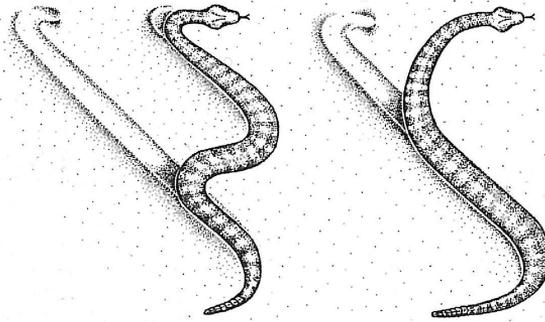


Figure 1.7: Sidewinding. **Image source: The Encyclopaedia of Snakes [12]**

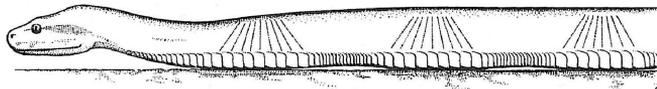


Figure 1.8: Rectilinear Crawling. **Image source: The Encyclopaedia of Snakes [12]**

- **Rectilinear Crawling:** In this type of locomotion pattern, the snake employs the edges of the scales which are located on the underside of its body as anchor points in order to propel itself forward on an almost straight line, see Figure 1.8. The operation consists of stretching forward and hooking the edges of the scales over small irregularities, then pulling the body up to this point [11]. Other parts of the body will perform this pulling and stretching action simultaneously as well. This type of locomotion pattern is often used by the snakes which have a heavy body, or in general by snakes during their approach to a prey to avoid altering it.

1.3 Preliminary Remarks and Literature Review

In this section, we review some of the most important and relevant literature on snake robots. Early empirical and analytical studies of biological snake locomotion were presented in the 1940's by Gray [13]. Since the 1970's, a significant amount of scientific results have been published, which have investigated snake robots and similar multi-link robotic systems, e.g. eel like or underwater snake robots. Consequently, in order to present an efficient literature review for snake robots, first we need to categorize these robots to some sub-classes.

In general, based on the kinematics and dynamics, snake robots can be categorized into two classes; snake robots which are subject to nonholonomic ve-

locity constraints, and snake robots which are not subject to any nonholonomic velocity constraint. Motion control of both classes of snake robots has been considered in several previous works. Furthermore, many previous works have derived controllers for snake robots based on various control approaches. However, due to complexities in the dynamical behaviour of snake robots and underactuation, very few works have presented formal stability proofs for the closed-loop systems. As a result, a majority of the previous works have validated their controllers using numerical simulations or by performing experiments on robotic snakes.

1.3.1 Holonomic versus Nonholonomic Constraints

In this subsection, based on [14], we briefly review the concepts of holonomic and nonholonomic constraints for mechanical systems. In particular, the motion of the constrained mechanical systems is subject to constraints that may arise from the structure of the mechanism, or from the way in which it is actuated and controlled [14].

Consider a mechanical system with the configuration vector q representing the configuration space \mathcal{Q} , which is an n -dimensional smooth manifold locally diffeomorphic¹ to the n -dimensional Euclidean space \mathbb{R}^n . We denote a trajectory in the configuration space as $q(t) \in \mathcal{Q}$. The vector of the generalized velocities is given by \dot{q} which takes values in the tangent space $T_q\mathcal{Q}$. For this system, the motion restrictions can be represented by time-independent equality constraints of the form

$$h_i(q) = 0, \quad i = 1, \dots, k < n \quad (1.1)$$

where $h_i : \mathcal{Q} \rightarrow \mathbb{R}$ are smooth functions, which are called **holonomic constraints**. In general, these constraints are introduced to mechanical systems through physical connections between different parts. A mechanical system whose constraints are all holonomic, is called a *holonomic system* [14]. These constraints confine the attainable configurations of the system to an $(n - k)$ -dimensional submanifold of \mathcal{Q} . As a result, the configuration of the system on this submanifold can completely be described by $(n - k)$ variables which are called *degrees of freedom* of the system.

Another type of constraints for mechanical systems can be described by velocity dependent equalities of the form

$$a_i(q, \dot{q}) = 0, \quad i = 1, \dots, k < n \quad (1.2)$$

¹There is a smooth and invertible function which maps one smooth manifold to another and the inverse of this function is smooth as well.

where $a_i : TQ \rightarrow \mathbb{R}^n$ are smooth and linearly independent vector-valued functions, which are called kinematic or velocity constraints. These constraints will limit the admissible motions of the system by restricting the set of generalized velocities that can be attained at a given configuration [14]. Velocity constraints are typically represented in the *Pfaffian form*, i.e. linear in the generalized velocities, as

$$a_i(q)^T \dot{q} = 0, \quad i = 1, \dots, k < n \quad (1.3)$$

Consider a single velocity constraint in the following Pfaffian form

$$a^T(q)\dot{q} = 0 \quad (1.4)$$

We call (1.4) a **nonholonomic constraint**, if it is not integrable into the holonomic form (1.1). Nonholonomic constraints affect the instantaneous mobility of the system, but they do not reduce the number of the generalized coordinates.

The class of snake robots that we consider throughout this thesis are *not* subject to nonholonomic velocity constraints.

1.3.2 Underactuated Mechanical Systems

The main reason that makes locomotion control problems for snake robots challenging is the underactuation. An **underactuated robot** is a robot with fewer actuators (control inputs) than the number of variables describing its configuration (degrees of freedom) [15]. In general, underactuation may arise due to the following reasons, see [16]:

- **Dynamics:** The way in which control forces and torques affect the motion of the system, i.e. the dynamics of the system. For example, ships, aircraft.
- **Design:** Presence of flexible components such as gearboxes and springs in the physical structure of a mechanical system makes it underactuated. Furthermore, special designs for reducing weight, cost, or energy consumption can make the system underactuated.
- **Actuator Failure:** If throughout an operation, the robot loses one or some of its actuators, then the robot can turn into an underactuated system.
- **Benchmark Example Robots used for Research:** Some robots are artificially made underactuated, and are used as benchmark examples for theoretical developments on underactuated control systems theory. Robots such as underactuated multi-link pendulums, Pendubot, Acrobot, and Cart-Pole are examples of these underactuated systems.

The underactuation leads to a constraint that can be described by acceleration dependent equalities of the form $a_i(q, \dot{q}, \ddot{q}) = 0$. Underactuated systems are thus also denoted second-order nonholonomic systems. The complexity of controlling underactuated mechanical systems is often associated with the fact that it is challenging to find an appropriate feedback transform and a change of coordinates to rewrite the dynamics in a linear format [103]. An N -link planar snake robot which moves on a horizontal and flat surface, and is not subject to nonholonomic velocity constraints, has $(N + 2)$ degrees of freedom. However, only $(N - 1)$ of these degrees of freedom are directly and independently actuated by motors which are mounted on the joints of the robot. Consequently, a planar snake robot without nonholonomic velocity constraints has three degrees of underactuation. In particular, the position of the center of mass and the orientation represent the underactuated degrees of freedom of the snake robot.

1.3.3 Snake Robots with Nonholonomic Velocity Constraints

The majority of previous works on snake robot control consider snake robots with nonholonomic velocity constraints, which is inspired by the world's first snake robot developed in 1972, see [1]. The nonholonomic constraints are in the form of sideslip constraints on the links of the robot, i.e. where each link is constrained from moving sideways, and are usually introduced to the system by installing passive wheels along the body of the robot, see Figure 1.9. These constraints allow the control input to be specified directly in terms of the desired propulsion of the snake robot, something which is employed in [19]–[21] for computed torque control of the position and heading of snake robots with nonholonomic velocity constraints. In [22]–[24], by using tools from differential geometry it is shown how the sinusoidal inputs can induce forward propulsion on snake robots, and also the controllability properties of snake robots with passive wheels are investigated. In [25] and [33], position and path following controllers are proposed for the case where some, but not all, of the snake robot links are subject to sideslip constraints. These constrained links can be lifted from the ground, which gives the system more degrees of freedom that can be utilized to follow a trajectory while simultaneously maintaining a high manipulability. Similar approaches are considered in [26], where strategies for sinus-lifting during the lateral undulatory motion are proposed. In [27], a path following controller for a snake robot with nonholonomic velocity constraints is proposed, and Lyapunov analysis is employed in order to analyze the controlled system. In [29], a continuum dynamic model of a 3D snake robot is derived, where the snake is modelled as a continuous curve that cannot move sideways. In [30]–[32], path following controllers for snake robots with passive wheels which move based on lateral undulation are proposed. In particular,

these controllers are derived based on minimizing the lateral constraint forces on the wheels of the robot. Path following controllers for wheeled snake robots which are derived using Lie bracket calculations and controllability analysis are proposed in [34]–[35]. In [36], a biomimetic approach which is based on the central pattern generator (CPG), i.e. a rhythmical motion generator that can generate self-induced oscillations, is proposed. Moreover, it is shown that the locomotion curvature of the robot and the motion velocity can be changed by adjusting the CPG parameters, and the theoretical results are validated using a wheeled snake robot.

Remark 1.1 *The common approach that is used for locomotion control of snake robots with nonholonomic velocity constraints is to employ the constraints to establish a connection between the internal body shape changes and the resulting locomotion of the robot, see e.g. [19,27,30,32,33]. Such approaches are the only known approaches for motion control of wheeled snake robots which infer some formal and model-based conclusions on the motion of the robot [37]. In particular, snake robots with passive wheels can be controlled at a kinematic level, i.e. control inputs can be derived in the form of time-differentiable velocities using the connection between the shape changes and locomotion. Note that even in this case the actual control inputs of the snake robots are accelerations which are obtained by time-differentiation of the velocities. Moreover, many of the previous works on wheeled-snake robots use numerical simulations and experiments for validating their control approaches, i.e. instead of presenting formal mathematical stability proofs.*

1.3.4 Snake Robots without Nonholonomic Velocity Constraints

Locomotion control of snake robots without nonholonomic velocity constraints is only considered in a few previous works. This class of snake robots includes wheel-less snake robots, for which the locomotion mechanisms are more similar to biological snakes, see e.g. [11]. Furthermore, wheel-less snake robots are interesting for traversing even more challenging environments where the passive wheels may slip or get tangled up in irregularities in the terrain.

In the previous subsection we stated that snake robots with passive wheels can be controlled at a kinematic level. However, snake robots without nonholonomic velocity constraints should be controlled in a dynamic level. In particular, due to the absence of passive wheels, i.e. lack of nonholonomic velocity constraints, there is no established connection between the body shape changes and the resulting locomotion of the center of mass of the robot. This can complicate the control design.

In [39], a control strategy is proposed for sinus-lifting during lateral undulation by solving a quadratic optimization problem. In [40], chain fitting and wave extraction are used in order to generate a variety of rolling and sidewinding gaits. Annealed chain fitting efficiently maps a continuous backbone curve describing the 3D shape of the robot by a set of joint angles, while wave extraction makes the joint angles fit to a sequence of backbone curves and identifies parameterized periodic functions that produce those sequences. Together, they allow a gait designer to conceive a motion in terms of three-dimensional shapes and translate them into easily manipulated wave functions [40]. In [41], path following control of swimming snake robots is achieved by moving the joints according to a predetermined gait pattern while introducing an angular offset in each joint to steer the robot to some desired path, i.e. controls the orientation. Methods based on numerical optimal control are considered in [42] for determining optimal gaits during positional control of snake robots. In [43], numerical simulations are used to study the properties of lateral undulation that are related to the optimality of motion of the snake robot. In [44] cascaded systems theory is employed to achieve path following control of a snake robot described by a simplified model. In this simplified model of the snake robot, the motion of the links is approximated as translational motion instead of rotational motion, an approximation which is valid for small joint angles. Also it is shown that by controlling the orientation of the robot to a reference angle defined by a Line-of-Sight guidance law, the robot converges to and follows a desired straight path. In [45], controllability and stability analysis of planar snake robot locomotion is considered, and the stability results for a path following controller based on numerical investigations using Poincaré maps are presented. All the above works consider snake locomotion on flat surfaces. In [91], averaging theory is employed to investigate the velocity dynamics of the snake robot during lateral undulatory motion, and to show that the average forward velocity will converge exponentially to a steady state value.

1.3.5 Snake Robots in Cluttered Environments

The idea of obstacle-aided locomotion [46] of snake robots, which is based on actively utilizing external objects to move forward in cluttered environments and on irregular surfaces, has been considered in several papers. In particular, in [46]–[49], hybrid dynamical models of obstacle-aided locomotion of snake robots are presented. Hybrid dynamical systems are a class of dynamical systems which contain both continuous and discontinuous state evolution, i.e. the state variables can both flow and jump, see e.g. [50]. A hybrid model of snake robot locomotion based on a timestepping method, i.e. without tracking discrete events, is presented in [46]. In [47], a shape curve-based approach



Figure 1.9: The snake robot ACM R3 (above) developed at Tokyo Institute of Technology. The robot is covered with passive wheels. **Image source:** [17]. The snake robot ACM R5 (below) developed at Tokyo Institute of Technology. The robot is covered by passive wheels and can swim under water. **Image source:** [18]

to obstacle aided locomotion is presented where new push-points for robot-obstacle contact are identified online and a corresponding robot shape curve is calculated. A hybrid model derived based on event-tracking is presented in [48], where also a hybrid control strategy employing measured contact forces is proposed which can maintain propulsion while simultaneously preventing the snake robot from getting jammed between obstacles. The results of [48] are experimentally validated in [49]. A curve fitting algorithm is proposed in [51], which determines the shape of the robot with respect to obstacles in the environment. Moreover, this shape is propagated backwards along the snake body and this pushes the robot forward. Among the first works on snake robot control in cluttered environments, in [1] an obstacle avoidance strategy is proposed that modifies the shape of a snake robot based on contact force measurements along the snake body to avoid obstacles. In [52], compliant locomotion that adapts automatically to the robot's surrounding terrain, is achieved by controlling the torques exerted by the joints of the robot.



Figure 1.10: The snake robot Mamba (above) developed at NTNU. The robot is equipped with contact force sensors along its body. **Image Source:** [38]. The Snake Arm (below) without passive wheels, which is a holonomic mechanical system. **Image courtesy of OC Robotics, www.ocrobotics.com.**

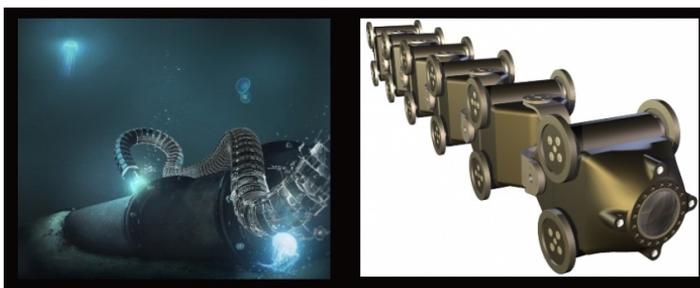


Figure 1.11: Snake robots can be used for pipeline inspections (left). The snake robot PIKo (right) developed for pipeline inspection. **Image source:** www.robotnor.com

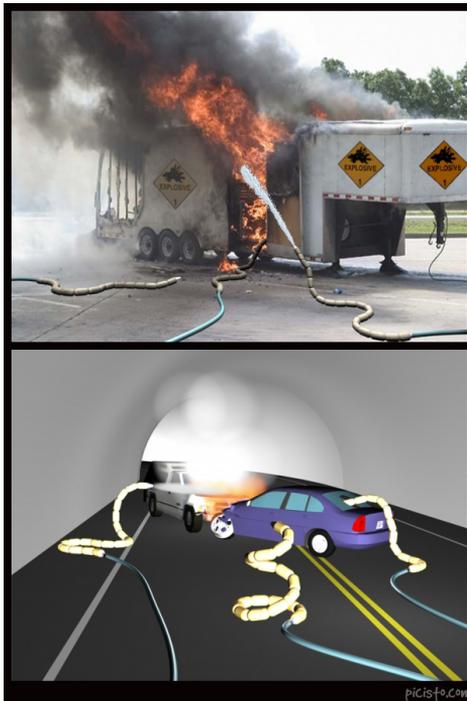


Figure 1.12: Snake robots can be used for fire fighting. Image source: [11].



Figure 1.13: The snake robot Anna Konda developed at NTNU for fire fighting operations. Image source: [11]

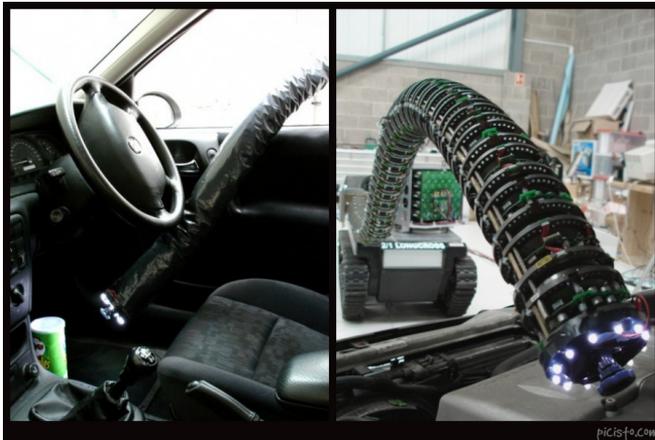


Figure 1.14: Due to their structural flexibility, snake arms can be used for inspection tasks where human presence is unsafe or unwanted. **Image courtesy of OC Robotics, www.ocrobotics.com.**

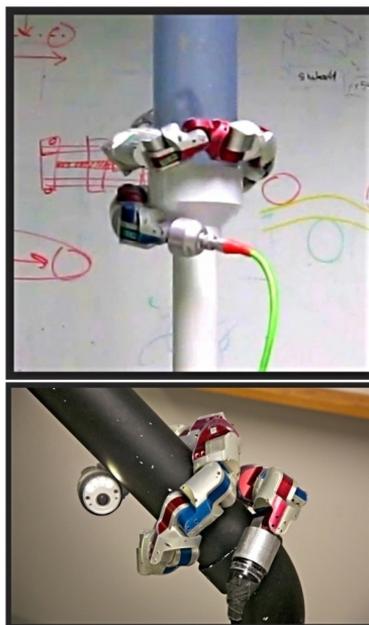


Figure 1.15: Snake robots can be used for pipeline inspections. **Image: Jiuguang Wang [58] (below), and D. Rollinson [2] (above)**

1.3.6 Real-Time Applications of Snake Robots

The unique characteristics of snake robots which were discussed in Section 1.1, such as the small cross section, hyper redundancy and high flexibility of the body that can be bent in one or more planes, have made these robots an excellent alternative agent for many real-time applications. In particular, many previous works focus on fire fighting, search and rescue, medical, pipeline inspection, and other industrial applications of snake robots.

For **fire fighting applications**, the idea is to build a self-propelled robotic fire hose which is inspired by biological snakes that can crawl into a burning building and extinguish a fire on its own without exposing human fire fighters to risk. The resulting robotic system works based on being connected to a stationary water supply through a flexible hose [53], see Figures 1.12–1.13. In [55], a control algorithm and design of a pipe crawling robot which can be used for the purpose of earthquake rescue and pipeline maintenance is presented. This robot is designed such that it can intelligently change its body shape in order to fit the pipe or tunnel-like voids within rubble.

Snake arm robots, i.e. snake robots mounted on a mobile base, see Figure 1.14, are a class of hyper-redundant manipulators which are basically characterized by their ability to navigate through complex and narrow environments. In [56], a sequential quadratic programming optimization approach is used to minimize a set of changes to the configuration of the snake arm that allows the robot to follow a desired trajectory with minimal error. In [57] and [67], design and control of a mobile hyper-redundant urban **search and rescue** robot in the form of an snake arm is considered. This system is capable of inspecting areas reachable by the mobile base, besides unreachable areas such as small cracks and pipes, using the camera and sensors mounted on its end-effector. Cooperation between dogs and snake robots for urban search and rescue missions is considered in [68].

Medical applications of snake robots and snake arms are reported in some previous works. In [59]–[62], image-guided surgery using snake robots and snake arms are considered. Open heart surgery using a snake robot is reported in [63]. In [64]–[65], the small-cross section and superior flexibility of snake robots are used for applications in minimally invasive surgeries. In [66], using Lie algebra, an algorithm for accurate estimation of the shape of a snake robot inside the body relative to the organs is presented.

Among various potential applications of snake robots in industry, the most relevant one is probably **pipeline inspection**, where the slenderness and high flexibility of the snake robot body is crucial, see Figure 1.15. Researchers at NTNU have developed a prototype named PIKo [69]–[71], see Figure 1.11, which is a robot with several identical joint modules that are equipped with motor-

ized, i.e. active, wheels. The system is designed to handle complex pipe network structures with corners, T-joints, vertical branching and varying diameter. In [72], a method of estimating a snake robot's motion inside and outside of straight pipes is presented which only uses the measurements of the robot's joint angles. This is done by introducing an approach to constructing a body-fixed frame for the robot which is aligned with the centerline of the straight pipe. In [73]–[74], pipe navigator snake robots are developed which employ planar sinusoidal gaits for moving through pipelines. In [75], development of an autonomous in-pipe robot is presented which performs online ultrasonic inspection for pipe wall thickness, in order to find the location of defect. In [76], a compliant controller for straight pipes is presented which allows the robot to automatically adapt to the shape of its environment while doing pipe navigation. The results of [76] are expanded in [5] to complex pipe networks where the compliant controller estimates the overall state of the robot in terms of the parameters of a gait. The controller then commands new gait parameters relative to that estimated state.

1.4 Scope of the Thesis

The following subjects form the scope of the research work that is presented in this thesis.

1.4.1 Motion on a Planar and Flat Surface

Throughout this thesis, we consider modelling and locomotion control of snake robots which move on a horizontal and flat surface. Although snakes in general move in 3D space, however, the essential principles of locomotion control of snake robots can be captured in 2D space. Furthermore, the locomotion of snake robots in 3D consists of a combination of link oscillations in a vertical and a horizontal surface.

1.4.2 Locomotion without Nonholonomic Velocity Constraints

The long-term goal of this project is to realize broad applications of snake robots in real-life. These applications mostly include using snake robots in unstructured environments where the surfaces are not flat. Consequently, in this thesis, we focus on locomotion control of snake robots without passive wheels, i.e. without nonholonomic velocity constraints, for which the links can move sideways. The application of this class of snake robots thus opens up important step

further towards real-time applications. Only few previous works have investigated locomotion control problems for this type of snake robots.

1.4.3 Model-based Locomotion Feedback Control

As we stated before, the main objective of this thesis is to contribute to development of model-based feedback control design for snake robots. In particular, based on kinematic and dynamic models of the snake robot locomotion, we derive solutions for various locomotion control problems, and we present formal stability proofs for the closed-loop systems. In particular, for the class of snake robots that we consider in this thesis, there have been few model-based feedback control designs presented so far. Motivated by the lack of analytical control approaches relying on formal stability proofs, in this thesis we develop feedback control approaches that are both verified through model-based analysis and simulations plus experiments.

1.4.4 Locomotion based on Lateral Undulation

Throughout this thesis, we will use lateral undulatory locomotion as the desired motion pattern for the snake robot. In particular, after stabilizing this gait pattern which leads to forward locomotion, we will use the gait parameters as additional control terms which are used in order to achieve additional control objectives, e.g. orientation and velocity control. This gait pattern is the most common form of locomotion among biological snakes. Inspired by this biological observation, we base the control designs on this type of snake locomotion.

1.5 Contributions of the Thesis

We titled this thesis as "*Model-based Locomotion Control of Underactuated Snake Robots*". As the title describes, the main objective of this thesis is to contribute to further theoretical developments of locomotion control theory for snake robots based on the dynamic model of the system. Following this main goal, we have made several contributions to snake robot control that are described in details below.

Chapter 2

Topic: We derive the dynamic model of the snake robot locomotion which is used to carry out model-based feedback control design in the subsequent chapters.

Contributions: In Chapter 2, we present three different dynamic models of snake robot locomotion. The derivation of the first and the second model are among the contributions of this thesis. However, the third model which represents a simplified kinematic and dynamic model of the snake robot, is previously derived in [11]. In particular, the first contribution of the chapter is to derive a dynamic model of the snake robot that describes the motion of the system in a Lagrangian framework. Using the Lagrangian function, we derive the equations of motion in the standard second-order form. Furthermore, we integrate the effects of anisotropic friction forces acting on the system by using the Jacobian matrices of the center of mass of the links. The second contribution of this chapter is to derive the equations of motion of the snake robot using the techniques of differential geometry. In particular, using the Riemannian metric, Christoffel symbols, and Jacobian matrices of the links, we derive the *natural representation* of the equations of motion of the robot. Moreover, we add the effects of parametric modelling uncertainties to these equations. In addition, we use partial feedback linearization to write this model in a form which is more amenable to model-based control design.

Chapter 3

Topic: In Chapter 3, we present body shape and orientation controllers which induce path following on the robot.

Contributions: The first contribution of this chapter is to design a body shape and orientation controller for the snake robot through a dynamic feedback control law. In particular, we control the body shape variables which define the internal configuration of the robot to a desired lateral undulatory gait pattern, and we choose the gait parameters using a dynamic compensator which controls the head angle of the robot to a reference head angle defined by a path following guidance law. The second contribution of this chapter is to design a robust body shape and orientation controller for the snake robot using sliding mode techniques. In particular, we stabilize sliding surfaces for the solutions of the joints and the head angle dynamics using sliding mode control techniques, which guarantee that the body shape and orientation control objectives for the robot are achieved even in the presence of strong nonlinear terms arising due to parametric modelling uncertainties. The third contribution of this chapter is to design a dynamic feedback control law for the body shape and orientation of the robot which guarantees the uniform boundedness of the states of the dynamic compensator which controls the head angle of the robot. In particular, the proposed control strategy controls the internal configuration of the robot

according to a desired lateral undulatory gait pattern, and the head angle of the robot in accordance with a reference angle provided by a path following guidance law. Using numerical simulations and experiments with a robotic snake, we show that the proposed control strategies in this chapter make the snake robot converge to and follow a desired geometric path.

Chapter 4

Topic: In Chapter 4, we address various locomotion control problems based on a simplified model of the snake robot.

Contributions: The first contribution of this chapter is to propose a direction following controller which is derived based on a simplified model of the snake robot locomotion. In particular, we solve the direction following control problem using the method of virtual holonomic constraints. In this method, we first stabilize the solutions of the dynamics of the body shape variables to a constraint manifold. This manifold is defined based on the well-known reference joint angle trajectories introduced in [1], which induce lateral undulatory locomotion on snake robots. Afterwards, we evaluate the dynamic of the robot on the exponentially stable constraint manifold where we use the gait parameters as two additional control terms which regulate the forward velocity and orientation of the robot to constant references. The second contribution of this chapter is a model-based solution to maneuvering control problem for the snake robot using the method of virtual holonomic constraints. In particular, we stabilize a constraint manifold for the body shape variables of the robot, and we employ two dynamic compensator which control the orientation and position of the robot on this manifold. Moreover, we control the orientation to a reference time-varying angle defined by a path following guidance law, and we utilize the frequency of the body oscillations, i.e. the gait pattern, as an additional control term in order to control the position of the robot along a desired path, and consequently we solve the maneuvering control problem. Extensive numerical simulations are presented which validate the theoretical control design approach.

Chapter 5

Topic: In Chapter 5, we address various locomotion control problems based on a complete dynamic model of the snake robot, which does not require the simplifying assumptions on the model used in Chapter 4.

Contributions: The first contribution of this chapter is to design a direction

following controller based on the complete dynamic model of the snake robot, which regulates the head angle and forward velocity of the robot to constant references. The second contribution of this chapter is to solve the maneuvering control problem for the snake robot based on the complete dynamic model. In particular, using the method of virtual holonomic constraints we stabilize a constraint manifold for the body shape variables, and we control the reduced dynamics of the system evaluated on the constraint manifold using two dynamic compensators. The maneuvering control problem is solved for general continuously differentiable curved paths. Extensive simulation results are presented which validate the theoretical approach.

Publications

The materials presented in this thesis are based on several conference and journal papers which are listed below.

- E. Rezapour, P. Liljebäck, **"Path Following Control of a Planar Snake Robot with an Exponentially Stabilizing Joint Control Law"**, in Proc. IAV 2013 - IFAC Symposium on Intelligent Autonomous Vehicles, Vol. 8, Part 1, pp. 28–35, Gold Coast, Australia, June 2013.

This paper considers the problem of path following control of a planar snake robot without sideslip constraints. We use Lagrangian mechanics to derive the dynamical equations of motion of the system. The possibility of controlling the orientation of the robot in the absence of external dissipative forces is investigated. An exponentially stabilizing joint control law for the directly actuated shape variables of the robot is presented. We analytically design a guidance-based path following control law for the snake robot, and we show that the trajectories of the heading error dynamics are ultimately bounded with a bound that can be made arbitrarily small. The efficiency of the control design is shown with numerical simulations.

- E. Rezapour, K. Y. Pettersen, P. Liljebäck, and J. T. Gravdahl, **"Path Following Control of Planar Snake Robots Using Virtual Holonomic Constraints"**, in Proc. 2013 IEEE International Conference on Robotics and Biomimetics (ROBIO 2013), pp. 530-537, Shenzhen, China, Dec. 12–14, 2013.

This paper considers path following control of planar snake robots using virtual holonomic constraints. We first derive the Euler-Lagrange equations of motion of the snake robot. Moreover, we integrate the effects

of friction forces into these equations. Subsequently, we define geometric relations among the generalized coordinates of the system, using the method of virtual holonomic constraints. These appropriately defined constraints shape the geometry of a constraint manifold for the system, which is a submanifold of the configuration space of the robot. In particular, we show that the constraint manifold can be made invariant by a suitable choice of feedback. Furthermore, we analytically design a smooth feedback control law to render the constraint manifold exponentially stable for the controlled system. We show that enforcing the appropriately defined virtual holonomic constraints implies that the robot converges to and follows a desired geometric path. Numerical simulations are presented to support the theoretical design.

- E. Rezapour, K. Y. Pettersen, P. Liljebäck, and J. T. Gravdahl, "**Differential Geometric Modelling and Robust Path Following Control of Snake Robots Using Sliding Mode Techniques**", in Proc. IEEE International Conference on Robotics and Automation (ICRA 2014), Hong Kong, China, 31 May – 5 June, 2014.

This paper considers straight line path following control of wheel-less planar snake robots using sliding mode techniques. We first derive the Poincaré representation of the equations of motion of the robot using the techniques of differential geometry. Furthermore, we use partial feedback linearization to linearize the directly actuated part of the system dynamics. Subsequently, we propose an analytical solution to the robust path following control problem in two steps. In the first step, we use sliding mode techniques to design a robust tracking controller for the joints of the robot to track a desired gait pattern. In the second step, we stabilize an appropriately defined sliding manifold for the underactuated configuration variables of the robot, thereby guaranteeing convergence of the robot to the desired straight path. The paper presents simulation results which validate the theoretical approach.

- E. Rezapour, K. Y. Pettersen, J. T. Gravdahl, and P. Liljebäck, "**Body Shape and Orientation Control for Locomotion of Biologically-Inspired Snake Robots**", in Proc. 5th IEEE RAS/EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob 2014), São Paulo, Brazil, Aug. 12–15, 2014.

This paper considers guidance-based motion control of planar snake robots using a dynamic feedback control law. We first present the Euler-Lagrange

equations of motion of the robot. Subsequently, we introduce a dynamic feedback control law for the joints of the robot to track a desired gait pattern. This tracking control law depends on the time evolution of the state variables of a dynamic compensator which is used for controlling the orientation of the robot. In particular, we employ the dynamic compensator to practically stabilize a reference head angle defined by a Line-of-Sight path following guidance law. Using an input-output stability analysis, we show the uniform boundedness of the solutions of the controlled system. Furthermore, we use a perturbation analysis to show that the orientation error is ultimately bounded by an arbitrarily small bound. Simulation results are presented to validate the theoretical results.

- E. Rezapour, A. Hofmann, K. Y. Pettersen, A. Mohammadi, and M. Maggiore, "**Virtual Holonomic Constraints Based Direction Following Control of Snake Robots Described by a Simplified Model**", in Proc. 2014 IEEE Multi-Conference on Systems and Control (IEEE MSC 2014), Antibes, France, Oct. 8–10, 2014.

This paper considers direction following control of planar snake robots for which the equations of motion are described based on a simplified model. In particular, we aim to regulate the orientation and the forward velocity of the robot to a constant vector, while guaranteeing the boundedness of the states of the controlled system. To this end, we first stabilize a constraint manifold for the fully-actuated body shape variables of the robot. The definition of the constraint manifold is inspired by the well-known reference joint angle trajectories which induce lateral undulatory motion for snake robots. Subsequently, we reduce the dynamics of the system to the invariant constraint manifold. Furthermore, we design two dynamic compensators which control the orientation and velocity of the robot on this manifold. Using numerical analysis and a formal stability proof, we show that the solutions of the dynamic compensators remain bounded. Numerical simulations are presented to validate the theoretical design.

- E. Rezapour, A. Hofmann, and K. Y. Pettersen, "**Maneuvering Control of Planar Snake Robots Based on a Simplified Model**", in Proc. 2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014), Bali, Indonesia, Dec. 5–10, 2014.

This paper considers maneuvering control of planar snake robots, for which the equations of motion are described based on a simplified model. In particular, we aim to stabilize a desired straight line path for the position of

the center of mass of the robot, and to regulate the forward velocity of the robot along the path to a constant reference velocity. In order to solve this problem, we first stabilize a desired gait pattern for the fully-actuated body shape variables of the robot. Furthermore, we choose the parameters of this gait pattern by means of two dynamic compensators which control orientation and position of the robot in the plane. In particular, by solving the maneuvering problem, we control the body shape, orientation, and planar position of the robot.

- 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 Conference on Decision and Control (CDC 2014), Los Angeles, CA, Dec. 15–17, 2014.

This paper investigates the problem of direction following for planar snake robots. The control objective is to regulate the linear velocity vector of the snake robot to a constant reference while guaranteeing boundedness of the system states. The proposed feedback control strategy enforces virtual constraints encoding a lateral undulatory gait, parametrized by states of dynamic compensators used to regulate the orientation and forward speed of the snake robot.

- 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*, SpringerOpen, 1:3, 2014.

This paper considers path following control of planar snake robots using virtual holonomic constraints. In order to present a model-based path following control design for the snake robot, we first derive the Euler-Lagrange equations of motion of the system. Subsequently, we define geometric relations among the generalized coordinates of the system, using the method of virtual holonomic constraints. These appropriately defined constraints shape the geometry of a constraint manifold for the system, which is a submanifold of the configuration space of the robot. Furthermore, we show that the constraint manifold can be made invariant by a suitable choice of feedback. In particular, we analytically design a smooth feedback control law to exponentially stabilize the constraint manifold. We show that enforcing the appropriately defined virtual holonomic constraints for the configuration variables implies that the robot converges to and follows a desired geometric path. Numerical simulations and exper-

imental results which are obtained using a robotic snake are presented to validate the proposed controllers.

- 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*, 2014. Submitted.

This paper investigates the problem of maneuvering control for planar snake robots. The control objective is to make the center of mass of the snake robot converge to a desired path and traverse the path with a desired velocity. The proposed feedback control strategy enforces virtual constraints encoding a lateral undulatory gait, parametrized by states of dynamic compensators used to regulate the orientation and forward speed of the snake robot.

Modelling Methods for Snake Robot Locomotion on Planar Surfaces

In this chapter, we present various modelling techniques for a planar snake robot without nonholonomic velocity constraints which moves on a horizontal and flat surface. Since throughout this thesis, we utilize a variety of model-based feedback control approaches in order to control the locomotion of the robot, then it is important to carefully analyse the kinematic and dynamic equations describing the motion of the robot in the plane. In particular, in this chapter we present three different techniques for modelling the snake robots which are not subject to nonholonomic velocity constraints. The first technique is based on the Lagrangian approach for modelling mechanical systems, which derives the dynamics describing the motion of the robot in the standard second-order form. The second technique includes the derivation of the equations of motion of the robot in a differential geometric framework. Moreover, we integrate the effects of parametric modelling uncertainties into the second model which will then be used for robust control design in the subsequent chapters. The third technique, which is previously presented in [11], derives a simplified kinematic and dynamic model of the snake robot locomotion. We will use these models in the subsequent chapters for presenting various model-based feedback control approaches for planar snake robot locomotion.

In particular, the first and the second model of the snake robot locomotion that we present in this chapter contain the complete characteristics of the nonlinear dynamics of the robot. Thus, we refer to these models as the *complex models* of the snake robot. In contrast, however, the third model only contains parts of the complete model which are essential to overall locomotion of the robot. We refer to this model of the snake robot where some of nonlinear terms are ap-

proximated by linear terms as the *simplified model* of the snake robot. In addition, for all three models we use the technique of partial feedback linearization in order to write the dynamics in a form which is more amenable to model-based control designs.

In particular, the following three different models of snake robot locomotion are presented in this chapter.

Dynamic Model Derived based on the Lagrangian Approach: The first model of the snake robot locomotion that we present in this chapter will be derived using a Lagrangian approach. In particular, based on the Lagrangian function of the robot, we will derive the Euler-Lagrange equations of motion of the robot. In contrast with the previous works, which derive the equations of motion of the snake robot mostly using a Newton-Euler formulation, this Lagrangian approach is simple to follow and is advantageous for complex systems such as multi-link robots [86]. However, these equations are naturally equivalent to the equations of motion which were derived based on a Newton's second law in many previous works such as [1] and [11].

Dynamic Model of the Snake Robots which are Subject to Parametric Modelling Uncertainties: The second model of snake robot locomotion which is presented in this chapter, includes the effects of nonlinear terms arising due to parametric modelling uncertainties. In particular, we use the techniques of differential geometry in order to derive the dynamic equations of motion of the robot, and we integrate the effects of parametric modelling uncertainties into the model. This model will be used for robust control design for the snake robot in the subsequent chapters.

A Simplified Model of the Snake Robot Locomotion: The third model that we present in this chapter is previously derived in [11], and will be used in order to design locomotion control approaches in the subsequent chapters. The main idea behind the simplified model of the snake robot dynamics, which we refer to as the simplified model, is to map the periodic body shape changes to forward propulsion, through mapping the rotational joint motion to translational link displacements, cf. Figure 2.2. Since the translational displacements are in general less complex than rotational motion, this will simplify the resulting dynamic model of the robot.

Contributions of this chapter: The first contribution of this chapter is to derive the equations of motion of the snake robot in a Lagrangian framework, i.e. treating the robot as a whole and performing the analysis using a Lagrangian

function [86], which is simple to follow and better suited for studying advanced mechanical phenomena such as elastic link deformations [86], which might be insightful for future research challenges on snake robots. Moreover, we integrate the anisotropic friction forces into these equations using the Jacobian matrices of the links, which give a straightforward mapping of these forces to the equations of motion. The second contribution of this chapter is to derive a partially feedback linearized Poincaré representation of the equations of motion of a snake robot without nonholonomic velocity constraints, which gives a detailed mathematical description of the system's behaviour that can be used for analysis and model-based control design. In particular, we believe that formulating the equations of motion of the system in a geometric mechanics setting can be particularly useful for effectively addressing problems regarding the fundamental properties of snake robot locomotion. For instance, it is interesting both for controllability analysis and motion planning algorithms which are derived based on differential geometric approaches to mechanics, see e.g. [87]. To our best knowledge, the only previous work which derives the dynamic model of unconstrained snake robots in a geometric mechanics framework is [88]. However, that work employs general affine differential geometry in contrast with the particular Poincaré representation in the present work. Furthermore, we add parametric modelling uncertainties due to changes in the friction coefficients to this model. We also present a partial feedback linearization of the resulting model that makes it more suitable for model-based control design.

Organization of this chapter: This chapter is organized as follows. In Section 2.1, we present the kinematics of the snake robot. In Section 2.2, we derive a complete model of the snake robot locomotion using Lagrangian mechanics. In Section 2.3, we formulate the equations of motion of the snake robot in a differential geometry framework. In Section 2.4, we present a simplified model of the snake robot locomotion.

Publications: The results of this chapter are based on the journal paper [84], and the conference papers [78] and [79].

Basic Notations

Following the notation in [11], we make use of the following matrices and vectors in presenting the kinematics and dynamics of the snake robot.

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

$$e = [1, \dots, 1]^T \in \mathbb{R}^N, \quad E = \begin{bmatrix} e & 0_{N \times 1} \\ 0_{N \times 1} & e \end{bmatrix} \in \mathbb{R}^{2N \times 2}$$

$$\bar{e} = [1, \dots, 1]^T \in \mathbb{R}^{N-1}, \quad \theta = [\theta_1, \dots, \theta_N]^T \in \mathbb{R}^N$$

$$\sin \theta = [\sin \theta_1, \dots, \sin \theta_N]^T \in \mathbb{R}^N, \quad \cos \theta = [\cos \theta_1, \dots, \cos \theta_N]^T \in \mathbb{R}^N$$

$$S_\theta = \text{diag}(\sin \theta) \in \mathbb{R}^{N \times N}, \quad C_\theta = \text{diag}(\cos \theta) \in \mathbb{R}^{N \times N}$$

$$\dot{\theta}^2 = [\dot{\theta}_1^2, \dots, \dot{\theta}_N^2]^T \in \mathbb{R}^N, \quad b = [0, \dots, 0, 1]^T \in \mathbb{R}^{N-1}$$

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

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

$$0_{N-1} = [0, \dots, 0]^T \in \mathbb{R}^{N-1}, \quad SC_\theta = \begin{bmatrix} K^T S_\theta \\ -K^T C_\theta \end{bmatrix}$$

Furthermore, N denotes the number of links, l denotes the length of the link, and m and J denote the uniformly distributed mass and moment of inertia of each link, respectively.

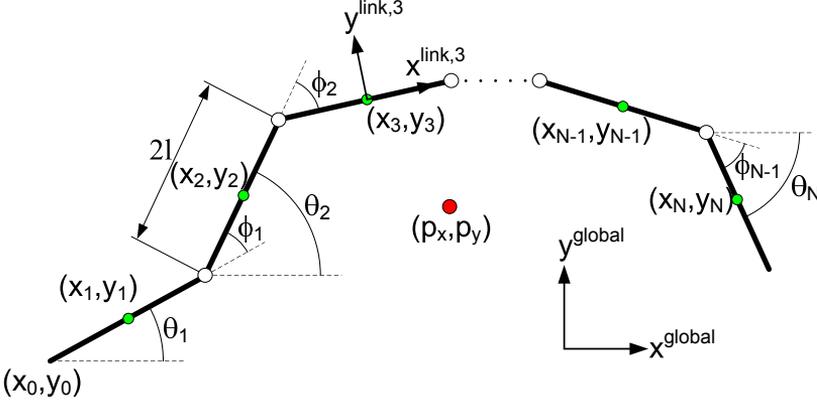


Figure 2.1: Kinematic parameters of the snake robot

2.1 Kinematics of the Snake Robot

In order to perform control design, we need to write the governing equations of the system in an implementable way. This is often done by choosing a local coordinate chart and writing the system equations with respect to these coordinates. According to the illustration of the snake robot in Figure 2.1, we choose the vector of the generalized coordinates of the N -link snake robot as

$$q = [\phi_1, \phi_2, \dots, \phi_{N-1}, \theta_N, p_x, p_y]^T \in \mathbb{R}^{N+2} \quad (2.1)$$

where ϕ_i with $i \in \{1, \dots, N-1\}$ denotes the i -th joint angle, θ_N denotes the head angle, and the pair (p_x, p_y) describes the position of the CM of the robot with respect to the global $x-y$ axes. The joint angles $(\phi_1, \phi_2, \dots, \phi_{N-1})$ of the snake robot are directly and independently actuated with motors, and thus represent the **fully-actuated** DOF of the system. However, there are no direct and independent control inputs for the head angle (θ_N) and the position of the CM (p_x, p_y) of the snake robot, and thus these are **underactuated** DOF of the system. Consequently, according to the actuation type, the configuration vector of the system can be divided into the following vectors

$$q_a = [\phi_1, \phi_2, \dots, \phi_{N-1}]^T \in \mathbb{R}^{N-1} \quad (2.2)$$

$$q_u = [\theta_N, p_x, p_y]^T \in \mathbb{R}^3 \quad (2.3)$$

Since the robot is not subject to nonholonomic velocity constraints, the vector of the generalized velocities is defined as the time-derivative of (2.1) as

$$\dot{q} = [\dot{\phi}_1, \dot{\phi}_2, \dots, \dot{\phi}_{N-1}, \dot{\theta}_N, \dot{p}_x, \dot{p}_y]^T \in \mathbb{R}^{N+2} \quad (2.4)$$

Using these coordinates, it is possible to specify the kinematic map of the robot. In this thesis, we denote the first N elements of the vector q , i.e. $(\phi_1, \dots, \phi_{N-1}, \theta_N)$, as the angular coordinates, and the corresponding dynamics as the angular dynamics of the system.

2.1.1 The Geometry of the Problem

The $(N + 2)$ -dimensional configuration space of the snake robot which we denote as $\mathcal{Q} = \mathcal{S} \times \mathcal{G}$, is composed of the shape space \mathcal{S} and a Lie group \mathcal{G} which is freely and properly acting on the configuration space. In particular, the **shape variables**, i.e. $q_a = (\phi_1, \phi_2, \dots, \phi_{N-1})$, which define the internal configuration of the robot and which we have direct control on, take values in \mathcal{S} . Moreover, the **position variables**, i.e. $q_u = (\theta_N, p_x, p_y)$, which are passive DOF of the system, lie in \mathcal{G} . The velocity space of the robot is defined as the differentiable $(2N + 4)$ -dimensional tangent bundle of \mathcal{Q} as $T\mathcal{Q} = \mathbb{T}^N \times \mathbb{R}^{N+4}$, where \mathbb{T}^N denotes the N -torus in which the angular coordinates live. The free Lagrangian function of the robot $\mathcal{L} : T\mathcal{Q} \rightarrow \mathbb{R}$ is invariant under the given action of \mathcal{G} on \mathcal{Q} . The coupling between the shape and the position variables causes the net displacement of the position variables, according to the cyclic motion of the shape variables, i.e. the *gait pattern*. Note that for simplicity of presentation, throughout this thesis we consider local representation of $T\mathcal{Q}$ embedded in an $(2N + 4)$ -dimensional open subset of the Euclidean space \mathbb{R}^{2N+4} .

2.1.2 The Forward Kinematic Map of the Snake Robot

Based on the kinematic parameters of the snake robot given in Figure 2.1, it is possible to write the coordinate representation of the forward kinematic map. The map between the absolute link angles θ_i and the relative joint angles ϕ_i is given by

$$\theta_i = \sum_{n=i}^{N-1} \phi_n + \theta_N \quad (2.5)$$

The **orientation** angle of the robot, can be defined as the average of the absolute link angles as

$$\bar{\theta} = \frac{1}{N} \sum_{i=1}^N \theta_i \quad (2.6)$$

The position of the CM of the i -th link with respect to the global $x - y$ axes can be, respectively, given as

$$x_i = x_0 + 2l \sum_{j=1}^{i-1} \cos \theta_j + l \cos \theta_i \quad (2.7)$$

$$y_i = y_0 + 2l \sum_{j=1}^{i-1} \sin \theta_j + l \sin \theta_i \quad (2.8)$$

where $2l$ denotes the length of each link, and (x_0, y_0) denotes the tail position, cf. Figure 2.1. The linear velocities of the CM of the i -th link with respect to the global $x - y$ axes can be found by taking the time-derivative of (2.7)–(2.8) which gives

$$\dot{x}_i = \dot{x}_0 - 2l \sum_{j=1}^{i-1} \sin \theta_j \dot{\theta}_j - l \sin \theta_i \dot{\theta}_i \quad (2.9)$$

$$\dot{y}_i = \dot{y}_0 + 2l \sum_{j=1}^{i-1} \cos \theta_j \dot{\theta}_j + l \cos \theta_i \dot{\theta}_i \quad (2.10)$$

Since all the links have equal length and mass, the position of the CM for the whole structure of the robot is defined as

$$(p_x, p_y) = \left(\frac{1}{N} \sum_{i=1}^N x_i, \frac{1}{N} \sum_{i=1}^N y_i \right) \quad (2.11)$$

To facilitate path following control of the CM of the snake robot, we replace the tail position (x_0, y_0) in (2.7)–(2.8) with the position of the CM of the robot (p_x, p_y) using the following change of coordinates

$$x_0 = p_x - \frac{1}{N} \sum_{i=1}^N \left(2l \sum_{j=1}^{i-1} \cos \theta_j + l \cos \theta_i \right) \quad (2.12)$$

$$y_0 = p_y - \frac{1}{N} \sum_{i=1}^N \left(2l \sum_{j=1}^{i-1} \sin \theta_j + l \sin \theta_i \right) \quad (2.13)$$

Substituting (2.12)–(2.13) along with their time-derivatives into (2.7)–(2.10) completes the derivation of the forward kinematic map of the snake robot with respect to the desired specified coordinate chart (q, \dot{q}) .

2.2 Complex Model of the Snake Robot Locomotion: The Lagrangian Approach

In this section, we present a complete model of the snake robot dynamics by using the Lagrangian approach.

2.2.1 Euler-Lagrange Equations of Motion of the Snake Robot

The majority of the previous literature on snake robots and similar mobile multi-links robotic structures, such as eel-like robots, have derived the equations of motion of these robots with a Newton-Euler formulation, i.e. where the equations describing the linear and angular motion of individual links are written separately, see e.g. [1] and [11]. This is due to the fact that it is usually not straightforward to integrate the anisotropic external dissipative forces, i.e. ground friction forces, acting on these complex robotic structures into their Euler-Lagrange equations of motion. However, ground friction forces have been proved to play a fundamental role in snake robot locomotion (see e.g. [11]). In this Section, we derive the equations of motion of the snake robot in a Lagrangian framework, i.e. treating the robot as a whole and performing the analysis using a Lagrangian function, which is simple to follow and better suited for studying advanced mechanical phenomena such as elastic link deformations [86], which might be insightful for future research challenges on snake robots. Moreover, we integrate the anisotropic friction forces into these equations using the Jacobian matrices of the links, which gives a straightforward mapping of these forces for the equations of motion.

Snake robots are a class of *simple* mechanical systems, where the **Lagrangian** $\mathcal{L}(q_a, \dot{q})$ is defined as the difference between the kinetic energy $\mathcal{K}(q_a, \dot{q})$ and potential energy $\mathcal{P}(q)$ of the system [87]. Since the planar snake robot is not subject to any potential field, i.e. $-\nabla\mathcal{P}(q) = 0$, we may write the Lagrangian equal to the kinetic energy, which is the sum of the translational and the rotational kinetic energy of the robot:

$$\mathcal{L}(q_a, \dot{q}) = \mathcal{K}(q_a, \dot{q}) = \frac{1}{2}m \sum_{i=1}^N (\dot{x}_i^2 + \dot{y}_i^2) + \frac{1}{2}J \sum_{i=1}^N \dot{\theta}_i^2 \quad (2.14)$$

where m and J denote the uniformly distributed mass and moment of inertia of the links, respectively. Using the Lagrangian function (2.14), we write the Euler-Lagrange equations of motion of the control system as

$$\frac{d}{dt} \left[\frac{\partial \mathcal{L}(q_a, \dot{q})}{\partial \dot{q}_i} \right] - \frac{\partial \mathcal{L}(q_a, \dot{q})}{\partial q_i} = (B(q)\tau - \tau_f)_i \quad (2.15)$$

where $i \in \{1, \dots, N+2\}$, $B(q) = [e_j] \in \mathbb{R}^{(N+2) \times (N-1)}$ is the full column rank actuator configuration matrix, where e_j denotes the j -th standard basis vector in \mathbb{R}^{N+2} . Moreover, $B(q)\tau \in \mathbb{R}^{N+2}$ with $\tau = [\tau_1, \dots, \tau_{N-1}]^T \in \mathbb{R}^{N-1}$ stands for the generalized forces resulting from the control inputs. Furthermore, $\tau_f = [\tau_f^1, \dots, \tau_f^{N+2}]^T \in \mathbb{R}^{N+2}$ denotes viscous and Coulomb friction forces acting on $(N+2)$ DOF of the system. The controlled Euler-Lagrange equations (2.15) can also be written in the form of a second-order differential equation as

$$M(q_a)\ddot{q} + C(q, \dot{q})\dot{q} = B(q)\tau - \tau_f \quad (2.16)$$

where $M(q_a) \in \mathbb{R}^{(N+2) \times (N+2)}$ is the positive definite symmetric inertia matrix, which is given by

$$M(q_a) = \begin{bmatrix} H^T M_\theta H & 0_{N \times 2} \\ 0_{2 \times N} & NmI_2 \end{bmatrix} \in \mathbb{R}^{(N+2) \times (N+2)} \quad (2.17)$$

where M_θ is given by

$$M_\theta = JI_N + ml^2 S_\theta V S_\theta + ml^2 C_\theta V C_\theta \quad (2.18)$$

and substituting θ_i according to (2.5). Furthermore, $C(q, \dot{q})\dot{q} \in \mathbb{R}^{N+2}$ denotes the generalized Coriolis and centripetal forces, which are given by

$$C(q, \dot{q})\dot{q} = \begin{bmatrix} H^T W_\theta \text{diag}(H\dot{\phi})\dot{\phi} \\ 0_{2 \times 1} \end{bmatrix} \in \mathbb{R}^{N+2} \quad (2.19)$$

where W_θ and $\dot{\phi}$ are given by the following equations, respectively,

$$W_\theta = ml^2 S_\theta V C_\theta - ml^2 C_\theta V S_\theta \quad (2.20)$$

$$\dot{\phi} = [\dot{\phi}_1, \dot{\phi}_1, \dots, \dot{\theta}_N]^T \in \mathbb{R}^N \quad (2.21)$$

and substituting θ_i according to (2.5). Moreover, the right-hand side terms denote the external forces (control forces and friction) acting on the system. The fact that the inertia matrix is only a function of the directly actuated shape variables q_a , is a direct consequence of the invariance of the Lagrangian function of the robot (2.14) with respect to the position variables q_u . Moreover, since $\text{rank}[B(q)] < \dim(q)$ the system is underactuated. This underactuation represents the lack of direct control on the head angle θ_N and the position of the CM (p_x, p_y) of the robot.

The standard second-order form of the equations of motion (2.16) perfectly agrees with the dynamic models of snake robots which are derived based on

the Newton-Euler formulation in previous works, see e.g. [11]. In order to validate the model, in the next chapter we present simulation results which are obtained using the dynamic model (2.16) together with experimental results for the locomotion of the robot which are obtained using a robotic snake, see Figure 3.14. The agreement between simulations and experiments shows that the dynamic model (2.16) accurately represents the motion of the robot.

2.2.2 The Ground Friction Model

In this subsection, both viscous and Coulomb friction models are used for capturing the essential properties of the anisotropic ground friction forces acting on the robot. For modelling the friction, we first define the rotation matrix for mapping from the global frame to the local frame of link i , cf. Figure 2.1, as

$$R_i = \begin{bmatrix} \cos \theta_i & -\sin \theta_i \\ \sin \theta_i & \cos \theta_i \end{bmatrix} \quad (2.22)$$

Note that we can map $\theta_i \mapsto (q_a, \theta_N)$ using (2.5). Using (2.9)–(2.10) and (2.22), the linear velocities of the links in the local link frames can be written in terms of the linear velocities of the links in the global frame as

$$v^{\text{link},i} = \begin{bmatrix} v_t^{\text{link},i} \\ v_n^{\text{link},i} \end{bmatrix} = R_i^T \begin{bmatrix} \dot{x}_i \\ \dot{y}_i \end{bmatrix} \quad (2.23)$$

where $v_t^{\text{link},i}$ and $v_n^{\text{link},i}$ denote the linear velocity of the CM of the i -th link in the tangential (along link x-axis) and normal (along link y-axis) direction of the link, respectively. The total friction force acting on link i is defined as the sum of the viscous and Coulomb friction forces, which are denoted by f_{v_i} and f_{c_i} , respectively, as

$$f^{\text{link},i} = f_{c_i} + f_{v_i} \quad (2.24)$$

Assuming equal friction coefficients for all the links, we write the model of the friction for each individual link i as

$$f_{c_i} = mg \begin{bmatrix} \mu_t \text{sgn}(v_t^{\text{link},i}) & \mu_n \text{sgn}(v_n^{\text{link},i}) \end{bmatrix}^T \in \mathbb{R}^2 \quad (2.25)$$

$$f_{v_i} = \begin{bmatrix} c_t v_t^{\text{link},i} & c_n v_n^{\text{link},i} \end{bmatrix}^T \in \mathbb{R}^2 \quad (2.26)$$

where $i \in \{1, \dots, N\}$, m denotes the mass of a link, g denotes the acceleration due to gravity, and μ_t and μ_n denote Coulomb friction coefficients in the tangential and normal direction of the link, respectively. Furthermore, c_t and c_n

denote viscous friction coefficients in the tangential and normal direction of the link, respectively. Thus, we map the friction force acting on the i -th link to the global $x - y$ frame as

$$f_{\text{global}}^{\text{link},i} = R_i f^{\text{link},i} \quad (2.27)$$

Finally, we can write τ_f in (2.16) as

$$\tau_f = \sum_{i=1}^N \mathcal{J}_i^T(q) f_{\text{global}}^{\text{link},i} \quad (2.28)$$

where

$$\mathcal{J}_i^T(q) = \begin{bmatrix} \frac{\partial \dot{x}_i}{\partial \dot{q}_j} & \frac{\partial \dot{y}_i}{\partial \dot{q}_j} \end{bmatrix} \in \mathbb{R}^{(N+2) \times 2}, \quad j \in \{1, \dots, N+2\} \quad (2.29)$$

denotes the transpose of the Jacobian matrix of the CM of the i -th link.

Remark 2.1 *As argued in [11], the motion of a snake robot with anisotropic viscous ground friction is qualitatively (but not quantitatively) similar as with anisotropic Coulomb friction. However, a viscous friction model is less complex with respect to control design and analysis. Accordingly, we employ a viscous friction model for the control design in this thesis.*

2.2.3 Partial Feedback Linearization of the Dynamic Model

A common method for control of mechanical systems is full-state feedback linearization. This approach is not applicable for snake robots due to the underactuation. However, it is still possible to linearize the dynamics of the directly actuated DOF of the robot, which is called collocated partial feedback linearization, and can simplify the analysis as well as the control design. A similar approach is considered in [11] but for the sake of completeness, we present the approach here. To this end, we separate the dynamic equations of the robot given by (2.16) into two subsets by taking $q = [q_a, q_u]^T \in \mathbb{R}^{N+2}$, with $q_a \in \mathbb{R}^{N-1}$ and $q_u \in \mathbb{R}^3$ which were defined in (2.2)–(2.3):

$$m_{11}(q_a) \ddot{q}_a + m_{12}(q_a) \ddot{q}_u + h_1(q, \dot{q}) = \psi \in \mathbb{R}^{N-1} \quad (2.30)$$

$$m_{21}(q_a) \ddot{q}_a + m_{22}(q_a) \ddot{q}_u + h_2(q, \dot{q}) = 0_{3 \times 1} \in \mathbb{R}^3 \quad (2.31)$$

where $m_{11} \in \mathbb{R}^{(N-1) \times (N-1)}$, $m_{12} \in \mathbb{R}^{(N-1) \times 3}$, $m_{21} \in \mathbb{R}^{3 \times (N-1)}$, and $m_{22} \in \mathbb{R}^{3 \times 3}$ denote the corresponding sub-matrices of the inertia matrix. Furthermore, vector-valued functions $h_1(q, \dot{q}) \in \mathbb{R}^{N-1}$ and $h_2(q, \dot{q}) \in \mathbb{R}^3$ include all the contributions of the Coriolis, centripetal, and friction forces. Moreover, $\psi = [\psi_1, \dots, \psi_{N-1}]^T \in$

\mathbb{R}^{N-1} denotes the non-zero part of the vector of control forces, i.e. $B(q)\tau = [\psi, 0_{3 \times 1}] \in \mathbb{R}^{N+2}$. From (2.31) we have

$$\ddot{q}_u = -m_{22}^{-1} (h_2 + m_{21}\ddot{q}_a) \in \mathbb{R}^3 \quad (2.32)$$

Substituting (2.32) into (2.30) yields

$$(m_{11} - m_{12}m_{22}^{-1}m_{21})\ddot{q}_a - (m_{12}m_{22}^{-1})h_2 + h_1 = \psi \quad (2.33)$$

For linearizing the dynamics of the directly actuated DOF, we apply the global transformation of the vector of control inputs as

$$\psi = (m_{11} - m_{12}m_{22}^{-1}m_{21})\vartheta - (m_{12}m_{22}^{-1})h_2 + h_1 \quad (2.34)$$

where $\vartheta = [\vartheta_1, \vartheta_2, \dots, \vartheta_{N-1}]^T \in \mathbb{R}^{N-1}$ is the vector of new control inputs. Consequently, the dynamic model (2.30)–(2.31) can be written in the following partially feedback linearized form

$$\ddot{q}_a = \vartheta \in \mathbb{R}^{N-1} \quad (2.35)$$

$$\ddot{q}_u = \mathcal{D}(q, \dot{q}) + \mathcal{C}(q_a)\vartheta \in \mathbb{R}^3 \quad (2.36)$$

with

$$\mathcal{D}(q, \dot{q}) = -m_{22}^{-1}(q_a)h_2(q, \dot{q}) = [f_{\theta_N}(q, \dot{q}), f_x(q, \dot{q}), f_y(q, \dot{q})]^T \in \mathbb{R}^3 \quad (2.37)$$

$$\mathcal{C}_i(q_a) = -m_{22}^{-1}(q_a)m_{21}(q_a) = [\beta_i(q_a), 0, 0]^T \in \mathbb{R}^3 \quad (2.38)$$

where \mathcal{C}_i denotes the i -th column of $\mathcal{C} \in \mathbb{R}^{3 \times (N-1)}$. Furthermore, $\beta_i(q_a) : \mathcal{Q} \rightarrow \mathbb{R}$ is a smooth scalar-valued function. It can be numerically shown that the value of β_i is negative at any configuration $q_a \in \mathcal{Q}$. Furthermore, f_{θ_N} , f_x , and f_y denote the friction forces acting on θ_N , p_x , and p_y , respectively (f_{θ_N} also contains Coriolis forces besides the friction forces). For the aim of analysis and model-based control design, we write (2.35)–(2.36) in a more detailed form:

$$\ddot{q}_a = \vartheta \in \mathbb{R}^{N-1} \quad (2.39)$$

$$\ddot{\theta}_N = f_{\theta_N}(q, \dot{q}) + \beta_i(q_a)\vartheta^i \in \mathbb{R} \quad (2.40)$$

$$\ddot{p}_x = f_x(q, \dot{q}) \in \mathbb{R} \quad (2.41)$$

$$\ddot{p}_y = f_y(q, \dot{q}) \in \mathbb{R} \quad (2.42)$$

where the summation convention is applied in (2.40), and henceforth, to all the equations which contain repeated upper-lower indices (i.e. whenever an expression contains a repeated index, one as a subscript and the other as a superscript, summation is implied over this index [87]). The dynamical system

(2.39)–(2.42) is in the form of a control-affine system with drift. In particular, the term

$$\mathcal{A}(q, \dot{q}) = [\dot{q}_a, \dot{q}_u, 0_{(N-1) \times 1}, \mathcal{D}(q, \dot{q})]^T \in \mathbb{R}^{2N+4} \quad (2.43)$$

is called the **drift vector field**, which specifies the dynamics of the robot when the control input is zero. Furthermore, the columns of the matrix

$$\mathcal{B}(q_a) = \begin{bmatrix} 0_{(N+2) \times (N-1)} \\ I_{N-1} \\ [\beta_1(q_a), \dots, \beta_{N-1}(q_a)] \\ 0_{2 \times (N-1)} \end{bmatrix} \in \mathbb{R}^{(2N+4) \times (N-1)} \quad (2.44)$$

are called the **control vector fields**, which enable us to control the internal configuration and consequently the orientation and the position of the robot in the plane.

Remark 2.2 *The last two rows of the control vectors in (2.44) are composed of zero entries. This implies that the control forces have no direct effect on the dynamics of the position of the CM of the robot, i.e. (2.41)–(2.42). Furthermore, the dynamics of the position of the CM are coupled with the dynamics of the directly actuated shape variables q_a , i.e. (2.39), only through the friction forces. Accordingly, in the absence of the friction forces the linear momentum of the robot is a conserved quantity, and the position of the CM of the robot is not controllable.*

Note that this coupling between the dynamics of the CM and the joint angles through friction forces is the essential mechanism underlying snake locomotion, and it is what makes the locomotion control problem challenging.

2.2.4 Dynamic Model via Absolute Link Angles

In this subsection, we review the kinematic and dynamic model of a snake robot presented in [11]. We consider a snake robot with N rigid links each of length $2l$. Each link is assumed to have uniformly distributed mass m and moment of inertia J . We denote the vector of absolute link angles by $\theta = [\theta_1, \dots, \theta_N]^T \in \mathbb{R}^N$, and the CM of the robot in inertial coordinates by $p = [p_x, p_y] \in \mathbb{R}^2$. Figure 2.1 illustrates the kinematic parameters of the snake robot. Following [11], the dynamic equations of the snake robot can be written as follows

$$M_\theta \ddot{\theta} + W_\theta \dot{\theta}^2 - l S C_\theta^T f_R(\theta, \dot{\theta}, \dot{p}) = D^T u \quad (2.45a)$$

$$N m \ddot{p} = E^T f_R(\theta, \dot{\theta}, \dot{p}) \quad (2.45b)$$

where $u \in \mathbb{R}^{N-1}$ is the vector of actuator torques, f_R is the vector of ground friction forces, and the remaining quantities are defined as follows:

$$M_\theta = JI_N + ml^2 S_\theta V S_\theta + ml^2 C_\theta V C_\theta, \quad (2.46a)$$

$$W_\theta = ml^2 S_\theta V C_\theta - ml^2 C_\theta V S_\theta. \quad (2.46b)$$

For simplicity, we assumed that the friction forces acting on the robot are viscous. We have:

$$\begin{aligned} f_R(\theta, \dot{\theta}, \dot{p}) &= \begin{bmatrix} f_{R,x} \\ f_{R,y} \end{bmatrix} = Q_\theta \begin{bmatrix} \dot{X} \\ \dot{Y} \end{bmatrix} \\ &= Q_\theta \begin{bmatrix} lK^T S_\theta \dot{\theta} + e\dot{p}_x \\ -lK^T C_\theta \dot{\theta} + e\dot{p}_y \end{bmatrix} = lQ_\theta S C_\theta \dot{\theta} + Q_\theta E \dot{p} \end{aligned} \quad (2.47)$$

where $X = [x_1, \dots, x_N] \in \mathbb{R}^N$, $Y = [y_1, \dots, y_N] \in \mathbb{R}^N$ are the vectors of inertial coordinates of the CM of the links of the robot. The matrix Q_θ maps the inertial frame velocities of the CM of the links to the inertial frame viscous friction forces acting on the links, and it is given by

$$Q_\theta = - \begin{bmatrix} c_t(C_\theta)^2 + c_n(S_\theta)^2 & (c_t - c_n)S_\theta C_\theta \\ (c_t - c_n)S_\theta C_\theta & c_t(S_\theta)^2 + c_n(C_\theta)^2 \end{bmatrix}, \quad (2.48)$$

where c_t and c_n denote the tangential and normal viscous friction coefficients of the links, respectively.

2.3 Complex Model of the Snake Robot Locomotion: Differential Geometric Approach

In this section, we present the complex model of the snake robot which will be derived in a differential geometric framework. Furthermore, we integrate the effects of parametric modelling uncertainties which are due to the changes in the friction coefficients into this model. We also present a partial feedback linearization of the resulting model that makes it more amenable to model-based control design.

2.3.1 Geometry and Kinematics of the Snake Robot

A planar N -link snake robot evolves naturally in the configuration space $\mathcal{Q} = \mathcal{S} \times \mathcal{G}$, which is composed of a shape space \mathcal{S} and a Lie group \mathcal{G} . In particular, the set of variables that define the internal configuration of the robot take values in \mathcal{S} . These are the relative joint angles of the robot which are equipped with DC motors as actuators, and which in coordinates we denote by $q_a = (\phi_1, \phi_2, \dots, \phi_{N-1}) \in \mathcal{S}$. Moreover, the position variables which define the orientation and the position of the robot in the plane, lie in \mathcal{G} . These are passive DOF of the system which in coordinates we denote by $q_u = (\theta_N, p_x, p_y) \in \mathcal{G}$. Consequently, the total configuration vector of the system is

$$q = [\phi_1, \phi_2, \dots, \phi_{N-1}, \theta_N, p_x, p_y]^T \in \mathcal{Q} \quad (2.49)$$

The velocity space of the system is the $(2N + 4)$ -dimensional tangent bundle of the configuration manifold which we denote by $T\mathcal{Q}$. Since in this thesis we consider a snake robot without nonholonomic velocity constraints, the velocity vector of the system has equal dimension to the configuration vector and is given by the time-derivative of (2.49) as

$$v = [\dot{\phi}_1, \dot{\phi}_2, \dots, \dot{\phi}_{N-1}, \dot{\theta}_N, \dot{p}_x, \dot{p}_y]^T \in T\mathcal{Q} \quad (2.50)$$

In the next subsection we will use the techniques of differential geometry in order to derive the natural representation of the equations of motion of the robot with respect to (q, v) .

2.3.2 Equations of Motion

In this subsection, we derive the Poincaré representation of the equations of motion of the snake robot. The majority of the previous works on snake robots have derived these equations based on a Newton-Euler formulation. However, we believe that formulating the equations of motion of the system in a geometric mechanics setting can be particularly useful for effectively addressing problems regarding the fundamental properties of snake robot motion. In particular, it is interesting both for controllability analysis and motion planning algorithms which are derived based on differential geometric approaches to mechanics, see e.g. [87].

Snake robots are a class of *simple* mechanical control systems, where the Lagrangian function $\mathcal{L} : T\mathcal{Q} \rightarrow \mathbb{R}$ is defined as the difference between the kinetic energy with respect to a Riemannian metric and the potential energy of the system. For geometric modelling of the system, we first write the kinetic energy of

the i -th link as the sum of the translational and rotational kinetic energy of the link

$$\mathcal{K}_i(q, v) = \frac{1}{2}m(\dot{x}_i^2 + \dot{y}_i^2) + \frac{1}{2}J\dot{\theta}_i^2 \quad (2.51)$$

where m and J denote the mass and moment of inertia of the link, respectively. Thus, the total kinetic energy for the N -link robot is defined as the sum of the kinetic energy of the individual links as

$$\mathcal{K}(q, v) = \sum_{i=1}^N \mathcal{K}_i(q, v) \quad (2.52)$$

The kinetic energy of the snake robot defines a Riemannian metric on the configuration space that can be written as

$$\mathbb{G}_{ij}(q) = \frac{\partial^2 \mathcal{K}(q, v)}{\partial v^i \partial v^j} \quad (2.53)$$

where \mathbb{G}_{ij} denotes the (i, j) component of the positive definite matrix-valued function \mathbb{G} . One can derive the Christoffel symbols of the second kind (see e.g. [87]) associated with the Riemannian metric of the robot in the form

$$\Gamma_{ij}^k(q) = \frac{1}{2} \mathbb{G}^{kl} \left(\frac{\partial \mathbb{G}_{il}}{\partial q^j} + \frac{\partial \mathbb{G}_{jl}}{\partial q^i} - \frac{\partial \mathbb{G}_{ij}}{\partial q^l} \right) \quad (2.54)$$

where $i, j, k, l \in \{1, \dots, N+2\}$, and \mathbb{G}^{kl} denotes the (k, l) component of \mathbb{G}^{-1} . Note that the summation convention is applied in (2.54). Using the Riemannian metric and the Christoffel symbols, it is possible to derive the equations of motion of the system on the configuration space with respect to (q, v) as

$$\dot{q}^i = v^i \quad (2.55)$$

$$\dot{v}^i = -\Gamma_{jk}^i v^j v^k - \mathbb{G}^{ik} \tau_{f,k} + \sum_{a=1}^{N-1} \mathbb{G}^{ik} F_k^a u^a \quad (2.56)$$

where $u = [u^1, \dots, u^{N-1}] \in \mathbb{R}^{N-1}$ denotes the vector of control inputs which take values in the control set $U = \mathbb{R}^{N-1}$. Moreover, $\tau_{f,k}$ denotes the k -th entry of the vector of friction forces τ_f given by (2.28) which take values in the cotangent bundle T^*Q (see e.g. [87]). Furthermore, $F = \{F^1, F^2, \dots, F^{N-1}\} = \{d\phi_1, d\phi_2, \dots, d\phi_{N-1}\}$ is the collection of the input covector fields of the system on Q . Since the codistribution generated by the $(N-1)$ input covector fields cannot span T^*Q , the snake robot is underactuated at any point of the configuration space. This underactuation reflects the fact that the input forces cannot set accelerations instantaneously in all directions of Q [87]. Note that since the robot

moves in the horizontal plane orthogonal to the direction of the gravitational field, there exists no gravitational term in the system dynamics (2.55)–(2.56). The above Poincaré representation is called the *natural representation*, and it has the property that the equations may also be written in the following second-order form [87]

$$\ddot{q}^i + \Gamma_{jk}^i \dot{q}^j \dot{q}^k + \mathbb{G}^{ik} \tau_{f,k} = \sum_{a=1}^{N-1} \mathbb{G}^{ik} F_k^a u^a \quad (2.57)$$

where $\ddot{q}^i = \dot{v}^i$, and where $\tau_{f,k}$ denotes the k -th element of the vector of friction forces given by (2.28).

2.3.3 Partial Feedback Linearization of the Geometric Model

For performing a model-based control design, we would like to write the model in the simplest possible form. Feedback linearization is a common technique that can simplify the model by cancellation of the nonlinear terms. However, due to the lack of direct independent control for some configuration variables of the system, this technique cannot be directly applied to the snake robot dynamics. Thus, in this case we use partial feedback linearization which linearizes the dynamics of the fully actuated configuration variables, i.e. dynamics of q_a . To this end, we separate the vector of the generalized coordinates q into two parts, in the form $q = [q_a, q_u]^T \in \mathbb{R}^{N+2}$, where $q_a \in \mathbb{R}^{N-1}$ and $q_u \in \mathbb{R}^3$ were defined in (2.2)–(2.3). Note that for clarity of presentation of the control design, henceforth we consider a local parametrization of the configuration space in an open subset of the Euclidean space. A partially feedback linearized Newton-Euler formulated model of snake robots was presented in [11], and we here extend it to a geometric model of the robot that is subject to parametric modelling uncertainties. This new model can be used for model-based robust control design for snake robots.

The dynamic model (2.57) is not suitable for partial feedback linearization. This is due to the presence of more than one input force in every scalar subsystem of (2.57). We note that this is the consequence of multiplying F by \mathbb{G}^{-1} in the right-hand side of (2.57). To obtain a suitable form for partial feedback linearization, we change the Christoffel symbols of the second kind (2.54), with those of the first kind using the relation

$$\Gamma_{jkl} = \sum_{i=1}^{N+2} \mathbb{G}_{li} \Gamma_{jk}^i \quad (2.58)$$

This changes (2.57) to the following form which is previously derived in [88],

where the model is called the *locomotion dynamics* of the snake robot given by

$$\sum_{j=1}^{N+2} \mathbb{G}_{kj} \ddot{q}^j + \sum_{i=1}^{N+2} \sum_{j=1}^{N+2} \Gamma_{ijk} \dot{q}^i \dot{q}^j = \tau_k - \tau_{f,k} \quad (2.59)$$

where $\tau = Fu = [\psi_1, \dots, \psi_{N-1}, 0, 0, 0] \in \mathbb{R}^{N+2}$ denotes the vector of input torques, in which ψ_i denotes the control torque provided by the actuator in the i -th robot joint. The dynamic model (2.59) is consistent with the well-known second-order Lagrangian equations of motion in the sense that in the left-hand side the first term is an acceleration related inertia term, the second term represents the Coriolis and centrifugal forces, and the right-hand side terms stand for external forces due to the controls and friction [88]. The dynamic model (2.59) is suitable for the aim of partial feedback linearization, since it can be separated into actuated and underactuated dynamical subsystems as

$$\sum_{j=1}^{N-1} \mathbb{G}_{mj}(q_a) \ddot{q}_a^j + \sum_{p=N}^{N+2} \mathbb{G}_{mp}(q_a) \ddot{q}_u^p + h_m(q, \dot{q}) = \psi_m \quad (2.60)$$

$$\sum_{j=1}^{N-1} \mathbb{G}_{kj}(q_a) \ddot{q}_a^j + \sum_{p=N}^{N+2} \mathbb{G}_{kp}(q_a) \ddot{q}_u^p + h_k(q, \dot{q}) = 0 \quad (2.61)$$

where $m \in \{1, \dots, N-1\}$, $k \in \{N, \dots, N+2\}$, and $h(q, \dot{q})$ contain all the contributions of the Coriolis, centrifugal and friction forces in (2.59).

Since we are going to use the dynamic model (2.59) in the subsequent chapters to develop a path following controller that is robust with respect to model uncertainties resulting from different friction properties, we will furthermore extend the locomotion dynamics model of [88], by adding these uncertainties to the model. In order to add parametric modelling uncertainties due to changes in the friction coefficients, we divide the vector function $h \in \mathbb{R}^{N+2}$ in accordance with $[q_a, q_u]^T \in \mathbb{R}^{N+2}$ into two parts as $h = [h_a, h_u]^T \in \mathbb{R}^{N+2}$, and present the following assumption.

Assumption 2.1 *The terms $h_a(q, \dot{q})$ and $h_u(q, \dot{q})$ are perturbed with multiplicative uncertainties in the form*

$$h_a = (I_1 + \Delta_1) \hat{h}_a \in \mathbb{R}^{N-1} \quad (2.62)$$

$$h_u = (I_2 + \Delta_2) \hat{h}_u \in \mathbb{R}^3 \quad (2.63)$$

where \hat{h}_a and \hat{h}_u are the estimations of the actual h_a and h_u , respectively. Moreover, $I_1 \in \mathbb{R}^{(N-1) \times (N-1)}$ and $I_2 \in \mathbb{R}^{3 \times 3}$ are identity matrices. Furthermore, $\Delta_1 \in \mathbb{R}^{(N-1) \times (N-1)}$ and $\Delta_2 \in \mathbb{R}^{3 \times 3}$, are measures of parametric modelling uncertainties due to the varying

friction properties on surfaces.

For partial feedback linearization, we divide the matrix representation of the Riemannian metric as

$$\mathbb{G} = \begin{bmatrix} \mathbb{G}_{aa} & \mathbb{G}_{ua} \\ \mathbb{G}_{au} & \mathbb{G}_{uu} \end{bmatrix} \in \mathbb{R}^{(N+2) \times (N+2)} \quad (2.64)$$

where $\mathbb{G}_{aa} \in \mathbb{R}^{(N-1) \times (N-1)}$, $\mathbb{G}_{au} \in \mathbb{R}^{(N-1) \times 3}$, $\mathbb{G}_{ua} \in \mathbb{R}^{3 \times (N-1)}$, and $\mathbb{G}_{uu} \in \mathbb{R}^{3 \times 3}$ denote the corresponding sub-matrices. We may now write (2.60)–(2.61) in the following matrix form:

$$\mathbb{G}_{aa}(q_a)\ddot{q}_a + \mathbb{G}_{au}(q_a)\ddot{q}_u + h_a(q, \dot{q}) = \psi \in \mathbb{R}^{N-1} \quad (2.65a)$$

$$\mathbb{G}_{ua}(q_a)\ddot{q}_a + \mathbb{G}_{uu}(q_a)\ddot{q}_u + h_u(q, \dot{q}) = 0_{3 \times 1} \in \mathbb{R}^3 \quad (2.65b)$$

where $\psi = [\psi_1, \dots, \psi_{N-1}]^T \in \mathbb{R}^{N-1}$. From (2.65b) we have

$$\ddot{q}_u = -\mathbb{G}_{uu}^{-1} \left(\mathbb{G}_{ua}\ddot{q}_a + (I_2 + \Delta_2)\hat{h}_u \right) \quad (2.66)$$

Inserting (2.66) into (2.65a) yields

$$(\mathbb{G}_{aa} - \mathbb{G}_{au}\mathbb{G}_{uu}^{-1}\mathbb{G}_{ua})\ddot{q}_a = \psi + \mathbb{G}_{au}\mathbb{G}_{uu}^{-1}(I_2 + \Delta_2)\hat{h}_u - (I_1 + \Delta_1)\hat{h}_a \quad (2.67)$$

To cancel out the nonlinear terms in the actuated subsystem of the equations of motion, we define the control inputs as

$$\psi = (\mathbb{G}_{aa} - \mathbb{G}_{au}\mathbb{G}_{uu}^{-1}\mathbb{G}_{ua})\vartheta - \mathbb{G}_{au}\mathbb{G}_{uu}^{-1}\hat{h}_u + \hat{h}_a \quad (2.68)$$

where $\vartheta = [\vartheta_1, \vartheta_2, \dots, \vartheta_{N-1}]^T \in \mathbb{R}^{N-1}$ is the new vector of control inputs. Finally, by inserting (2.68) into (2.67), we obtain the dynamics of the system in the following control-affine with drift form

$$\ddot{q}_a = \vartheta + G_a(q, \dot{q}) \in \mathbb{R}^{N-1} \quad (2.69)$$

$$\ddot{q}_u = f(q, \dot{q}) + G_u(q, \dot{q}) + g(q_a)\vartheta \in \mathbb{R}^3 \quad (2.70)$$

with

$$f = -\mathbb{G}_{uu}^{-1}\hat{h}_u = [f_{\theta_N}(q, \dot{q}), f_x(q, \dot{q}), f_y(q, \dot{q})]^T \in \mathbb{R}^3 \quad (2.71)$$

$$g_i = -\mathbb{G}_{uu}^{-1}\mathbb{G}_{ua} = [\beta_i(q_a), 0, 0]^T \in \mathbb{R}^3 \quad (2.72)$$

$$G_a = \Delta_1\hat{h}_1 - \mathbb{G}_{au}\mathbb{G}_{uu}^{-1}\Delta_2\hat{h}_u = [G_1(q, \dot{q}), \dots, G_{N-1}(q, \dot{q})]^T \in \mathbb{R}^{N-1} \quad (2.73)$$

$$G_u = -\mathbb{G}_{uu}^{-1}\Delta_2\hat{h}_u = [G_{\theta_N}(q, \dot{q}), G_x(q, \dot{q}), G_y(q, \dot{q})]^T \in \mathbb{R}^3 \quad (2.74)$$

where g_i denotes the i -th column of $g \in \mathbb{R}^{3 \times (N-1)}$ in which $\beta_i(q_a) : \mathbb{R}^{N-1} \rightarrow \mathbb{R}_{<0}$ is a smooth function. Moreover, in dynamic model (2.69)–(2.70), f_{θ_N} , f_x , and f_y denote the friction forces acting on θ_N , p_x , and p_y , respectively. (f_{θ_N} also contains Coriolis and centrifugal forces besides the friction forces). Furthermore, G_a and G_u denote the nonlinear terms due to parametric modelling uncertainties in the dynamics of the fully-actuated internal configuration variables and the dynamics of the underactuated position variables of the robot, respectively. In particular, the uncertain terms are upper bounded by some known positive-valued vector function which we denote by

$$\varrho(q, \dot{q}) = [\varrho_1(q, \dot{q}), \varrho_2(q, \dot{q}), \dots, \varrho_{N+2}(q, \dot{q})]^T \in \mathbb{R}_{>0}^{N+2} \quad (2.75)$$

This implies that the following inequality holds

$$\|G_i(q, \dot{q})\| \leq \varrho_i(q, \dot{q}) \quad (2.76)$$

for every $i \in \{1, \dots, N+2\}$.

The partially feedback linearized model (2.69)–(2.70) is suitable for analysis and robust control design for snake robot. For the aim model-based control design, we write (2.69)–(2.70) in a detailed form

$$\ddot{q}_a = \vartheta + G_a(q, \dot{q}) \in \mathbb{R}^{N-1} \quad (2.77)$$

$$\ddot{\theta}_N = f_{\theta_N}(q, \dot{q}) + \beta_i(q_a)\vartheta^i + G_{\theta_N}(q, \dot{q}) \in \mathbb{R} \quad (2.78)$$

$$\ddot{p}_x = f_x(q, \dot{q}) + G_x(q, \dot{q}) \in \mathbb{R} \quad (2.79)$$

$$\ddot{p}_y = f_y(q, \dot{q}) + G_y(q, \dot{q}) \in \mathbb{R} \quad (2.80)$$

where $i \in \{1, \dots, N-1\}$. The dynamic model (2.76)–(2.79) which contains the effects of nonlinear terms due to parametric modelling uncertainties, will be used for robust control design for the snake robot in the subsequent chapters.

2.4 Simplified Modelling Approach

In this section, we present a simplified model of the snake robot dynamics that can effectively be used for the model-based control design for snake robots. This model is previously presented in [11], where it is validated both through numerical simulations and real time experiments. Furthermore, in [11] it is shown that the fundamental properties of the simplified model such as stabilizability and controllability, are essentially the same as the more complex models presented in several previous works, see e.g. [1] and [105].

2.4.1 An Overview of the Simplified Modelling Approach

In this subsection, we briefly review the simplified modelling approach presented in [11]. Kinematic and dynamic models of snake robots are previously derived in several works (see e.g. [1], [84], [105]). All these models share the same property that they are very complex for analytical investigations. The derivation of the simplified model of snake robot dynamics in [11] is motivated by the attractive idea that these complex dynamic models contain some nonlinear dynamics that are not essential to the overall locomotion of the robot. Moreover, proper approximations of these nonlinear dynamics with simpler mathematical descriptions can significantly simplify the analysis and model-based control design for snake robots. In particular, it is seen in [11] that lateral undulation mainly consists of link displacements which are transversal to the direction of motion. Moreover, it is this transversal link displacement that induces the forward motion of snake robots, cf. Figure 2.3. The main idea behind the simplified model of the snake robot dynamics is to map the periodic body shape changes to forward propulsion, through mapping the rotational joint motion to translational link displacements, cf. Figure 2.2. Since the translational displacements are in general less complex than rotational motion, this mapping will simplify the resulting dynamic model of the robot.

2.4.2 Simplified Kinematics and Dynamics of the Snake Robot

In this subsection, we present the simplified kinematic and dynamic models of a snake robot without nonholonomic velocity constraints, which moves on a horizontal and flat surface. The configuration space \mathcal{Q} of the snake robot is a $(N + 2)$ -dimensional smooth manifold, locally diffeomorphic to an open subset of \mathbb{R}^{N+2} . Based on the illustrations of the robot in Figures 2.2–2.3, we choose the elements of the vector of the generalized coordinates, which represent \mathcal{Q} , as

$$q = [\phi_1, \dots, \phi_{N-1}, \theta, p_x, p_y]^T \in \mathbb{R}^{N+2} \quad (2.81)$$

where ϕ_i denotes the i -th joint coordinate, θ denotes the orientation, and (p_x, p_y) denotes the planar position of the CM of the robot. We denote the vector of the joint coordinates of the robot by $\phi = [\phi_1, \dots, \phi_{N-1}]^T \in \mathbb{R}^{N-1}$. The elements of ϕ are called the body shape variables, which define the internal configuration of the robot. The vector of the generalized velocities is defined as the time-derivative of (2.81) which we denote as

$$\dot{q} = [v_{\phi_1}, \dots, v_{\phi_{N-1}}, v_\theta, \dot{p}_x, \dot{p}_y]^T \in \mathbb{R}^{N+2} \quad (2.82)$$

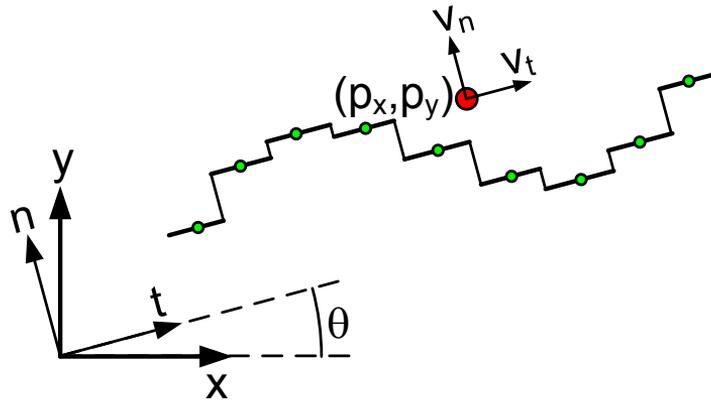


Figure 2.2: Illustration of two coordinate frames used in the simplified model. The $x - y$ frame is fixed, and the $t - n$ frame is always aligned with the snake robot.

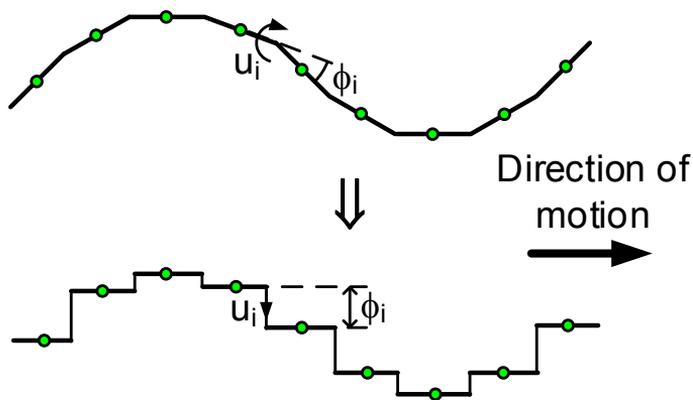


Figure 2.3: The snake robot is modelled using a series of prismatic joints which move the robot forward by translational displacements. u_i is the exerted torque or force in the i -th joint of the robot.

We denote the vector of the joint velocities by $v_\phi = [v_{\phi_1}, \dots, v_{\phi_{N-1}}]^T \in \mathbb{R}^{N-1}$. Since we aim to control the forward and normal velocities of the robot, we define the mapping between the inertial velocity of the CM of the robot and the velocity in the $t-n$ frame which is always aligned with the robot, cf. Figure 2.2, as

$$\dot{p}_x = v_t \cos(\theta) - v_n \sin(\theta) \quad (2.83)$$

$$\dot{p}_y = v_t \sin(\theta) + v_n \cos(\theta) \quad (2.84)$$

where $v_t \in \mathbb{R}$ and $v_n \in \mathbb{R}$ denote the tangential and normal components of the inertial velocity of the CM mapped into the direction of motion of the robot, respectively. The simplified dynamic model of the robot with respect to (q, \dot{q}) can be represented as [11]

$$\dot{\phi} = v_\phi \quad (2.85)$$

$$\dot{\theta} = v_\theta \quad (2.86)$$

$$\dot{p}_t = v_t \quad (2.87)$$

$$\dot{p}_n = v_n \quad (2.88)$$

$$\dot{p}_x = v_t \cos(\theta) - v_n \sin(\theta) \quad (2.89)$$

$$\dot{p}_y = v_t \sin(\theta) + v_n \cos(\theta) \quad (2.90)$$

$$\dot{v}_\phi = -\frac{c_n}{m} v_\phi + \frac{c_p}{m} v_t A D^T \phi + \frac{1}{m} D D^T u \quad (2.91)$$

$$\dot{v}_\theta = -\lambda_1 v_\theta + \frac{\lambda_2}{N-1} v_t \bar{e}^T \phi \quad (2.92)$$

$$\dot{v}_t = -\frac{c_t}{m} v_t + \frac{2c_p}{Nm} v_n \bar{e}^T \phi - \frac{c_p}{Nm} \phi^T A \bar{D} v_\phi \quad (2.93)$$

$$\dot{v}_n = -\frac{c_n}{m} v_n + \frac{2c_p}{Nm} v_t \bar{e}^T \phi \quad (2.94)$$

where $c_n \in \mathbb{R}_{>0}$ and $c_t \in \mathbb{R}_{>0}$ denote the viscous friction coefficients in the normal and tangential direction of motion of the links, respectively. Note that $c_n > c_t$ is a general controllability condition for planar snake robots, see [11], i.e. the friction in normal direction of motion of the link should in general be larger than the friction in tangential direction of motion of the link. Furthermore, $\lambda_1 \in \mathbb{R}_{>0}$ and $\lambda_2 \in \mathbb{R}_{>0}$ are used to describe the mapping from the rotational motion to the prismatic motion (see [11]). These coefficients are chosen such that the simplified model quantitatively behaves as the complex model of the snake robot. Furthermore, $c_p \in \mathbb{R}_{>0}$ is defined as

$$c_p = \frac{c_n - c_t}{2l}. \quad (2.95)$$

In order to linearize the dynamics of the fully-actuated degrees of freedom of the robot, i.e. the joint angles ϕ , we use the following change of the vector of the control inputs:

$$u = m (DD^T)^{-1} \left(\bar{u} + \frac{c_n}{m} v_\phi - \frac{c_p}{m} v_t AD^T \phi \right) \quad (2.96)$$

where $\bar{u} = [\bar{u}_1, \dots, \bar{u}_{N-1}]^T \in \mathbb{R}^{N-1}$ is the new set of control inputs. Inserting (2.96) into (2.91), transforms the dynamics of the joint angles into the following linear form

$$\dot{v}_\phi = \bar{u} \quad (2.97)$$

The linearized simplified dynamics in (2.85)–(2.97) will be used for locomotion control of the snake robot in Chapter 4.

2.4.3 Model Transformation

In this subsection, we present a coordinate transformation which can simplify the model-based locomotion control design for the snake robot. In particular, we note that the joint angles ϕ are present in both the dynamics of the angular velocity v_θ and the sideways velocity v_n . This coupling complicates the control design. In order to remove this coupling, we use the following coordinate transformation [11]

$$\bar{p}_y = p_y + \epsilon \sin(\theta) \quad (2.98)$$

$$\bar{v}_n = v_n + \epsilon v_\theta \quad (2.99)$$

where

$$\epsilon = -\frac{2(N-1)c_p}{Nm\lambda_2} \quad (2.100)$$

is a negative constant. This change of coordinates transforms the dynamics of the position of the CM of the system into

$$\dot{\bar{p}}_y = v_t \sin(\theta) + v_n \cos(\theta) \quad (2.101)$$

$$\dot{\bar{v}}_n = X v_\theta - Y \bar{v}_n \quad (2.102)$$

where

$$X = \epsilon \left(\frac{c_n}{m} - \lambda_1 \right) \quad (2.103)$$

$$Y = \frac{c_n}{m} \quad (2.104)$$

The joint angle coupling is removed from the dynamic model, and the resulting model is suitable for model-based locomotion control design which is the subject of the subsequent chapters.

Chapter Summary

- We presented the kinematic model for the snake robot locomotion in this chapter.
- In this chapter, we derived three models of the snake robot dynamics which will be used for model-based feedback control designs in the subsequent chapters.
- The first dynamic model of the snake robot that we derived in this chapter, presented complete characteristics of the snake robot locomotion on horizontal and flat surfaces. We derived this model using the Lagrangian approach for modelling mechanical systems, and we integrated the effects of external dissipative forces into this model using Jacobian matrices of the links.
- In this chapter, using the techniques of differential geometry, we derived a dynamic model of the snake robot locomotion which contained the effects of parametric modelling uncertainties due to varying friction properties on different surfaces. In particular, this model can effectively be used for robust model-based feedback control design for the snake robot.
- In the last part of this chapter we presented a simplified kinematic and dynamic model of the snake robot locomotion which was previously presented in [11]. In this model, the rotational joint motions are mapped into translational link displacements, and the resulting dynamics is simpler for model-based control design.

Body Shape and Orientation Control for Locomotion of Snake Robots

In this chapter, we investigate model-based feedback control approaches for body shape and orientation control of snake robots. In particular, based on the dynamic models of snake robots which were presented in Chapter 2, we design three different feedback control laws which control the body shape of the robot to a desired gait pattern. Furthermore, we use the parameters of this desired gait pattern in order to control the orientation of the robot in the plane. To this end, we design a dynamic compensator which controls the orientation of the robot to a reference angle defined by a path following guidance law. Moreover, using numerical simulations along with experimental results, we show that the body shape and orientation controllers can make the snake robot move towards a desired planar path and drive it along this path. Additionally, using an sliding mode control design, we show that the control goals can be achieved even in the presence of parametric modelling uncertainties, which arise due to the varying friction properties on different surfaces. It is noteworthy to mention that the orientation of the snake robot is an underactuated degree of freedom of the system. This is due to the fact that for one of the links, the head link of the robot, there is no direct and independent control input. Consequently, controlling the orientation of the snake robot is a challenging underactuated control problem. However, we present formal stability proofs for all the controllers in this chapter by using various tools from nonlinear and underactuated control theory. To our best knowledge, the application of model-based motion control approaches which rely on formal stability proofs for snake robots is very restricted in the previous literature.

This chapter, however, considers both body shape and orientation control of planar snake robots without nonholonomic velocity constraints. In particular, based on the Euler-Lagrange equations describing the dynamics and kinematics of the snake robot, we design a guidance-based feedback control strategy using a dynamic feedback control law. Guidance-based control strategies are in general based on defining a reference heading angle for the vehicle through a guidance law, and subsequently designing a tracking controller to track this angle [89]. The motivation for this guidance-based control strategy is to solve the path following problem for the snake robot. To our best knowledge, the only previous work which considers guidance-based path following control of planar snake robots without nonholonomic velocity constraints is presented in [44]. In [44], however, the control design is based on the simplified model of the snake robot which is valid for small joint angles. In the present work we carry out a model-based control design for the snake robot based on the more accurate model of the robot presented in Section 2.2–2.3, i.e. the complex model, which does not consider such simplifying assumptions.

Contributions of this chapter: The first contribution of this chapter is a body shape and orientation controller for the snake robot, which is designed by use of dynamic virtual holonomic constraints (VHC), which is a particularly useful concept for control of oscillations, (see e.g. [92]–[96]). Using this approach, we constrain the state evolution of the system to an appropriately defined submanifold of the configuration space, which is called the constraint manifold. This manifold is defined based on the specified geometric relations among the generalized coordinates of the system which are called VHC. The proposed feedback control law is designed to exponentially stabilize the constraint manifold, i.e. to enforce the VHC, which allows the convergence of the snake robot to a desired path. In this chapter we also employ design and analysis tools from finite-gain \mathcal{L} stability which enable us to analytically show that the body shape variables achieve perfect tracking, the orientation error converges to an arbitrarily small neighbourhood of the origin, and the states of the dynamic compensator which controls the underactuated head angle of the robot remain bounded.

The second contribution of this chapter is to design a robust guidance based controller for the snake robot using sliding mode techniques. In particular, we will employ the dynamic model subject to parametric modelling uncertainties which was presented in Chapter 2, in order to derive robust feedback control laws for the body shape and orientation of the snake robot. To our best knowledge, a robust guidance-based control strategy has never been proposed for snake robots before.

Organization of this chapter: This chapter is organized as follows. In Section 3.1, we state the control design objectives for the controllers derived throughout this chapter. In Section 3.2, we use the method of VHC in order to control the body shape and orientation of a planar snake robot. In Section 3.3, we design a robust body shape and orientation controller for the robot using sliding mode techniques. In Section 3.4, we design a body shape and orientation controller for the snake robot, and using an input-output stability analysis we show that the solutions of the dynamic compensator which controls the orientation of the robot remain uniformly bounded. In Section 3.5, we present simulation results along with experimental results which are obtained using a robotic snake in order to validate the control approaches of this chapter.

Publications: The results of this chapter are based on the journal paper [84], and the conference papers [77], [78], [79], and [80].

3.1 Control Design Objectives

In this chapter, we have two main control design objectives. The first objective is to control the internal configuration, i.e. the body shape, of the snake robot to provide a desired gait pattern. The second objective is to control the orientation of the robot, which is an underactuated degree of freedom.

To achieve these objectives, we first want to stabilize a lateral undulatory gait pattern for the shape variables of the robot. In particular, we define a tracking error variable for the i -th joint angle ϕ_i of the robot as

$$\tilde{\phi}_i = \phi_i - \phi_{\text{ref},i} \quad (3.1)$$

where $\phi_{\text{ref},i}$ denotes the reference trajectory for the i -th joint which provides the desired gait pattern (the desired gait pattern will be defined in the next section). We denote the vector of the joint tracking errors as $\tilde{\Phi} = [\tilde{\phi}_1, \dots, \tilde{\phi}_{N-1}]^T \in \mathbb{R}^{N-1}$. The first part of the control objective is then defined as asymptotic trajectory tracking for the joint angles of the robot such that

$$\lim_{t \rightarrow \infty} \|\tilde{\phi}_i(t)\| = 0 \quad (3.2)$$

for every $i \in \{1, \dots, N-1\}$.

In order to control the orientation of the robot, we then need to control the head angle of the robot. Note that according to (2.5), the head angle θ_N , together with the joint angles ϕ_i , define the orientation of the snake robot through (2.6). We define the tracking error variable for the head angle of the robot as

$$\tilde{\theta} = \theta_N - \theta_{\text{ref}} \quad (3.3)$$

where θ_{ref} denotes the reference head angle which will be defined later in this chapter. Since we only have $(N-1)$ independent control inputs, which will be used to control $\tilde{\Phi}$, stabilizing the passive degree of freedom (3.3) is challenging. The second part of the control objective is to asymptotically stabilize the head angle $\theta_N \rightarrow \theta_{\text{ref}}$ such that

$$\lim_{t \rightarrow \infty} \|\tilde{\theta}(t)\| = 0 \quad (3.4)$$

Moreover, by choosing a proper reference head angle defined by a path following guidance law, we will show that the snake robot converges to and follows a desired planar path. In particular, we define a desired straight path for the position of the CM (p_x, p_y) of the robot, as a smooth one-dimensional manifold $\mathcal{P} \subset \mathbb{R}^2$, with coordinates in the $x-y$ plane given by the pair (p_{xd}, p_{yd}) , which are parameterized by a scalar time-dependent variable $\Theta(t)$ as

$$\mathcal{P} = \{(p_{xd}(\Theta), p_{yd}(\Theta)) \in \mathbb{R}^2 : \Theta \geq 0\} \quad (3.5)$$

We define the vector of the path following error variables for the position of the CM of the robot as $\tilde{p} = [p_x(t) - p_{xd}(\Theta), p_y(t) - p_{yd}(\Theta)]^T \in \mathbb{R}^2$. Subsequently, the auxiliary part of the control objectives is defined as practical convergence, see e.g. [102], of the position of the CM of the robot to the desired path such that

$$\limsup_{t \rightarrow \infty} \|\tilde{p}(t)\| \leq \varepsilon_p \quad (3.6)$$

where $\varepsilon_p \in \mathbb{R}_{>0}$ is an arbitrary positive scalar. Moreover, we require that $\dot{\Theta}(t) \geq 0$ and $\lim_{t \rightarrow \infty} \Theta(t) = \infty$ (forward motion along the path), and boundedness of the states of the controlled system.

3.2 Virtual Holonomic Constraints Based Body Shape and Orientation Control of Snake Robots

3.2.1 The idea of Virtual Holonomic Constraints

Throughout this thesis, we frequently use the method of virtual holonomic constraints in order to solve the locomotion control problems for snake robots. virtual holonomic constraints are a constructive tool for feedback stabilization that have previously been used for motion control of mechanical systems in several works, see e.g. [92]–[96]. In this approach, we confine the state evolution of a mechanical system to a feedback invariant **constraint manifold**. This manifold is defined based on specified geometric relationships among the configuration variables of the system, which are called **virtual holonomic constraints (VHC)**. The VHC are defined such that the controlled system possesses the desired structural properties. These constraints are **virtual** because they do not arise from a physical connection between two variables but rather from the actions of a feedback controller [92]. It is noteworthy to mention that by using the VHC approach, we completely remove the time-dependence from the feedback loop. We do this by confining the time-evolution of the state variables of the system to state-dependent constraint functions, see e.g. [93]. Enforcing the specified VHC for the configuration variables of the system implies that in the kinematic level the system behaves similar to a system which has the same physical constraints, although the dynamic behaviour of these two systems are different. This difference is because of the extra power which the controller needs to inject to the closed-loop system in order to keep the VHC relations invariant, see [92]. Note that in case of *real* holonomic constraints which are typically introduced through physical connections between the variables of the system, the control system does not need to inject this extra power, i.e. physical constraints do not work on the system.

The fundamental theory of VHC are presented in e.g. [92], [93], and [95]. This method is used for stabilizing a curved line path for the position of the CM of a bicycle in [94]. In [96], VHC are used in order to control the oscillations of an underactuated double link pendulum. The mathematical definitions of VHC that we use in this chapter will be presented in the subsequent sections.

3.2.2 Path Following Control with Virtual Holonomic Constraints

The idea of VHC is particularly a useful concept for control of oscillations (see e.g. [92]–[96]). In this section, we show how this approach can be used to solve the path following control problem of snake robots. In particular, we show how by designing the joint reference trajectories in (3.1) using VHC, and combining this with VHC which are motivated by Line-of-Sight (LOS) guidance for the head angle in (3.3), we are able to solve the path following control problem, i.e. achieving (3.6). Our main motivation for using this approach is the fact that while performing the gait pattern lateral undulation which consists of fixed periodic body motions, all the solutions of the snake robot dynamics have inherent oscillatory behaviour. Moreover, we will show how this behaviour can be analytically and *constructively* controlled based on VHC. In particular, we use the word constructive in the sense that through the feedback action we shape the dynamics of the system such that it possesses the desired structural properties, i.e. exponential stability of an appropriately defined constraint manifold. To this end, we define a constraint manifold for the system, and we design the control input of the fully-actuated joint angles of the robot to exponentially stabilize the constraint manifold. The geometry of this manifold is defined based on specified geometric relations among the variables of the system which are called the constraint functions.

Trajectory Planning by Virtual Holonomic Constraints

VHC are specified through C^1 coordinate-dependent functions $\Phi_i : \mathcal{Q} \rightarrow \mathbb{R}$ which are called the **constraint functions**, in the relations of the form $\Phi_i(q) = 0$, which can be **enforced** through the feedback action. In particular, for the snake robot we define a vector-valued function

$$\Phi = [\phi_{\text{ref},1}, \phi_{\text{ref},2}, \dots, \phi_{\text{ref},N-1}, \theta_{\text{ref}}]^T \in \mathbb{R}^N \quad (3.7)$$

in which every entry, i.e. Φ_i where $i \in \{1, \dots, N\}$, defines one constraint function for the corresponding angular coordinate of the system.

To perform trajectory planning using VHC, we augment the state vector of the system with three new states that in the following will be used in the control

design. The introduction of these new variables to the state vector of the system, which will be used as constraint variables, is inspired by the notion of **dynamic VHC** [93], i.e. VHC which depend on the solutions of dynamic compensators. The idea is to make the VHC to depend on the variations of a dynamic parameter, which is used for controlling the system on the constraint manifold. The purpose of these additional states is explained below.

1. We introduce two new states $[\phi_o, \dot{\phi}_o]^T \in \mathbb{R}^2$ where the second order time-derivative of ϕ_o will be used as an additional control input that drives the snake robot towards the desired path by modifying the orientation of the robot in accordance with a reference angle defined by a path following guidance law.
2. In the previous section we defined the control objective for the joints and the head angle of the robot as a trajectory tracking problem. However, it is known that the holonomic constraints are coordinate-dependent equality constraints of the form $\Phi_i(q) = 0$, where $\Phi_i : \mathcal{Q} \rightarrow \mathbb{R}$ is a time-independent function [86]. Thus, we remove this explicit time-dependency from the reference joint trajectories by augmenting the state vector of the system with a new variable λ , with $\dot{\lambda} = 2\pi/T$ and $\lambda(0) = 0$, where T denotes the period of the cyclic motion of the shape variables of the robot, i.e. the gait pattern.

Subsequently, we denote the augmented coordinate vector of the system by

$$\hat{q} = [\phi_1, \dots, \phi_{N-1}, \theta_N, p_x, p_y, \phi_o, \lambda]^T \in \mathbb{R}^{N+4} \quad (3.8)$$

and the corresponding augmented state space by $T\hat{\mathcal{Q}}$.

Virtual Holonomic Constraints for the Joint Angles

A fundamental work in the area of snake robots was presented by Hirose [1]. In this work Hirose considers empirical studies of biological snakes to derive a mathematical approximation of the most common gait pattern among biological snakes, known as lateral undulation. In particular, the shape of a snake conducting lateral undulation can be described by a planar curve (the serpenoid curve) with coordinates in the $x - y$ plane along the curve at arc length s given by

$$x(s) = \int_0^s \cos(a \cos(bz) + cz) dz \quad (3.9)$$

$$y(s) = \int_0^s \sin(a \cos(bz) + cz) dz \quad (3.10)$$

where a , b , and c are positive scalars. Locomotion of a snake-like structure in accordance with the serpenoid curve, i.e. lateral undulation, is achieved if the joints of the robot move according to the reference joint trajectories in the form of a sinusoidal function with specified amplitude, frequency, and phase shift. In particular, using the foregoing defined new states, we define a constraint function for the i -th joint of the snake robot by

$$\Phi_i = \alpha \sin(\lambda + (i - 1)\delta) + \phi_o \quad (3.11)$$

where $i \in \{1, \dots, N-1\}$, α denotes the amplitude of the sinusoidal joint motion, and δ is a phase shift that is used to keep the joints *out of phase*. Moreover, ϕ_o is an offset value that is identical for all of the joints. It was illustrated in [11] how the offset value ϕ_o affects the orientation of the snake robot in the plane. Building further on this insight, we consider the second-order time-derivative of ϕ_o in the form of a dynamic compensator, which will be used to control the orientation of the robot. In particular, through this control term we modify the orientation of the robot in accordance with a reference orientation. This will be done by adding an offset angle to the reference trajectory of each joint. We will show that this will steer the position of the CM of the robot towards the desired path. The constraint function (3.11) is dynamic, since it depends on the solution of a dynamic compensator which will be defined to control the orientation of the robot.

Virtual Holonomic Constraint for the Head Link Angle

In this subsection, we define a constraint function for the head angle of the robot. In particular, we use a Line-of-Sight (LOS) guidance law as the reference angle for the head link. LOS guidance is a much-used method in marine control systems, (see e.g. [89]). In general, guidance-based control strategies are based on defining a reference heading angle for the vehicle through a guidance law, and designing a controller to track this angle [89]. Motivated by marine control literature, in [44] based on a simplified model of the snake robot, using cascade systems theory it was shown that if the heading angle of the snake robot was controlled to the LOS angle, then also the position of the CM of the robot would converge to the desired path. We will show that a similar guidance-based control strategy can successfully steer the robot towards the desired path. However, we perform the model-based feedback control design based on a more accurate model of the snake robot which does not contain the simplifying assumptions of [44] which are valid for small joint angles.

To define the guidance law, without loss of generality, we assign the global coordinate system such that the global x-axis is aligned with the desired path. Consequently, the position of the CM of the robot along the y-axis denoted by

p_y , defines the shortest distance between the robot and the desired path, often referred to as the *cross-track error*. In order to solve the path following problem, we use the LOS guidance law as a VHC, which defines the desired head angle as a function of the cross-track error as

$$\Phi_N = -\tan^{-1}\left(\frac{p_y}{\Delta}\right) \quad (3.12)$$

where $\Delta > 0$ is a design parameter known as the look-ahead-distance. The idea is that steering the head angle of the snake robot such that it is headed towards a point located at a distance Δ ahead of the robot along the desired path, will make the snake robot move towards the path and follow it.

Defining a Constraint Manifold

We collect all the defined constraint functions in the following vector-valued function

$$\Phi = \left[\alpha \sin(\lambda) + \phi_o, \dots, \alpha \sin(\lambda + (N-1)\delta) + \phi_o, \tan^{-1}\left(\frac{p_y}{\Delta}\right) \right]^T \in \mathbb{R}^N \quad (3.13)$$

For trajectory planning using VHC, we define the constraint manifold associated with the constraint functions (3.13) as

$$\Gamma = \left\{ (\hat{q}, \dot{\hat{q}}) \in T\hat{Q} : \phi_i = \Phi_i(\lambda, \phi_o), \theta_N = \Phi_N(p_y), \dot{\phi}_i = \dot{\lambda} \frac{\partial \Phi_i}{\partial \lambda} + \dot{\phi}_o \frac{\partial \Phi_i}{\partial \phi_o}, \dot{\theta}_N = \dot{p}_y \frac{\partial \Phi_N}{\partial p_y} \right\} \quad (3.14)$$

where $i \in \{1, \dots, N-1\}$. Since there are three different constraint variables, i.e. (λ, ϕ_o, p_y) , then the constraint manifold (3.14) is a 6-dimensional submanifold of \hat{Q} . The goal of the control input is to enforce the VHC (3.13), by making Γ exponentially stable for the closed-loop system, and thereby achieving the control objectives (3.2) and (3.4). To this end, we define the elements of a controlled output vector $y \in \mathbb{R}^N$ for the dynamical system (2.39)–(2.42) as the difference between the angular coordinates, i.e. $(\phi_1, \dots, \phi_{N-1}, \theta_N)$, and their corresponding constraint functions as

$$y = [\phi_1 - \Phi_1(\lambda, \phi_o), \dots, \phi_{N-1} - \Phi_{N-1}(\lambda, \phi_o), \theta_N - \Phi_N(p_y)]^T \in \mathbb{R}^N \quad (3.15)$$

We will achieve our control design objectives which we defined in the previous section, by designing the control inputs ϑ and $\ddot{\phi}_o$ such that $(y_i, \dot{y}_i) \rightarrow (0, 0)$ for all $i \in \{1, \dots, N\}$. To this end, we first need to ensure that the given relations in (3.13) are stabilizable, i.e. a suitable choice of feedback can stabilize the constraint manifold for the closed-loop system. For simplicity of notation, we

denote the following differentials:

$$d\Phi_i = \dot{\lambda} \frac{\partial \Phi_i}{\partial \lambda} + \dot{\phi}_o \frac{\partial \Phi_i}{\partial \phi_o} \quad (3.16a)$$

$$d\Phi_N = \dot{p}_y \frac{\partial \Phi_N}{\partial p_y} \quad (3.16b)$$

$$d^2\Phi_i = \ddot{\lambda} \frac{\partial \Phi_i}{\partial \lambda} + \dot{\lambda}^2 \frac{\partial^2 \Phi_i}{\partial \lambda^2} + \ddot{\phi}_o \frac{\partial \Phi_i}{\partial \phi_o} + \dot{\phi}_o^2 \frac{\partial^2 \Phi_i}{\partial \phi_o^2} \quad (3.16c)$$

$$d^2\Phi_N = \ddot{p}_y \frac{\partial \Phi_N}{\partial p_y} + \dot{p}_y^2 \frac{\partial^2 \Phi_N}{\partial p_y^2} \quad (3.16d)$$

where $i \in \{1, \dots, N-1\}$. The Lie derivative¹ of (3.15) along (2.39)–(2.42) is of the form

$$\dot{y} = \left[\dot{\phi}_1 - d\Phi_1, \dots, \dot{\phi}_{N-1} - d\Phi_{N-1}, \dot{\theta}_N - d\Phi_N \right]^T \in \mathbb{R}^N \quad (3.17)$$

which lacks the control inputs $(\vartheta, \ddot{\phi}_o)$. The Lie derivative of (3.17) along (2.39)–(2.42) is of the form

$$\ddot{y} = \left[\vartheta_1 - d^2\Phi_1, \dots, \vartheta_{N-1} - d^2\Phi_{N-1}, f_{\theta_N} + \beta_i \vartheta^i - d^2\Phi_N \right]^T \in \mathbb{R}^N \quad (3.18)$$

which contains the control inputs. Consequently, the controlled output vector (3.15) yields a well-defined vector relative degree $\{2, 2, \dots, 2\}$ everywhere on the configuration space. The VHC satisfying this vector relative degree condition are called regular, and regular constraints are always feasible [93], i.e. there exists a smooth feedback such that Γ is positively invariant for the closed-loop system. Furthermore, regular constraints in parametric form (3.13) are always stabilizable [93].

The well-defined vector relative degree $\{2, 2, \dots, 2\}$ on Γ implies that the system (2.39)–(2.42) with the controlled output function (3.15) is input-output feedback linearizable. Consequently, we can stabilize Γ with an input-output feedback linearizing controller.

Output Regulation via Input–Output Linearization

In this subsection, we will derive a control law for the joint angle dynamics (2.39) such that the constraint manifold (3.14) with the constraint functions defined in (3.13) is globally exponentially stable for the closed-loop system. In particular, we use input-output linearization to stabilize the constraint manifold. The exponential stability of the constraint manifold Γ implies that the control objectives (3.2) and (3.4) will be achieved. Furthermore, using numerical

¹For the dynamical system $\dot{x} = f(x) + g(x)u$, with output function $y = h(x)$, the Lie derivative of h along f is given by $L_f h(x) = \frac{\partial h(x)}{\partial x} f(x)$.

simulations and experiments on a robotic snake, we show that the exponential stability of Γ induces path following on the snake robot.

In order to stabilize $\Phi_i(\lambda, \phi_o)$ for the i -th joint such that $(y_i, \dot{y}_i) \rightarrow (0, 0)$ for all $i \in \{1, \dots, N-1\}$, we define an exponentially stabilizing joint control law. The second-order time-derivative of the i -th joint tracking error, i.e. the i -th entry of (3.18), is of the form

$$\ddot{y}_i = \vartheta_i - d^2\Phi_i \quad (3.19)$$

We define the control input for the i -th joint in (2.39) as

$$\vartheta_i = d^2\Phi_i - k_p y_i - k_d \dot{y}_i \quad (3.20)$$

where $k_p > 0$ and $k_d > 0$ are the joint controller gains. These gains are chosen similar for all the joints since the links have similar inertial parameters. Inserting (3.20) into (3.19) yields

$$\ddot{y}_i + k_d \dot{y}_i + k_p y_i = 0 \quad (3.21)$$

The tracking error dynamics of the i -th joint angle (3.21) clearly has a globally exponentially stable equilibrium at the origin $(y_i, \dot{y}_i) = (0, 0)$, which implies that every i -th control input (3.20) exponentially stabilizes the constraint manifold for the solutions of the dynamics of the i -th joint, and the control objective (3.2) is achieved.

In continue, we use the dynamic compensator $\ddot{\phi}_o$ in order to stabilize the solutions of the dynamics of the head angle (2.40) to the constraint manifold, such that $(y_N, \dot{y}_N) \rightarrow (0, 0)$. The head angle error corresponds to the N -th element of the controlled output vector (3.15), and its second-order time-derivative (i.e. the head angle error dynamics) is given by

$$\ddot{y}_N = f_{\theta_N} + \beta_i \vartheta^i - d^2\Phi_N \quad (3.22)$$

Inserting ϑ_i from (3.20) into (3.22) we can obtain the closed-loop dynamics of the head angle, which gives

$$\ddot{y}_N = f_{\theta_N} + \sum_{i=1}^{N-1} \beta_i (d^2\Phi_i - k_p y_i - k_d \dot{y}_i) - d^2\Phi_N \quad (3.23)$$

By considering the notation which was introduced in (3.16c), we can write (3.23) in the following equivalent form

$$\ddot{y}_N = f_{\theta_N} + \sum_{i=1}^{N-1} \beta_i \left(\ddot{\lambda} \frac{\partial \Phi_i}{\partial \lambda} + \dot{\lambda}^2 \frac{\partial^2 \Phi_i}{\partial \lambda^2} + \ddot{\phi}_o \frac{\partial \Phi_i}{\partial \phi_o} + \dot{\phi}_o^2 \frac{\partial^2 \Phi_i}{\partial \phi_o^2} - k_p y_i - k_d \dot{y}_i \right) - d^2\Phi_N \quad (3.24)$$

For simplicity of notation, we denote the constraint function for the i -th joint angle of the robot by $\Phi_i = S_i + \phi_o$, where $S_i = \alpha \sin(\lambda + (i-1)\delta)$. Subsequently, based on the specified constraint functions in (3.13), i.e. since $\ddot{\lambda} = 0$ and $\frac{\partial^2 \phi_i}{\partial \phi_o^2} = 0$, we can write (3.24) as

$$\ddot{y}_N = f_{\theta_N} + \sum_{i=1}^{N-1} \beta_i \left(-\dot{\lambda}^2 S_i + \ddot{\phi}_o - k_p y_i - k_d \dot{y}_i \right) - d^2 \Phi_N \quad (3.25)$$

In order to stabilize the constraint function $\Phi_N(p_y)$ for the head angle, we define the second-order time-derivative of the augmented coordinate ϕ_o in the form of a dynamic compensator as

$$\ddot{\phi}_o = \left(\sum_{i=1}^{N-1} \beta_i \right)^{-1} \left(\sum_{i=1}^{N-1} \beta_i \left(\dot{\lambda}^2 S_i + k_p y_i + k_d \dot{y}_i \right) + d^2 \Phi_N - f_{\theta_N} - k_{p,\theta_N} y_N - k_{d,\theta_N} \dot{y}_N \right) \quad (3.26)$$

where $k_{p,\theta_N} > 0$ and $k_{d,\theta_N} > 0$ are the head angle controller gains. Notice that since β_i is negative-valued in any configuration, (3.26) is globally well-defined. For the control design method presented in this section, through numerical simulations and experimental results we show that the states of the dynamic compensator (3.26), i.e. $(\phi_o, \dot{\phi}_o)$, remain bounded. However, in Section 3.4, we use an input-output stability analysis in order to show the boundedness of $(\phi_o, \dot{\phi}_o)$. By inserting (3.26) into (3.25), the error dynamics of the head angle takes the form

$$\ddot{y}_N + k_{d,\theta_N} \dot{y}_N + k_{p,\theta_N} y_N = 0 \quad (3.27)$$

which clearly has a globally exponentially stable equilibrium at the origin, i.e. $(y_N, \dot{y}_N) = (0, 0)$. Consequently, we have that $(y_N, \dot{y}_N) \rightarrow (0, 0)$ from any initial condition, and the control objective (3.4) will be achieved.

Finally, we conjecture that while the output trajectories of the system (2.39)–(2.42) are evolving on the constraint manifold (3.14), the internal dynamics given by (2.41)–(2.42), which has the form

$$\ddot{p}_x = f_x(\Phi, p_x, p_y, d\Phi, \dot{p}_x, \dot{p}_y) \quad (3.28)$$

$$\ddot{p}_y = f_y(\Phi, p_x, p_y, d\Phi, \dot{p}_x, \dot{p}_y) \quad (3.29)$$

converge to and follow the desired planar path. Analytically investigating the convergence of the snake robot position to the desired path is a topic of future chapters. As a preliminary support of this conjecture, in this chapter we provide simulation and experimental results which show that the snake robot successfully converges to and follows the desired path using the guidance-based control strategy presented in this section.

3.3 Robust Locomotion Control of Snake Robots Using Sliding Mode Techniques

In this section, we extend the results of the previous section to the dynamic model of the snake robot (2.77)–(2.80) which is subject to parametric modelling uncertainties. In particular, we show how sliding mode control techniques can be used in order to control the locomotion of the snake robots which move on various surfaces. Our main motivation for using this technique for motion control of snake robots is the fact that these robots often move on different surfaces with different friction properties. Accordingly, the necessity of developing control methods for snake robots which are robust with respect to changes in the environment of the robot is well-justified. To design the sliding mode controller, we use the dynamic model (2.77)–(2.80), which included the effects of parametric modelling uncertainties due to varying friction properties on the motion of the robot. Using this dynamic model, we propose an analytical solution to the robust path following control problem through the following two steps.

1. In the first step, we use sliding mode techniques to design a robust tracking controller for the joints of the robot to track a desired gait pattern.
2. In the second step, using robust dynamic compensation, we stabilize an appropriately defined sliding manifold for the underactuated head angle of the robot, thereby achieving the convergence of the robot to the desired straight path.

The control objectives for this section are same as the previous section which are given in (3.2), (3.4), and (3.6). In particular, we aim to control the body shape of the robot to a desired gait pattern, to control the head angle of the robot to a reference head angle defined by the LOS path following guidance law, and to make the CM of the robot to converge to and follow a desired geometric path. However, in the section we use sliding mode techniques to achieve these goals in the presence of nonlinear terms arising due to the parametric modelling uncertainties. At the end of this chapter we present simulation results for the sliding mode controller which validate the theoretical approach.

3.3.1 Sliding Mode Tracking Control of the Joint Angles

In this section, we achieve the control objective (3.2) by defining the reference trajectories for the joints of the snake robot (i.e. the gait pattern), and by using sliding mode techniques to design a robust joint angle tracking control law.

To define the desired gait pattern, we use the reference joint trajectories (3.11), which induce lateral undulatory locomotion on the robot. Moreover, motivated by [11], where it is shown how ϕ_o can be used to steer the heading angle of the robot, we use the second-order time-derivative of ϕ_o as an additional guidance control term for the underactuated head angle of the robot.

The powerful feature of sliding mode control is its robustness with respect to model uncertainties. For snake robots, this robustness is useful for performing robust control tasks on surfaces with varying friction properties. To design a robust tracking controller for the joints of the robot, we first define the vector of the reference joint trajectories as

$$\Phi(t) = [\phi_{\text{ref},1}, \phi_{\text{ref},2}, \dots, \phi_{\text{ref},N-1}]^T \in \mathbb{R}^{N-1} \quad (3.30)$$

Thus, the vector of the joint angle tracking errors is defined as

$$\tilde{\Phi}(t) = [\tilde{\phi}_1, \tilde{\phi}_2, \dots, \tilde{\phi}_{N-1}]^T \in \mathbb{R}^{N-1} \quad (3.31)$$

where $\tilde{\phi}_i = \phi_i - \phi_{\text{ref},i}$ denotes the i -th joint angle tracking error variable. Based on the dynamic model (2.77)–(2.80), the dynamics of the joint angles, i.e. $q_a = (\phi_1, \phi_2, \dots, \phi_{N-1})$ of the robot which is subject to parametric modelling uncertainties is given by

$$\ddot{q}_a = \vartheta + G_a(q, \dot{q}) \in \mathbb{R}^{N-1} \quad (3.32)$$

where $\vartheta = [\vartheta_1, \dots, \vartheta_{N-1}]^T \in \mathbb{R}^{N-1}$ denotes the vector of the joint control inputs and the vector $G_a(q, \dot{q}) \in \mathbb{R}^{N-1}$ denotes the effects of parametric modelling uncertainties on the dynamics of the joint angles. We define the sliding mode variable for the joint angles of the robot as

$$s = \dot{\tilde{\Phi}} + K\tilde{\Phi} \in \mathbb{R}^{N-1} \quad (3.33)$$

where $K = \text{diag}\{k_i\}_{i=1}^{N-1} \in \mathbb{R}^{(N-1) \times (N-1)}$ is a diagonal matrix of positive constant gains. The time-derivative of the sliding mode variable (3.33) is given by

$$\dot{s} = \ddot{\tilde{\Phi}} + K\dot{\tilde{\Phi}} \in \mathbb{R}^{N-1} \quad (3.34)$$

The control objective in (3.2) is achieved by stabilizing the sliding manifold $s = 0_{N-1}$ in finite time, and remaining on the manifold for all future time. To this end, we select a Lyapunov function candidate for (3.34) as

$$V = \frac{1}{2} s^T s \quad (3.35)$$

Taking the time-derivative of V along the solutions of (3.34) gives

$$\begin{aligned}\dot{V} &= s^T \dot{s} = s^T \left(\ddot{\Phi} + K \dot{\Phi} \right) \\ &= s^T \left(\ddot{q}_a - \ddot{\Phi} + K \dot{\Phi} \right) = s^T \left([\vartheta + G_a(q, \dot{q})] - \ddot{\Phi} + K \dot{\Phi} \right)\end{aligned}\quad (3.36)$$

The typical structure of a robust controller is composed of a nominal part similar to a feedback linearizing or inverse control law, and of an additional term aimed at dealing with model uncertainty [97]. Consequently, we take the joint angle tracking control law as

$$\vartheta = \vartheta_{\text{nom}} + \vartheta_{\text{add}} \quad (3.37)$$

where

$$\vartheta_{\text{nom}} = \left(\ddot{\Phi} - K \dot{\Phi} \right) \in \mathbb{R}^{N-1} \quad (3.38)$$

$$\vartheta_{\text{add}} = -\gamma \operatorname{sgn}(s) \in \mathbb{R}^{N-1} \quad (3.39)$$

and where $\gamma = \operatorname{diag}\{\gamma_i\}_{i=1}^{N-1} \in \mathbb{R}^{(N-1) \times (N-1)}$ is a diagonal matrix of positive constants. Moreover, we define $\operatorname{sgn}(s) = [\operatorname{sgn}(s_1), \dots, \operatorname{sgn}(s_{N-1})]^T \in \mathbb{R}^{N-1}$. Substituting (3.37) into (3.36) yields

$$\dot{V} = s^T [-\gamma \operatorname{sgn}(s) + G_a(q, \dot{q})] \quad (3.40)$$

We take $\gamma_i = \gamma_0 + \varrho_i(q, \dot{q})$, where $\gamma_0 > 0$ is a constant, and $\varrho_i(q, \dot{q})$ denotes the upper-bound on the i -th nonlinear term arising due to parametric modelling uncertainties. We note that the i -th term of \dot{V} denoted by \dot{V}_i is of the form

$$\begin{aligned}\dot{V}_i &= s_i [-\gamma_i \operatorname{sgn}(s_i) + G_i(q, \dot{q})] \leq \\ &\quad - [\gamma_0 + \varrho_i(q, \dot{q})] s_i \operatorname{sgn}(s_i) + |s_i| \varrho_i(q, \dot{q}) \leq -\gamma_0 |s_i|\end{aligned}\quad (3.41)$$

Consequently², $\dot{V} = -\sum_{i=1}^{N-1} \gamma_0 |s_i|$ is negative-definite. This implies that the sliding manifold $s = 0_{N-1}$ is a positively invariant set for (3.34). The positive invariance property of $s = 0_{N-1}$ implies that once the solutions of (3.34) reach the sliding manifold, they cannot leave it and the motion of the joints will be restricted to this manifold. To show that the solutions of (3.34) reach the sliding manifold in finite time we use the *comparison lemma*, see e.g. [99]. In particular, we take $W = \sum_{i=1}^{N-1} \sqrt{2V_i} = \sum_{i=1}^{N-1} |s_i|$. The upper right derivative $D^+W = \sum_{i=1}^{N-1} \dot{V}_i \frac{1}{\sqrt{2V_i}}$ satisfies the differential inequality

$$D^+W \leq -(N-1)\gamma_0 \quad (3.42)$$

²Note that $s_i \operatorname{sgn}(s_i) = |s_i|$.

Using the comparison lemma we have that

$$W(s(t)) \leq W(s(0)) - (N-1)\gamma_0 \quad (3.43)$$

which implies that $W = \sum_{i=1}^{N-1} \sqrt{2V_i} = 0$ must reach $V_i = 0$ in finite time. Accordingly, the solutions of (3.34) starting off the positively invariant manifold $s = 0_{N-1}$ will reach it in finite time.

We summarize the results of the foregoing arguments in the following theorem.

Theorem 3.1 *With the robust joint tracking control law (3.37), the solutions of (3.34) reach the sliding manifold $s = 0_{N-1}$ in finite time. The positive invariance of this manifold, which is shown by (3.41), implies that these solutions will remain on the sliding manifold for all future time. Moreover, exponential stability of the origin of the joint tracking error dynamics $\dot{\tilde{\Phi}} = -K\tilde{\Phi}$ on the sliding manifold, implies that the joint tracking errors converge exponentially to zero during the sliding phase, see [99], and the control objective (3.2) will be achieved.*

Remark 3.1 *The discontinuous $\text{sgn}(\cdot)$ function in the sliding mode controller may lead to issues related to existence and uniqueness of solutions, issues related to the validity of the Lyapunov analysis, and chattering (see [99]). To avoid these issues, a common approach is to approximate the (discontinuous) $\text{sgn}(s)$ function with a high slope (continuous) saturation function $\text{sat}(s/\epsilon)$. However, with this approximation the best we can achieve is ultimate boundedness of the tracking errors with an ultimate bound that can be controlled by the design parameter ϵ [99]. Also note that in the case that the nonlinear terms arising due to parametric modelling uncertainties $G_a(q, \dot{q})$ are non-vanishing in the origin $(\tilde{\Phi}, \dot{\tilde{\Phi}}) = (0_{N-1}, 0_{N-1})$, then the origin is not an equilibrium point that can be made asymptotically stable.*

3.3.2 Underactuated Tracking Control via Sliding Mode Design

In this section, we design a head angle controller for the snake robot in order to achieve the second control objective (3.4). In particular, we analytically show that robust tracking control of the head angle can be achieved by using ϕ_o as an additional control term for the underactuated head angle of the robot. To this end, we use $\ddot{\phi}_o$ as a dynamic compensator which adds a similar extra offset angle to the sinusoidal parts of the reference joint trajectories such that the position of the CM of the robot converges to the desired path. Furthermore, we use sliding mode techniques to design the dynamic compensator in a way that

this convergence will be achieved even in the presence of parametric modelling uncertainties which are due to friction changes.

Sliding Mode Control of the Head Angle

With the gait pattern lateral undulation (3.11), the reference joint trajectories are composed of non-identical sinusoidal parts, and an identical offset term. Let us denote $S_i = \alpha \sin(\omega t + (i - 1)\delta)$. Thus, the reference trajectory of the i -th joint can be denoted as

$$\phi_{\text{ref},i} = S_i + \phi_o \quad (3.44)$$

In (2.78), the head angle dynamics subject to nonlinear terms arising due to the parametric modelling uncertainties was given by

$$\ddot{\theta}_N = f_{\theta_N}(q, \dot{q}) + \sum_{i=1}^{N-1} \beta_i(q_a) \vartheta_i + G_{\theta_N}(q, \dot{q}) \quad (3.45)$$

The head angle dynamics in closed-loop form can be obtained by substituting the joint control law (3.37) into the head angle dynamics (3.45), which gives (arguments are excluded for notational convenience)

$$\begin{aligned} \ddot{\theta}_N = f_{\theta_N} + G_{\theta_N} + \sum_{i=1}^{N-1} \beta_i \vartheta_i = \\ f_{\theta_N} + G_{\theta_N} + \sum_{i=1}^{N-1} \beta_i \left(\ddot{S}_i + k_i \dot{S}_i - k_i \dot{\phi}_i + \ddot{\phi}_o + k_i \dot{\phi}_o - \gamma_i \text{sgn}(s_i) \right) \end{aligned} \quad (3.46)$$

The goal of the control design is to make the head angle exponentially converge to a reference head angle. In the following, we show that this convergence can be achieved by using the additional control input $\ddot{\phi}_o$. To this end, we first define the error variable for the head angle of the robot as $\tilde{\theta} = \theta_N - \theta_{\text{ref}}$, where θ_{ref} denotes the reference head angle of the robot which was defined by a LOS guidance law in (3.12). Moreover, we define the sliding mode variable for the head angle in the form

$$s_\theta = \dot{\tilde{\theta}} + \Lambda \tilde{\theta} \quad (3.47)$$

where $\Lambda > 0$ is a constant gain. The time-derivative of the sliding mode variable (3.47) is given by

$$\dot{s}_\theta = \ddot{\tilde{\theta}} + \Lambda \dot{\tilde{\theta}} \quad (3.48)$$

To stabilize the sliding manifold $s_\theta = 0$, we select a Lyapunov function candidate for (3.48) as

$$V_\theta = \frac{1}{2} s_\theta^2 \quad (3.49)$$

The time-derivative of V_θ along the solutions of (3.48) gives

$$\dot{V}_\theta = s_\theta \dot{s}_\theta = s_\theta \left(\ddot{\theta} + \Lambda \dot{\theta} \right) = s_\theta \left(\ddot{\theta}_N - \ddot{\theta}_{\text{ref}} + \Lambda \dot{\theta} \right) \quad (3.50)$$

By inserting $\ddot{\theta}_N$ from (3.46) into (3.50), we obtain

$$\dot{V}_\theta = s_\theta \left[f_{\theta_N} + G_{\theta_N} + \sum_{i=1}^{N-1} \beta_i \left(\ddot{S}_i + k_i \dot{S}_i - k_i \dot{\phi}_i + \ddot{\phi}_o + k_i \dot{\phi}_o - \gamma_i \text{sgn}(s_i) \right) - \ddot{\theta}_{\text{ref}} + \Lambda \dot{\theta} \right] \quad (3.51)$$

In order to stabilize the sliding manifold $s_\theta = 0$ we define the dynamic compensator as

$$\ddot{\phi}_o = \frac{1}{\sum_{i=1}^{N-1} \beta_i} \left[-f_{\theta_N} - \sum_{i=1}^{N-1} \beta_i \left(\ddot{S}_i + k_i \dot{S}_i - k_i \dot{\phi}_i + k_i \dot{\phi}_o + \gamma_i \text{sgn}(s_i) \right) + \ddot{\theta}_{\text{ref}} - \Lambda \dot{\theta} - \gamma_\theta \text{sgn}(s_\theta) \right] \quad (3.52)$$

where $\gamma_\theta > 0$ is a constant gain. Since β_i is a negative-valued function for all $i \in \{1, \dots, N-1\}$, (3.52) is globally well-defined. For the controller of this section, through numerical simulations we show that the states of the dynamic compensator (3.52), i.e. $(\phi_o, \dot{\phi}_o)$, remain bounded, however, a formal proof of this boundedness will be presented in the following section. We define $\gamma_\theta = \gamma_{\theta_0} + \varrho_N(q, \dot{q})$ for some $\gamma_{\theta_0} > 0$. Note that $\varrho_N(q, \dot{q}) \in \mathbb{R}_{>0}$ denotes the upper-bound on $G_{\theta_N}(q, \dot{q}) \in \mathbb{R}$. Inserting (3.52) into (3.51) yields

$$\dot{V}_\theta = s_\theta \left(-\gamma_\theta \text{sgn}(s_\theta) + G_{\theta_N} \right) \leq -(\gamma_{\theta_0} + \varrho_N(q, \dot{q})) s_\theta \text{sgn}(s_\theta) + |s_\theta| \varrho_N(q, \dot{q}) \leq -\gamma_{\theta_0} |s_\theta| \quad (3.53)$$

The negative-definiteness of \dot{V}_θ implies the positive invariance of the sliding manifold $s_\theta = 0$ for the dynamical system (3.48). This implies that once solutions of (3.48) reach $s_\theta = 0$, they will remain there for all future time.

The results of the foregoing arguments is summarized in the following theorem.

Theorem 3.2 *Inequality (3.53) along with the comparison lemma, imply that all solutions of (3.48) starting off the manifold $s_\theta = 0$, will reach it in finite time, and solutions on the manifold cannot leave it. Moreover, the exponential stability of the origin of the head angle error dynamics $\dot{\theta} = -\Lambda \tilde{\theta}$ on the sliding manifold, implies that during the sliding phase, the head angle error converges exponentially to zero, and the control design objective (3.4) will be achieved.*

We conjecture that the proposed guidance-based sliding mode path following control strategy, steers the position of the CM of the robot towards the desired path, and drives it along the path even in the presence of parametric modelling uncertainties. In Section 3.5 we present simulation results which support this conjecture. However, a formal proof of this conjecture remains a topic of next chapters.

3.4 Stability Analysis of the Controlled Systems

So far in this chapter, based on the complex dynamic model of the snake robot locomotion, we have derived two body shape and orientation controllers which exponentially stabilize a desired gait pattern and a reference orientation angle for the robot. However, the major drawback of the proposed controllers was the absence of an analytical proof for the boundedness of the solutions of the dynamic compensator which was used in order to control the orientation of the robot. In this section, we propose a controller which guarantees the boundedness of the solutions of the controlled system. In particular, using an input-output stability analysis we will show that the solutions of the dynamic compensator which controls the head angle of the robot remain uniformly bounded.

Remark 3.2 *In Section 3.2, using the method of VHC we derived a body shape and orientation controller which led to a strong stability result. In particular, by (3.27) we showed that the orientation controller (3.26) globally exponentially stabilizes $\theta_N \rightarrow \theta_{ref}$. However, this strong stability result for the orientation was obtained using the dynamic compensator (3.26) which has a drawback. The drawback is that there is no term in (3.26) to control the solutions of the dynamic compensator so that it guarantees the uniform boundedness of these solutions. In this section, however, we change the control design in the way that we add a term to the dynamic compensator which ensures the boundedness of its solutions. However, this term acts as a non-vanishing perturbing term for the orientation error dynamics, and consequently we will obtain a weaker stability result for the orientation angle. Instead, we guarantee that the solutions of the controlled system remain uniformly bounded.*

To perform control design, we first present two realistic assumptions regarding the dynamics of the robot, and then we state a new control design objective for the head angle of the robot. Subsequently, we design a new guidance-based control strategy for the snake robot which ensures the boundedness of the solutions of the closed-loop system. In particular, we employ design and analysis tools from finite-gain \mathcal{L} stability which enable us to analytically show that the body shape variables achieve perfect tracking, the orientation error converges

to an arbitrarily small neighbourhood of the origin, and the states of the dynamic compensator remain bounded.

Assumption 3.1 We assume that $\sum_{i=1}^{N-1} \beta_i$ is a negative constant.

Remark 3.2 Both through numerical simulations and experiments, see e.g. Figures 3.2 and 3.15, it can be verified that $\beta_i(q_a)$ is negative-valued for all $i \in \{1, \dots, N-1\}$ in any configuration of the robot. This follows from the uniform positive-definiteness of the inertia matrix of the robot. Moreover, $\sum_{i=1}^{N-1} \beta_i$ shows oscillations with a very small magnitude about a negative constant. This negative constant depends on the inertial parameters of the robot.

We denote the vector of friction forces acting on the underactuated degrees of freedom of the robot by $f = [f_{\theta_N}, f_x, f_y] \in \mathbb{R}^3$, which is used in the following assumption.

Assumption 3.2 Throughout this section we assume that $\sup_{t \geq 0} \|f(q(t), \dot{q}(t))\| < \infty$.

Remark 3.3 Assumption 3.2 is a realistic assumption, since snake robots often move very slowly, and the external forces due to friction acting on the system will be bounded.

New Control Design Objective for the Head Angle

In order to control the orientation of the robot, we then need to control the head angle of the robot. Please note that the head angle θ_N , together with the joint angles that give the orientation of the links, $(\theta_1, \dots, \theta_{N-1})$, give the orientation of the snake robot through (2.6). We define the tracking error variable for the head angle of the robot as

$$\tilde{\theta} = \theta_N - \theta_{\text{ref}} \quad (3.54)$$

where θ_{ref} denotes the reference head angle defined by (3.12). Since we only have $(N-1)$ independent control inputs, which will be used to control the joint angles, stabilizing the passive degree of freedom (3.54) is challenging. We aim to achieve practical stability³ (see e.g. [18]) for this degree of freedom. Thus, the new control objective for the head angle of the robot is to practically stabilize $\theta_N \rightarrow \theta_{\text{ref}}$ such that

$$\limsup_{t \rightarrow \infty} \|\tilde{\theta}(t)\| \leq \epsilon_\theta \quad (3.55)$$

³Practical stability means that we can drive the error (3.54) to any arbitrary small neighbourhood of zero.

where $\epsilon_\theta \in \mathbb{R}_{>0}$ is any positive constant. We also require that all the states of the controlled system, i.e. including the solutions $(\phi_o, \dot{\phi}_o)$ of the dynamic compensator, remain bounded.

3.4.1 Guidance-based Control Strategy for the Snake Robot

In the following, we will present a guidance-based control strategy in order to control the CM of the snake robot to a desired path through controlling the head angle of the robot to the reference angle defined by the LOS path following guidance law. Furthermore, using an input-output stability analysis, we show the uniform boundedness of the solutions of the controlled system. Moreover, we use a perturbation analysis to show that the orientation error is ultimately bounded by an arbitrarily small bound containing the origin.

Body Shape Control

In this subsection, we define a dynamic tracking control law for the joint angles of the robot. To this end, we define the tracking error for the i -th joint as

$$\tilde{\phi}_i = \phi_i - \phi_{\text{ref},i} \quad (3.56)$$

where the reference i -joint trajectory $\phi_{\text{ref},i}$ is defined based on the work of Hirose [1] as

$$\phi_{\text{ref},i} = \alpha \sin(\omega t + (i-1)\delta) + \phi_o \quad (3.57)$$

For the i -th joint we define the tracking control law

$$\vartheta_i = \ddot{\phi}_{\text{ref},i} - k_d \dot{\tilde{\phi}}_i - k_p \tilde{\phi}_i \quad (3.58)$$

where $k_p, k_d > 0$ denote the joint controller gains. By inserting (3.58) into (2.39)⁴, the tracking error dynamics for the i -th joint takes the following exponentially stable form

$$\ddot{\tilde{\phi}}_i + k_d \dot{\tilde{\phi}}_i + k_p \tilde{\phi}_i = 0 \quad (3.59)$$

which implies that the joint tracking errors converges exponentially to zero from any initial conditions.

⁴Note that the partially feedback linearized dynamics of the i -th joint angle in (2.39) was presented as $\ddot{\phi}_i = \vartheta_i$.

Head Angle Control

For the head angle control design, we will use the idea that we presented before; using $\ddot{\phi}_o$ in the form of a dynamic compensator which reorients the robot in the plane through adding an appropriately defined offset angle to each link of the robot. In particular, we design this term to practically stabilize a reference head angle for the robot, and thereby achieve the control objective (3.55).

For simplicity, by taking $S_i = \alpha \sin(\omega t + (i - 1)\delta)$, the reference trajectory of the i -th joint can be denoted as

$$\phi_{\text{ref},i} = S_i + \phi_o \quad (3.60)$$

The closed-loop dynamics of the head angle can be obtained by inserting the control law (3.58) into the head angle dynamics (2.40) which gives (arguments are excluded for notational convenience)

$$\begin{aligned} \ddot{\theta}_N &= f_{\theta_N} + \sum_{i=1}^{N-1} \beta_i \vartheta_i = \\ & f_{\theta_N} + \sum_{i=1}^{N-1} \beta_i \left(\ddot{S}_i + k_d \dot{S}_i + k_p S_i - k_p \phi_i - k_d \dot{\phi}_i \right) + \\ & \sum_{i=1}^{N-1} \beta_i \left(\ddot{\phi}_o + k_d \dot{\phi}_o + k_p \phi_o \right) \end{aligned} \quad (3.61)$$

We choose $\ddot{\phi}_o$, utilizing that this can be used as an additional control input, in the form

$$\begin{aligned} \ddot{\phi}_o &= \\ & \frac{1}{\sum_{i=1}^{N-1} \beta_i} \left(-f_{\theta_N} - \sum_{i=1}^{N-1} \beta_i \left(\ddot{S}_i + k_d \dot{S}_i + k_p S_i - k_p \phi_i - k_d \dot{\phi}_i + 2k_p \phi_o + 2k_d \dot{\phi}_o + \sigma \right) \right) \end{aligned} \quad (3.62)$$

where σ is a new control input which will be defined later in this section. Note that β_i is negative-valued for all $i \in \{1, \dots, N - 1\}$, which implies that (3.62) is globally well-defined. The global exponential stability of the origin of the joint angle error dynamics in (3.59) implies that the joint tracking errors $(\tilde{\phi}_i, \dot{\tilde{\phi}}_i)$ converge exponentially fast to zero. Consequently, the reduced form of (3.62) to the invariant manifold where $(\tilde{\phi}_i, \dot{\tilde{\phi}}_i) = (0, 0)$ for all $i \in \{1, \dots, N - 1\}$, i.e. where $(\phi_i, \dot{\phi}_i) = (S_i + \phi_o, \dot{S}_i + \dot{\phi}_o)$, can be written as

$$\ddot{\phi}_o + k_d \dot{\phi}_o + k_p \phi_o = f_{\Phi} \quad (3.63)$$

where the right-hand side function is of the form

$$f_{\Phi}(t, \tilde{\theta}, \dot{\tilde{\theta}}) = \frac{1}{\sum_{i=1}^{N-1} \beta_i} \left(-f_{\theta_N} - \sum_{i=1}^{N-1} \beta_i \ddot{S}_i + \sigma \right) \quad (3.64)$$

We denote (3.64) as the Φ -subsystem. Note that the tracking control law (3.58) then is a dynamic feedback law, in the sense that it depends on the time evolution of $(\phi_o, \dot{\phi}_o)$, which are the solutions of the dynamical system (3.63).

In order to control the head angle of the robot, we define the control term σ in (3.62) as

$$\sigma = \ddot{\theta}_{\text{ref}} - k_{d,\theta_N} \dot{\tilde{\theta}} - k_{p,\theta_N} \tilde{\theta} \quad (3.65)$$

where $k_{p,\theta_N} > 0$ and $k_{d,\theta_N} > 0$ are the head angle controller gains. By inserting (3.65) into (3.62), and then the resulting equation into (3.61), the reduced form of the error dynamics equation for the head angle of the robot evaluated on the invariant manifold where $(\tilde{\phi}_i, \dot{\tilde{\phi}}_i) = (0, 0)$, can be written as

$$\ddot{\tilde{\theta}} + k_{d,\theta_N} \dot{\tilde{\theta}} + k_{p,\theta_N} \tilde{\theta} = f_{\Theta} \quad (3.66)$$

We denote (3.66) as the Θ -subsystem, where the perturbing term on the right-hand side is of the form

$$f_{\Theta}(\phi_o, \dot{\phi}_o) = -k_p \phi_o - k_d \dot{\phi}_o \quad (3.67)$$

For the aim of analysis, we divide the input to the Φ -subsystem f_{Φ} given by (3.64), into two parts. In particular, one part depends on the solutions of the Θ -subsystem, which are the head angle tracking errors, and the other part includes uniformly bounded friction forces and time-dependent reference signals. Consequently, we divide it into

$$f_{\Phi} = f_{\Phi_1} + f_{\Phi_2} \quad (3.68)$$

where

$$\begin{aligned} f_{\Phi_1} &= \frac{1}{\sum_{i=1}^{N-1} \beta_i} \left(-k_{\theta,d} \dot{\tilde{\theta}} - k_{\theta,p} \tilde{\theta} \right) \\ f_{\Phi_2} &= \frac{1}{\sum_{i=1}^{N-1} \beta_i} \left(-f_{\theta_N} - \sum_{i=1}^{N-1} \beta_i \ddot{S}_i + \ddot{\theta}_{\text{ref}} \right) \end{aligned} \quad (3.69)$$

Since the input to the Φ -subsystem depends on the solutions of the Θ -subsystem and vice versa, one may verify that the $(\Phi - \Theta)$ -subsystems are feedback connected. This interconnection is illustrated in Figure 3.1, and can be represented as

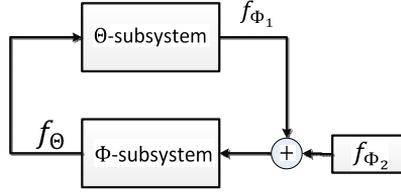


Figure 3.1: Illustration of the feedback connection (3.70)

$$\Sigma_{\text{int}} \begin{cases} \ddot{\tilde{\theta}} &= -k_{d,\theta_N} \dot{\tilde{\theta}} - k_{p,\theta_N} \tilde{\theta} + f_{\Theta}(\phi_o, \dot{\phi}_o) \\ \ddot{\phi}_o &= -k_d \dot{\phi}_o - k_p \phi_o + f_{\Phi}(t, \tilde{\theta}, \dot{\tilde{\theta}}) \end{cases} \quad (3.70)$$

The feedback connected system Σ_{int} is the dynamical system which governs the interconnection between the actuated and underactuated dynamics of the robot in closed-loop. In particular, for the Φ -subsystem the objective is to keep the solutions bounded, while for the Θ -subsystem the objective is to drive the solutions to a small neighbourhood of the origin, i.e. to make the head angle error arbitrarily small. To achieve these objectives, in the following we analyze the conditions under which the feedback connection remains stable.

Input-Output Stability of the Feedback Connected System

The feedback connected system Σ_{int} is composed of two subsystems given by

$$\frac{d}{dt} \begin{bmatrix} \tilde{\theta} \\ \dot{\tilde{\theta}} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k_{p,\theta_N} & -k_{d,\theta_N} \end{bmatrix} \begin{bmatrix} \tilde{\theta} \\ \dot{\tilde{\theta}} \end{bmatrix} + \begin{bmatrix} 0 \\ f_{\Theta} \end{bmatrix} \quad (3.71)$$

$$\frac{d}{dt} \begin{bmatrix} \phi_o \\ \dot{\phi}_o \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k_p & -k_d \end{bmatrix} \begin{bmatrix} \phi_o \\ \dot{\phi}_o \end{bmatrix} + \begin{bmatrix} 0 \\ f_{\Phi} \end{bmatrix} \quad (3.72)$$

To investigate the input-output stability of Σ_{int} , we introduce the augmented state vector $\hat{x} = [\tilde{\theta}, \phi_o, \dot{\tilde{\theta}}, \dot{\phi}_o]^T \in \mathbb{R}^4$, and the following augmented linear time-invariant system

$$\dot{\hat{x}} = \hat{A}\hat{x} + \hat{B}u \quad (3.73)$$

$$y = \hat{C}\hat{x} \quad (3.74)$$

where \hat{A} denotes the following matrix

$$\hat{A} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -k_{p,\theta_N} & -k_p & -k_{d,\theta_N} & -k_d \\ \frac{-k_{p,\theta_N}}{\sum_{i=1}^{N-1} \beta_i} & -k_p & \frac{-k_{d,\theta_N}}{\sum_{i=1}^{N-1} \beta_i} & -k_d \end{bmatrix} \quad (3.75)$$

and the input u is given by the following uniformly bounded scalar-valued function

$$u = f_{\Phi_2} \quad (3.76)$$

The input matrix \hat{B} and the output matrix \hat{C} are, respectively, given by

$$\hat{B} = [0 \ 0 \ 0 \ 1]^T \quad (3.77)$$

$$\hat{C} = [1 \ 1 \ 1 \ 1] \quad (3.78)$$

The following theorem investigates the input-output stability of the augmented dynamical system (3.73), with the output function (3.74).

Theorem 3.3 *The augmented dynamical system (3.73)–(3.74), i.e. the feedback connected system Σ_{int} is finite-gain \mathcal{L}_2 stable.*

Proof: It can be verified that all the eigenvalues of matrix \hat{A} have negative real parts, i.e. that matrix \hat{A} is Hurwitz, when $k_p, k_d, k_{p,\theta_N}, k_{d,\theta_N} > 0$. Consequently, by [99, Corollary 5.2], we conclude that (3.73)–(3.74) is finite-gain \mathcal{L}_p stable, for each $p \in [1, \infty]$, and the finite-gain is given by

$$\gamma^* = \frac{2\lambda_{\max}^2(\hat{P})\|\hat{B}\|_2\|\hat{C}\|_2}{\lambda_{\min}(\hat{P})} \quad (3.79)$$

where λ_{\max} and λ_{\min} denote, respectively, the maximum and minimum eigenvalues of $\hat{P} \in \mathbb{R}^{2 \times 2}$ which is the symmetric positive definite matrix solution of the Lyapunov equation

$$\hat{A}^T \hat{P} + \hat{P} \hat{A} = -I \quad (3.80)$$

where $I \in \mathbb{R}^{2 \times 2}$ denotes the identity matrix. ■

Remark 3.4 *Based on the finite-gain \mathcal{L}_2 stability of Σ_{int} , and uniform boundedness of the exogenous input f_{Φ_2} , we can conclude that the solutions of Σ_{int} are uniformly bounded by*

$$\|y(t)\|_2 \leq \gamma^* \|u(t)\|_2 \quad (3.81)$$

Thus, if we denote the upper-bound on the exogenous input f_{Φ_2} as $\zeta^* = \sup_{t \geq 0} \{f_{\Phi_2}(t)\}$, then we can derive the upper-bound on the solutions $\Phi_s = [\phi_o, \dot{\phi}_o]^T \in \mathbb{R}^2$ of the dynamic compensator, i.e. the Φ -subsystem, as

$$\zeta = \|\Phi_s(t)\|_2 \leq \gamma^* \zeta^* \in \mathbb{R}_{>0} \quad (3.82)$$

This upper-bound will be used in the stability analysis presented in the next subsection.

Theorem 3.3 guarantees the boundedness of the solutions of the feedback connected system Σ_{int} , and thus the requirement for the boundedness of the states of the dynamic compensator is fulfilled. It remains to show that under the controllers (3.58) and (3.62) the head angle error can be made arbitrarily small, which is the subject of the next subsection.

Practical Stability of the Head Angle Error Dynamics

We denote the state vector of (3.71) by $\Theta_s = [\tilde{\theta}, \dot{\tilde{\theta}}]^T \in \mathbb{R}^2$, and the state vector of (3.72) by $\Phi_s = [\phi_o, \dot{\phi}_o]^T \in \mathbb{R}^2$. The dynamical system (3.71) with $f_{\Theta} \equiv 0$ denotes the nominal part of the Θ -subsystem, and the dynamical system (3.72) with $f_{\Phi} \equiv 0$ denotes the nominal part of the Φ -subsystem. The nominal part of the Θ -subsystem (3.71), which characterizes the dynamics of the head angle tracking error, has a globally exponentially stable equilibrium at the origin $(\tilde{\theta}, \dot{\tilde{\theta}}) = (0, 0)$ because the following matrix is Hurwitz

$$A = \begin{bmatrix} 0 & 1 \\ -k_{p,\theta_N} & -k_{d,\theta_N} \end{bmatrix} \quad (3.83)$$

However, this nominal part is perturbed by the bounded non-vanishing perturbation term f_{Θ} . Through the following theorem, we investigate the practical stability of the origin of (3.71) in the presence of f_{Θ} .

Theorem 3.4 *Given the feedback connected system (3.70), the head angle error $(\tilde{\theta}, \dot{\tilde{\theta}})$ is uniformly ultimately bounded. Furthermore, it is possible to make the ultimate bound arbitrarily small by choosing sufficiently large gains $(k_{p,\theta_N}, k_{d,\theta_N})$.*

Proof: We select a quadratic Lyapunov function in the form

$$V = \frac{1}{2} \Theta_s^T P \Theta_s \quad (3.84)$$

where $P \in \mathbb{R}^{2 \times 2}$ is the solution of the Lyapunov equation

$$A^T P + P A = -Q \quad (3.85)$$

Since the matrix A in (3.83) is Hurwitz, there will always be a unique, symmetric and positive definite solution P to (3.85) for any positive definite matrix Q [99, Th. 4.6]. In order to reflect that the convergence rate of the linear system given by (3.83) will depend on the chosen gain parameters, we choose the following $Q \in \mathbb{R}^{2 \times 2}$ positive definite matrix

$$Q = \begin{bmatrix} k_{p,\theta_N} & 0 \\ 0 & k_{d,\theta_N} \end{bmatrix} \quad (3.86)$$

We denote the minimum eigenvalue of Q by $\lambda_{\min}(Q)$, which is characterized by the choice of the head angle controller gains $(k_{p,\theta_N}, k_{d,\theta_N})$. By a *Converse Lyapunov Theorem* [99], exponential stability of the nominal part of (3.71) implies that (3.84) satisfies the following inequalities [99, Ch. 9.1]

$$\lambda_{\min}(P)\|\Theta_s\|_2^2 \leq V(\Theta_s) \leq \lambda_{\max}(P)\|\Theta_s\|_2^2 \quad (3.87)$$

$$\frac{\partial V(\Theta_s)}{\partial \Theta_s} A \Theta_s \leq -\lambda_{\min}(Q)\|\Theta_s\|_2^2 \quad (3.88)$$

$$\left\| \frac{\partial V(\Theta_s)}{\partial \Theta_s} \right\|_2 \leq 2\lambda_{\max}(P)\|\Theta_s\|_2 \quad (3.89)$$

where $\lambda_{\min}(P)$ and $\lambda_{\max}(P)$ denote the minimum and maximum eigenvalues of P , respectively. Furthermore, we select V as a Lyapunov function *candidate* for the perturbed system (3.71). Taking the time-derivative of V along the solutions of (3.71), and utilizing the properties (3.87)–(3.89), gives

$$\dot{V} = -\frac{\partial V}{\partial \Theta_s} A \Theta_s + \frac{\partial V}{\partial \Theta_s} f_{\Theta} \quad (3.90)$$

The first right-hand side term in (3.90) denotes the time-derivative of V along the solutions of the nominal part of (3.71), and the second right-hand side term is the effect of the perturbing term f_{Θ} . Using the inequalities in (3.87)–(3.89), we obtain

$$\dot{V} \leq -\lambda_{\min}(Q)\|\Theta_s\|_2^2 + 2\lambda_{\max}(P)\|\Theta_s\|_2\|f_{\Theta}\|_2 \quad (3.91)$$

Moreover, from the definition of f_{Θ} in (3.67) and using Cauchy-Schwartz inequality, see e.g. [99], we have that

$$\|f_{\Theta}\|_2 \leq \left(\sqrt{k_p^2 + k_d^2} \right) \|\Phi_s\|_2 \quad (3.92)$$

To simplify the analysis, we choose the controller gains as

$$k_p = \frac{k_p^*}{2\lambda_{\max}(P)}, \quad k_d = \frac{k_d^*}{2\lambda_{\max}(P)} \quad (3.93)$$

where $k_p^* > 0$ and $k_d^* > 0$. With this choice of the controller gains, and also based on the upper-bound on the solutions of the dynamic compensator $\zeta \in \mathbb{R}_{>0}$ in (3.82), inequality (3.91) takes the form

$$\dot{V} \leq -\lambda_{\min}(Q) \|\Theta_s\|_2^2 + \|\Theta_s\|_2 \left(\sqrt{k_p^{*2} + k_d^{*2}} \right) \zeta \quad (3.94)$$

For the second term in the right-hand side of (3.94), we use Young's inequality where we have that

$$ab \leq \frac{\gamma a^2}{2} + \frac{b^2}{2\gamma} \quad (3.95)$$

where $a, b \in \mathbb{R}$, and $\gamma > 0$ is any positive constant [101]. In particular, by taking

$$a = \|\Theta_s\|_2 \left(\sqrt{k_p^{*2} + k_d^{*2}} \right), \quad b = \zeta \quad (3.96)$$

one can write (3.94) in the form

$$\dot{V} \leq \left(-\lambda_{\min}(Q) + \gamma [k_p^{*2} + k_d^{*2}] \right) \|\Theta_s\|_2^2 + \frac{\zeta^2}{2\gamma} \quad (3.97)$$

With any choice of γ , k_p^* , and k_d^* , we can choose the elements of Q , i.e. $(k_{p,\theta_N}, k_{d,\theta_N})$, sufficiently large, so that

$$\alpha^* = \left(-\lambda_{\min}(Q) + \gamma [k_p^{*2} + k_d^{*2}] \right) \quad (3.98)$$

is negative. In this case, for a sufficiently small positive constant λ the following inequality holds

$$\dot{V} \leq -\alpha^* \|\Theta_s\|_2^2 + \frac{\zeta^2}{2\gamma} \leq -\lambda (\lambda_{\max}(P) \|\Theta_s\|_2^2) + \frac{\zeta^2}{2\gamma} \quad (3.99)$$

Based on the inequality in (3.87), we can also derive the following inequality

$$-\lambda V \geq -\lambda (\lambda_{\max}(P) \|\Theta_s\|_2^2) \quad (3.100)$$

and using this in (3.99) yields

$$\dot{V} \leq -\lambda V + \frac{\zeta^2}{2\gamma} \quad (3.101)$$

Consequently, a straightforward application of the comparison lemma yields

$$V(t) \leq e^{-\lambda t} V(0) + \frac{\zeta^2}{2\gamma\lambda} \quad (3.102)$$

From (3.102) we conclude the ultimate boundedness of the head angle error, because the first term on the right-hand side of (3.102) vanishes as $t \rightarrow \infty$, and the second term is uniformly bounded. However, boundedness of the head angle error is not sufficient to achieve the control objective (3.55). We also need to show that the ultimate bound can be made arbitrarily small. To this end, we notice that based on (3.102) V converges to a ball of radius $\zeta^2/2\gamma\lambda$. Consequently, based on the inequality $\lambda_{\min}(P)\|\Theta_s\|_2^2 \leq V$ in (3.87), we conclude that $\|\Theta_s\|_2$ also converges to a ball of radius

$$r = \frac{\zeta}{\sqrt{2\lambda_{\min}(P)\gamma\lambda}} \quad (3.103)$$

Moreover, we can drive $\|\Theta_s\|_2$ to any arbitrary small neighbourhood of the origin ϵ_θ , by choosing

$$\gamma = \frac{\zeta^2}{2\lambda_{\min}(P)\lambda\epsilon_\theta^2} \quad (3.104)$$

which can be seen by inserting (3.104) into (3.103). Furthermore, from (3.98) it can be seen that for any value of γ in (3.104), it is always possible to make α^* negative by making $\lambda_{\min}(Q)$ large enough, i.e. by choosing $(k_{p,\theta_N}, k_{d,\theta_N})$ sufficiently large. Consequently, an arbitrarily small ultimate bound for the head angle error can be achieved by properly choosing the gains $(k_{p,\theta_N}, k_{d,\theta_N})$, and the control objective (3.55) is achieved. This completes the proof of Theorem 3.4. ■

Remark 3.5 *By the result of Theorem 3.3, the feedback connection (3.70) is finite-gain \mathcal{L}_2 stable when $k_p, k_d, k_{p,\theta_N}, k_{d,\theta_N} > 0$. Furthermore, from (3.93) we see that we need to choose the gains of the orientation controller $(k_{p,\theta_N}, k_{d,\theta_N})$ sufficiently larger than the gains of the dynamic compensator (k_p, k_d) , in order to guarantee that the head angle error converges to a small neighbourhood of the origin. This can also be interpreted based on the fact that (k_p, k_d) increase the strength of the perturbing term f_Θ on the right-hand side of (3.71). In other words, by decreasing the strength of f_Θ and increasing $(k_{p,\theta_N}, k_{d,\theta_N})$, any small ultimate bound on the head angle error can be achieved.*

Such as previous controllers of this chapter, through numerical simulations for this controller we will show that the proposed guidance-based control strategy of this section, successfully make the snake robot converge to the desired path and then drives it along the path.

3.5 Simulation and Experimental Results

3.5.1 Simulation and Experimental Results for the VHC based Body Shape and Orientation Controller

In this section, we present simulation and experimental results which illustrate the performance of the proposed path following controller in Section 3.2. We considered a snake robot with $N = 11$ links, $m = 1$ kg, $l = 0.07$ m, and $J = 0.0016$ kgm². The friction coefficients were $c_n = 10$, $c_t = 1$. The parameters of the joint constraint functions (3.11) were $\alpha = \pi/6$ rad, $\eta = 70\pi t/180$ rad, and $\delta = 36\pi/180$ rad. The controller gains in (3.20) and (3.26) were tuned as $k_p = 10$, $k_d = 5$, $k_{p,\theta_N} = 20$, $k_{d,\theta_N} = 1$, and $\Delta = 1.4$ m. In order to calculate $\dot{\Phi}_N$ and $\ddot{\Phi}_N$, we employed the approach taken in [89] by passing Φ_N through a low-pass filter of the form

$$\frac{d}{dt} \begin{bmatrix} \Phi_N \\ \dot{\Phi}_N \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\psi_f\omega_n \end{bmatrix} \begin{bmatrix} \Phi_N \\ \dot{\Phi}_N \end{bmatrix} + \begin{bmatrix} 0 \\ \omega_n^2 \end{bmatrix} \Phi_N \quad (3.105)$$

with natural frequency $\omega_n = \pi/2$ rad, damping ratio $\psi_f = 1$, and zero initial conditions. As seen from the simulation results which are presented in Figures 3.2–3.7, the snake robot successfully converges to and follows the desired path. In particular, Figure 3.2 shows that the quantity $\sum_{i=1}^{10} \beta_i$ is uniformly bounded away from zero, which implies that (3.26) is globally well-defined. Figure 3.3 shows that the solutions of the dynamic compensator (3.26) remain bounded. Figure 3.4 shows that the joint angles track the reference angles provided by the constraint functions (3.11), while the tracking errors converge exponentially to zero. Figure 3.5 shows that the head angle tracks the reference head angle provided by the constraint function (3.12), while the tracking error converges to zero exponentially. Figure 3.6 shows that the CM of the robot converges to and follows the desired straight path. Moreover, in order to show the performance of the proposed tracking control law (3.20) in the presence of angular position measurement noise, we subjected every i -th joint angle ϕ_i to an additive noise by using Matlab function `randn()` which generates normally distributed pseudorandom numbers that can be considered as measurement noise for the joint angles. In particular, we added `randn(1)` to each joint angle ϕ_i in each integration step. The result of simulation is presented in Figure 3.7, which shows that the joint tracking errors converge to a very small neighbourhood of zero in the presence of measurement noise.

Experimental Setup

The experiment was carried out using the snake robot Wheeko, see [11]. The robot, which is shown in Figure 3.8, has 10 identical joint modules, i.e. $N = 11$ links. Each joint module is equipped with a set of passive wheels which give the robot anisotropic ground friction properties during motion on flat surfaces. The wheels are able to slip sideways, and thus do not introduce nonholonomic velocity constraints in the system. Each joint is driven by a *Hitec* servo motor (HS-5955TG), and the joint angles are measured using magnetic rotary encoders. The motion of the snake robot was measured using a camera-based motion capture system from *Optitrack* of type Flex 13. The system consists of 16 cameras which are sampled at 120 frames per second and which allow reflective markers to be tracked on a sub-millimeter level. During the experiment, reflective markers were mounted on the head link of the snake robot in order to measure the position (x_N, y_N) and orientation (θ_N) of the head. These measurements were combined with the measured joint angles $(\phi_1, \dots, \phi_{N-1})$ of the snake robot in order to measure the absolute link angles (2.5) and the position of the CM (p_x, p_y) of the robot. In order to obtain the derivatives of the reference head angle (3.12), we used the same technique as in the simulations, i.e. passing Φ_N through a low-pass filter of the form (3.105). The parameters of the low-pass filter were set to $\omega_n = \pi/2$ and $\psi_f = 1$.

In the following, we elaborate on a few adjustments that were made in the implemented path following controller in order to comply with the particular properties and capabilities of the physical snake robot employed in the experiment. We conjecture that these adjustments only marginally affected the overall motion of the robot. The successful path following behaviour of the robot demonstrated below supports this claim. Since the experimental setup only provided measurements of the joint angles and the position and orientation of the head link, we chose to implement the joint controller in (3.20) as

$$\vartheta_i = -k_p y_i \quad (3.106)$$

where $i \in \{1, \dots, 10\}$. We conjecture that eliminating the joint angular velocity terms from (3.20) did not significantly change the dynamic behaviour of the system since the joint motion was relatively slow during the experiment. The main consequence of excluding the velocity terms from (3.20) is that we potentially introduce a steady-state error in the tracking of the joint angles. Consequently, since with the joint control law (3.106) the derivative terms in (3.20) are identically zero, they need not to be linearized in the head angle dynamics by the dynamic compensator. As the result, we implemented the dynamic compen-

sator of the form

$$\ddot{\phi}_o = \left(\sum_{i=1}^{N-1} \beta_i \right)^{-1} \left(-f_{\theta_N} + d^2 \Phi_N - k_{p,\theta_N} y_N - k_{d,\theta_N} \dot{y}_N \right) - k_p \phi_o - k_d \dot{\phi}_o \quad (3.107)$$

where the controller gains were $k_{p,\theta_N} = 20$, $k_{d,\theta_N} = 1$, $k_p = 10$, and $k_d = 5$. We saturated the joint angle offset ϕ_o according to $\phi_o \in [-\pi/6, \pi/6]$, in order to keep the joint reference angles within reasonable bounds with respect to the maximum allowable joint angles of the physical snake robot. Moreover, from Figure 3.8, it can be seen that the head link of the snake robot does not touch the ground since the ground contact points occur at the location of the joints. As a results, we implemented (3.107) with $f_{\theta_N} \equiv 0$. The solutions of the dynamic compensator (3.107) were obtained by numerical integration in *LabVIEW* which was used as the development environment. We chose the look-ahead-distance of the path following controller as $\Delta = 1.4$ m. The initial values for the configuration variables of the snake robot were $\phi_i = 0$ rad, $\theta_N = -\pi/2$ rad, $p_x = 0.3$ m and $p_y = 1.7$ m, i.e. the snake robot was initially headed towards the desired path (the x-axis), and the initial distance from the CM to the desired path was 1.7 m. Furthermore, the parameters of the constraint functions for the joint angles (3.11), were $\alpha = \pi/6$, $\eta = 70\pi t/180$, and $\delta = 36\pi/180$, and the ground friction coefficients were $c_t = 1$ and $c_n = 10$, identical to the simulation parameters.

Experimental Results

The results of the experiments are illustrated in Figures 3.9–3.14. In particular, Figure 3.9 shows screen shots of the experiments. Figure 3.10 shows that the solution of the dynamic compensator remained bounded. Figure 3.11 shows that the joints of the robot tracked the sinusoidal reference angles provided by the constraint functions (3.11), and that the tracking error converged to a neighbourhood of the origin. As discussed above, this is probably due to the modification of the joint controller (3.106) due to the lack of velocity measurements in the lab. Figure 3.12 shows that the head angle of the robot tracked the reference head angle defined by the constraint function (3.12), and that the tracking error converged to a neighbourhood of the origin. Figure 3.13 shows the motion of the CM of the robot in the x – y plane, which converged to and followed the desired path. Figure 3.14 compares the motion of the CM during the simulation and the experiment, which were performed using the same controller parameters in order to obtain comparable results. In particular, from Figure 3.14 it can be seen that the physical snake and the simulated snake follow almost the same path. However, due to precise measurement and a more accurate joint control law for the simulated snake, the path following error converges to a smaller neighbourhood of the origin.

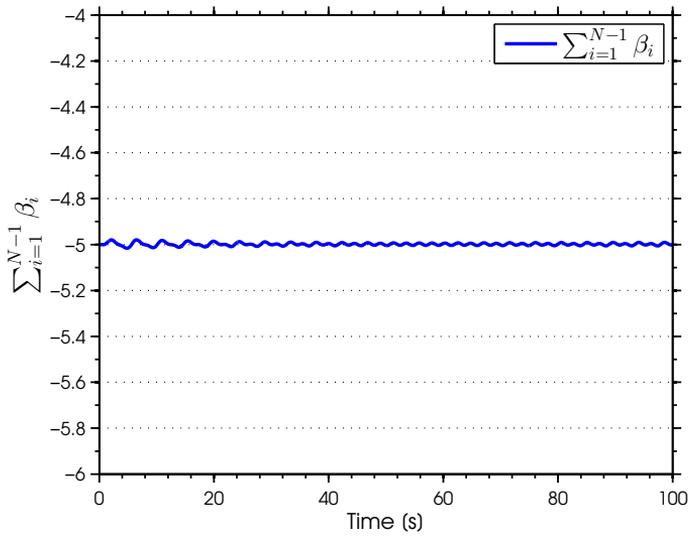


Figure 3.2: The term $\sum_{i=1}^{N-1} \beta$ is bounded away from zero.

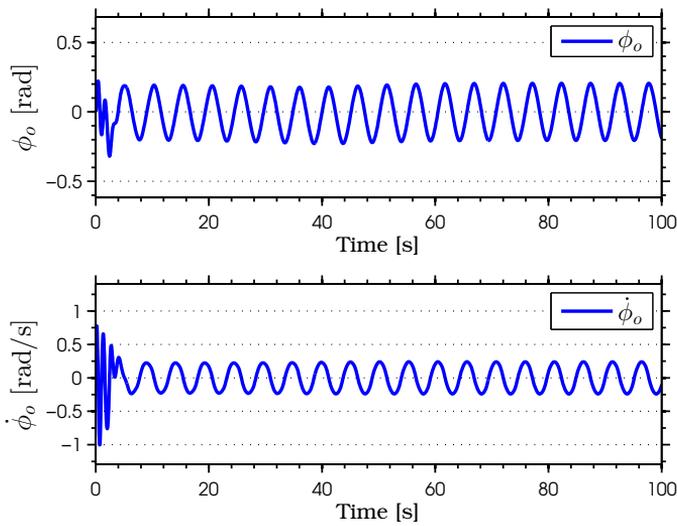


Figure 3.3: The solutions of the dynamic compensator remain bounded, and converge to a neighbourhood of the origin after compensating for the head angle error.

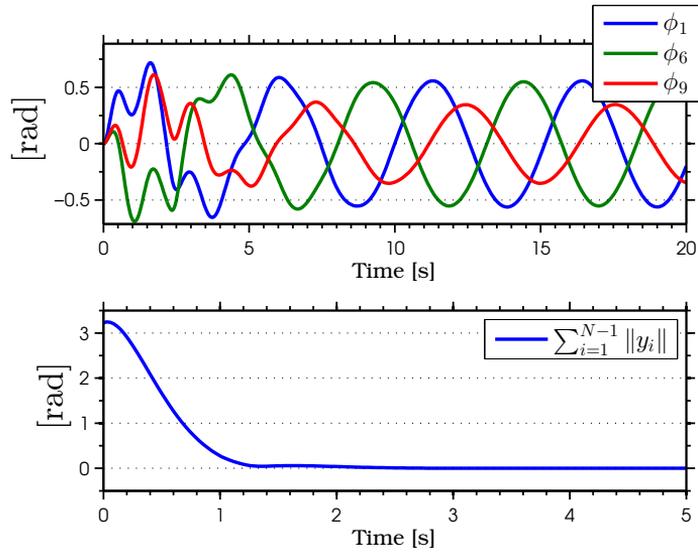


Figure 3.4: The joint angles track the reference joint angles provided by constraint functions (3.11) (above). The tracking error converges exponentially to zero (below).

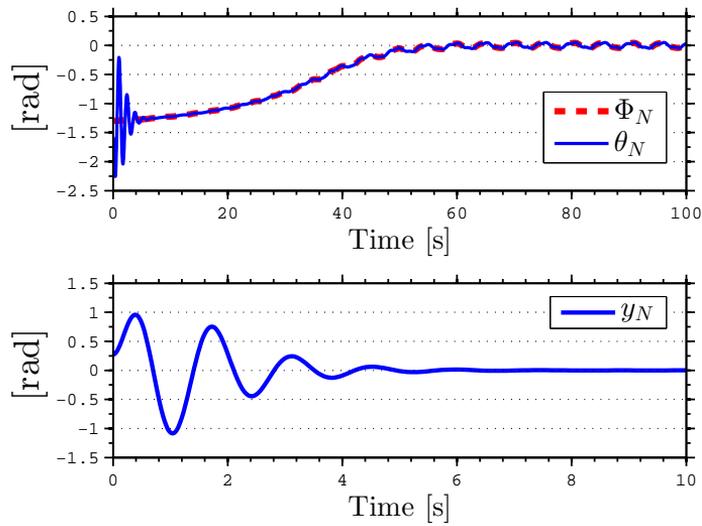


Figure 3.5: The head angle tracks the reference head angle provided by (3.12) (above), while the tracking error converges exponentially to zero (below).

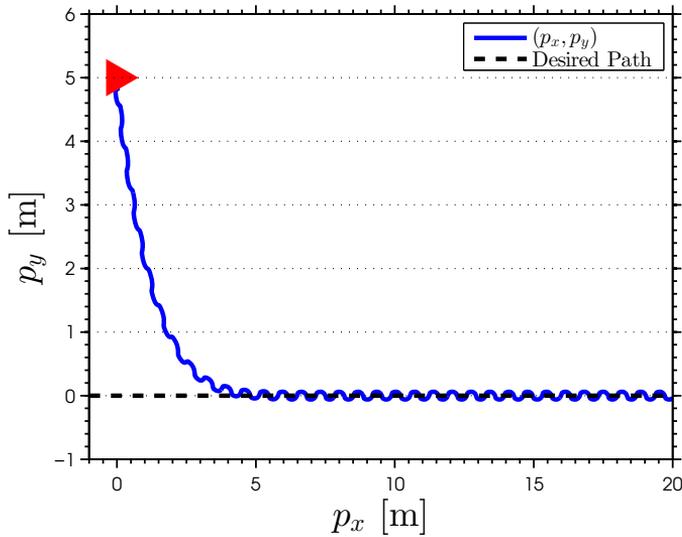


Figure 3.6: The simulations verify that the position of the CM of the robot (blue) converges to and follows the desired path (dashed black).

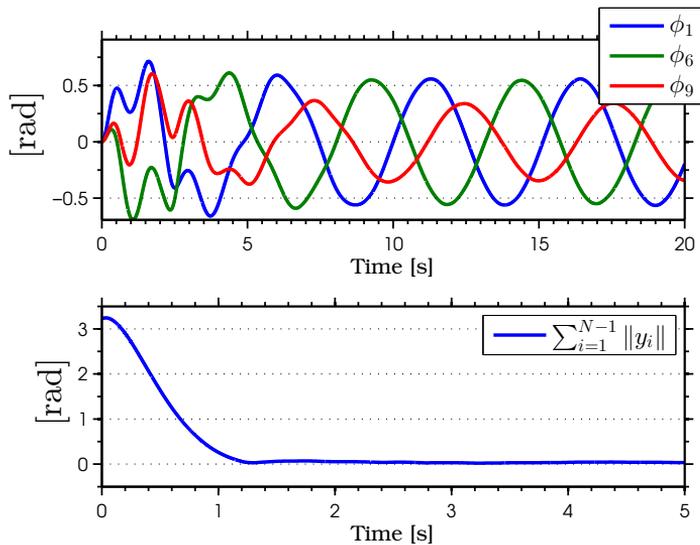


Figure 3.7: The joint angles track the reference joint angles (3.11) in the presence of measurement noise (above), the tracking error converges to a neighbourhood of zero (below).



Figure 3.8: Snake robot Wheeko was used for the experiments.

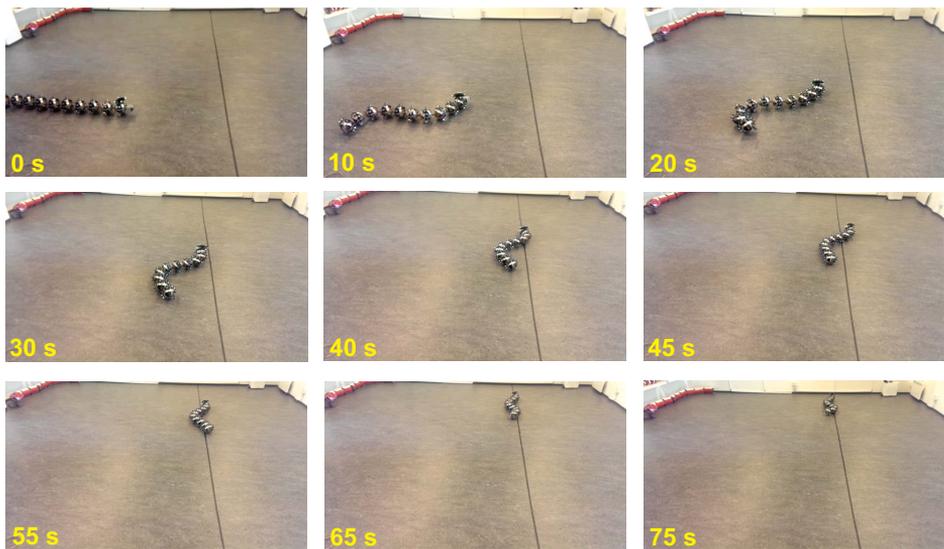


Figure 3.9: Screen-shots of the motion of the robot during the experiments, which shows that the robot converges to and follows the desired path.

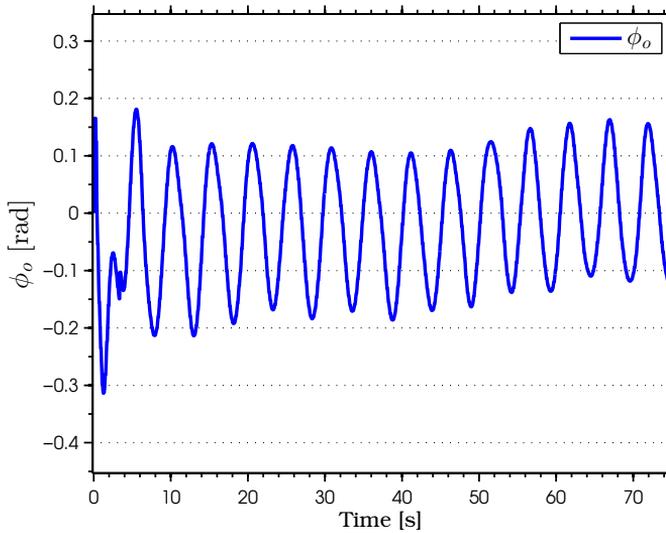


Figure 3.10: The solution of the dynamic compensator remains bounded during the experiments.

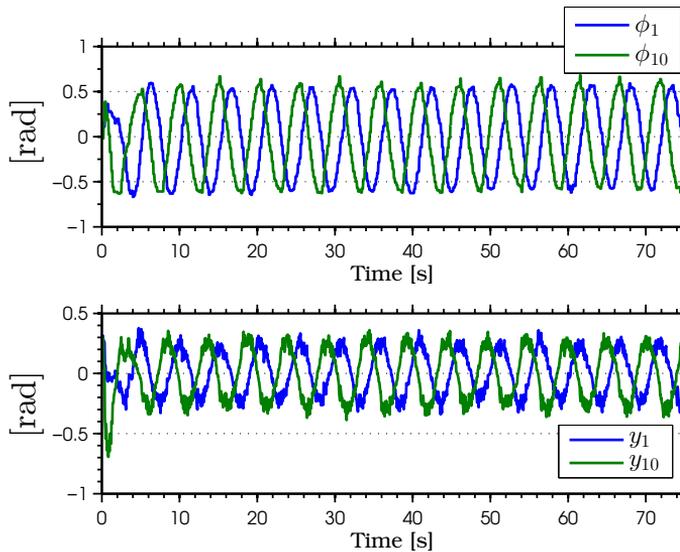


Figure 3.11: The joints of the robot track the sinusoidal references provided by the constraint functions (3.11) (above), the tracking error converges to a neighbourhood of zero (below).

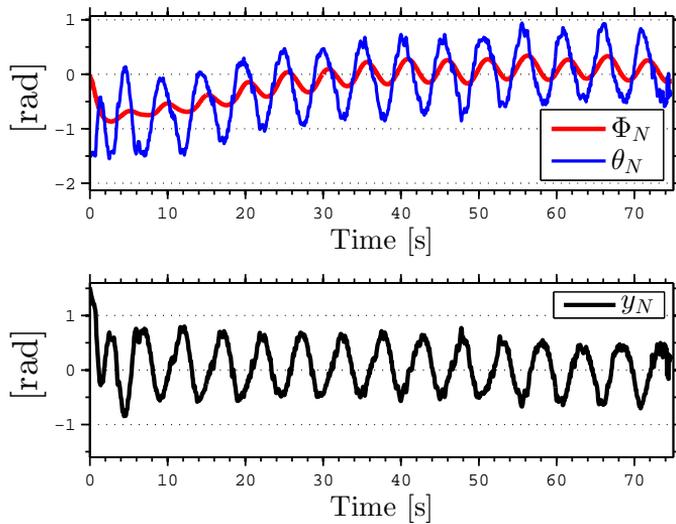


Figure 3.12: The head angle tracks the reference head angle defined by the LOS guidance law (3.12) (above), the tracking error converges to a neighbourhood of zero (below).

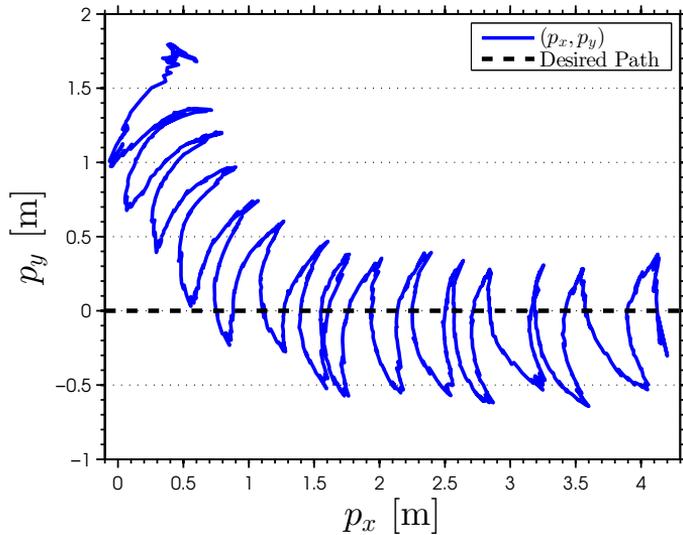


Figure 3.13: The position of the CM of the robot (blue) converges to and follows the desired path (dashed black) during the experiments.

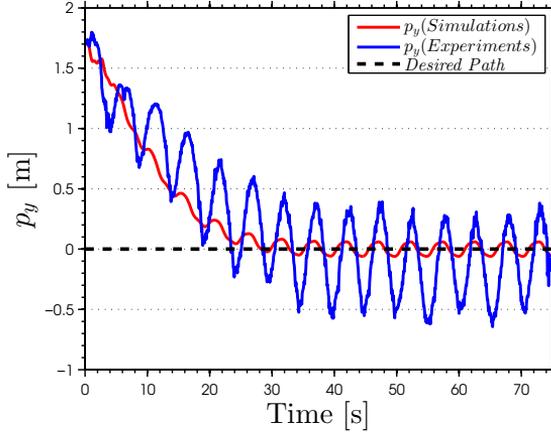


Figure 3.14: Comparison between the motion of the CM of the robot during the simulations (red) and the experiments (blue).

3.5.2 Simulation Results for the Sliding Mode Controller

In this subsection, the simulation results for the sliding mode controller are presented. For the simulations, the inertial parameters of the $N = 11$ links robot were $m = 0.5$ kg, $l = 0.1$ m, and $J = 0.0016$ kgm². The parameters of the joint reference trajectory in (3.11) were $\alpha = \pi/6$ rad, $\lambda = 80t\pi/180$ rad/s, and $\delta = 2\pi/3$ rad. The sliding mode controller gains were tuned as $K = \text{diag}\{2\}_{i=1}^3$, $\gamma = \text{diag}\{15\}_{i=1}^3$, $\Lambda = 4$, $\gamma_\theta = 10$, and $\Delta = 2$ m. All the initial conditions were set to zero, except $p_y = 5$ meters initial cross-track error. In order to show the robustness of the proposed feedback control law with respect to the modelling uncertainties due to friction forces, we assumed that the identified ground friction coefficients were $c_n = 10$, and $c_t = 1$, while the actual coefficients were $c_n = 20$, and $c_t = 2$. This implies that exact linearization will not be performed by the nominal part ϑ_{nom} of the sliding mode tracking feedback control law (3.37). Consequently, the dynamic model will be subject to nonlinear terms which arise due to this parameter uncertainties. Throughout the simulation study, we aim to show that the body shape and orientation controller derived in Section 3.3 will successfully achieve the control objectives (3.2), (3.4), and (3.6) in the presence of these nonlinear terms. In particular, Figure 3.16 shows that the states of the dynamic compensator (3.52) remain bounded. Figure 3.17 shows that the joint angles track the reference trajectories (3.11). Figure 3.18 shows that the CM of the robot converges to and follows the desired straight path, and Figure 3.19 shows the convergence of the head angle error to zero.

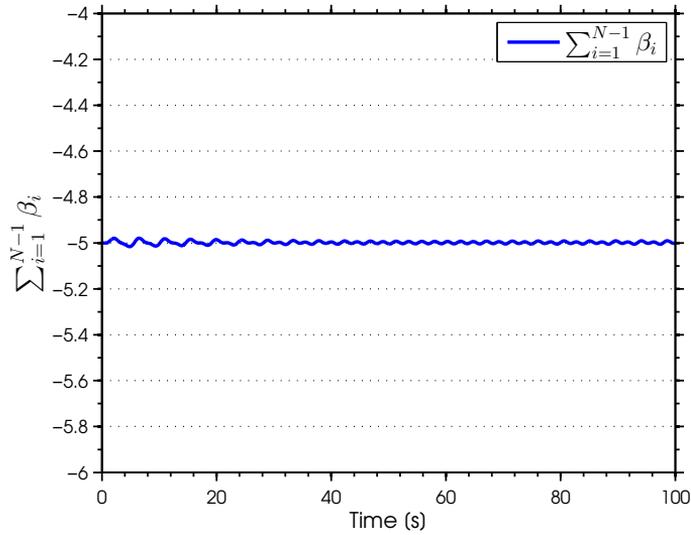


Figure 3.15: The term $\sum_{i=1}^{N-1} \beta_i$ is uniformly bounded away from zero, which implies that the dynamic equation (3.52) is globally well-defined.

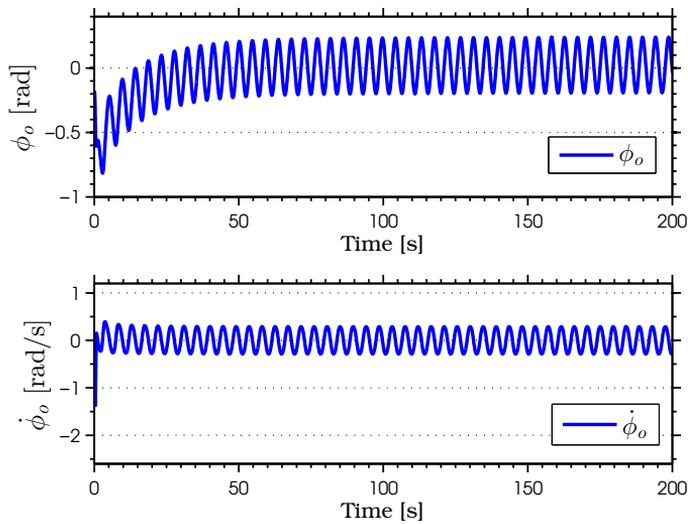


Figure 3.16: The solutions of the dynamic compensator converge to a neighbourhood of the origin, after compensating for the head angle error

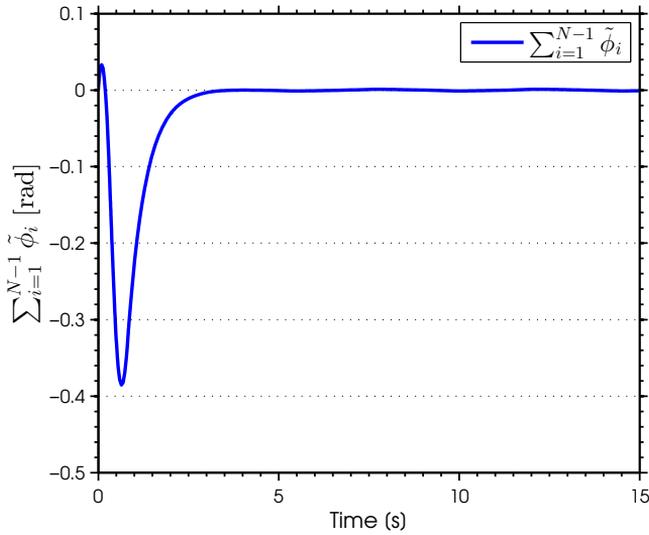


Figure 3.17: The joints of the robot track the serpentine motions while the joint tracking errors converge to zero exponentially.

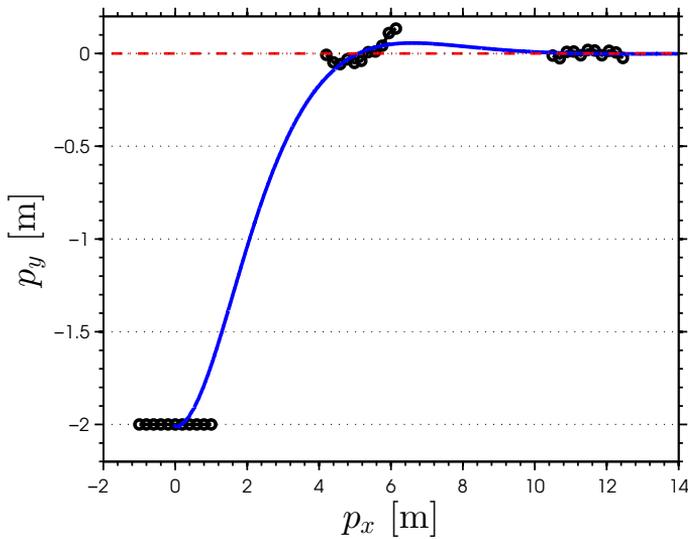


Figure 3.18: The position of the CM of the robot (blue) follows the desired straight line path (dash-dotted red).

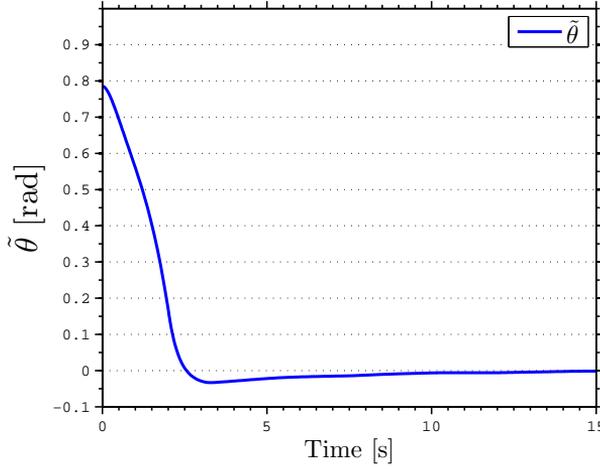


Figure 3.19: The head angle tracking error converges to zero exponentially

3.5.3 Simulation Results for the Dynamic Feedback Controller with Analysis of the Controlled System

In this subsection, we present the results of numerical simulations which illustrate the performance of the proposed control design. We considered a snake robot with $N = 4$ links and with inertial link parameters $m = 0.3$ kg, $2l = 0.14$ m, and $J = 0.0016$ kgm². The friction coefficients were $c_n = 10$, and $c_t = 1$. The parameters of the reference joint trajectories were $\alpha = 30\pi/180$ rad, $\omega = \pi$ rad/s, and $\delta = 120\pi/180$ rad. The dynamic feedback controller gains were tuned as $k_p = 5$, $k_d = 5$, $k_{p,\theta_N} = 500$, and $k_{d,\theta_N} = 500$. Note that we have chosen the gains in accordance with Remark 3.5, such that the system is finite-gain \mathcal{L}_2 stable, and the orientation error goes to a small neighbourhood of the origin. The look-ahead-distance was $\Delta = 3$ m.

As seen from the simulation results, the snake robot successfully tracks the reference body shape and orientation, and thereby converges to and follows the desired path. In particular, Figure 3.20 shows that the states of the dynamic compensator remain bounded. Figure 3.21 shows that perfect tracking is achieved for the body shape of the robot, and Figure 3.22 shows the time evolution of the link angles. Moreover, in Figure 3.22 the head angle tracks the LOS guidance law, while the tracking error converges to a neighbourhood of zero as shown in Figure 3.23. Finally, Figure 3.24 shows how the snake robot successfully converges to and follows the desired path.

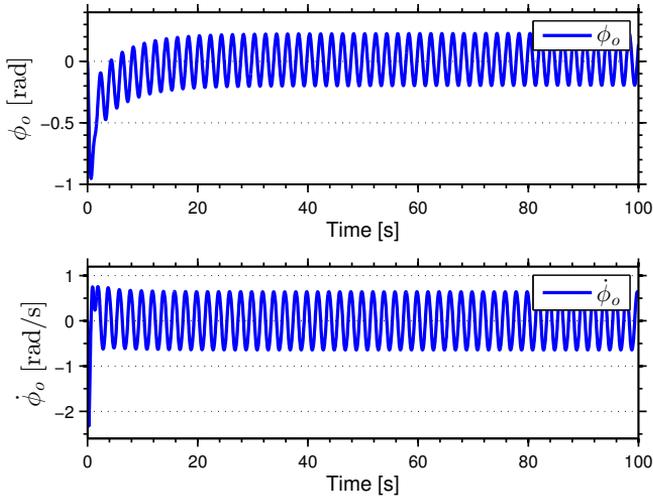


Figure 3.20: The solutions of the Φ -subsystem converge to a neighbourhood of the origin after compensating for the head angle error.

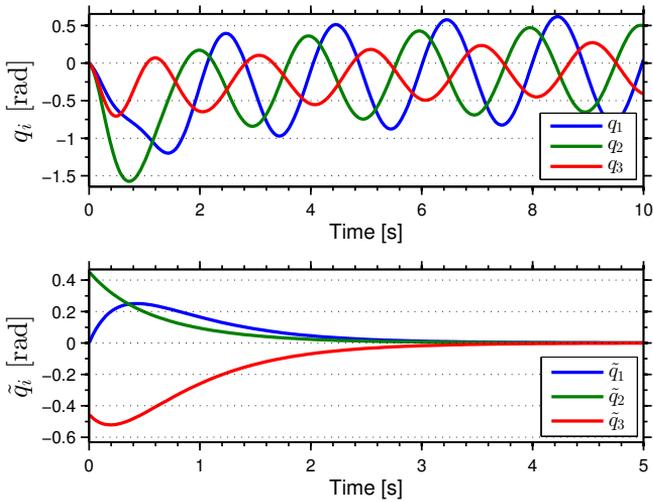


Figure 3.21: The joints track the sinusoidal motions (above), the tracking errors converge exponentially to zero (below).

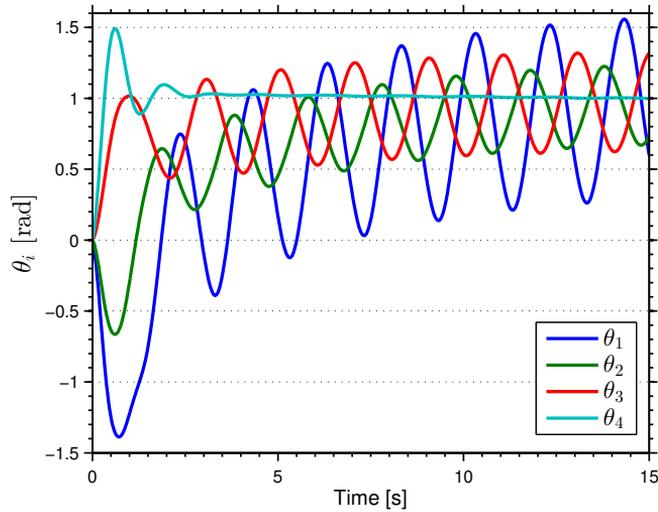


Figure 3.22: Oscillations in the links of the robot which are induced by sinusoidal joint motions. The head angle (cyan) tracks the guidance law, and thus does not oscillate.

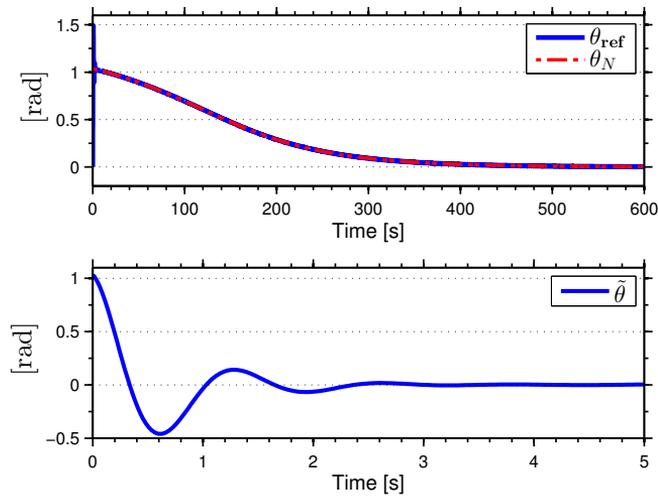


Figure 3.23: The head angle tracks the reference head angle (above), the tracking error converges to a neighbourhood of the origin (below).

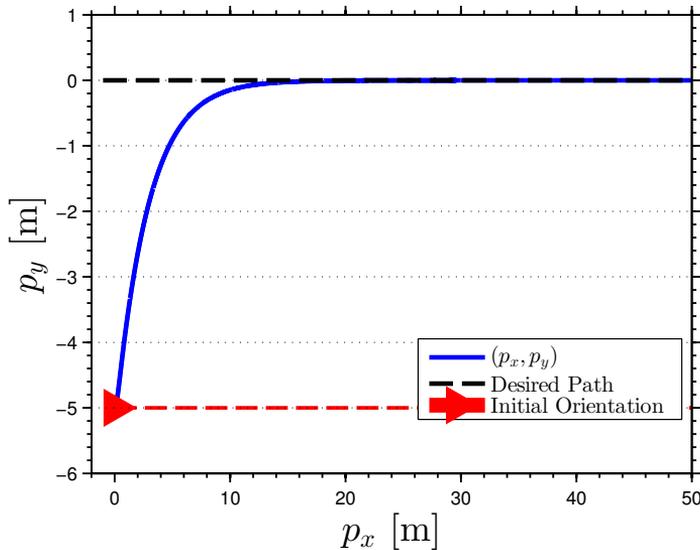


Figure 3.24: The robot is initially headed away from the desired path. The position of the CM of the robot (blue) converges to and follows the desired straight line path (the x -axis).

Chapter Summary

- We considered body shape and orientation control of planar snake robots by use of VHC. In particular, we introduced VHC that defined a constraint manifold for the robot. We designed an input-output feedback linearizing control law to exponentially stabilize the constraint manifold for the system.
- Using sliding mode techniques, and based on the dynamic model which was subject to nonlinearities that were due to parametric modelling uncertainties, we designed a robust body shape and orientation controller, which induced path following on the robot. In particular, we designed a dynamic compensator to control the head angle of the robot to an angle defined by a LOS Guidance law.
- We used an input-output stability analysis to show that the solutions of the dynamic compensator which was used in order to control the head angle of the robot remain uniformly bounded.

- A formal proof of the convergence of the position of the CM of the robot to the desired path remains a topic of subsequent chapters. However, we presented simulation and experimental results which validated the theoretical approach for the controllers in this chapter. In particular, these results showed that the robot successfully converged to and followed a desired straight path.

Model-based Locomotion Control Approaches for Snake Robots Part I: The Simplified Model

In this chapter, we present various control methods for the position of the CM of the snake robot in the plane. In particular, we use the simplified kinematic and dynamic model of the snake robot which was presented in Chapter 2, in order to solve two locomotion control problems. These control problems consist of direction following and maneuvering control of snake robots. In particular, to address these problems we need to control the body shape, orientation, and velocity of the snake robot to given references. Note that the orientation and position of the CM of the snake robot are underactuated degrees of freedom for which the control design is challenging. For instance, it is often difficult or even impossible for underactuated systems to find an appropriate feedback transform along with a change of coordinates in order to write the governing dynamics of the passive degrees of freedom in a linear format [103]. As a result, for snake robots which have at least three degrees of underactuation, there are very few model-based feedback control approaches which rely on formal stability proofs. However, in this chapter we utilize the simplified model of the snake robot locomotion in order to present formal stability proofs for the closed-loop dynamics of the underactuated position variables of the robot. In particular, we use the method of virtual holonomic constraints (VHC) to address the direction following and maneuvering control problems for the snake robot. To this end, we first stabilize a constraint manifold for the fully-actuated body shape variables of the robot. The definition of the constraint manifold is inspired by the well-known reference joint angle trajectories which induce lateral undulatory motion on snake robots. Moreover, we then will show that the dynamical system evaluated on the invariant constraint manifold can effectively be controlled

using dynamic and static compensators. In particular, we will use the design freedom that the parameters of the desired gait patterns give, and choose these parameters by using dynamic and static compensators, which control the orientation and planar position of the robot. Furthermore, we will show that the solutions of these compensators remain uniformly bounded.

In this chapter, we will investigate two locomotion control problems for snake robots which move on a horizontal and flat surface. These control problems consist of direction following, and maneuvering control of snake robots. The general definitions of these control problems are given below.

- **Direction Following:** In the direction following problem, the goal is to regulate the orientation and the forward velocity of the robot to a constant value, while guaranteeing the boundedness of the states of the controlled system.
- **Maneuvering:** The maneuvering problem consists of two tasks (see e.g. [104]). The first task, which is called the *geometric task*, is to converge to and follow a desired geometric path. The second task, which is called the *dynamic task*, consists of satisfying dynamical constraints, e.g. a desired velocity profile, along the desired path.

In the subsequent sections, we will state the mathematical formulations for the above control problems, and we use nonlinear control theory in order to derive the controllers. Furthermore, we present formal stability proofs for the controlled systems.

Contributions of this chapter: Using the simplified model, this chapter makes three contributions to snake robot locomotion control. The first contribution of this chapter is to propose a direction following controller for the snake robot. The second contribution of this chapter is to propose a maneuvering controller for the snake robot along straight paths. In particular, by solving the maneuvering problem, we control the body shape, orientation, and planar position of the robot. The third contribution of this chapter is to propose a maneuvering controller for the snake robot along general curved paths.

In particular, in this chapter we aim to control the orientation and the forward velocity of the robot to given references (either constant or time-varying), while guaranteeing the boundedness of all the states of the controlled system. To this end, we first stabilize a constraint manifold for the fully-actuated body shape variables of the robot. The definition of the constraint manifold is inspired by the well-known reference joint angle trajectories which induce lateral undulatory motion for snake robots. Subsequently, we reduce the dynamics of the system to the invariant constraint manifold. Furthermore, we design two

dynamic compensators which control the orientation and velocity of the robot on this manifold. Using numerical analysis and a formal stability proof, we show that the solutions of the compensators remain bounded.

To our best knowledge, similar control problems have never been considered for snake robots. However, both in terms of theoretical developments and practical aspects, this is an important step forward for locomotion control of snake robots. In particular, theoretical control challenges which arise due to the complicated dynamic model of snake robots, which have at least three degrees of underactuation, can help to an increased understanding of motion control of underactuated mechanical systems. Furthermore, for real-time applications of snake robots, it is crucial to automate the locomotion of the robot in the environments where the human presence is unsafe or impossible, and also to remove the necessary communication system with a human operator which can reduce the cost and sensor equipments.

Using the simplified model, a guidance-based *straight line* path following control strategy for the snake robot is previously presented in [44] using a cascaded systems theory. However, [44] does not provide any results regarding the forward velocity v_t of the robot, and only assumes that the velocity of the robot is inside a positive constant range, i.e. $v_t \in [v_{\max}, v_{\min}]$ where v_{\max} and v_{\min} denote the maximum and minimum of the forward velocity, respectively. However, this is a restricting assumption since the velocity can change sign or go unstable. Furthermore, in order to solve the direction following and the maneuvering control problems, we need to control the velocity and position of the robot along the general curved paths. In this work, using the method of VHC we achieve these goals. Moreover, using numerical analysis and formal stability proofs we show that all the states of the controlled system remain bounded.

Organization of this chapter: This chapter is organized as follows. In Section 4.1, we solve the direction following control problem for the snake robot using the method of VHC. In Section 4.2, we will solve the maneuvering control problem along straight paths using the VHC approach. In Section 4.3, a control strategy for maneuvering along general curved paths is presented. Finally, in Section 4.4, simulation results for the proposed controllers of this chapter are presented which validate the theoretical approaches.

Publications: The results of this chapter are based on the conference papers [81], and [82].

Preliminary Theory: Reduction Problem for Asymptotic Stability

Throughout Chapters 4 and 5, we use the reduction theorems for stability of closed sets given in [98]. In this part, we present for completeness the version of the reduction theorem that we use in the thesis. This theorem may be used in applications in which the designer must simultaneously meet control specifications that can be formulated hierarchically [98]. Consider the dynamical system

$$\Sigma : \dot{x} = f(x, u) \quad (4.1)$$

with the state space $\mathcal{X} \subset \mathbb{R}^n$, where f is locally Lipschitz on \mathcal{X} , with the solution $\phi(t, x_0)$ at time t and initial condition $x(0) = x_0$, and $u(x)$ is a locally Lipschitz feedback which makes the sets $\Gamma_1 \subset \Gamma_2$ positively invariant for the closed-loop system. This invariance property implies that for all $x_0 \in \Gamma_i$, $i = 1, 2$, and for all $t \geq 0$, $\phi(t, x_0) \in \Gamma_i$. Furthermore, we say that the set Γ_1 is (globally) asymptotically stable *relative* to Γ_2 for Σ , provided that whenever $x_0 \in \Gamma_2$ then Γ_1 is (globally) asymptotically stable. In particular [98, Definition 4] states the definition of the stability of two sets relative to each other as follows.

Definition 4.1 *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 $\varepsilon > 0$, there exists a neighbourhood $\mathcal{N}(\Gamma_1)$ such that $\phi(\mathbb{R}_{>0}, \mathcal{N}(\Gamma_1) \cap \Gamma_2) \subset \mathcal{B}_\varepsilon(\Gamma_1)$.*

Note that in the above definition, \mathcal{B}_ε denotes the ε -ball given by the set $\mathcal{B}_\varepsilon = \{x \in \mathcal{X} : \|x\|_{\Gamma_1} < \varepsilon\}$, where $\|x\|_{\Gamma_1}$ denotes the point-to-set distance, see [98].

Now suppose that $\Gamma_1 \subset \Gamma_2 \subset \dots \subset \Gamma_l$ is a nested sequence of closed subsets of \mathcal{X} which represent hierarchical control specifications. According to [98], we say that specification 1, \dots , specification l are met when $x \in \Gamma_i$, i.e. the states enter the corresponding set which represents the specification. The property that $\Gamma_i \subset \Gamma_{i+1}$ induces a hierarchy of control specifications, where specification i is met only if specification $i + 1$ is met, and thus specification $i + 1$ has higher priority than specification i [98].

We state Part a of [98, Proposition 14], and we will use this to carry out parts of the control design in Chapter 4 and 5 of this thesis.

Proposition 4.1 [98] *Consider system (4.1), and assume that there exists a locally Lipschitz feedback $\bar{u}(x)$ making the sets $\Gamma_1 \subset \Gamma_2 \subset \dots \subset \Gamma_l$, positively invariant for the closed-loop system. Let $\Gamma_{l+1} := \mathcal{X}$. If, for $i = 1, \dots, l$, Γ_i is asymptotically stable relative to Γ_{i+1} for the closed-loop system, and Γ_1 is compact, then Γ_1 is asymptotically stable for the closed-loop system $\dot{x} = f(x, \bar{u}(x))$.*

Remark 4.1 For the case of a snake robot, the highest priority is the body shape control which induces the forward motion on the robot based on a desired gait pattern, and then the second priority is the orientation control which can drive the robot towards a target in the plane, and the third priority is the position control which makes the robot move along the desired path with a desired velocity profile. We will develop a hierarchical control approach based on these control objective priorities for the snake robot.

Review of the Simplified Model

In this part, we recall the simplified kinematic and dynamic models derived in [11] that were presented in Section 2.3. In this chapter, this model will be used for direction following and maneuvering control design for the snake robot. In particular, using the chosen generalized coordinates which were given as $q = [\phi_1, \dots, \phi_{N-1}, \theta, p_x, p_y]^T \in \mathbb{R}^{N+2}$, where ϕ_i denotes the i -th joint coordinate, θ denotes the orientation, and (p_x, p_y) denotes the planar position of the CM of the robot, the dynamic model of the robot is given as

$$\dot{\phi} = v_\phi \quad (4.2)$$

$$\dot{\theta} = v_\theta \quad (4.3)$$

$$\dot{p}_x = v_t \cos(\theta) - v_n \sin(\theta) \quad (4.4)$$

$$\dot{p}_y = v_t \sin(\theta) + v_n \cos(\theta) \quad (4.5)$$

$$\dot{v}_\phi = \bar{u} \quad (4.6)$$

$$\dot{v}_\theta = -\lambda_1 v_\theta + \frac{\lambda_2}{N-1} v_t \bar{e}^T \phi \quad (4.7)$$

$$\dot{v}_t = -\frac{c_t}{m} v_t + \frac{2c_p}{Nm} v_n \bar{e}^T \phi - \frac{c_p}{Nm} \phi^T A \bar{D} v_\phi \quad (4.8)$$

$$\dot{v}_n = -\frac{c_n}{m} v_n + \frac{2c_p}{Nm} v_t \bar{e}^T \phi \quad (4.9)$$

The simplified model (4.2)–(4.9) includes the essential components of the complex model of the snake robot. Furthermore, due to the approximation of the many of the nonlinear terms of the complex model with simpler linear terms, the simplified model is more amenable to the model-based locomotion control design which will be presented in the following sections.

4.1 Direction Following Control of Snake Robots based on a Simplified Model

4.1.1 Control Design Objectives

In this subsection, we present the control design objectives for the direction following controller which will be derived in this section. In direction following control (DFC), the objective is to regulate the linear velocity of the snake robot to a constant reference, while guaranteeing the boundedness of the system states. According to this definition, we define the following control objectives for the snake robot.

The first control objective concerns the body shape of the robot. Given the desired periodic body motions, i.e., a desired gait pattern, which we denote by $\phi_{\text{ref}}(t) : \mathbb{R}_{>0} \rightarrow \mathbb{R}^{N-1}$, we aim to asymptotically stabilize $\phi \rightarrow \phi_{\text{ref}}$ such that

$$\lim_{t \rightarrow \infty} \|\phi(t) - \phi_{\text{ref}}(t)\| = 0 \quad (4.10)$$

Furthermore, we aim to regulate the orientation of the robot to a constant reference orientation which we denote by $\theta_{\text{ref}} \in \mathbb{R}$. Thus, the second control objective is to asymptotically stabilize the orientation of the robot to the reference orientation such that

$$\lim_{t \rightarrow \infty} \|\theta(t) - \theta_{\text{ref}}\| = 0 \quad (4.11)$$

The third control objective concerns the velocity of the robot. In particular, we aim to practically stabilize (see e.g. [100]) the forward velocity of the robot to a constant reference forward velocity such that

$$\limsup_{t \rightarrow \infty} \|v_t(t) - v_{t,\text{ref}}\| \leq \epsilon_t \quad (4.12)$$

where $\epsilon_t \in \mathbb{R}_{>0}$ is any arbitrary positive constant. Meanwhile, we aim to drive the normal velocity to a small neighbourhood of the origin such that

$$\lim_{t \rightarrow \infty} \|v_n(t)\| \leq \epsilon_n \quad (4.13)$$

where $\epsilon_n \in \mathbb{R}_{>0}$ is a constant. Finally, we require that all the solutions of the controlled system remain bounded.

Remark 4.2 *Note that because of the oscillations that are necessary in order to control snake robots [11], the forward velocity of the CM will typically not be constant but will instead oscillate, and this is the reason why we instead of stabilization aim to practically stabilize it around the desired forward velocity.*

4.1.2 Body Shape Control

In this subsection, we propose a feedback control law for the body shape of the snake robot. In particular, we stabilize a desired gait pattern for the body shape variables, which induces lateral undulatory locomotion on the robot.

As we discussed in Chapter 3, it is well-known [1] that the gait pattern lateral undulation for an N -link snake robot will be achieved if every i -th joint of the robot moves in accordance with the reference joint trajectory given by

$$\phi_{\text{ref},i}(t) = \alpha \sin(\omega t + (i-1)\delta) + \phi_o \quad (4.14)$$

where α denotes the amplitude of the sinusoidal joint motion, ω denotes the frequency of the joint oscillations, and δ denotes a phase shift which is used to keep the joints out of phase. Furthermore, ϕ_o is an offset term which can be used for controlling the orientation of the robot in the plane.

In [11], based on analytical investigations using the averaging theory, it was shown that the forward velocity of a snake robot which moves based on the lateral undulatory gait induced by (4.14), is affected by the gait parameters (α, ω, δ) . Consequently, inspired by the work of [1] and [11], we introduce the following reference for the joint angles of the snake robot,

$$\phi_{\text{ref},i}(\lambda, \phi_o) = \alpha \sin(\lambda + (i-1)\delta) + \phi_o \quad (4.15)$$

where λ and ϕ_o are the solutions of two compensators which will be defined later in this chapter. In particular, we will use these compensators in order to control the forward velocity and orientation of the robot, respectively.

Inspired by the idea of VHC that has effectively been used for motion control of mechanical systems (see e.g. [92]–[96] for various examples), we consider (4.15) as a VHC for the body shape variables of the snake robot. Furthermore, these VHC will be enforced through the control input \bar{u} in (4.6). In particular, (4.15) is a dynamic VHC in that it depends on the state-evolution of dynamic compensators.

Associated with constraint functions (4.15), is the following constraint manifold

$$\Gamma = \left\{ (q, \dot{q}, \phi_o, \dot{\phi}_o, \lambda, \dot{\lambda}) \in \mathbb{R}^{2N+8} : \phi_i = \phi_{\text{ref},i}(\lambda, \phi_o), v_{\phi_i} = \dot{\lambda} \frac{\partial \phi_{\text{ref},i}}{\partial \lambda} + \dot{\phi}_o \frac{\partial \phi_{\text{ref},i}}{\partial \phi_o} \right\} \quad (4.16)$$

where $i \in \{1, \dots, N-1\}$. Our direction following control design approach for the snake robot is given in the following two steps:

1. In the first step, we use the control input \bar{u} in (4.6) to stabilize the solutions of the joint angle dynamics to the constraint manifold (4.16). This

stabilizes a lateral undulatory gait pattern among the shape variables of the robot. Consequently, this induces a forward motion based on the gait pattern lateral undulation for the robot.

2. In the second step, we restrict the dynamics of the system to the invariant constraint manifold (4.16), where we use λ and ϕ_o as two additional control terms, which will be used to control the velocity and orientation of the robot, respectively, cf. Figure 4.1.

4.1.3 Enforcing the VHC for the Shape Variables of the Robot

In order to stabilize the shape variables ϕ to the constraint manifold, we define the following controlled output vector

$$\tilde{\phi} = [\phi_1 - \phi_{\text{ref},1}, \dots, \phi_{N-1} - \phi_{\text{ref},N-1}]^T \in \mathbb{R}^{N-1} \quad (4.17)$$

The controlled output vector (4.17) yields a well-defined vector relative degree $\{2, \dots, 2\}$ everywhere on the configuration space. Consequently, we can stabilize the constraint manifold using an input-output linearizing feedback control law [93]. We define this control law as

$$\bar{u} = \ddot{\phi}_{\text{ref}} - K_d \dot{\tilde{\phi}} - K_p \tilde{\phi} \quad (4.18)$$

where $K_p = \text{diag}\{k_{p_i}\}_{i=1}^{N-1}$ and $K_d = \text{diag}\{k_{d_i}\}_{i=1}^{N-1}$ denote the positive definite diagonal matrices of the joint proportional and derivative controller gains, respectively. By inserting (4.18) into (4.6), the error dynamics equation for the joint angles of the robot takes the form

$$\ddot{\tilde{\phi}} + K_d \dot{\tilde{\phi}} + K_p \tilde{\phi} = 0 \quad (4.19)$$

which clearly has a globally exponentially stable equilibrium at the origin, i.e. $(\tilde{\phi}, \dot{\tilde{\phi}}) = (0_{N-1}, 0_{N-1})$. This implies that joint angle errors converge exponentially to zero, i.e. the constraint manifold is a globally exponentially stable manifold for the dynamical system (4.2)–(4.9), and the control objective (4.10) will be achieved.

4.1.4 Orientation Control

In this subsection, we control the orientation of the robot by using $\ddot{\phi}_o$ as an additional control input on the exponentially stable constraint manifold. To this end, we define the orientation error as

$$\tilde{\theta} = \theta - \theta_{\text{ref}} \quad (4.20)$$

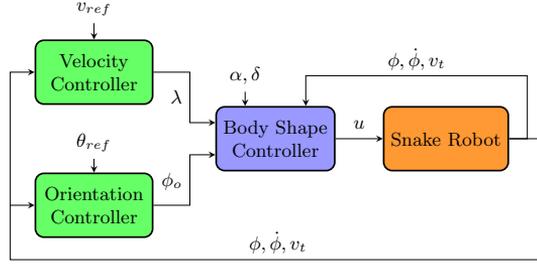


Figure 4.1: The Structure of the direction following controller.

where $\theta_{\text{ref}} \in \mathbb{R}$ denotes the constant reference orientation for the robot. Furthermore, we derive the orientation error dynamics of the robot evaluated on the constraint manifold. This can be done by writing (4.3) and (4.7) in the error coordinates $(\tilde{\phi}_1, \dots, \tilde{\phi}_{N-1}, \tilde{\theta})$, and then restricting it to the invariant manifold where $(\tilde{\phi}, \dot{\tilde{\phi}}) = (0_{N-1}, 0_{N-1})$. The resulting error dynamics has the form

$$\ddot{\tilde{\theta}} = -\lambda_1 \dot{\tilde{\theta}} + \frac{\lambda_2}{N-1} v_t \bar{e}^T S + \lambda_2 v_t \phi_o \quad (4.21)$$

where $S \in \mathbb{R}^{N-1}$ denotes a vector-valued function which is composed of the sinusoidal parts of the reference joint angles (4.15):

$$S = [\alpha \sin(\lambda), \dots, \alpha \sin(\lambda + (i-1)\delta)]^T \in \mathbb{R}^{N-1} \quad (4.22)$$

Motivated by the work of [11], where ϕ_o is used as the control input for the orientation dynamics of the snake robot, we use $\ddot{\phi}_o$ as a dynamic compensator which controls the orientation of the robot. In particular, since $\ddot{\phi}_o$ is needed for the joint control law (4.18), then it is more suitable to use this term rather than ϕ_o as the control input for the orientation. Note that if we choose ϕ_o as the control input, then we need to compute $(\dot{\phi}_o, \ddot{\phi}_o)$ which are very complex functions of time, numerically. We take the derivatives of (4.21) until the control input $\ddot{\phi}_o$ appears. The resulting dynamics is of the form

$$\tilde{\theta}^{(4)} = -\lambda_1 \tilde{\theta}^{(3)} + \psi_1(v_t, \phi_o) \ddot{\phi}_o + \psi_2(v_t, v_n, \phi_o, \dot{\phi}_o, \lambda, \dot{\lambda}, \ddot{\lambda}) \quad (4.23)$$

where $\psi_1(\cdot) \in \mathbb{R}$ and $\psi_2(\cdot) \in \mathbb{R}$ are two scalar-valued functions. Note that it is straightforward to derive $\psi_1(\cdot)$ and $\psi_2(\cdot)$ by taking the time-derivatives of (4.21), however, for clarity of presentation, we write them in the symbolic form. We define the input-output linearizing control law

$$\ddot{\phi}_o = \frac{1}{\psi_1} (\lambda_1 \tilde{\theta}^{(3)} - \psi_2 + \sigma) \quad (4.24)$$

where $\sigma \in \mathbb{R}$ is a new control input which we define as

$$\sigma = -k_3\tilde{\theta}^{(3)} - k_2\tilde{\theta}^{(2)} - k_1\tilde{\theta}^{(1)} - k_0\tilde{\theta} \quad (4.25)$$

where $k_0, k_1, k_2, k_3 > 0$ denote the orientation controller gains. It can be numerically verified that ψ_1 is bounded away from zero except for very small values of the forward velocity v_t , and this agrees well with the fact that the orientation is not controllable if the forward velocity of the snake robot is zero [45]. We stabilize the origin, i.e. $\tilde{\theta}^{(i)} = 0$ for all $i \in \{0, \dots, 4\}$, of the orientation error dynamics (4.21) by properly choosing the gains k_i , for instance according to the Routh-Hurwitz stability criterion. Furthermore, we show the boundedness of the solutions of the dynamic compensator (4.24) for the direction following controller through numerical simulations. We denote this bound by

$$\|[\phi_o, \dot{\phi}_o]\| \leq \varepsilon \quad (4.26)$$

where $\varepsilon \in \mathbb{R}_{>0}$ is a constant. In particular, we denote the upper-bound on each i -th reference joint angle, which is composed of a bounded sinusoidal part and the offset term ϕ_o , as

$$\|\phi_{\text{ref},i}\| \leq \varepsilon^* \quad (4.27)$$

where $\varepsilon^* \in \mathbb{R}_{>0}$ is a constant. We will employ these bounds for the stability analysis of the velocity controller in the next subsection.

Remark 4.3 *A formal proof for the boundedness of the solutions of the dynamic compensators which are used in order to control the orientation of the robot and are derived based on the simplified model of the snake robot remains as a theoretical gap in this thesis. However, at the end of this chapter we derive a static compensator for orientation control of the robot for which we analytically show that the solutions remain bounded. Moreover, in the subsequent chapter we derive dynamic compensators based on the complex model of the snake robot and we present formal proofs for the boundedness of the solutions of the dynamic compensator which controls the orientation of the robot.*

4.1.5 Velocity Regulation

In this subsection, we use the frequency of the joint oscillations as an additional control term to regulate the forward velocity v_t of the robot to a constant reference. Furthermore, we show that the normal velocity v_n of the robot converges to a small neighbourhood of the origin. To this end, we define the velocity errors for the normal and tangential components of the velocity of the CM of the robot as

$$\tilde{v}_t = v_t - v_{t,\text{ref}} \quad (4.28)$$

$$\tilde{v}_n = v_n - v_{n,\text{ref}} \quad (4.29)$$

where $v_{t,\text{ref}} \in \mathbb{R}_{>0}$ and $v_{n,\text{ref}} = 0$ denote the reference tangential and normal velocities, respectively. Furthermore, we derive the velocity error dynamics of the robot evaluated on the constraint manifold by writing (4.8)–(4.9) in the error coordinates, i.e. $(\tilde{\phi}_1, \dots, \tilde{\phi}_{N-1}, \tilde{v}_t, \tilde{v}_n)$, and then restricting them to the invariant constraint manifold, i.e. where $(\tilde{\phi}, \dot{\tilde{\phi}}) = (0_{N-1}, 0_{N-1})$, which yields

$$\dot{\tilde{v}}_t = -\frac{c_t}{m}(\tilde{v}_t + v_{t,\text{ref}}) + \frac{2c_p}{Nm}\tilde{v}_n\bar{e}^T\Phi_{\text{ref}} + \eta(\dot{\lambda}C + \dot{\phi}_o\bar{e}) \quad (4.30)$$

$$\dot{\tilde{v}}_n = -\frac{c_n}{m}\tilde{v}_n + \frac{c_p}{Nm}(\tilde{v}_t + v_{t,\text{ref}})\bar{e}^T\Phi_{\text{ref}} \quad (4.31)$$

where η , C , and Φ_{ref} denote the following vector-valued functions,

$$\eta = -\frac{c_p}{Nm}\Phi_{\text{ref}}^T A\bar{D} \in \mathbb{R}^{N-1} \quad (4.32)$$

$$C = [\alpha \cos(\lambda), \dots, \alpha \cos(\lambda + (i-1)\delta)]^T \in \mathbb{R}^{N-1} \quad (4.33)$$

$$\Phi_{\text{ref}} = [\phi_{\text{ref},1}, \dots, \phi_{\text{ref},N-1}]^T \in \mathbb{R}^{N-1} \quad (4.34)$$

In the following, we use

$$u_\lambda = \ddot{\lambda} \quad (4.35)$$

as a control input to regulate the linear velocities of the robot $[v_t, v_n]^T \in \mathbb{R}^2$ to the constant reference vector $v_{\text{ref}} = [v_{t,\text{ref}}, 0]^T \in \mathbb{R}^2$. In particular, we take

$$\dot{\lambda} = \frac{1}{\delta_1} \left(\frac{c_t}{m}v_{t,\text{ref}} - \frac{2c_p}{Nm}\tilde{v}_n\bar{e}^T\Phi_{\text{ref}} - k_\lambda\tilde{v}_t \right) \quad (4.36)$$

where

$$\delta_1(\phi_o, \lambda) = -\frac{c_p}{Nm}\Phi_{\text{ref}}^T A\bar{D}C = \eta C \quad (4.37)$$

and where $k_\lambda > 0$ denotes the proportional forward velocity controller gain. It can be numerically verified that $\delta_1(\cdot)$ is uniformly bounded away from zero, and this is because of the phase shift between the link references in (4.15). The following theorem investigates the stability of the origin of (4.30)–(4.31).

Theorem 4.1 *Stability characteristics of the origin of (4.30)–(4.31) are as follows:*

- (a) *The origin of the system (4.30)–(4.31) with λ given by the dynamic compensator (4.35)–(4.37) is stabilized provided that $k_\lambda > 0$ is chosen sufficiently high. Furthermore, $\dot{\lambda}$ remains uniformly bounded.*
- (b) *The practical stability of the origin $\tilde{v}_t = 0$ of (4.30), and convergence of the normal velocity error \tilde{v}_n , which is governed by the dynamical system (4.31), to a neighbourhood of the origin is achieved with the dynamic compensator (4.35)–(4.37), provided that $k_\lambda > 0$ is chosen sufficiently high. Furthermore, $\dot{\lambda}$ remains uniformly bounded.*

Proof of part (a): In order to prove the arguments of part (a), we iteratively introduce control-Lyapunov functions (CLF) borrowing from the techniques of backstepping (see e.g. [99]), and in addition including a dynamic compensator. In particular, we select the first CLF as

$$V_1 = \frac{1}{2} \tilde{v}_t^2 \quad (4.38)$$

Taking the time-derivative of (4.38) along the solutions of (4.30)–(4.31) yields

$$\begin{aligned} \dot{V}_1 &= \tilde{v}_t \dot{\tilde{v}}_t \\ &= \tilde{v}_t \left[-\frac{c_t}{m} v_{t,\text{ref}} - \frac{c_t}{m} \tilde{v}_t + \frac{2c_p}{Nm} \tilde{v}_n \bar{e}^T \Phi_{\text{ref}} + \eta (\dot{\lambda} C + \dot{\phi}_o \bar{e}) \right] \end{aligned} \quad (4.39)$$

We take $\dot{\lambda}$ defined in (4.36) as a virtual control input that we use to make (4.39) negative. For simplicity, we denote

$$\delta_2(\lambda, \phi_o, \tilde{v}_n, \tilde{v}_t) = \frac{1}{\delta_1} \left(\frac{c_t}{m} v_{t,\text{ref}} - \frac{2c_p}{Nm} \tilde{v}_n \bar{e}^T \Phi_{\text{ref}} - k_\lambda \tilde{v}_t \right) \quad (4.40)$$

For backstepping, we introduce the error variable

$$z = \dot{\lambda} - \delta_2(\lambda, \phi_o, \tilde{v}_n, \tilde{v}_t) \quad (4.41)$$

which we would like to drive to zero. The dynamic equation of the error variable (4.41) is given by

$$\dot{z} = u_\lambda - \dot{\delta}_2(\lambda, \dot{\lambda}, \phi_o, \dot{\phi}_o, \tilde{v}_n, \tilde{v}_t) \quad (4.42)$$

Note that it is straightforward to derive an analytical expression for $\dot{\delta}_2(\cdot)$, however, for the clarity of presentation we write it in the symbolic form. Furthermore, inserting $\dot{\lambda} = z + \delta_2(\cdot)$ into (4.39) yields

$$\dot{V}_1 = -k_\lambda \tilde{v}_t^2 - \frac{c_t}{m} \tilde{v}_t^2 + z \tilde{v}_t \delta_1 + \tilde{v}_t \eta \dot{\phi}_o \bar{e} \quad (4.43)$$

We introduce an augmented CLF of the form

$$V_2 = V_1 + \frac{1}{2} z^2 + \frac{1}{2} \tilde{v}_n^2 \quad (4.44)$$

Taking the time-derivative of V_2 along the solutions of (4.30)–(4.31) gives

$$\begin{aligned} \dot{V}_2 &= \dot{V}_1 + z \dot{z} + \tilde{v}_n \dot{\tilde{v}}_n \\ &= -k_\lambda \tilde{v}_t^2 - \frac{c_t}{m} \tilde{v}_t^2 + \tilde{v}_t \eta \dot{\phi}_o \bar{e} + z \left(u_\lambda - \dot{\delta}_2 + \tilde{v}_t \delta_1 \right) + \tilde{v}_n \dot{\tilde{v}}_n \end{aligned} \quad (4.45)$$

We define the control input u_λ in (4.45) as

$$u_\lambda = \dot{\delta}_2 - \tilde{v}_t \delta_1 - k_z z \quad (4.46)$$

where $k_z > 0$ is a constant gain. Inserting (4.46) into (4.45) yields

$$\dot{V}_2 = -k_\lambda \tilde{v}_t^2 - \frac{c_t}{m} \tilde{v}_t^2 - k_z z^2 + \tilde{v}_t \eta \dot{\phi}_o \bar{e} + \tilde{v}_n \dot{\tilde{v}}_n \quad (4.47)$$

The last two terms in (4.47) have indefinite signs. In particular, $\tilde{v}_n \dot{\tilde{v}}_n$ is of the form

$$\begin{aligned} \tilde{v}_n \dot{\tilde{v}}_n &= \tilde{v}_n \left(-\frac{c_n}{m} \tilde{v}_n + \frac{c_p}{Nm} (\tilde{v}_t + v_{t,\text{ref}}) \bar{e}^T \Phi_{\text{ref}} \right) \\ &= -\frac{c_n}{m} \tilde{v}_n^2 + \frac{c_p}{Nm} (\tilde{v}_n \tilde{v}_t + \tilde{v}_n v_{t,\text{ref}}) \bar{e}^T \Phi_{\text{ref}} \end{aligned} \quad (4.48)$$

Using the upper-bound (4.27), we can write (4.48) as

$$\tilde{v}_n \dot{\tilde{v}}_n \leq -\frac{c_n}{m} \tilde{v}_n^2 + \frac{2c_p}{Nm} (|\tilde{v}_n| |\tilde{v}_t| + |\tilde{v}_n| v_{t,\text{ref}}) (N-1) \varepsilon^* \quad (4.49)$$

In (4.49), we apply Young's inequality [101] where we have that

$$ab \leq \frac{\gamma a^2}{2} + \frac{b^2}{2\gamma} \quad (4.50)$$

where $a, b > 0$, and $\gamma \in \mathbb{R}_{>0}$ is any positive constant. Using this inequality, we can write (4.49) as

$$\tilde{v}_n \dot{\tilde{v}}_n \leq -\frac{c_n}{m} \tilde{v}_n^2 + \frac{2c_p}{Nm} \left(\frac{\gamma \tilde{v}_n^2}{2} + \frac{\tilde{v}_t^2}{2\gamma} + \frac{\gamma \tilde{v}_n^2}{2} + \frac{v_{t,\text{ref}}^2}{2\gamma} \right) (N-1) \varepsilon^* \quad (4.51)$$

Moreover, using the Young's inequality for the term $(\tilde{v}_t \eta \dot{\phi}_o \bar{e})$ in (4.47) we have that

$$\tilde{v}_t \eta \dot{\phi}_o \bar{e} \leq \left| \frac{c_p}{Nm} \bar{e}^T A \bar{D} \bar{e} \right| \varepsilon \varepsilon^* \left(\frac{\tilde{v}_t^2}{2\gamma} + \frac{\gamma}{2} \right) \quad (4.52)$$

For simplicity we denote

$$\left| \frac{c_p}{Nm} \bar{e}^T A \bar{D} \bar{e} \right| = \zeta \quad (4.53)$$

where $\zeta > 0$ is a constant. Using the inequalities (4.51)–(4.52) in (4.47), we obtain

$$\dot{V}_2 \leq -\left(k_\lambda + \frac{c_t}{m} - \frac{c_p(N-1)\alpha}{Nm\gamma} - \frac{\zeta}{2\gamma} \varepsilon \varepsilon^* \right) \tilde{v}_t^2 - k_z z^2 - \left(\frac{c_n}{m} - \frac{\gamma c_p(N-1)\varepsilon^*}{Nm} \right) \tilde{v}_n^2 + \varepsilon \quad (4.54)$$

where ϵ denotes the following constant

$$\epsilon = \frac{c_p(N-1)\varepsilon^*v_{t,\text{ref}}^2}{Nm\gamma} + \gamma\zeta\varepsilon\varepsilon^* \quad (4.55)$$

In order to make the coefficient of \tilde{v}_n^2 in (4.54) negative, we need to choose $\gamma < \frac{c_n N}{c_p(N-1)\varepsilon^*}$. For this choice of γ , we can always choose a sufficiently large k_λ such that the coefficient of \tilde{v}_t^2 in (4.54) will be negative as well. In this case we conclude that there exist a sufficiently small positive constant $\beta \in \mathbb{R}_{>0}$ such that the following inequality holds

$$\dot{V}_2 \leq -\beta V_2 + \epsilon \quad (4.56)$$

Consequently, a straightforward application of the Comparison Lemma gives

$$V_2(t) \leq V_2(0)e^{-\beta t} + \frac{\epsilon}{\beta} \quad (4.57)$$

From (4.57) we conclude that \tilde{v}_n , \tilde{v}_t , and z remain bounded. This implies that the solution exists globally. Moreover, according to (4.41) $\dot{\lambda}$ remains uniformly bounded, i.e. since z and δ_2 are bounded. Furthermore, V_2 converges to a ball of radius ϵ/β . Because of the quadratic form of (4.44), $\|\tilde{v}_n\|$ and $\|\tilde{v}_t\|$ converge to a ball of radius

$$r = \sqrt{2\epsilon/\beta} \quad (4.58)$$

Consequently, by choosing k_λ sufficiently high, we can drive $\|\tilde{v}_n\|$ and $\|\tilde{v}_t\|$ to a neighbourhood of the origin. This completes the proof of part (a) of Theorem 4.1.

So far we have shown that the origin $(\tilde{v}_t, \tilde{v}_n) = (0, 0)$ of (4.30)–(4.31) is stable. However, this is not enough to prove part (b) of Theorem 4.1. To prove part (b), in the following we will show that not only $(\tilde{v}_t, \tilde{v}_n) = (0, 0)$ is stable, but also it is possible to make $\|\tilde{v}_t\|$ converge to *any arbitrary small* neighbourhood of the origin, i.e. to be practically stable, and to make $\|\tilde{v}_n\|$ converge to a neighbourhood of the origin. We will show that this can be achieved by choosing k_λ sufficiently large.

Proof of part (b): Using the Comparison Lemma, in (4.57) we have shown that \tilde{v}_n , \tilde{v}_t , and z are bounded. We denote these bounds by

$$\|\tilde{v}_n\| \leq \delta_{v_n}, \quad \|\tilde{v}_t\| \leq \delta_{v_t}, \quad \|z\| \leq \delta_z \quad (4.59)$$

where δ_{v_n} , δ_{v_t} , and δ_z are positive constants. Using these bounds, the time-derivative of V_1 in (4.43) can be rewritten as

$$\dot{V}_1 \leq -k_\lambda \tilde{v}_t^2 - \frac{c_t}{m} \tilde{v}_t^2 + \delta_z |\tilde{v}_t| \epsilon_1 + |\tilde{v}_t \eta \dot{\phi}_o \bar{e}| \quad (4.60)$$

where ϵ_1 is a positive constant which denotes the upper-bound on δ_1 . Using Young's inequality for (4.60) we have that

$$\dot{V}_1 \leq -k_\lambda \tilde{v}_t^2 - \frac{c_t}{m} \tilde{v}_t^2 + \frac{(\delta_z \epsilon_1)^2}{2\gamma} + \frac{\gamma \tilde{v}_t^2}{2} + \zeta \epsilon \epsilon^* \left(\frac{\tilde{v}_t^2}{2\gamma} + \frac{\gamma}{2} \right) \quad (4.61)$$

By collecting the coefficients of \tilde{v}_t^2 , (4.61) can be written as

$$\begin{aligned} \dot{V}_1 &\leq - \left(k_\lambda + \frac{c_t}{m} - \frac{\gamma}{2} - \frac{\zeta \epsilon \epsilon^*}{2\gamma} \right) \tilde{v}_t^2 + \frac{(\delta_z \epsilon_1)^2}{2\gamma} + \frac{\gamma \zeta \epsilon \epsilon^*}{2} \\ &= -\beta^* V_1 + \frac{(\delta_z \epsilon_1)^2}{2\gamma} + \frac{\gamma \zeta \epsilon \epsilon^*}{2} \end{aligned} \quad (4.62)$$

where

$$\beta^* = 2 \left(k_\lambda + \frac{c_t}{m} - \frac{\gamma}{2} - \frac{\zeta \epsilon \epsilon^*}{2\gamma} \right) \quad (4.63)$$

is a positive constant. Consequently, a straightforward application of the Comparison Lemma yields

$$V_1(t) \leq V_1(0) e^{-\beta^* t} + \frac{(\delta_z \epsilon_1)^2}{2\gamma \beta^*} + \frac{\gamma \zeta \epsilon \epsilon^*}{2\beta^*} \quad (4.64)$$

From (4.64), it can be seen that V_1 converges to a ball of radius $\frac{(\delta_z \epsilon_1)^2}{2\gamma \beta^*} + \frac{\gamma \zeta \epsilon \epsilon^*}{2\beta^*}$. Because of (4.38), $\|\tilde{v}_t\|$ converges to a ball of the radius

$$r_1 = \sqrt{\frac{(\delta_z \epsilon_1)^2}{\gamma \beta^*} + \frac{\gamma \zeta \epsilon \epsilon^*}{\beta^*}} \quad (4.65)$$

Furthermore, we can choose k_λ sufficiently large to drive $\|\tilde{v}_t\|$ to any arbitrary small neighbourhood of the origin $\epsilon_t > 0$, i.e. by making β^* sufficiently large. This completes the proof of part (b) of Theorem 4.1, and the control objectives (4.12)–(4.13) will be achieved. ■

Remark 4.4 *It is interesting to note from (4.54) that it is the friction, given by the parameter c_n , that stabilizes the velocity in the normal direction v_n . We have no direct control over v_n , as the snake robot is underactuated, and the oscillations (4.15) that are induced by the $(N - 1)$ actuators, create a sideways velocity v_n . Thus, (4.54) indicates that the friction coefficient c_n needs to be sufficiently large for the system to be stable. This complies with the results concerning controllability of snake robots presented in [45].*

4.2 Maneuvering Control of Snake Robots along Straight Paths

In this section, we address the maneuvering control of planar snake robots based on the simplified dynamic model (4.2)–(4.9). In particular, we aim to make the robot converge to and follow a desired straight line path, and to regulate the forward velocity of the robot along the path to a constant reference velocity. In order to address this problem, we first stabilize the desired lateral undulatory gait pattern given by (4.15) for the fully-actuated body shape variables of the robot. Furthermore, we choose the gait parameters (λ, ϕ_o) using two dynamic compensators which control the orientation and the position of the robot in the plane. In particular, by solving the maneuvering problem, we control the body shape, orientation, and planar position of the robot along a straight path.

Model Transformation

For the maneuvering control design, we use the transformed form of the dynamics of the CM of the robot which was presented in Section 2.4. In particular, we use the following coordinate transformation [11]

$$\bar{p}_y = p_y + \epsilon \sin(\theta) \quad (4.66)$$

$$\bar{v}_n = v_n + \epsilon v_\theta \quad (4.67)$$

where

$$\epsilon = -\frac{2(N-1)c_p}{Nm\lambda_2} \quad (4.68)$$

is a negative constant. This change of coordinates transforms the dynamics of the position of the CM of the system into the form

$$\dot{\bar{p}}_y = v_t \sin(\theta) + v_n \cos(\theta) \quad (4.69)$$

$$\dot{\bar{v}}_n = Xv_\theta - Y\bar{v}_n \quad (4.70)$$

where

$$X = \epsilon \left(\frac{c_n}{m} - \lambda_1 \right) \quad (4.71)$$

$$Y = \frac{c_n}{m} \quad (4.72)$$

This change of coordinates removes the joint angle coupling from the dynamic model, and the resulting model is suitable for model-based maneuvering control design which is the subject of the subsequent subsections.

4.2.1 Control Design Objectives

In this subsection, we formulate the maneuvering control objectives for the proposed controllers in the following subsections. In general, the maneuvering problem consists of two tasks (see e.g. [104]). The first task is to converge to and follow a desired geometric path. This task is called the geometric task. The second task, which is called the dynamic task, consists of satisfying dynamical constraints, e.g. a desired velocity profile, along the desired path. In order to solve the maneuvering problem for the snake robot, we need to control the body shape, orientation, position, and velocity of the robot in the plane.

We start the control design formulation by defining a desired gait pattern, which is given by the vector function $\phi_{\text{ref}}(t) : \mathbb{R} \rightarrow \mathbb{R}^{N-1}$, among the fully-actuated body shape variables of the robot. In particular, we aim to asymptotically stabilize $\phi \rightarrow \phi_{\text{ref}}$ such that

$$\lim_{t \rightarrow \infty} \|\phi(t) - \phi_{\text{ref}}(t)\| = 0 \quad (4.73)$$

which is equivalent to body shape control of the robot.

The second control objective concerns the orientation of the robot in the plane. In particular, given a reference orientation $\theta_{\text{ref}}(t) : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}$, we aim to asymptotically stabilize $\theta \rightarrow \theta_{\text{ref}}$ such that

$$\lim_{t \rightarrow \infty} \|\theta(t) - \theta_{\text{ref}}(t)\| = 0 \quad (4.74)$$

which is equivalent to orientation control of the robot.

The third control objective concerns the planar position and linear velocity of the CM of the robot, i.e. the maneuvering control. In particular, we aim to solve the geometric task, i.e. to stabilize the position of the CM of the robot to a desired planar path. In order to formulate the maneuvering control objectives, we first define a desired straight line path, as a one dimensional manifold $\mathcal{P} \subset \mathbb{R}^2$, with coordinates in the $x - y$ plane given by the pair (p_{xd}, p_{yd}) . These coordinates are parametrized by a time-dependent variable $\Theta(t) : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$. Consequently, the desired path is defined as

$$\mathcal{P} = \{(p_{xd}(\Theta), p_{yd}(\Theta)) \in \mathbb{R}^2 : \Theta \geq 0\} \quad (4.75)$$

Furthermore, without loss of generality, we assume that the global x -axis is always aligned with the desired straight line path, i.e. $p_{yd}(\Theta) \equiv 0$. Thus, the geometric task is formulated as the convergence to the desired path such that

$$\lim_{t \rightarrow \infty} \|p_y(t)\| = 0 \quad (4.76)$$

To formulate the dynamic task, i.e. to regulate the linear velocity of the robot along the desired path to a desired positive constant velocity profile $v_{t,\text{ref}} \in \mathbb{R}_{>0}$, we define a reference position along the desired path $p_{t,\text{ref}}(t) \in \mathbb{R}$ with $\dot{p}_{t,\text{ref}} = v_{t,\text{ref}}$. The dynamic task is defined as

$$\lim_{t \rightarrow \infty} \|p_t(t) - p_{t,\text{ref}}(t)\| = 0. \quad (4.77)$$

Achieving (4.76)–(4.77) is equivalent to position and velocity control for the robot. This can also be regarded as an output trajectory tracking objective, but since we focus on controlling the motion *along the path* we think that maneuvering control is the most adequate term. Finally, we require that all the solutions of the controlled system remain uniformly bounded.

Solution Methodology

The blueprint of the maneuvering control design approach for the snake robot is given in the following four steps:

1. In the first step, we use the control input \bar{u} in (4.6) to stabilize the solutions of the dynamics of the body shape variables ϕ to the constraint manifold (4.16). This induces a forward motion based on the gait pattern lateral undulation on the robot.
2. In the second step, we reduce the dynamics of the system to the globally invariant constraint manifold (4.16), where we use ϕ_o as an additional control term, which will be used to control the orientation of the robot.
3. In the third step, we use the frequency of the periodic body motion, i.e. the gait pattern, as an additional control term to control forward velocity of the robot. This solves the dynamic task.
4. In the fourth step, we use the reference orientation such that the convergence of the position of the robot to the desired path is guaranteed. This solves the geometric task.

4.2.2 Body Shape and Orientation Control

Body Shape Control

For the body shape control of the snake robot, we use the same control law that we derived in (4.18). In particular, we consider the VHC defined in (4.15) as the reference joint angles for the robot. Associated with these VHC is the constraint

manifold (4.16) which we aim to exponentially stabilize for the solutions of the joint dynamics (4.6).

Using the control law (4.18), the error dynamics equation for the joint angles of the robot takes the exponentially stable form (4.19). This implies that joint angle errors exponentially converge to zero, i.e. the constraint manifold is a globally exponentially stable manifold for (4.6), and the control objective (4.73) will be achieved.

Orientation Control

In the following, we control the orientation of the robot to a reference angle defined a path following guidance law. To this end, we use ϕ_o as an additional control term on the exponentially stable constraint manifold. In particular, we use the following LOS guidance law to define the reference orientation for the robot

$$\theta_{\text{ref}} = -\text{atan2}\left(\frac{\bar{p}_y}{\Delta}\right) \quad (4.78)$$

where $\Delta > 0$ is a design parameter that is called the look-ahead-distance. The idea of the guidance law (4.78) is presented in Figure 4.2. A similar guidance law for snake robots is previously used in [44] where the path following control of snake robots is considered. In contrast, here we relax a restricting assumption in [44] on the forward velocity of the robot by regulating the forward velocity of the robot by using a dynamic compensator, and we solve the maneuvering control problem.

In order to control the orientation of the robot to the reference orientation defined by the LOS guidance law (4.78) we define the orientation error as

$$\tilde{\theta} = \theta - \theta_{\text{ref}} \quad (4.79)$$

Furthermore, we derive the orientation error dynamics of the robot evaluated on the constraint manifold. This can be done by writing (4.3) and (4.7) in the error coordinates, i.e. $(\tilde{\phi}_1, \dots, \tilde{\phi}_{N-1}, \tilde{\theta})$, and then reducing them to the invariant manifold where $(\tilde{\phi}, \dot{\tilde{\phi}}) = (0_{N-1}, 0_{N-1})$. The resulting error dynamics has the form

$$\ddot{\tilde{\theta}} = -\lambda_1 \ddot{\tilde{\theta}} - \lambda_1 \dot{\theta}_{\text{ref}} + \frac{\lambda_2}{N-1} v_t \bar{e}^T S + \lambda_2 v_t \phi_o - \ddot{\theta}_{\text{ref}} \quad (4.80)$$

where $S \in \mathbb{R}^{N-1}$ denotes the following vector which is composed of the sinusoidal parts of the reference joint angles (4.15):

$$S = [\alpha \sin(\lambda), \dots, \alpha \sin(\lambda + (i-1)\delta)]^T \in \mathbb{R}^{N-1} \quad (4.81)$$

In order to control the orientation of the robot, we use $\ddot{\phi}_o$ as a dynamic compensator. To this end, we take the time-derivatives of (4.80) until the control input

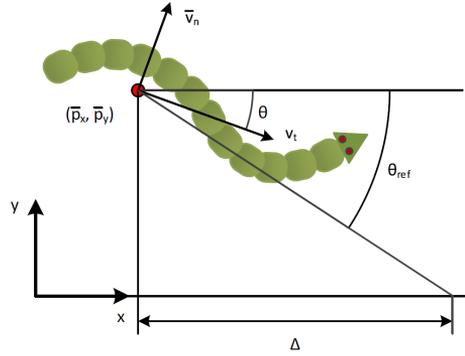


Figure 4.2: The LOS guidance law

$\ddot{\phi}_o$ appears. The resulting dynamics is of the form

$$\tilde{\theta}^{(4)} = -\lambda_1 \tilde{\theta}^{(3)} + \psi_1(v_t, \phi_o) \ddot{\phi}_o + \psi_2(v_t, v_n, \phi_o, \dot{\phi}_o, \lambda, \dot{\lambda}, \ddot{\lambda}) \quad (4.82)$$

Note that it is straightforward to derive $\psi_1(\cdot)$ and $\psi_2(\cdot)$ by taking the time-derivatives of (4.80), but for clarity of presentation, we write them in the symbolic form. We define the input-output linearizing control law

$$\ddot{\phi}_o = \frac{1}{\psi_1} (\lambda_1 \tilde{\theta}^{(3)} - \psi_2 + \sigma) \quad (4.83)$$

where $\sigma \in \mathbb{R}$ is a new control input which we define as

$$\sigma = -k_3 \tilde{\theta}^{(3)} - k_2 \tilde{\theta}^{(2)} - k_1 \tilde{\theta}^{(1)} - k_0 \tilde{\theta} \quad (4.84)$$

where $k_0, k_1, k_2, k_3 > 0$ denote the orientation controller gains. It can be numerically verified that $\psi_1(\cdot)$ is bounded away from zero except for very small values of the forward velocity v_t , and this agrees well with the fact that the orientation is not controllable if the forward velocity of the snake robot is zero [45]. We stabilize the origin, i.e. $\tilde{\theta}^{(i)} = 0$ for all $i \in \{0, \dots, 4\}$, of the orientation error dynamics by properly choosing the gains k_i . Furthermore, we show the boundedness of the solutions of the dynamic compensator (4.83) through numerical simulations. We denote this bound by

$$\|[\phi_o, \dot{\phi}_o]\| \leq \varepsilon \quad (4.85)$$

where $\varepsilon \in \mathbb{R}_{>0}$. In particular, we denote the upper-bound on each i -th reference joint angle, which is composed of a bounded sinusoidal part and the offset term

ϕ_o , as

$$\|\phi_{\text{ref},i}\| \leq \varepsilon^* \quad (4.86)$$

where $\varepsilon^* \in \mathbb{R}_{>0}$ is a constant.

4.2.3 Maneuvering Control

In this section, we perform the dynamic task of the maneuvering problem by utilizing the idea of velocity control given in [83]. To this end, we derive a dynamic compensator which controls the velocity and position of the robot along the desired path by using the frequency of the joint oscillations as an extra freedom in the control design. Moreover, following [44], we perform the geometric task by using the look-ahead-distance Δ in (4.78) as a control term. However, using our velocity controller we relax the restricting assumption in [44] on the forward velocity of the robot.

As a preliminary condition for the stability analysis of the maneuvering controller, we need to show that the normal velocity of the robot is uniformly bounded, which is the subject of the next theorem.

Theorem 4.2. *Under the joint controller (4.18) and the orientation controller (4.83), the normal velocity v_n of the robot is uniformly bounded.*

Proof: In order to show the boundedness of normal velocity v_n , we select the Lyapunov function

$$V = \frac{1}{2} \bar{v}_n^2 \quad (4.87)$$

Using (4.70), the time-derivative of (4.87) is given by

$$\dot{V} = \bar{v}_n \dot{\bar{v}}_n = \bar{v}_n (X v_\theta - Y \bar{v}_n) = X \bar{v}_n v_\theta - Y \bar{v}_n^2 \quad (4.88)$$

For the first right-hand side term we apply Young's inequality which yields

$$\dot{V} \leq -Y \bar{v}_n^2 + |X| \left(\frac{\gamma \bar{v}_n^2}{2} + \frac{v_\theta^2}{2\gamma} \right) = - \left(Y - \frac{\gamma |X|}{2} \right) \bar{v}_n^2 + \frac{|X| v_\theta^2}{2\gamma} \quad (4.89)$$

From the stability result of the previous subsection, and assuming that v_t has no finite escape time (we will clarify this assumption in Remark 4.5) we conclude that the second term in right-hand side of (4.89) is uniformly bounded. We denote this bound by

$$\frac{|X| v_\theta^2}{2\gamma} \leq \beta_1 \quad \forall t \geq 0 \quad (4.90)$$

where $\beta_1 \in \mathbb{R}_{>0}$ is a constant. By (4.72) Y is positive. Suppose now that we choose γ sufficiently small so that the coefficient of \bar{v}_n^2 is negative. In this case we conclude that there is a positive constant β_2 such that

$$\dot{V} \leq -\beta_2 V + \beta_1 \quad (4.91)$$

Therefore, it is straightforward to conclude from the Comparison Lemma that

$$V(t) \leq e^{-\beta_2 t} V(0) + \frac{\beta_1}{\beta_2} \quad (4.92)$$

From (4.92) we conclude that \bar{v}_n remains bounded, and converges to a ball of radius $\sqrt{2\frac{\beta_1}{\beta_2}}$. Since $v_n = \bar{v}_n + \epsilon v_\theta$, and v_θ is bounded, this also implies that v_n remains bounded. We denote this bound by $\|v_n\| \leq \epsilon_n$. ■

We will use this fact in the stability analysis of the forward velocity error in the following.

Dynamic Task along Straight Paths

In this subsection, we solve the dynamic task by controlling the position and velocity of the robot along the path. In particular, we use the frequency of the joint angle oscillations as an additional control term in order to regulate the forward velocity of the robot to a constant reference. To this end, we define the tangential position and velocity errors as

$$\tilde{p}_t = p_t - p_{t,\text{ref}}, \quad (4.93)$$

$$\tilde{v}_t = v_t - v_{t,\text{ref}} \quad (4.94)$$

Using (4.93)–(4.94), we derive the position and velocity error dynamics evaluated on the constraint manifold (4.16) as

$$\dot{\tilde{p}}_t = \tilde{v}_t \quad (4.95a)$$

$$\dot{\tilde{v}}_t = -\frac{c_t}{m}(\tilde{v}_t + v_{t,\text{ref}}) + \frac{2c_p}{Nm}v_n \bar{e}^T \Phi_{\text{ref}} + \eta (\dot{\lambda} C + \dot{\phi}_o \bar{e}) \quad (4.95b)$$

where η , C , and Φ_{ref} are defined in (4.32)–(4.34). In the following, we use

$$u_\lambda = \ddot{\lambda} \quad (4.96)$$

as a control input to stabilize the origin $(\tilde{p}_t, \tilde{v}_t) = (0, 0)$ of (4.95a)–(4.95b). In the following, we use the techniques of backstepping by selecting the first CLF of the form

$$V_1 = \frac{1}{2} \tilde{p}_t^2 \quad (4.97)$$

Taking the time-derivative of (4.97) along the solutions of (4.95a)–(4.95b) yields

$$\dot{V}_1 = \tilde{p}_t \tilde{v}_t \quad (4.98)$$

We take \tilde{v}_t as a virtual control input which we utilize to make (4.98) negative. In particular, we define

$$\tilde{v}_t = -k_{z_1} \tilde{p}_t \quad (4.99)$$

where $k_{z_1} > 0$ is a constant gain. We define the error variable

$$z_1 = \tilde{v}_t + k_{z_1} \tilde{p}_t \quad (4.100)$$

that we aim to drive to zero. Thus, we can rewrite (4.98) as

$$\dot{V}_1 = -k_{z_1} \tilde{p}_t^2 + z_1 \tilde{p}_t \quad (4.101)$$

To perform backstepping for z_1 , we write the error dynamics for the error variable which has the form

$$\dot{z}_1 = \dot{\tilde{v}}_t + k_{z_1} \tilde{v}_t \quad (4.102)$$

We choose an augmented CLF of the form

$$V_2 = V_1 + \frac{1}{2} z_1^2 \quad (4.103)$$

Taking the time-derivative of V_2 along the solutions of (4.95a)–(4.95b) yields

$$\begin{aligned} \dot{V}_2 &= -k_{z_1} \tilde{p}_t^2 + z_1 (\tilde{p}_t + \dot{\tilde{v}}_t + k_{z_1} \tilde{v}_t) \\ &= -k_{z_1} \tilde{p}_t^2 + z_1 (\tilde{p}_t - \frac{c_t}{m} z_1 + \frac{c_t}{m} k_{z_1} \tilde{p}_t - \frac{c_t}{m} v_{t,\text{ref}} + \frac{2c_p}{Nm} v_n \bar{e} \Phi_{\text{ref}} + \eta C \dot{\lambda} \\ &\quad + \eta \bar{e} \dot{\phi}_o + k_{z_1} \tilde{v}_t) \end{aligned} \quad (4.104)$$

We denote

$$\delta_1(\phi_o, \lambda) = \eta C \quad (4.105)$$

It can be numerically verified that $\delta_1(\cdot)$ is uniformly bounded away from zero, and this is because of the phase shift between the link references in (4.15). We take $\dot{\lambda}$ as a virtual control input that we use to make (4.104) negative:

$$\dot{\lambda} = \frac{1}{\delta_1} \left(-\tilde{p}_t + \frac{c_t}{m} v_{t,\text{ref}} - \frac{c_t}{m} k_{z_1} \tilde{p}_t - \frac{2c_p}{Nm} v_n \bar{e} \Phi_{\text{ref}} - k_{z_1} \tilde{v}_t - k_{z_2} z_1 \right) \quad (4.106)$$

where $k_{z_2} > 0$ is a constant gain. For simplicity, we denote

$$\delta_2(\phi_o, \lambda, p_t, v_t) = \frac{1}{\delta_1} \left(-\tilde{p}_t + \frac{c_t}{m} v_{t,\text{ref}} - \frac{c_t}{m} k_{z_1} \tilde{p}_t - \frac{2c_p}{Nm} v_n \bar{e} \Phi_{\text{ref}} - k_{z_1} \tilde{v}_t - k_{z_2} z_1 \right) \quad (4.107)$$

We define the second error variable as

$$z_2 = \dot{\lambda} - \delta_2 \quad (4.108)$$

which we aim to drive to zero. Inserting $\dot{\lambda} = z_2 + \delta_2$ into (4.104) yields

$$\dot{V}_2 = -k_{z_1} \tilde{p}_t^2 - \left(\frac{c_t}{m} + k_{z_2} \right) z_1^2 + z_1 z_2 \delta_1 + z_1 \eta \bar{e} \dot{\phi}_o \quad (4.109)$$

To perform backstepping for z_2 , we write the error dynamics for the error variable z_2 which has the form

$$\dot{z}_2 = u_\lambda - \dot{\delta}_2 \quad (4.110)$$

We choose the augmented CLF in the form

$$V_3 = V_2 + \frac{1}{2} z_2^2 \quad (4.111)$$

The time-derivative of V_3 along the solutions of (4.95a)–(4.95b) is of the form

$$\dot{V}_3 = -k_{z_1} \tilde{p}_t^2 - \left(\frac{c_t}{m} + k_{z_2} \right) z_1^2 + z_2 \left(z_1 \delta_1 + u_\lambda - \dot{\delta}_2 \right) + z_1 \eta \bar{e} \dot{\phi}_o \quad (4.112)$$

We define the control input u_λ as

$$u_\lambda = -z_1 \delta_1 + \dot{\delta}_2 - k_{z_3} z_2 \quad (4.113)$$

where $k_{z_3} > 0$ is a constant gain. Inserting (4.113) into (4.112) yields

$$\dot{V}_3 = -k_{z_1} \tilde{p}_t^2 - \left(\frac{c_t}{m} + k_{z_2} \right) z_1^2 - k_{z_3} z_2^2 + z_1 \eta \bar{e} \dot{\phi}_o \quad (4.114)$$

Only the last term in (4.114) has indefinite sign. For this term, we apply Young's inequality and we can write

$$|z_1| |\eta \bar{e} \dot{\phi}_o| \leq \zeta \left(\frac{\gamma_1}{2} + \frac{z_1^2}{2\gamma_1} \right) \quad (4.115)$$

where $\gamma_1 > 0$ is a constant, and

$$\zeta = \left| -\frac{c_p}{Nm} \bar{e}^T A \bar{D} \bar{e} \right| \varepsilon \varepsilon^* \quad (4.116)$$

is a positive constant. Consequently, (4.114) will be of the form

$$\dot{V}_3 \leq -k_{z_1} \tilde{p}_t^2 - \left(\frac{c_t}{m} + k_{z_2} - \frac{\zeta}{2\gamma_1} \right) z_1^2 - k_{z_3} z_2^2 + \frac{\gamma_1 \zeta}{2} \quad (4.117)$$

From (4.117) we conclude that there exists a sufficiently small positive constant $\beta \in \mathbb{R}_{>0}$ such that

$$\dot{V}_3 \leq -\beta V_3 + \frac{\gamma_1 \zeta}{2} \quad (4.118)$$

Consequently, a straightforward application of the Comparison Lemma implies that

$$V_3(t) \leq V_3(0)e^{-\beta t} + \frac{\gamma_1 \zeta}{2\beta} \quad (4.119)$$

From (4.119) we conclude that V_3 converges to a ball of radius

$$r = \frac{\gamma_1 \zeta}{2\beta} \quad (4.120)$$

Furthermore, because of (4.111), z_2 , z_1 , and \tilde{p}_t converge to a ball of radius $\sqrt{2r}$. Moreover, because of the boundedness of z_2 and $\delta_2(\cdot)$ in (4.107), $\dot{\lambda}$ remains bounded. In addition, we can drive the position and velocity errors to any arbitrarily small neighbourhood of the origin $\epsilon^* \in \mathbb{R}$ by taking

$$\gamma_1 = \frac{2\beta\epsilon^{*2}}{\zeta} \quad (4.121)$$

along with choosing a sufficiently large k_{z_2} , i.e. such that the coefficient of z_1^2 in (4.117) will be negative. This implies that the origin $(\tilde{p}_t, \tilde{v}_t) = (0, 0)$ of (4.95a)–(4.95b) is practically stable, and the control objective (4.77), i.e. the dynamic task, will be achieved.

Remark 4.5 Under the the controllers (4.18), (4.83), and (4.113), with the augmented state vector

$$x = [p_t, v_t, p_n, v_n, \phi_o, \dot{\phi}_o, \lambda, \dot{\lambda}] \in \mathbb{R}^2 \quad (4.122)$$

the closed-loop tangential and normal position dynamics along with the dynamics of the compensators take the form $\dot{x} = f(x)$. Since all the functions in the closed-loop were shown to remain bounded, then it can be verified that

$$\|f(x)\| \leq K(1 + \|x\|) \quad (4.123)$$

where $K \in \mathbb{R}_{>0}$ is a constant. This linear growth condition implies that any of the components of the state vector (4.122) have no finite escape time, which validates the results presented in this section which were derived based on this assumption.

Geometric Task for Straight Paths

In this subsection, we use the look-ahead distance Δ as an additional control term to address the geometric task, i.e. to stabilize the position of the CM of the robot to the desired path \mathcal{P} such that $(\bar{p}_y, \bar{v}_n) \rightarrow (0, 0)$. To this end we define the position and velocity cross-track errors

$$\tilde{p}_y = \bar{p}_y \quad (4.124)$$

$$\tilde{v}_n = \bar{v}_n \quad (4.125)$$

which we would like to drive to zero and thereby achieve control objective (4.76). This will imply that the robot will converge to the global x-axis which we defined as the desired straight line path. A similar approach is used in [44], where cascaded systems theory is used to show that for a properly chosen look-ahead distance Δ , the desired path is globally asymptotically stable for (4.69)–(4.70). However, [44] does not provide any results regarding the forward velocity of the robot, and only assumes that the tangential velocity of the robot is inside a positive constant range, i.e. $v_t \in [v_{\max}, v_{\min}]$ where v_{\max} and v_{\min} denote the maximum and minimum of the forward velocity, respectively. In the previous subsection, we designed a dynamic compensator which enables us to control the forward velocity of the robot, and thus we relax this assumption. The following theorem is a reformed version of the theorem given in [44].

Theorem 4.3 *The controller defined by (4.18), (4.83), and (4.113) globally asymptotically and locally exponentially stabilizes the desired straight path \mathcal{P} (4.75) for the dynamical system (4.2)–(4.9), provided that the look-ahead distance Δ satisfies*

$$\Delta > \frac{|X|}{|Y|} \left(1 + \frac{v_{\max}}{v_{\min}} \right) \quad (4.126)$$

Proof: The proof of Theorem 4.3 follows from the proof of [11, Th. 8.2], together with the proof of stabilization of \tilde{v}_t given in the previous subsection. ■

4.3 Maneuvering Control of Snake Robots along Curved Paths

The principal goal of this section is to design a feedback control law for a snake robots to converge to and follow a desired continuously differentiable curved path while satisfying a desired speed profile. To this end, we use the dynamic

feedback control law (4.18) which controls the body shape of the robot to the desired gait pattern provided by (4.15). Furthermore, we define the parameters of this gait pattern using a static and a dynamic compensator which will be used for controlling the orientation and position of the robot in the plane. In order to solve the maneuvering control problem along curved line paths, in this section we use the reduction theory for asymptotic stability of the closed sets from [98] that we reviewed in the beginning of this chapter.

4.3.1 Control Design Objectives

We start the control objectives formulation by defining a desired gait pattern given by the vector function $\phi_{\text{ref}}(t) : \mathbb{R} \rightarrow \mathbb{R}^{N-1}$, for the fully-actuated body shape variables of the robot. In particular, we aim to asymptotically stabilize $\phi \rightarrow \phi_{\text{ref}}$ such that

$$\lim_{t \rightarrow \infty} \|\phi(t) - \phi_{\text{ref}}(t)\| = 0 \quad (4.127)$$

Achieving (4.127) is equivalent to body shape control of the robot.

The second control objective concerns the orientation of the robot in the plane. In particular, given a reference orientation $\theta_{\text{ref}}(t) : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}$, we aim to asymptotically stabilize $\theta \rightarrow \theta_{\text{ref}}$ such that

$$\lim_{t \rightarrow \infty} \|\theta(t) - \theta_{\text{ref}}(t)\| = 0 \quad (4.128)$$

Achieving (4.128) is equivalent to orientation control of the robot.

The third control objective concerns the planar position and linear velocity of the CM of the robot, i.e. the maneuvering control. In particular, we aim to solve the geometric task, i.e. to stabilize a desired planar path, for the position of the robot. In order to formulate the maneuvering control objectives, we first define a desired path, as a one-dimensional manifold $\mathcal{P} \subset \mathbb{R}^2$, with coordinates in the $x - y$ plane given by the pair $\xi = (p_{xd}, p_{yd}) \in \mathbb{R}^2$. These coordinates are parametrized by a time-dependent variable $\Theta(t) : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$. Consequently, the desired path is defined as

$$\mathcal{P} = \{(p_{xd}(\Theta), p_{yd}(\Theta)) \in \mathbb{R}^2 : \Theta \geq 0\} \quad (4.129)$$

where $\dot{\Theta} > 0$, i.e. forward motion along the path. Thus, the geometric task is formulated as practical convergence to the desired path such that

$$\lim_{t \rightarrow \infty} \sup \|\bar{p}_y(t) - \bar{p}_{yd}(\Theta)\| \leq \epsilon_p \quad (4.130)$$

where $\bar{p}_{yd} = p_{yd} + \epsilon \sin \theta$ is the projection of the desired path, which implies that the normal distance of the position of the CM of the robot with the desired path is zero.

Furthermore, we aim to control the velocity of the robot along the path. To regulate the forward velocity of the robot along the desired path to a desired velocity profile $v_{t,\text{ref}}(t) : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$, we define a reference position along the desired path $p_{t,\text{ref}}(t) : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ with $\dot{p}_{t,\text{ref}}(t) = v_{t,\text{ref}}(t)$. The dynamic task is defined as

$$\lim_{t \rightarrow \infty} \|p_t(t) - p_{t,\text{ref}}(t)\| = 0 \quad (4.131)$$

Achieving (4.128), (4.130) and (4.131) *all together* is equivalent to position and velocity control for the robot. Note that because of (4.128) the robot is aligned with the path, and because of (4.130) the normal distance of the robot with the path is zero, and because of (4.131) the robot will move along the path based on a desired velocity profile, i.e. the tangential position and velocity errors converge to zero.

Finally, we require that all the solutions of the controlled system remain uniformly bounded.

4.3.2 Body Shape and Orientation Control

In this subsection, we control the body shape of the robot to the desired gait pattern provided by the reference joint angle (4.15). Furthermore, we use the gait parameter ϕ_o as an additional input which will be designed in the form of a *static* compensator in order to control the orientation of the robot to a reference orientation angle defined by a path following guidance law. A similar orientation controller is used in [11]. However, in contrast with [11], here we present a stability proof in order to show that the solution of this compensator remains bounded.

Body Shape Control

In order to control the body shape of the robot to the desired lateral undulatory gait pattern provided by (4.15), we utilize the exponentially stabilizing joint control law (4.18). In particular, using this control law, we exponentially stabilize the closed-loop solutions of the joint angles dynamics (4.6) to the constraint manifold which is defined as

$$\Gamma_4 = \left\{ (q, \dot{q}, \phi_o, \dot{\phi}_o, \lambda, \dot{\lambda}) \in \mathbb{R}^{2N+8} : \phi_i = \phi_{\text{ref},i}(\lambda, \phi_o), v_{\phi_i} = \dot{\lambda} \frac{\partial \phi_{\text{ref},i}}{\partial \lambda} + \dot{\phi}_o \frac{\partial \phi_{\text{ref},i}}{\partial \phi_o} \right\} \quad (4.132)$$

where $i \in \{1, \dots, N-1\}$. Exponential stability of the constraint manifold (4.132) implies that the robot moves based on the gait pattern lateral undulation, and the control objective (4.127) will be achieved.

The Path Following Guidance Law

In this part, we define a reference orientation for the robot by using a LOS guidance law. A guidance-based path following control strategy for snake robots is previously presented in [11] which is valid for straight line paths. The approach presented in this section is motivated by [11], but in contrast we here solve the path following problem, i.e. the geometric task, for general curved line paths. Moreover, we relax a restricting assumption in [11] on the forward velocity of the robot by controlling the forward velocity by using a dynamic compensator, and we extend the path following approach to solve the maneuvering control problem.

To define the LOS path following guidance law, we take the normal distance between the CM of the robot along the y -axis with the desired path, which is defined as

$$\tilde{p}_y = \bar{p}_y - \bar{p}_{yd}(\Theta) \quad (4.133)$$

We then define the LOS path following guidance law, giving the reference orientation for the robot, as a function of the cross-track error as

$$\theta_{\text{ref}} = -\text{atan2}\left(\frac{\tilde{p}_y}{\Delta}\right) \quad (4.134)$$

where $\Delta > 0$ is a design parameter that will be used as a control term for the position of the robot in this section.

Stabilizing the Reference Orientation using a Static Compensator

In this part, we control the orientation of the robot by using ϕ_o as an additional control input on the exponentially stable constraint manifold. To this end, we define the orientation error as

$$\tilde{\theta} = \theta - \theta_{\text{ref}} \quad (4.135)$$

Furthermore, we derive the orientation error dynamics of the robot evaluated on the constraint manifold which has the form

$$\ddot{\tilde{\theta}} = -\lambda_1 \dot{\tilde{\theta}} - \lambda_1 \dot{\theta}_{\text{ref}} + \frac{\lambda_2}{N-1} v_t \bar{e}^T S + \lambda_2 v_t \phi_o - \ddot{\theta}_{\text{ref}} \quad (4.136)$$

where $S \in \mathbb{R}^{N-1}$ denotes the following vector which is composed of the sinusoidal parts of the reference joint angles (4.15):

$$S = [\alpha \sin(\lambda), \dots, \alpha \sin(\lambda + (i-1)\delta)]^T \in \mathbb{R}^{N-1} \quad (4.137)$$

We define the following *orientation control manifold*, which we aim to exponentially stabilize relative to Γ_4 ,

$$\Gamma_3 = \left\{ (\theta, v_\theta, \phi_o, \dot{\phi}_o, v_t, \lambda) \in \Gamma_4 : \begin{pmatrix} \tilde{\theta} \\ \dot{\tilde{\theta}} \end{pmatrix} = (0, 0), \left\| [\phi_o, \dot{\phi}_o] \right\| \leq \epsilon_\phi \right\} \quad (4.138)$$

where $\epsilon_\phi > 0$ is a positive constant. Note that stabilizing Γ_3 relative to Γ_4 implies that the orientation error converges exponentially to zero on the constraint manifold, and our control goal with the second priority will be achieved. Furthermore, we will show that the solutions of the static compensator which controls the orientation of the robot remain uniformly bounded.

In order to stabilize the origin $(\tilde{\theta}, \dot{\tilde{\theta}}) = (0, 0)$ of (4.136), we define the additional control input ϕ_o as

$$\phi_o = \frac{1}{\lambda_2 v_t} \left(-\frac{\lambda_2}{N-1} v_t \bar{e}^T S + \lambda_1 \dot{\theta}_{\text{ref}} + \ddot{\theta}_{\text{ref}} - k_\theta \tilde{\theta} \right) \quad (4.139)$$

where $k_\theta > 0$ denotes the proportional orientation controller gain. Note that on the constraint manifold, where a lateral undulatory gait is stabilized, the tangential velocity v_t is a positive constant and consequently (4.137) is well-defined. By inserting (4.139) into (4.136), the controlled orientation error dynamics of the robot evaluated on the constraint manifold takes the form

$$\ddot{\tilde{\theta}} + \lambda_1 \dot{\tilde{\theta}} + k_\theta \tilde{\theta} = 0 \quad (4.140)$$

which clearly has a globally exponentially stable equilibrium at the origin $(\tilde{\theta}, \dot{\tilde{\theta}}) = (0, 0)$. This implies that the control objective (4.126) will be achieved.

Remark 4.6 *Following the approach presented in [89], in order to compute the derivatives $(\dot{\theta}_{\text{ref}}, \ddot{\theta}_{\text{ref}})$ of the reference orientation (4.134) we use the low-pass filtering reference model given by*

$$\frac{d}{dt} \begin{bmatrix} \theta_{\text{ref}} \\ \dot{\theta}_{\text{ref}} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\psi_f \omega_n \end{bmatrix} \begin{bmatrix} \theta_{\text{ref}} \\ \dot{\theta}_{\text{ref}} \end{bmatrix} + \begin{bmatrix} 0 \\ \omega_n^2 \end{bmatrix} \theta_{\text{ref}} \quad (4.141)$$

with natural frequency ω_n , and damping ratio ψ_f . This filter is an input-to-state stable (ISS) system. Since θ_{ref} is uniformly bounded, this ISS property implies that the outputs $(\dot{\theta}_{\text{ref}}, \ddot{\theta}_{\text{ref}})$ are bounded. We denote this bound by $\left\| [\dot{\theta}_{\text{ref}}, \ddot{\theta}_{\text{ref}}] \right\| \leq \epsilon_r$, where $\epsilon_r \in \mathbb{R}_{>0}$ is a positive constant.

Remark 4.7 Assuming that v_t has no finite-escape time, see Remark 4.9, and since the variables $(\hat{\theta}_{\text{ref}}, \dot{\theta}_{\text{ref}})$ are bounded, the solutions of the static compensator (4.139) will be uniformly ultimately bounded by the following ultimate bound

$$\|\phi_o\| \leq \alpha + \epsilon_r \quad (4.142)$$

where α denotes the amplitude of the reference joint angles (4.15).

Remark 4.8 The second order time-derivative of the control input ϕ_o is needed for the joint control law (4.18). However, $\dot{\phi}_o$ and $\ddot{\phi}_o$ are complex functions of time that cannot be easily computed analytically, see [11]. In order to compute this term, we take the approach given in [89], by using a second order low-pass filtering reference model. In particular, we compute these time-derivatives by passing ϕ_o through a low-pass filter of the form

$$\frac{d}{dt} \begin{bmatrix} \phi_o \\ \dot{\phi}_o \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\psi_f\omega_n \end{bmatrix} \begin{bmatrix} \phi_o \\ \dot{\phi}_o \end{bmatrix} + \begin{bmatrix} 0 \\ \omega_n^2 \end{bmatrix} \phi_o \quad (4.143)$$

This filter is an ISS system, see e.g. [99]. This implies that the output $\dot{\phi}_o$ remains bounded. Consequently, for the two other dynamical subsystems which govern the dynamics of the position of the CM of the robot, i.e. (4.8)–(4.9), we take $\dot{\phi}_o$ as a bounded exogenous signal which will be cancelled through the action of a dynamic compensator given by (4.148) which will be designed to control the position of the robot in the next subsection.

We collect the results of the arguments presented in this subsection in the following theorem.

Theorem 4.4 The control law governed by the solution of the static compensator (4.139), asymptotically stabilizes Γ_3 relative to Γ_4 . Furthermore, provided that v_t has no finite-escape time, see Remark 4.9, the solutions of the static compensator (4.139) remain uniformly ultimately bounded by the ultimate bound (4.142).

4.3.3 Maneuvering Control along Curved Paths: The Dynamic Task

In this part, we perform the dynamic task of the maneuvering problem by utilizing the idea of velocity control for snake robots given in [83]. To this end, we derive a dynamic compensator which controls the velocity and position of the robot along the desired path by using the frequency of the joint oscillations as an additional control term.

In particular, we define a *velocity control manifold* which we aim to exponentially stabilize relative to the constraint manifold Γ_4 for the closed-loop system as

$$\Gamma_2 = \left\{ \left(\theta, \dot{\theta}, p_t, v_t, v_n, \phi_o, \dot{\phi}_o, \lambda, \dot{\lambda} \right) \in \Gamma_4 : \right. \\ \left. \left(\tilde{\theta}, \dot{\tilde{\theta}} \right) = (0, 0), (\tilde{p}_t, \tilde{v}_t) = (0, 0), \|v_n\| \leq \epsilon_n, \left\| \left[\phi_o, \dot{\phi}_o \right] \right\| \leq \epsilon_\phi, \left\| \left[\lambda, \dot{\lambda} \right] \right\| \leq \epsilon_\lambda \right\} \quad (4.144)$$

where $\epsilon_n > 0$, $\epsilon_\phi > 0$, and $\epsilon_\lambda > 0$ are positive constants. Thus, stabilizing Γ_2 relative to the constraint manifold Γ_4 implies that the robot will follow the reference orientation defined by (4.134), and a reference velocity which will be defined below. Furthermore, the solutions of the static compensator which controls the orientation of the robot remain bounded. Moreover, the solutions of the dynamic compensator which will be designed in this section to control the forward velocity of the robot will remain bounded.

Using the same approach as presented in Theorem 4.2, it can be shown that under the controllers (4.18) and (4.139), the normal velocity of the robot remains bounded. We denote this bound by $\|v_n\| < \epsilon_n$.

Performing the Dynamic Task

In this part, we address the dynamic task by controlling the position and velocity of the robot along the desired path. In particular, we use the frequency of the joint angle oscillations as an additional control term in order to control the forward velocity of the robot to a reference velocity. To this end, we define the tangential position and velocity errors as

$$\tilde{p}_t = p_t - p_{t,\text{ref}} \quad (4.145)$$

$$\tilde{v}_t = v_t - v_{t,\text{ref}} \quad (4.146)$$

Furthermore, we derive the position and velocity error dynamics evaluated on the constraint manifold (4.132) as

$$\dot{\tilde{p}}_t = \tilde{v}_t \\ \dot{\tilde{v}}_t = -\frac{c_t}{m}(\tilde{v}_t + v_{t,\text{ref}}) + \frac{2c_p}{Nm}v_n \bar{e}^T \Phi_{\text{ref}} + \eta \left(\dot{\lambda} C + \dot{\phi}_o \bar{e} \right) - \dot{v}_{t,\text{ref}} \quad (4.147)$$

where η , C , and Φ_{ref} , are given by (4.32)–(4.34). In the following, we use

$$u_\lambda = \ddot{\lambda} \quad (4.148)$$

as the control input to stabilize the origin $(\tilde{p}_t, \tilde{v}_t) = (0, 0)$ of (4.147). In order to use the backstepping techniques, we select the first CLF of the form

$$V_1 = \frac{1}{2} \tilde{p}_t^2 \quad (4.149)$$

Taking the time-derivative of (4.149) along the solutions of (4.147) yields

$$\dot{V}_1 = \tilde{p}_t \tilde{v}_t \quad (4.150)$$

We take \tilde{v}_t as a virtual control input which we utilize to make (4.150) negative. In particular, we define

$$\tilde{v}_t = -k_{z_0} \tilde{p}_t \quad (4.151)$$

where $k_{z_0} > 0$ is a constant gain. We define the error variable

$$z_1 = \tilde{v}_t + k_{z_0} \tilde{p}_t \quad (4.152)$$

that we aim to drive to zero. Thus, we can rewrite (4.150) as

$$\dot{V}_1 = -k_{z_0} \tilde{p}_t^2 + z_1 \tilde{p}_t \quad (4.153)$$

To perform backstepping for z_1 , we write the error dynamics for the error variable which has the form

$$\dot{z}_1 = \dot{\tilde{v}}_t + k_{z_0} \tilde{v}_t \quad (4.154)$$

We choose an augmented CLF of the form

$$V_2 = V_1 + \frac{1}{2} z_1^2 \quad (4.155)$$

Taking the time-derivative of V_2 along the solutions of (4.147) yields

$$\begin{aligned} \dot{V}_2 &= -k_{z_0} \tilde{p}_t^2 + z_1 (\tilde{p}_t + \dot{\tilde{v}}_t + k_{z_0} \tilde{v}_t) \\ &= -k_{z_0} \tilde{p}_t^2 + z_1 (\tilde{p}_t - \frac{c_t}{m} z_1 - \frac{c_t}{m} v_{t,\text{ref}} + \frac{c_t}{m} k_{z_0} \tilde{p}_t + \frac{2c_p}{Nm} v_n \bar{e} \Phi_{\text{ref}} + \eta C \dot{\lambda} + \eta \bar{e} \dot{\phi}_o \\ &\quad - \dot{v}_{t,\text{ref}} + k_{z_0} \tilde{v}_t) \end{aligned} \quad (4.156)$$

We denote

$$\delta_1(\phi_o, \lambda) = \eta C \quad (4.157)$$

It can be numerically verified that $\delta_1(\cdot)$ is uniformly bounded away from zero, and this is because of the phase shift between the link references in (4.15). We take $\dot{\lambda}$ as a virtual control input that we use to make (4.156) negative:

$$\dot{\lambda} = \frac{1}{\delta_1} \left(-\tilde{p}_t + \frac{c_t}{m} v_{t,\text{ref}} - \frac{c_t}{m} k_{z_0} \tilde{p}_t - \frac{2c_p}{Nm} v_n \bar{e} \Phi_{\text{ref}} - \eta \bar{e} \dot{\phi}_o + \dot{v}_{t,\text{ref}} - k_{z_0} \tilde{v}_t - k_{z_1} z_1 \right) \quad (4.158)$$

where $k_{z_1} > 0$ is a constant gain. For simplicity, we denote

$$\begin{aligned} \delta_2(\phi_o, \dot{\phi}_o, \lambda, p_t, v_t) &= \\ \frac{1}{\delta_1} \left(-\tilde{p}_t + \frac{c_t}{m} v_{t,\text{ref}} - \frac{c_t}{m} k_{z_0} \tilde{p}_t - \frac{2c_p}{Nm} v_n \bar{e} \Phi_{\text{ref}} - \eta \bar{e} \dot{\phi}_o + \dot{v}_{t,\text{ref}} - k_{z_0} \tilde{v}_t - k_{z_1} z_1 \right) \end{aligned} \quad (4.159)$$

We could have chosen $\dot{\lambda}$ to be given by the compensator in (4.158). However, since (4.158) represents the frequency of the lateral undulation motion given in (4.15) it is desirable from a practical implementation point of view to smooth the frequency function. We thus introduce an extra step of backstepping, and define the second error variable as

$$z_2 = \dot{\lambda} - \delta_2 \quad (4.160)$$

which we aim to drive to zero. Inserting $\dot{\lambda} = z_2 + \delta_2(\cdot)$ into (4.156) yields

$$\dot{V}_2 = -k_{z_0}\tilde{p}_t^2 - \left(\frac{c_t}{m} + k_{z_1}\right)z_1^2 + z_1z_2\delta_1 \quad (4.161)$$

To perform backstepping for z_2 , we write the error dynamics for the error variable z_2 which has the form

$$\dot{z}_2 = u_\lambda - \dot{\delta}_2 \quad (4.162)$$

We choose an augmented CLF in the form

$$V_3 = V_2 + \frac{1}{2}z_2^2 \quad (4.163)$$

The time-derivative of V_3 along the solutions of (4.147) is

$$\dot{V}_3 = -k_{z_0}\tilde{p}_t^2 - \left(\frac{c_t}{m} + k_{z_1}\right)z_1^2 + z_2(z_1\delta_1 + u_\lambda - \dot{\delta}_2) \quad (4.164)$$

We define the control input u_λ as

$$u_\lambda = -z_1\delta_1 + \dot{\delta}_2 - k_{z_2}z_2 \quad (4.165)$$

where $k_{z_2} > 0$ is a constant gain. Inserting (4.165) into (4.164) yields

$$\dot{V}_3 = -k_{z_0}\tilde{p}_t^2 - \left(\frac{c_t}{m} + k_{z_1}\right)z_1^2 - k_{z_2}z_2^2 \quad (4.166)$$

From (4.166) it can be shown that

$$\dot{V}_3 \leq -\beta_3 V_3 \quad (4.167)$$

where $\beta_3 \in \mathbb{R}_{>0}$ is a small enough positive constant. This implies that the origin $(\tilde{p}_t, \tilde{v}_t) = (0, 0)$ of (4.147) is exponentially stable, i.e. since we have a guaranteed rate of convergence to zero, and the control objective (4.131) will be achieved. Furthermore, since in (4.160) z_2 converges to zero and $\delta_2(\cdot)$ is uniformly bounded, then $\dot{\lambda}$ remains uniformly bounded. We denote the bound on the solutions of the dynamic compensator (4.165) by $\|[\lambda, \dot{\lambda}]\| \leq \epsilon_\lambda$.

Remark 4.9 Under the the controllers (4.18), (4.139), and (4.165), with the augmented state vector

$$x = [p_t, v_t, p_n, v_n, \phi_o, \dot{\phi}_o, \lambda, \dot{\lambda}] \in \mathbb{R}^2 \quad (4.168)$$

the closed-loop tangential and normal position dynamics along with the dynamics of the compensators take the form $\dot{x} = f(x)$. Since throughout our stability proof we showed that all the functions in the closed-loop remain bounded, then it can be shown that

$$\|f(x)\| \leq K(1 + \|x\|) \quad (4.169)$$

where $K \in \mathbb{R}_{>0}$ is a constant. This linear growth condition implies that any of the components of the state vector (4.165) have no finite escape time.

We collect the results of the arguments presented in this subsection in the following theorem.

Theorem 4.5 Under the controllers (4.18), (4.139), and (4.165), the velocity control manifold Γ_2 is asymptotically stable relative to the constraint manifold Γ_4 .

4.3.4 Maneuvering Control along Curved Paths: The Geometric Task

So far we have controlled the body shape, orientation, and the position of the robot along the tangential axis of the $t - n$ frame. The last step of our maneuvering control design is to stabilize the normal position \bar{p}_y of the robot to the desired path. Note that this will imply the convergence of the cross-track error to zero. Also note that we have already proved the boundedness of the normal velocity \bar{v}_n of the robot in the previous section. For clarity of presentation, we summarize the stability results that we have established so far in the following:

- Under the joint control law (4.18), the constraint manifold Γ_4 is globally exponentially stable for the solutions of (4.6), i.e. which implies that $\phi \rightarrow \phi_{\text{ref}}$.
- Under the joint control law (4.18) and the static compensator (4.139), the orientation control manifold Γ_3 is asymptotically stable relative to the constraint manifold Γ_4 , which implies that $\theta \rightarrow \theta_{\text{ref}}$.
- The solution of the static compensator (4.139) remains uniformly ultimately bounded. The ultimate bound on the solutions is given by $\|\phi_o\| \leq \alpha + \epsilon_r$ where α denotes the amplitude of the joint angle oscillations, and $\epsilon_r > 0$ is a constant.

- Under the controllers (4.18) and (4.139), the normal velocity \bar{v}_n of the robot remains uniformly bounded. We denoted this bound by $\|v_n\| \leq \epsilon_n$ where $\epsilon_n \in \mathbb{R}_{>0}$ is a constant.
- Using the dynamic compensator (4.165), the velocity control manifold Γ_2 is asymptotically stable relative to the constraint manifold Γ_4 , which implies that the tangential position and tangential velocity of the robot converge asymptotically to the reference tangential position and velocity, respectively, i.e. which implies that $(p_t, v_t) \rightarrow (p_{t,\text{ref}}, v_{t,\text{ref}})$.
- The solutions $(\lambda, \dot{\lambda})$ of the dynamic compensator (4.165) remain uniformly ultimately bounded. We denoted this ultimate bound by $\|[\lambda, \dot{\lambda}]\| \leq \epsilon_\lambda$ where $\epsilon_\lambda > 0$ is a constant.

Using the stability results above, we solve the geometric task and we ensure that the normal position of the robot \bar{p}_y converges to the desired path. Also we define the path following manifold, i.e. the manifold on which the geometric task is achieved, as

$$\Gamma_1 = \left\{ (\theta, \dot{\theta}, p_t, v_t, \bar{p}_y, v_n, \phi_o, \dot{\phi}_o, \lambda, \dot{\lambda}) \in \Gamma_2 : \tilde{p}_y \leq \epsilon_p \right\} \quad (4.170)$$

where $\epsilon_p \in \mathbb{R}_{>0}$ is any positive constant.

In order to stabilize $\bar{p}_y \rightarrow p_{yd}$, we define the position and velocity cross track errors as

$$\tilde{p}_y = \bar{p}_y - p_{yd} \quad (4.171)$$

$$\tilde{v}_n = \bar{v}_n \quad (4.172)$$

which we would like to drive to zero and thereby achieve the control objective (4.130). From (4.69)–(4.70), the dynamics of the position of the robot is given by

$$\dot{\bar{p}}_y = v_t \sin(\theta) + v_n \cos(\theta) \quad (4.173)$$

which in the error coordinates can be written as

$$\dot{\tilde{p}}_y = (\tilde{v}_t + v_{t,\text{ref}}) \sin(\tilde{\theta} + \theta_{\text{ref}}) + \tilde{v}_n \cos(\tilde{\theta} + \theta_{\text{ref}}) - \dot{\Theta} \frac{\partial p_{yd}}{\partial \Theta} \quad (4.174)$$

The reduced dynamics of the position of the CM evaluated on the exponentially stable manifold Γ_2 , is of the form

$$\dot{\tilde{p}}_y = v_{t,\text{ref}} \sin(\theta_{\text{ref}}) + \tilde{v}_n \cos(\theta_{\text{ref}}) - \dot{\Theta} \frac{\partial p_{yd}}{\partial \Theta} \quad (4.175)$$

where $\frac{\partial p_{yd}}{\partial \Theta}$ is associated with the curvature of the desired path. By using the relations

$$\sin\left(-\operatorname{atan2}\left(\frac{\tilde{p}_y}{\Delta}\right)\right) = -\frac{\tilde{p}_y}{\sqrt{\tilde{p}_y^2 + \Delta^2}} \quad (4.176)$$

$$\cos\left(-\operatorname{atan2}\left(\frac{\tilde{p}_y}{\Delta}\right)\right) = \frac{\Delta}{\sqrt{\tilde{p}_y^2 + \Delta^2}} \quad (4.177)$$

we can rewrite (4.175) as

$$\dot{\tilde{p}}_y = -\frac{v_{t,\text{ref}}\tilde{p}_y}{\sqrt{\tilde{p}_y^2 + \Delta^2}} + \frac{\tilde{v}_n\Delta}{\sqrt{\tilde{p}_y^2 + \Delta^2}} - \dot{\Theta}\frac{\partial p_{yd}}{\partial \Theta} \quad (4.178)$$

We select a Lyapunov function candidate of the form

$$V = \frac{1}{2}\tilde{p}_y^2 \quad (4.179)$$

Taking the time-derivative of (4.179) along the solutions of (4.178), and utilizing the stability results above, yields

$$\begin{aligned} \dot{V} &= \tilde{p}_y \dot{\tilde{p}}_y \\ &= \tilde{p}_y \left(-\frac{v_{t,\text{ref}}\tilde{p}_y}{\sqrt{\tilde{p}_y^2 + \Delta^2}} + \frac{\tilde{v}_n\Delta}{\sqrt{\tilde{p}_y^2 + \Delta^2}} - \dot{\Theta}\frac{\partial p_{yd}}{\partial \Theta} \right) \\ &\leq -\left(\frac{v_{t,\text{ref}}}{\sqrt{\tilde{p}_y^2 + \Delta^2}}\right)\tilde{p}_y^2 + \epsilon_n\|\tilde{p}_y\| + \|\dot{\Theta}\|\left\|\frac{\partial p_{yd}}{\partial \Theta}\right\|\|\tilde{p}_y\| \\ &\leq -\left(\frac{v_{\min}}{\sqrt{\tilde{p}_y^2 + \Delta^2}}\right)\tilde{p}_y^2 + \epsilon_n\left(\frac{\gamma\tilde{p}_y^2}{2} + \frac{1}{2\gamma}\right) + \epsilon_\Theta\left(\frac{\gamma\tilde{p}_y^2}{2} + \frac{1}{2\gamma}\left\|\frac{\partial p_{yd}}{\partial \Theta}\right\|^2\right) \end{aligned} \quad (4.180)$$

where we used Young's inequality, and where v_{\min} denotes the minimum desired forward velocity of the robot. Note that according to the results of [45], a snake robot with zero forward velocity is not controllable, and (4.180) concurs with this result. Finally, we have that

$$\dot{V} \leq \left(-\frac{v_{\min}}{\sqrt{\tilde{p}_y^2 + \Delta^2}} + \frac{\epsilon_n\gamma}{2} + \frac{\epsilon_\Theta\gamma}{2}\right)\tilde{p}_y^2 + \eta \quad (4.181)$$

where

$$\eta = \frac{\epsilon_n}{2\gamma} + \frac{\epsilon_\Theta}{2\gamma}\left\|\frac{\partial p_{yd}}{\partial \Theta}\right\|^2 \quad (4.182)$$

We investigate two possible scenarios for the time-derivative of the Lyapunov function candidate given in (4.181).

1. In the first scenario, it can be seen that for given parameters $(v_{\min}, \Delta, \epsilon_n, \epsilon_\theta)$, we can always choose a sufficiently small γ such that the coefficient of \tilde{p}_y^2 in (4.181) is negative. In this case, we conclude that there exist a sufficiently small positive constant $\beta \in \mathbb{R}_{>0}$, such that

$$\dot{V} \leq -\beta V + \eta \quad (4.183)$$

Using the Comparison Lemma, we have that

$$V(t) \leq V(0)e^{-\beta t} + \frac{\eta}{\beta} \quad (4.184)$$

This implies that V converges to a ball of radius $\frac{\eta}{\beta}$. Furthermore, because of (4.179), we can conclude that \tilde{p}_y converges to a ball of the radius $\sqrt{\frac{2\eta}{\beta}}$.

2. In the second scenario, we assume that we would like to drive the cross-track error \tilde{p}_y , to an arbitrary small neighbourhood of zero which we denote by $\epsilon_p \in \mathbb{R}_{>0}$ for any positive constant ϵ_p , i.e. we seek practical stability for the origin of (4.178). In this case, we choose

$$\gamma \geq \frac{\epsilon_n + \epsilon_\theta}{2} \left\| \frac{\partial p_{yd}}{\partial \Theta} \right\|^2 \frac{\beta}{\epsilon_p^2} \quad (4.185)$$

Substituting (4.185) into (4.182) and then (4.184) yields

$$V(t) \leq V(0)e^{-\beta t} + \epsilon_p^2 \quad (4.186)$$

which implies that \tilde{p}_y converges to a ball of the radius ϵ_p . In this case we must have the following conditions on the parameters $(v_{\min}, \Delta, \epsilon_n, \epsilon_\theta)$ such that the coefficient of \tilde{p}_y^2 is negative.

- (a) The minimum tangential velocity v_{\min} should be sufficiently large.
- (b) The look-ahead distance Δ should be sufficiently small.
- (c) The upper-bound on the normal velocity v_n should be sufficiently small. From (4.89) and (4.72), this implies that the friction coefficient in the normal direction of motion c_n must be sufficiently large, i.e. in order to damp the sideways velocity.
- (d) The upper-bound on the curvature of the path should be sufficiently small. This implies that the desired path is sufficiently smooth.

The above conditions guarantee that the path following error \tilde{p}_y converges to an arbitrarily small neighbourhood of the origin, which readily implies that

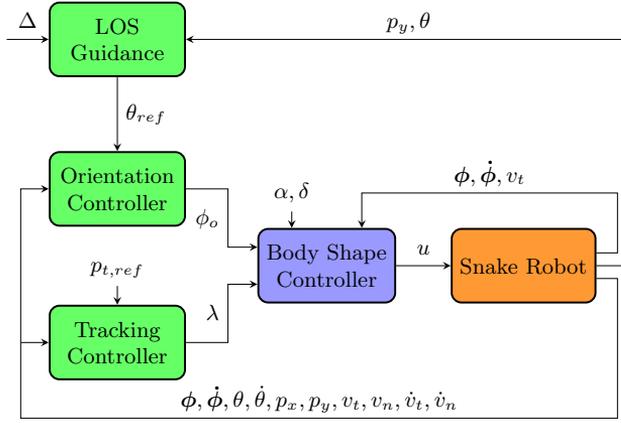


Figure 4.3: The structure of the maneuvering controller

we will solve the geometric task. Note that due to the oscillatory motion of the snake robot, that is necessary in order to achieve forward motion, we cannot expect the path following error to converge to zero, but rather to a neighbourhood of the origin. Figure 4.3 shows the structure of the proposed maneuvering controller.

Remark 4.10 *The path following set Γ_1 is a compact set. This is due to the fact that all the variables $(\theta, \dot{\theta}, p_t, v_t, \bar{p}_y, v_n, \phi_o, \dot{\phi}_o, \lambda, \dot{\lambda})$ used to define this set were proved to be bounded, which implies the compactness of the set.*

Finally, we collect all the established stability results in the following theorem, which states that the proposed maneuvering controller solves the dynamic and geometric tasks.

Theorem 4.6 (Main Result) *Consider the positively invariant sets Γ_4, Γ_3 , and Γ_2 and Γ_1 in (4.132), (4.138), (4.144), and (4.170), respectively. Note that $\Gamma_1 \subset \Gamma_2 \subset \Gamma_3 \subset \Gamma_4 \subset \mathcal{Q}$. The set Γ_1 is a compact set. For $i = 1 \dots 3$, the set Γ_i was asymptotically stable relative to Γ_{i+1} . Consequently, according to Proposition 4.1, the set Γ_1 is asymptotically stable for the controlled system. This implies that all the control objectives (4.127)–(4.131) will be achieved, and all the solutions of the controlled system remain uniformly bounded.*

4.4 Simulation Results

Simulation Results for the Direction Following Controller

In this section, we present simulation results for the proposed direction following control approach. We considered a snake robot with $N = 10$ links of length $l = 0.14$ m, and mass $m = 1$ kg. The ground friction coefficients were $c_t = 1$ and $c_n = 3$, and the rotation parameters were $\lambda_1 = 0.5$ and $\lambda_2 = 20$. The rotation parameters are chosen such that the simplified model quantitatively behaves similar to the complex model derived in many previous works such as [1] and [11]. We chose $\alpha = 4.5$ cm, and $\delta = 40\pi/180$. Employing the Routh-Hurwitz stability criterion, the gains of the exponentially stabilizing joint controller in (4.18) were set to $k_p = 20$, and $k_d = 5$. The orientation controller gain in (4.24) were tuned as $k_0 = 5$, $k_1 = 26$, $k_2 = 39$, and $k_3 = 20$. The gains of the dynamic compensator which was used to control the velocity (4.46) were tuned as $k_\lambda = 20$ and $k_z = 0.5$. The tangential reference velocity was $v_{t,\text{ref}} = 0.2$ m/s, and the orientation reference angle was $\theta_{\text{ref}} = \pi/4$. Since on the constraint manifold, where a lateral undulatory gait is stabilized, v_t is positive [11], the initial tangential velocity was chosen as $v_t(0) = 0.1$ m/s, see the arguments after (4.25), yet all other states are set initially to zero.

The simulation results are shown in Figures 4.4–4.10. In Figure 4.4 the solutions $\dot{\lambda}$ and ϕ_o of the dynamic compensators are shown. In particular, the frequency of the joint oscillations converges to a positive constant, which according to the work of [1] implies a forward motion for the robot, and ϕ_o which is the state of the dynamic compensator (4.24) which controls the orientation of the robot, remain uniformly bounded. Figure 4.5 illustrates the motion of the snake robot in the $x - y$ plane, and how the snake robot follows a path heading in the direction given by the reference. Figure 4.6 shows the forward velocity v_t converges to the constant reference velocity $v_{t,\text{ref}}$. Figure 4.7 shows that the normal velocity converges to a small neighbourhood of the origin. In Figure 4.8 it can be seen that the body shape variables follow the reference joint angles provided by (4.15), while the norm of the error converges exponentially to zero. Figure 4.9 shows that the proposed orientation controller (4.24) successfully reorients the robot in accordance with θ_{ref} . Finally, Figure 4.10 shows that the coefficients of the virtual control inputs in (4.24) and (4.36) are bounded away from zero and consequently the control inputs are globally well-defined.

Simulation Results for the Maneuvering Controllers

To illustrate the performance of the proposed maneuvering controllers, in this section we present simulation results for a snake robot which moves along a straight line path. Note that any shape of path which fulfils the condition (d) on curvature of the path which was presented in Subsection 4.3.4, can be used for the proposed maneuvering controller. We considered a snake robot with $N = 10$ links of length $l = 0.14$ m, mass $m = 1$ kg, and anisotropic ground friction coefficients $c_t = 1$ and $c_n = 3$. We chose the rotation parameters such that the simplified model qualitatively and quantitatively behaves similar to the complex model. In particular, we defined $\lambda_1 = 0.5$ and $\lambda_2 = 20$. The gait parameters were $\alpha = 4.5$ cm and $\delta = 40\pi/180$, and the joint controller gains were $k_p = 20$ and $k_d = 5$. To stabilize the orientation error dynamics we chose the orientation controller gain in (4.83) as $k_0 = 5$, $k_1 = 26$, $k_2 = 39$, and $k_3 = 20$. The gains of the position tracking controller were tuned to $k_{z_1} = 0.5$, $k_{z_2} = 0.5$ and $k_{z_3} = 0.1$. The reference forward velocity was $v_{t,\text{ref}} = 0.2$ m/s, and the position reference was $p_{t,\text{ref}} = \int v_{t,\text{ref}} dt$. The look ahead distance was chosen as twice the length of the robot $\Delta = 2.8$ m. To avoid singularities the the initial tangential velocity was set to $v_t(0) = 0.1$ m/s, see the arguments after (4.83). All the other states were set initially to zero.

The results of the simulation are shown in Figures 4.11-4.15. In particular, Figure 4.11 illustrates the exponential stability of the joint error, hence the joints move in accordance with the lateral undulatory gait provided by (4.15). The convergence of the orientation to the reference orientation and the corresponding error can be seen in Figure 4.12. Figure 4.13 shows that the controller regulates the tangential velocity to the desired constant forward velocity, and the normal velocity to zero. Figure 4.14 shows that the joint oscillation frequency $\dot{\lambda}$ converges to a positive constant and the solution of the orientation controller ϕ_o is bounded and becomes zero when the robot is on the path. Finally, Figure 4.15 illustrates that the snake robot converges to and follows the desired path based on the desired velocity profile.

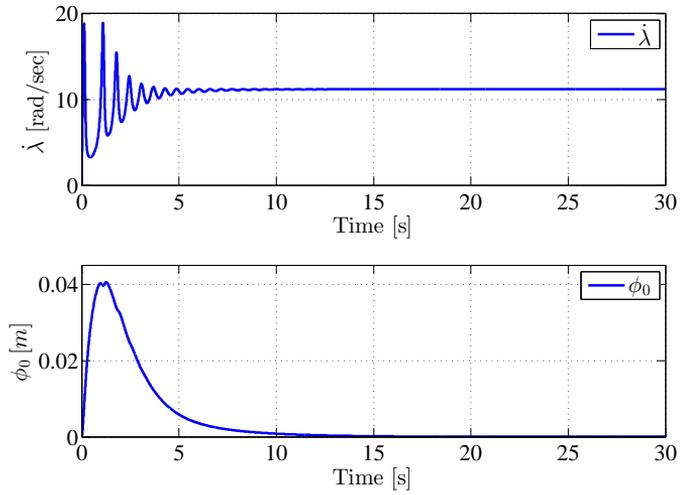


Figure 4.4: The solutions of the dynamic compensators remain bounded.

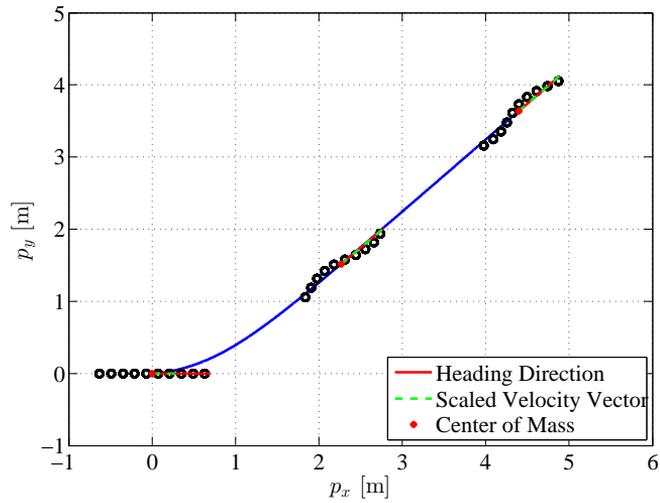


Figure 4.5: A 10-links snake robot follows the desired orientation.

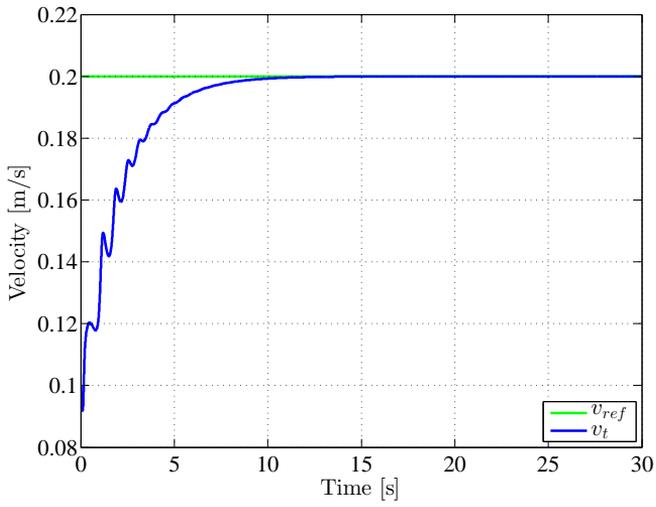


Figure 4.6: The forward velocity of the robot is regulated to the constant reference velocity.

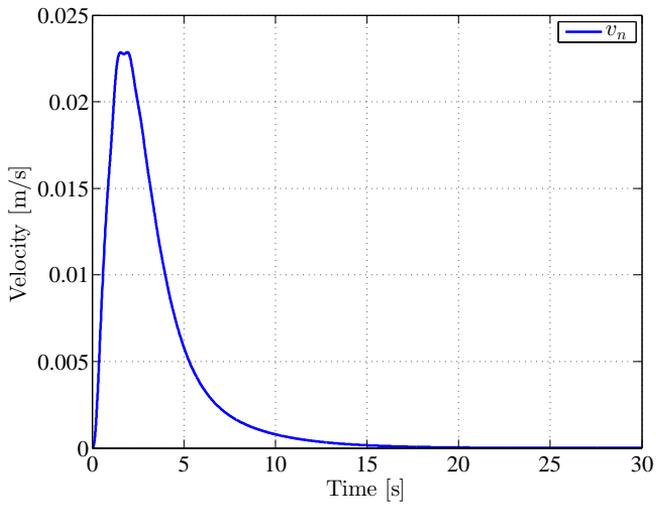


Figure 4.7: The normal velocity of the robot converges to a small neighbourhood of the origin.

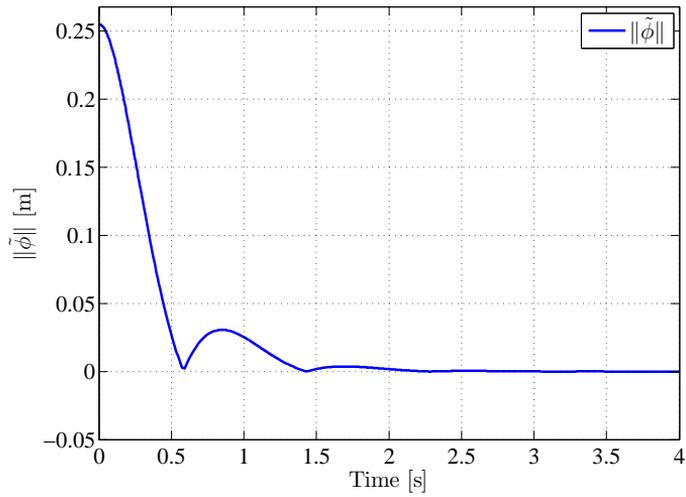


Figure 4.8: The joints angles converge to the reference joint angles.

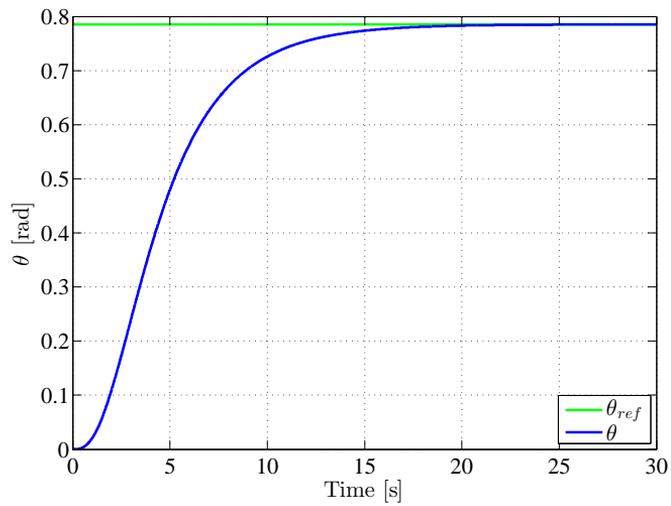


Figure 4.9: The orientation of the robot converges to the reference orientation.

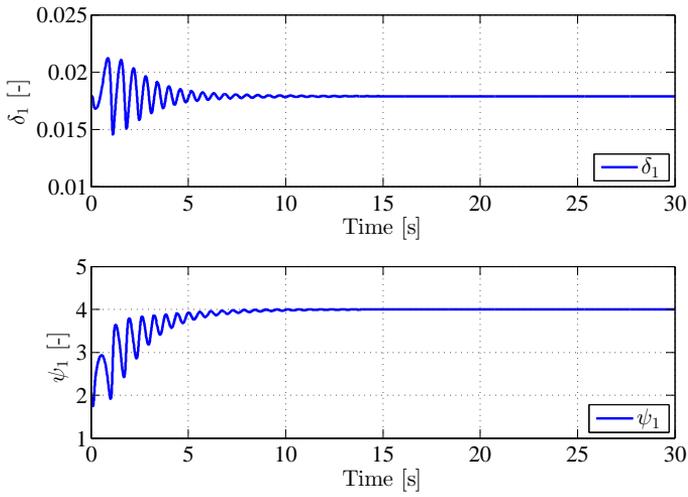


Figure 4.10: The coefficients of the control inputs in (34,43) are globally well-defined.

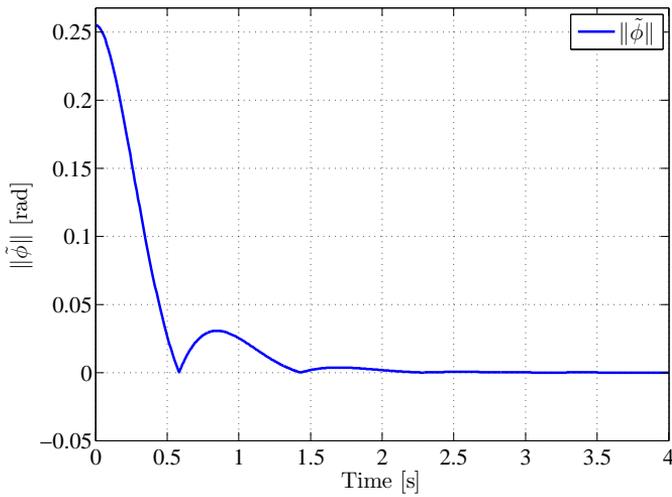


Figure 4.11: Exponential stability of the joints tracking errors.

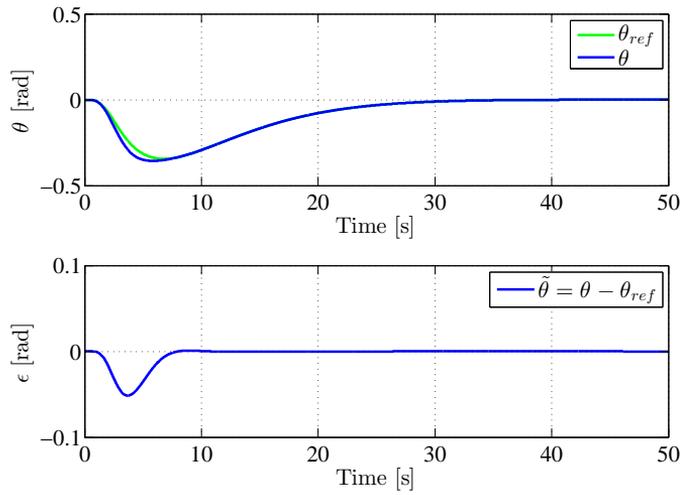


Figure 4.12: Orientation reference tracking and orientation error.

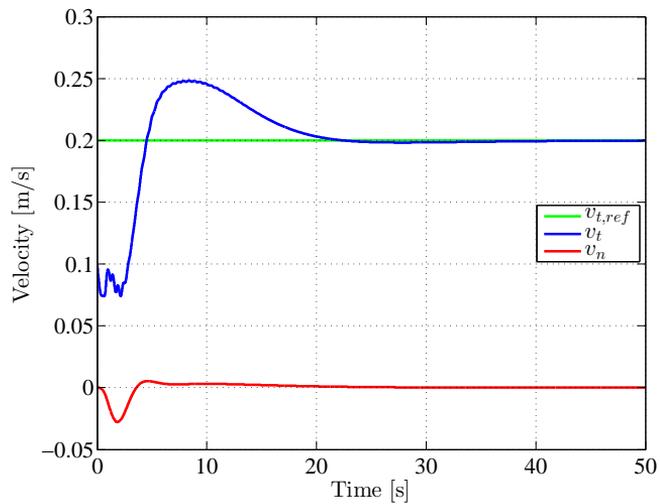


Figure 4.13: Forward and sideways velocities with references.

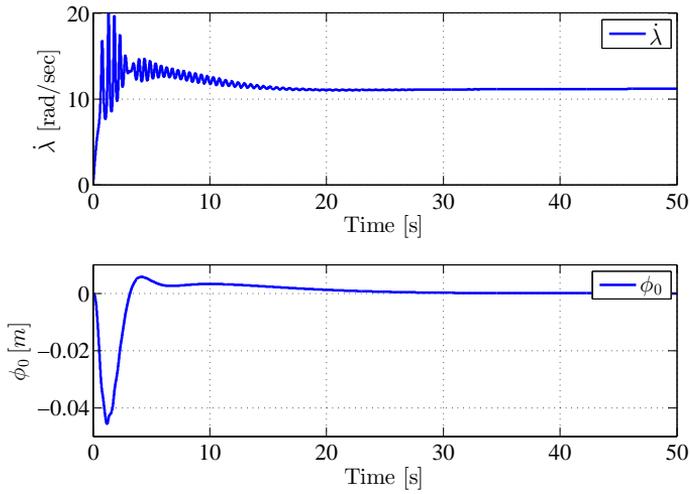


Figure 4.14: The joint oscillation frequency $\dot{\lambda}$ converges to a positive constant and the joint offset ϕ_0 is bounded and becomes zero.

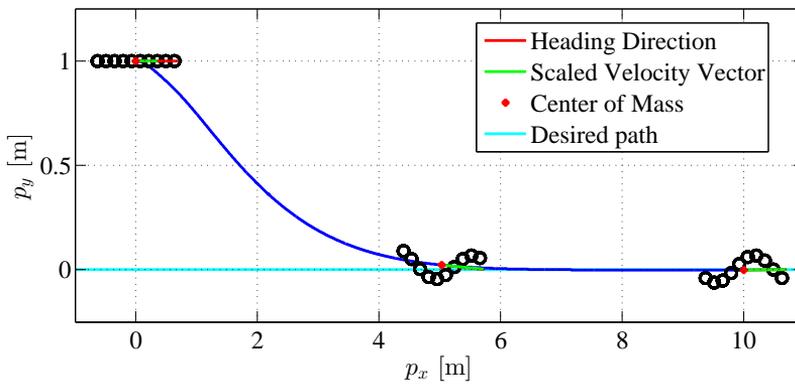


Figure 4.15: A snake robot following the desired path.

Chapter Summary

- We considered direction following control of planar snake robots using the method of virtual holonomic constraints, based on a simplified dynamic model developed for control design and analysis purposes in [11]. We enforced virtual holonomic constraints for the body shape variables of the robot. These constraints were inspired by the well-known reference joint angle trajectories which induce a lateral undulatory gait pattern on snake robots. Furthermore, we removed the explicit time-dependence of the reference joint angles, and rather made them a function of the solutions of two dynamic compensators. Subsequently, we reduced the dynamics of the system to the invariant constraint manifold, where we used the dynamic compensators to control the velocity and orientation of the robot.
- We considered maneuvering control of planar snake robots along straight and curved paths. We started the hierarchical control design by enforcing virtual holonomic constraints for the body shape variables of the robot which induced lateral undulatory gait pattern on snake robot. Moreover, we derived the reduced dynamics of the system evaluated on the constraint manifold associated with virtual holonomic constraints. Furthermore, we used choose the frequency of the desired gait pattern through a dynamic compensator which controlled the velocity of the robot, and we introduced an offset angle through a static compensator to control the orientation of the robot on this manifold.
- We presented extensive simulation results which showed the performance of the proposed controllers.

Model-based Locomotion Control Approaches for Snake Robots Part II: The Complex Model

At the beginning of this thesis, we mentioned the main objective of this work as bridging the gap between the theory and practice for snake robots by proposing model-based control approaches which are derived based on formal stability proofs. These analytical control approaches can rarely be found in the previous snake robot literature. Actually, underactuation, complex dynamical behaviour, and strong nonlinearities present in the dynamic model are among the main reasons of the lack of analytical control approaches for snake robots.

In this chapter, we address locomotion control problems for the snake robot based on the complex model of the robot which we derived in an Euler-Lagrange framework in Chapter 2. In particular, we will solve the direction following and maneuvering control problems for the snake robot. It is noteworthy to mention that similar problems have never been addressed for the snake robot in the previous literature. This is due to the fact that many challenges arise for the model-based control design based on the complex model, that make the application of analytical control approaches and formal stability proofs for these robots challenging. However, in this chapter we will provide formal stability proofs throughout the derivations of control laws based on the complex model, and we solve the direction following and maneuvering control problems.

In particular, in this chapter we will propose hierarchical control design approaches, which solve these control problems using the following steps.

1. In the first step, we will stabilize a desired gait pattern for the robot. Consequently, the robot starts locomotion based on the desired gait pattern.
2. In the second step, we use the gait parameters in order to control the orientation and planar position of the robot which leads to solving the direction following and maneuvering control problems.

We will provide formal stability proofs for all the arguments given in the foregoing two steps, and will show the performance of the proposed control strategies by using extensive numerical simulations.

We believe that this is an important step forward in locomotion control of snake robots, which can help to realize the ultimate objectives regarding the extensive application of these robots in real-time operations. Furthermore, here we will present the core of the control design approach, and since the arguments here are derived using mathematical analysis based on the complex model, then it is straightforward to extend these results to various other control frameworks, such as robust and adaptive control design approaches.

Contributions of this chapter: The first contribution of this chapter is to solve the direction following control problem for the snake robot based on the complex model. In particular, we regulate the orientation and forward velocity of the robot to constant references. To this end, we stabilize a desired lateral undulatory gait pattern among the body shape variables of the robot. Moreover, we choose the gait parameters using two dynamic compensators which will be employed to control the orientation and forward velocity of the robot. Furthermore, we show that the solutions of the controlled system remain uniformly ultimately bounded.

The second contribution of this chapter is to solve the path following control problem for the snake robot based on the complex model. To this end, after controlling the body shape, orientation, and the forward velocity of the robot according to the design steps given above, we will define the reference velocity for the robot in a way that enables us to practically stabilize general curved line paths for the position of the CM of the robot. To our best knowledge, path following control along general curved paths has never been considered for the snake robots before based on the complex model of the snake robot locomotion.

Organization of this chapter: This chapter is organized as follows. In Section 5.1, we propose a direction following controller for the robot. In Section 5.2, we solve the path following problem for the snake robot. In Section 5.3 we present the results of simulations for the controllers proposed in this chapter.

Publications: The results of this chapter are based on the journal paper [85], and the conference paper [83].

Review of the Complex Model

In this section, we briefly review the complex model of the snake robot which we derived in Chapter 2. We use this model for the model-based control design in the subsequent sections. Among the different (and equivalent) representations of the complex dynamic model which we presented in Chapter 2, we choose the dynamic model (2.45a)–(2.45b) for the model-based locomotion control design in this chapter. This choice is purely due to the notation convenience in the derivation of the control laws. Consequently, the analytical derivations and the stability proofs which are derived based on this model can easily be used for the other models as well. This dynamic model of the N -link robot is represented as

$$M_\theta \ddot{\theta} + W_\theta \dot{\theta}^2 - lSC_\theta^T f_R(\theta, \dot{\theta}, \dot{p}) = D^T u \quad (5.1a)$$

$$Nm\ddot{p} = E^T f_R(\theta, \dot{\theta}, \dot{p}) \quad (5.1b)$$

where $\theta = [\theta_1, \dots, \theta_N]^T \in \mathbb{R}^N$ denotes the vector of the absolute link angles, $\dot{p} = [\dot{p}_x, \dot{p}_y]^T \in \mathbb{R}^2$ denotes the vector of the linear velocities of the CM of the robot, $u \in \mathbb{R}^{N-1}$ is the vector of the actuator torques, and f_R is the vector of ground friction forces. The remaining quantities of the dynamic model (5.1a)–(5.1b) were defined in Chapter 2, through (2.46)–(2.48). Finally, letting

$$u_{\theta_N} = [\cos \theta_N, \sin \theta_N]^T \quad (5.2a)$$

$$v_{\theta_N} = [-\sin \theta_N, \cos \theta_N]^T \quad (5.2b)$$

we define

$$v_t = u_{\theta_N}^T \dot{p} \quad (5.3a)$$

$$v_n = v_{\theta_N}^T \dot{p} \quad (5.3b)$$

The scalars v_t and v_n defined above are the components of the inertial velocity of the center of mass parallel and perpendicular to the angle of the head, respectively.

Lateral Undulation

One of the basic gait patterns through which biological snakes achieve forward motion is called lateral undulation [11]. During lateral undulation, the snake undergoes periodic shape changes that resemble a wave traveling backward along its body, from head to tail. As a result of this motion, the snake body

traces out a periodic curve on the plane, which Hirose [1] mathematically represented as a *serpenoid*. Thinking of a snake robot as a discrete approximation of a biological snake, researchers (see e.g. [1] and [105]) have observed that the serpenoid curve can be well-approximated by imposing the sinusoidal reference signal for the i -th joint angle

$$\phi_{\text{ref},i}(t) = \alpha \sin(\omega t + (i - 1)\delta) + \phi_o, \quad (5.4)$$

where α denotes the amplitude of the sinusoid, ω denotes the frequency of the joint oscillations, δ denotes the phase shift between two consecutive joints, and ϕ_o is a joint offset used to control the direction of locomotion.

5.1 Direction Following Control of Snake Robots

This section investigates the problem of direction following for planar snake robots. The control objective is to regulate the linear velocity vector of the snake robot to a constant reference while guaranteeing boundedness of the system states. The proposed feedback control strategy enforces virtual constraints encoding a lateral undulatory gait pattern. This gait is parametrized by states of dynamic compensators which are used to regulate the orientation and forward speed of the snake robot.

5.1.1 Control Design Objectives

In this section, we present the blueprint of our control design. We begin by stating the control specifications.

Direction Following Problem (DFP): Given a desired constant velocity vector \dot{p}_{ref} with polar representation $(r, \theta) = (v_{\text{ref}}, \theta_{\text{ref}})$, design a smooth feedback controller achieving the following specifications:

- (i) Practical stabilization¹ of the head angle θ_N to θ_{ref} .
- (ii) Practical stabilization of the tangential velocity $v_t = u_{\theta_N}^T \dot{p}$ to v_{ref} .
- (iii) Uniform ultimate boundedness of the normal velocity $v_n = v_{\theta_N}^T \dot{p}$ with a small ultimate bound, and ultimate boundedness of the joint dynamics and all controller states of the controlled system.

¹Practical stabilization of a variable means that by a suitable choice of controller parameters the variable is made to converge to an arbitrarily small neighborhood of its desired value.

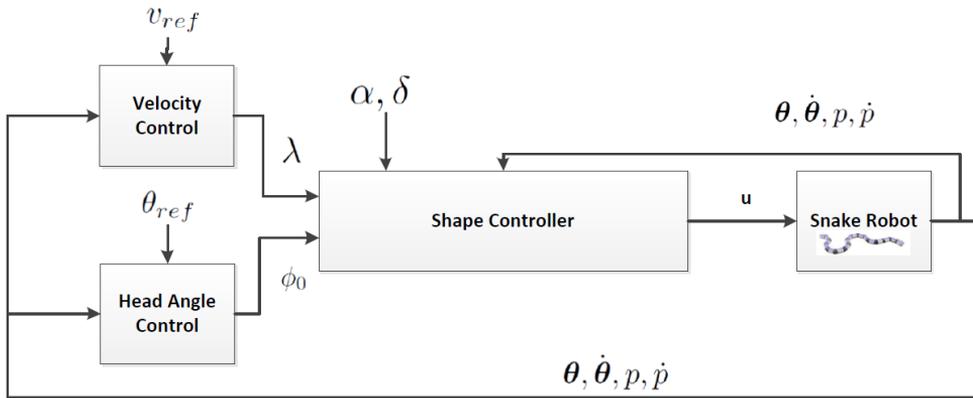


Figure 5.1: The structure of the direction following controller.

The above problem formulation relies on the observation that if $\theta_N = \theta_{ref}$, then making $\dot{p} \rightarrow \dot{p}_{ref}$ is equivalent to making $(v_t, v_n) \rightarrow (v_{ref}, 0)$.

Solution Methodology:

In order to solve the DFP, we stabilize a lateral undulatory gait for the shape variables of the robot. Our control design approach is given in three stages.

Stage 1: Body Shape Control. We use the controls u in (5.1a) to stabilize a virtual constraint encoding a lateral undulatory gait similar to (5.4), in which ωt is replaced by a state λ , and ϕ_o affects only the head angle θ_N . The evolution of λ, ϕ_o is governed by two compensators, $\ddot{\phi}_o = u_{\phi_o}$ and $\ddot{\lambda} = u_\lambda$.

Stage 2: Head Angle Control. Inspired by the biological observation that snakes keep their head pointed towards a target while their body undulates behind the head, we design u_{ϕ_o} to practically stabilize $\theta_N \rightarrow \theta_{ref}$ while guaranteeing that $(\phi_o, \dot{\phi}_o)$ is uniformly ultimately bounded.

Stage 3: Velocity Regulation. We design u_λ to practically stabilize $v_t \rightarrow v_{ref}$ while guaranteeing that v_n converges to a small neighborhood of the origin and $\dot{\lambda}$ remains uniformly ultimately bounded.

Figure 5.1 depicts the structure of the proposed direction following controller.

5.1.2 Body Shape Control

In this section, we stabilize the lateral undulatory gait for the shape variables of the robot by using the joint control inputs u in (5.1a). In particular, inspired by the lateral undulatory gait, we stabilize the relations:

$$\theta_i - \theta_{i+1} = \alpha \sin(\lambda + (i - 1)\delta), \quad i = 1, \dots, N - 2, \quad (5.5a)$$

$$\theta_{N-1} - \theta_N = \alpha \sin(\lambda + (N - 2)\delta) + \phi_o \quad (5.5b)$$

where (α, δ) are positive constants referred to as the gait parameters and $(\lambda, \phi_o) \in \mathbb{S}^1 \times \mathbb{R}$ are the states of two dynamic compensators

$$\ddot{\lambda} = u_\lambda, \quad \ddot{\phi}_o = u_{\phi_o}, \quad (5.6)$$

which will be designed later in order to control the forward velocity and the head angle of the robot, respectively. The relations (5.5a)–(5.5b) are referred to as virtual holonomic constraints (VHC), see e.g. [92]–[96], and they have the property that they can be made invariant through feedback control. These VHC are dynamic, since they are parametrized by the states of the dynamic compensators (5.6). Note that the difference between the reference joint angles given in (5.5a)–(5.5b) with the reference joint angles given in the previous chapters is that here we add the offset term ϕ_o only to the last joint's reference. This implies that the overall motion of the robot will be smoother, since the transient behaviour of ϕ_o does not affect all the joints.

In continue, we denote

$$\Phi_i(\lambda) = \alpha \sin(\lambda + (i - 1)\delta), \quad i = 1, \dots, N - 1 \quad (5.7)$$

and

$$\Phi(\lambda) = [\Phi_1(\lambda), \dots, \Phi_{N-1}(\lambda)]^T \in \mathbb{R}^{N-1} \quad (5.8)$$

Since $\theta = HD\theta + e\theta_N$, the relations in (5.5a)–(5.5b) can be expressed in the vector form as:

$$\theta = e\theta_N + H\Phi(\lambda) + Hb\phi_o. \quad (5.9)$$

The relations (5.9) can also be written as $h(\lambda, \phi_o, \theta) = 0$, where

$$h(\lambda, \phi_o, \theta) = D\theta - \Phi(\lambda) - b\phi_o. \quad (5.10)$$

If we view $h(\lambda, \phi_o, \theta)$ as an controlled output vector function for system (5.1a)–(5.1b) augmented with compensators (5.5), then this output vector yields a vector relative degree $\{2, \dots, 2\}$ everywhere on the configuration space because we

have that $\text{rank}(DM_\theta^{-1}D^T) = N - 1$. Consequently, the zero dynamics manifold associated with output (5.10) is the set

$$\Gamma = \left\{ \left(\theta, \dot{\theta}, p, \dot{p}, \lambda, \dot{\lambda}, \phi_o, \dot{\phi}_o \right) \in \mathbb{R}^{2N+8} : D\theta = \Phi(\lambda) + b\phi_o, D\dot{\theta} = \Phi'(\lambda)\dot{\lambda} + b\dot{\phi}_o \right\}. \quad (5.11)$$

We refer to Γ as the constraint manifold associated with the VHC (5.5a)–(5.5b). Stabilizing the VHC (5.5a)–(5.5b) corresponds to stabilizing the set Γ . Furthermore, the vector relative degree $\{2, \dots, 2\}$ implies that we can stabilize Γ by using the input-output linearizing control law

$$\begin{aligned} u = & (DM_\theta^{-1}D^T)^{-1} \{ DM_\theta^{-1}W_\theta\dot{\theta}^2 \\ & - lDM_\theta^{-1}SC_\theta^T f_R + \Phi''(\lambda)\dot{\lambda}^2 + \Phi'(\lambda)u_\lambda \\ & + bu_{\phi_o} - K_P[D\theta - \Phi(\lambda) - b\phi_o] \\ & - K_D[D\dot{\theta} - \Phi'(\lambda)\dot{\lambda} - b\dot{\phi}_o] \}, \end{aligned} \quad (5.12)$$

where K_D, K_P are positive definite diagonal matrices containing the joint controller gains.

After asymptotically stabilizing Γ , we are left with two control inputs, (u_λ, u_{ϕ_o}) to solve the direction following problem. In particular, we use the dynamic compensators to regulate the head angle and the velocity of the robot to desired values. To this end, we first derive the reduced dynamics of the robot, i.e., we reduce the system to the invariant manifold Γ . By left multiplying both sides of (5.1a) by e^T , which is a left annihilator of the control input matrix D^T , and evaluating the result on the virtual constraint manifold Γ , the dynamics of the snake robot on the virtual constraint manifold Γ read as

$$\ddot{\theta}_N = \Psi_1(\theta_N, \dot{\theta}_N, \lambda, \dot{\lambda}, \phi_o, \dot{\phi}_o, p, \dot{p}) + \Psi_2(\theta_N, \lambda, \phi_o)u_\lambda + \Psi_3(\theta_N, \lambda, \phi_o)u_{\phi_o}, \quad (5.13a)$$

$$\ddot{p} = \Psi_4(\theta_N, \lambda, \phi_o)\dot{p} + \Psi_5(\theta_N, \lambda, \phi_o)\dot{\theta}_N + \Psi_6(\theta_N, \lambda, \phi_o)\dot{\lambda} + \Psi_7(\theta_N, \lambda, \phi_o)\dot{\phi}_o, \quad (5.13b)$$

$$\ddot{\phi}_o = u_{\phi_o}, \quad (5.13c)$$

$$\ddot{\lambda} = u_\lambda, \quad (5.13d)$$

where

$$\Psi_1(\cdot) = -\frac{e^T M_\theta H \Phi''(\lambda)}{e^T M_\theta e} \dot{\lambda}^2 - \frac{1}{e^T M_\theta e} \{W_\theta \dot{\theta}^2 - l S C_\theta^T f_R\},$$

$$\Psi_2(\cdot) = -\frac{e^T M_\theta H \Phi'(\lambda)}{e^T M_\theta e}, \quad (5.14a)$$

$$\Psi_3(\cdot) = -\frac{e^T M_\theta H b}{e^T M_\theta e}, \quad (5.14b)$$

$$\Psi_4(\cdot) = \frac{1}{Nm} E^T Q_\theta E, \quad (5.14c)$$

$$\Psi_5(\cdot) = \frac{l}{Nm} E^T Q_\theta S C_\theta e, \quad (5.14d)$$

$$\Psi_6(\cdot) = \frac{l}{Nm} E^T Q_\theta S C_\theta H \Phi'(\lambda), \quad (5.14e)$$

$$\Psi_7(\cdot) = \frac{l}{Nm} E^T Q_\theta S C_\theta H b. \quad (5.14f)$$

In the above, each function $\Psi_i(\cdot)$ is evaluated on the constraint manifold Γ . The equations in (5.13a)–(5.13d) describe a control system with two inputs, (u_{ϕ_o}, u_λ) . This system completely describes the motion of the snake once the VHC (5.5) have been enforced. The control objective for system (5.12) is to stabilize θ_N to an arbitrarily small neighborhood of θ_{ref} ; to stabilize $v_t = u_{\theta_N}^T \dot{p}$ to an arbitrarily small neighborhood of v_{ref} ; and finally, to guarantee that $v_n = v_{\theta_N}^T \dot{p}$ converges to a neighborhood of the origin. Meanwhile, we also require that the solutions of the dynamic compensators $(\dot{\lambda}, \phi_o, \dot{\phi}_o)$ remain uniformly bounded.

In the process of developing controllers for the reduced dynamics system, we will require some knowledge of each $\Psi_i(\cdot)$ which is summarized in the following remark.

Remark 5.1 *We make some numerical observations that are important in the subsequent development of our control laws. It can be numerically verified that for all gait parameters (α, δ) :*

- (a) $\Psi_3(\cdot) = -e^T M_\theta H b / e^T M_\theta e$ is bounded away from zero and negative for all $\theta_N, \lambda, \phi_o$;
- (b) $v_{\theta_N}^T \Psi_4(\cdot) v_{\theta_N} \approx -c_n / m$ for all $\theta_N, \lambda, \phi_o$;
- (c) There exists $\gamma_6 > 0$ such that $-u_{\theta_N}^T \Psi_6(\cdot) < -\gamma_6$ for all θ_N, λ and small values of ϕ_o and for $c_n > c_i$;
- (d) There exists $\epsilon_0 > 0$ such that we have $|v_{\theta_N}^T \Psi_6(\cdot)| \leq \alpha \epsilon_0$ for all $\theta_N, \lambda, \phi_o$ where α denotes the amplitude of sinusoidal joint motion in (5.5a)–(5.5b);

(e) $\|\Psi_4(\cdot)\| \leq c_n/m$ for all $\theta_N, \lambda, \phi_o$;

(f) There exists $\gamma_7 > 0$ such that $\|\Psi_7(\cdot)\| \leq \gamma_7$ for all $\theta_N, \lambda, \phi_o$;

(g) $|v_{\theta_N}^T \Psi_4(\cdot) u_{\theta_N}| < c_t/m$ for all $\theta_N, \lambda, \phi_o$.

Note that the above observations are independent of the parameters N, m, l, J .

5.1.3 Head Angle Control

In this section, we consider the head angle control for the snake robot. In particular, using the control input u_{ϕ_o} , we control the head angle of the robot by controlling the states $(\theta_N, \dot{\theta}_N, \phi_o, \dot{\phi}_o)$ of the constrained system (5.12a)-(5.12d). In order to do so, we design a high-gain feedback of the form $u_{\phi_o}(\theta_N, \dot{\theta}_N, \phi_o, \dot{\phi}_o)$ that makes $(\theta_N - \theta_{ref}, \dot{\theta}_N)$ converge to an arbitrarily small neighborhood of the origin and $(\phi_o, \dot{\phi}_o)$ uniformly ultimately bounded. We make this analysis independent of the choice of the input u_λ , by using time scale separation.

Through (5.13a) and (5.13c), the dynamic equations which govern the states $(\theta_N, \dot{\theta}_N, \phi_o, \dot{\phi}_o)$ of the constrained system can be written as

$$\begin{aligned} \ddot{\theta}_N &= f_1 \left(\theta_N, \dot{\theta}_N, \lambda, \dot{\lambda}, \phi_o, \dot{\phi}_o, u_\lambda \right) + \Psi_3(\cdot) u_{\phi_o}, \\ \ddot{\phi}_o &= u_{\phi_o}. \end{aligned} \tag{5.15}$$

Proposition 5.1 Consider the following head angle control law for system (5.15)

$$u_{\phi_o} = \frac{1}{\epsilon} \left[\dot{\theta}_N + k_N (\theta_N - \theta_{ref}) \right] - k_1 \phi_o - k_2 \dot{\phi}_o. \tag{5.16}$$

Also, assume that $u_\lambda(t), \dot{\lambda}(t)$ are defined for all $t \geq 0$. Then for any $k_N, k_1, k_2 > 0$, there exist $\epsilon^*, k > 0$ such that for all $\epsilon \in (0, \epsilon^*)$

$$\limsup_{t \rightarrow +\infty} |\theta_N(t) - \theta_{ref}| = k\epsilon \tag{5.17}$$

$$\limsup_{t \rightarrow +\infty} |\dot{\theta}_N(t)| = k\epsilon. \tag{5.18}$$

Moreover, the states $(\phi_o, \dot{\phi}_o)$ are uniformly ultimately bounded.

Remark 5.2 Under the control law (5.15), the head angle error can be made arbitrarily small provided that ϵ is chosen to be sufficiently small.

Remark 5.3 In the next section we define a feedback controller u_λ guaranteeing that for any initial condition, the closed-loop system has no finite escape time (see Remark 5.4). This will guarantee that the above proposition is applicable.

Proof: Viewing the states $\lambda(t), \dot{\lambda}(t)$, and the input $u_\lambda(t)$ as exogenous signals, the control system (5.15) can be viewed as a time-varying system with states $(\theta_N, \dot{\theta}_N, \phi_o, \dot{\phi}_o)$. Under the control input (5.16), the closed-loop dynamics of system (5.15) in the standard singular perturbation form become

$$\dot{\tilde{\theta}}_N = \tilde{\omega}_N \quad (5.19a)$$

$$\epsilon \dot{\tilde{\omega}}_N = \epsilon \left[g_1 \left(t, \phi_o, \dot{\phi}_o, \theta_N, \dot{\theta}_N \right) - k_1 \phi_o - k_2 \dot{\phi}_o \right] + \Psi_3(\cdot) \left(\tilde{\omega}_N + k_N \tilde{\theta}_N \right) \quad (5.19b)$$

where $\tilde{\theta}_N = \theta_N - \theta_{\text{ref}}$ denotes the head angle error, and where

$$g_1 \left(t, \phi_o, \dot{\phi}_o, \theta_N, \dot{\theta}_N \right) = f_1 \left(\theta_N, \dot{\theta}_N, \lambda(t), \dot{\lambda}(t), \phi_o, \dot{\phi}_o, u_\lambda(t) \right). \quad (5.20)$$

Here we use time-scale separation to make the analysis independent of the choice of u_λ . Note that (5.19a)–(5.19b) is a singularly perturbed system with reduced dynamics

$$\dot{\tilde{\theta}}_N = -k_N \tilde{\theta}_N \quad (5.21)$$

and the boundary-layer dynamics is given by

$$\frac{dy}{d\tau} = \Psi_3(\cdot)y \quad (5.22)$$

where $y = \tilde{\omega}_N + k_N \tilde{\theta}_N$. The origin is an exponentially stable equilibrium point of the reduced system. Also, the origin is an exponentially stable equilibrium point of the boundary-layer system because, by Remark 5.1, for some $\gamma_0 > 0$, $\Psi_3(\cdot) \leq -\gamma_0 < 0$ uniformly in t . According to the singular perturbation theorem on an infinite interval (see Th. 11.2 in [99]), for all $\tilde{\theta}_N(t_0), y(t_0) \in \mathbb{R}$ and $t_0 \geq 0$, the singularly perturbed system (5.19a)–(5.19b) has a unique solution $(\tilde{\theta}_N(t, \epsilon), \tilde{\omega}_N(t, \epsilon))$ such that

$$\begin{aligned} & \tilde{\theta}_N(t, \epsilon) - \exp(-k_N(t - t_0)) \tilde{\theta}_N(t_0) \\ & = O(\epsilon), \end{aligned} \quad (5.23a)$$

$$\begin{aligned} & \tilde{\omega}_N(t, \epsilon) + k_N \exp(-k_N(t - t_0)) \tilde{\theta}_N(t_0) \\ & - \exp\left(\int_{t_0}^{t/\epsilon} \Psi_3(\cdot) d\tau\right) y(t_0) = O(\epsilon), \end{aligned} \quad (5.23b)$$

for all $t \in [t_0, \infty)$. This proves the first part of the proposition. For the second part, note that the closed-loop dynamics governing the states $(\phi_o, \dot{\phi}_o)$ become

$$\ddot{\phi}_o + k_2 \dot{\phi}_o + k_1 \phi_o = \frac{1}{\epsilon} \underbrace{\left(\tilde{\omega}_N(t, \epsilon) + k_N \tilde{\theta}_N(t, \epsilon) \right)}_{f_N(t, \epsilon)}. \quad (5.24)$$

From (5.23a)–(5.23b), it can be seen that $f_N(t, \epsilon)$ is uniformly bounded and of order $O(1)$. Since the unforced system $\ddot{\phi}_o + k_2 \dot{\phi}_o + k_1 \phi_o = 0$ is an LTI system and has a globally exponentially stable equilibrium point at the origin $(\phi_o, \dot{\phi}_o) = (0, 0)$, the system (5.24) is input-to-state stable. This proves the second part of the proposition. ■

5.1.4 Velocity Regulation

Consider the reduced dynamics (5.13a)–(5.13d). In the previous section, we controlled the states $(\theta_N, \dot{\theta}_N, \phi_o, \dot{\phi}_o)$. Now, we are left with the states $(p, \dot{p}, \lambda, \dot{\lambda})$. The map $\dot{p} \mapsto (v_t, v_n)$ is a diffeomorphism so for velocity control we may consider the subsystem with states $(\tilde{v}_t, v_n, \lambda, \dot{\lambda})$, where $\tilde{v}_t = v_t - v_{\text{ref}}$ denotes the forward velocity error. In order to obtain the tangential and normal velocity dynamics, we take the time derivatives of equations (5.3a)–(5.3b), which using (5.13b) yields

$$\begin{aligned} \dot{v}_t = & u_{\theta_N}^T \Psi_4(\cdot) u_{\theta_N} v_t + u_{\theta_N}^T \Psi_4(\cdot) v_{\theta_N} v_n + \dot{\theta}_N v_n + \\ & u_{\theta_N}^T \Psi_5(\cdot) \dot{\theta}_N + u_{\theta_N}^T \Psi_6(\cdot) \dot{\lambda} + u_{\theta_N}^T \Psi_7(\cdot) \dot{\phi}_o \end{aligned} \quad (5.25a)$$

$$\begin{aligned} \dot{v}_n = & v_{\theta_N}^T \Psi_4(\cdot) u_{\theta_N} v_t + v_{\theta_N}^T \Psi_4(\cdot) v_{\theta_N} v_n - \dot{\theta}_N v_t + \\ & v_{\theta_N}^T \Psi_5(\cdot) \dot{\theta}_N + v_{\theta_N}^T \Psi_6(\cdot) \dot{\lambda} + v_{\theta_N}^T \Psi_7(\cdot) \dot{\phi}_o. \end{aligned} \quad (5.25b)$$

Thus, the velocity error dynamics have the form

$$\begin{aligned} \dot{\tilde{v}}_t = & f_2 \left(\theta_N, \dot{\theta}_N, \lambda, \dot{\lambda}, \phi_o, \dot{\phi}_o, \tilde{v}_t, v_n \right) + \\ & u_{\theta_N}^T \Psi_6(\cdot) \dot{\lambda} \end{aligned} \quad (5.26a)$$

$$\begin{aligned} \dot{v}_n = & f_3 \left(\theta_N, \dot{\theta}_N, \lambda, \dot{\lambda}, \phi_o, \dot{\phi}_o, \tilde{v}_t, v_n \right) + \\ & v_{\theta_N}^T \Psi_4(\cdot) v_{\theta_N} v_n \end{aligned} \quad (5.26b)$$

$$\ddot{\lambda} = u_\lambda \quad (5.26c)$$

In order to stabilize the solutions of (5.26a)–(5.26b) to a neighborhood of the origin $(\tilde{v}_t, v_n) = (0, 0)$, we iteratively introduce control-Lyapunov functions (CLF)

using the techniques of backstepping [99]. To this end, we start by defining the first CLF in the form

$$V_1 = \frac{1}{2}\tilde{v}_t^2, \quad (5.27)$$

and taking its time derivative along the solutions of (5.26a)–(5.26c) to obtain

$$\dot{V}_1 = \tilde{v}_t \dot{\tilde{v}}_t = \tilde{v}_t \left(u_{\theta_N}^T \Psi_6(\cdot) \dot{\lambda} + f_2(\cdot) \right). \quad (5.28)$$

We use $\dot{\lambda}$ as a virtual control input which we employ to make (5.28) negative. In particular, we define

$$\dot{\lambda} = -k_\lambda \tilde{v}_t \quad (5.29)$$

where $k_\lambda > 0$ is a constant. We introduce the error variable

$$z = \dot{\lambda} + k_\lambda \tilde{v}_t, \quad (5.30)$$

that we would like to drive to zero, and rewrite (5.28) as

$$\dot{V}_1 = -k_\lambda u_{\theta_N}^T \Psi_6(\cdot) \tilde{v}_t^2 + \tilde{v}_t u_{\theta_N}^T \Psi_6(\cdot) z + \tilde{v}_t f_2(\cdot). \quad (5.31)$$

To perform backstepping for z , we define a composite CLF of the form

$$V_2 = V_1 + \frac{1}{2}z^2 + \frac{1}{2}v_n^2. \quad (5.32)$$

Taking the time derivative of (5.32) along the solutions of (5.26a)–(5.26c), we have

$$\begin{aligned} \dot{V}_2 &= -u_{\theta_N}^T \Psi_6(\cdot) k_\lambda \tilde{v}_t^2 + z \left(u_\lambda + k_\lambda \dot{\tilde{v}}_t + \tilde{v}_t u_{\theta_N}^T \Psi_6(\cdot) \right) \\ &\quad + v_{\theta_N}^T \Psi_4(\cdot) v_{\theta_N} v_n^2 + v_n \left(f_3(\cdot) - v_{\theta_N}^T \Psi_6(\cdot) \dot{\lambda} \right) \\ &\quad + v_n v_{\theta_N}^T \Psi_6(\cdot) (z - k_\lambda \tilde{v}_t). \end{aligned} \quad (5.33)$$

In order to achieve the velocity control objective, we define the feedback controller

$$\begin{aligned} u_\lambda &= -k_\lambda \underbrace{\left\{ f_2(\cdot) + u_{\theta_N}^T \Psi_6(\cdot) \dot{\lambda} \right\}}_{\dot{\tilde{v}}_t} - K_z z \\ &\quad - \tilde{v}_t u_{\theta_N}^T \Psi_6(\cdot) - v_n v_{\theta_N}^T \Psi_6(\cdot). \end{aligned} \quad (5.34)$$

where $K_z > 0$ is a constant gain. The following remark investigates the boundedness of the solutions of the controlled system.

Remark 5.4 Consider the state vector $x = [v_t, v_n, \lambda, \dot{\lambda}, \phi_o, \dot{\phi}_o]^T$. Under the control laws (5.34) and (5.16), we have $\dot{x} = f(x)$ for the closed-loop system. Because of the uniform bounds on $\Psi_i, i = 2, \dots, 7$, it can be seen that

$$\|f(x)\| \leq B(1 + \|x\|) \quad (5.35)$$

for some constant B . Because of this linear growth condition, there is no finite escape time and the signals $\dot{\lambda}(t), u_\lambda(t)$ are defined for all $t \geq 0$ as required by Proposition 5.1.

We have the following proposition regarding the velocity control system.

Proposition 5.2 Consider the control system (5.26a)–(5.26c) under the controller (5.34). If the ultimate bound on ϕ_o from Proposition 5.1 is small enough such that $u_{\theta_N}^T \Psi_6(\cdot)$ is bounded away from zero, then for all $\epsilon > 0$, there exists a controller gain $k_\lambda > 0$ and positive constants α^*, c^* such that, for all $\alpha \in (0, \alpha^*)$ and all $c_n - c_t > c^*$, the set

$$\Gamma' = \left\{ (\lambda, \dot{\lambda}, v_t, v_n) \mid |\tilde{v}_t| < \epsilon \right\} \quad (5.36)$$

is asymptotically stable. Moreover, $\dot{\lambda}$ and v_n are uniformly ultimately bounded.

Remark 5.5 Under (5.34), the velocity error \tilde{v}_t can be made arbitrarily small provided that the gain k_λ is chosen to be sufficiently large.

Proof: Substituting (5.34) into (5.33) yields

$$\begin{aligned} \dot{V}_2 &= -u_{\theta_N}^T \Psi_6(\cdot) k_\lambda \tilde{v}_t^2 - K_z z^2 + v_{\theta_N}^T \Psi_4(\cdot) v_{\theta_N} v_n^2 \\ &+ v_n \left(f_3(\cdot) - v_{\theta_N}^T \Psi_6(\cdot) \dot{\lambda} \right) - k_\lambda v_n v_{\theta_N}^T \Psi_6(\cdot) \tilde{v}_t \end{aligned} \quad (5.37)$$

Therefore, by parts (b) and (c) of Remark 5.1, for small enough ϕ_o , we have

$$\begin{aligned} \dot{V}_2 &\leq -\gamma_6 k_\lambda \tilde{v}_t^2 - K_z z^2 - \frac{c_n}{m} v_n^2 \\ &+ v_n \underbrace{\left(v_{\theta_N}^T \Psi_4(\cdot) u_{\theta_N} \tilde{v}_t + v_{\theta_N}^T \Psi_7(\cdot) \dot{\phi}_o + v_{\theta_N}^T \Psi_4(\cdot) u_{\theta_N} v_{\text{ref}} \right)}_{f_3(\cdot) - v_{\theta_N}^T \Psi_6(\cdot) \dot{\lambda}} \\ &- k_\lambda v_n v_{\theta_N}^T \Psi_6(\cdot) \tilde{v}_t \end{aligned} \quad (5.38)$$

By Proposition 5.1, $\dot{\phi}_o$ is uniformly ultimately bounded by a positive constant $\delta_{\dot{\phi}_o} > 0$. By parts (d), (e), (f), (g) of Remark 5.1, we have $\|\Psi_4\| \leq c_n/m, \|\Psi_7\| \leq \gamma_7$,

$|v_{\theta_N}^T \Psi_4 u_{\theta_N}| < c_t/m$, and $|v_{\theta_N}^T \Psi_6(\cdot)| \leq \alpha \epsilon_0$, and thus

$$\begin{aligned} \dot{V}_2 &\leq -\gamma_6 k_\lambda \tilde{v}_t^2 - K_z z^2 - \frac{c_n}{m} v_n^2 \\ &\quad + \frac{c_t}{m} |v_n| |\tilde{v}_t| + \gamma_7 |v_n| |\dot{\phi}_o| + \\ &\quad \frac{c_n}{m} v_{\text{ref}} |v_n| + k_\lambda \epsilon_0 \alpha |v_n| |\tilde{v}_t|. \end{aligned} \quad (5.39)$$

Using the fact that, for any $\gamma > 0$, $ab \leq (\gamma/2)a^2 + (1/2\gamma)b^2$, we have

$$\begin{aligned} \dot{V}_2 &\leq - \left[k_\lambda \gamma_6 - \frac{1}{2} \left(\frac{c_t}{m} + k_\lambda \alpha \epsilon_0 \right) \right] \tilde{v}_t^2 - K_z z^2 \\ &\quad - \left[\frac{c_n}{m} - \left(\frac{c_t}{m} + k_\lambda \alpha \epsilon_0 \right) \frac{1}{2} - \frac{\gamma_7 \gamma}{2} - \frac{c_n \gamma'}{2m} \right] v_n^2 + \frac{\gamma_7}{2\gamma} \delta_{\phi_o}^2 \\ &\quad + \frac{c_n}{2m\gamma'} v_{\text{ref}}^2, \end{aligned} \quad (5.40)$$

where γ and γ' are arbitrary positive numbers. Pick $\alpha < \gamma_6/\epsilon_0$ and $k_\lambda > c_t/(m\gamma_6)$. Then, the coefficient pre-multiplying \tilde{v}_t^2 in (5.40) is negative. Moreover, if $c_n > (c_t/2) + mk_\lambda \gamma_6$, then for sufficiently small $\gamma, \gamma' > 0$, the coefficient of v_n^2 in (5.40) is also negative. Consequently, for sufficiently large $c_n - c_t$ and k_λ , and sufficiently small α , there exists $\beta > 0$ such that

$$\dot{V}_2 \leq -\beta V_2 + \frac{\gamma_7}{2\gamma} \delta_{\phi_o}^2 + \frac{c_n}{2m\gamma'} v_{\text{ref}}^2, \quad (5.41)$$

from which it follows, by the Comparison Lemma [99], that

$$V_2(t) \leq V_2(0) \exp(-\beta t) + \frac{\left(\frac{\gamma_7}{2\gamma} \delta_{\phi_o}^2 + \frac{c_n}{2m\gamma'} v_{\text{ref}}^2 \right)}{\beta}. \quad (5.42)$$

This implies that the solutions of (5.26a)–(5.26c), i.e., $\tilde{v}_t, v_n, \dot{\lambda}$, remain bounded, V_2 converges to a ball of radius

$$r_2 = \frac{\left(\frac{\gamma_7}{2\gamma} \delta_{\phi_o}^2 + \frac{c_n}{2m\gamma'} v_{\text{ref}}^2 \right)}{\beta} \quad (5.43)$$

and therefore $\|[\tilde{v}_t, v_n, \dot{\lambda}]^T\|$ converges to a neighborhood of the origin given by

$$r = \sqrt{\frac{\left(\frac{\gamma_7}{2\gamma} \delta_{\phi_o}^2 + \frac{c_n}{2m\gamma'} v_{\text{ref}}^2 \right)}{2\beta}}. \quad (5.44)$$

Moreover, we can use the first CLF, i.e., $V_1 = 1/2\tilde{v}_t^2$, to show the practical stability of the tangential velocity. Taking its time derivative along the solutions of (5.26a)–(5.26c) given in (5.28) and using the fact that $u_{\theta_N}^T \Psi_6$ is uniformly bounded we have

$$\dot{V}_1 \leq -k_\lambda \gamma_6 \tilde{v}_t^2 + \Upsilon_6 |\tilde{v}_t| |z| + |\tilde{v}_t| |f_2(\cdot)| \quad (5.45)$$

By the previous argument, $|z|$ is ultimately bounded. We denote the ultimate bound by $\delta_z > 0$. Also, there exists $\delta_2 > 0$ such that $\|f_2(\cdot)\| < \delta_2$, so that

$$\dot{V}_1 \leq -\left(k_\lambda \gamma_6 - \frac{\Upsilon_6}{2} - \frac{1}{2}\right) \tilde{v}_t^2 + \underbrace{\frac{\Upsilon_6 \delta_z^2}{2} + \frac{1}{2} \delta_2^2}_d. \quad (5.46)$$

For sufficiently large k_λ , there exists $\beta > 0$ such that

$$\dot{V}_1 \leq -2\beta V_1 + d, \quad (5.47)$$

from which we get

$$V_1(t) \leq \exp(-2\beta t) V_1(0) + \frac{1}{2\beta} d, \quad t \geq 0. \quad (5.48)$$

Therefore, \tilde{v}_t converges to a ball of radius $\sqrt{\frac{1}{\beta}d}$. Since $\beta = k_\lambda \gamma_6 - \frac{\Upsilon_6 \gamma}{2} - \frac{1}{2}$, choosing k_λ large enough makes the ultimate bound of \tilde{v}_t less than ϵ for any desired $\epsilon > 0$. This completes the proof for practical stability of the origin $\tilde{v}_t = 0$ of the tangential velocity error dynamics (5.26a) under the control law (5.34) with a sufficiently large controller gain k_λ . ■

Remark 5.6 Note that c_n/m must be sufficiently large to dominate the positive term. The physical interpretation of this inequality is that it is only the friction that damps out the normal velocity component. No active control is used here. And this normal friction coefficient must be large enough (w.r.t. the mass m) in order to dominate the velocity increasing contribution that comes from the excitation that the sinusoidal gait pattern gives.

5.2 Maneuvering Control of Snake Robots

This section investigates the problem of maneuvering control for planar snake robots. The control objective is to make the CM of the snake robot converge to a desired path and traverse the path with a desired velocity. The proposed feedback control strategy enforces virtual constraints encoding a lateral undulatory gait, parametrized by states of dynamic compensators which are employed to regulate the orientation and position of the snake robot.

5.2.1 Control Design Objectives

In this subsection, we present the blueprint of our maneuvering control design for the snake robot. We begin by stating the control specifications which we aim to achieve throughout the control design.

Velocity Control Problem (VCP): Given a desired velocity vector $\mu(p)$ with polar representation $(r, \theta) = (v_{\text{ref}}(p), \theta_{\text{ref}}(p))$, design a smooth feedback controller achieving the following specifications:

- (i) Practical stabilization of the head angle θ_N to $\theta_{\text{ref}}(p)$.
- (ii) Practical stabilization of the tangential velocity $v_t = u_{\theta_N}^T \dot{p}$ to $v_{\text{ref}}(p)$.
- (iii) Uniform ultimate boundedness of the normal velocity $v_n = v_{\theta_N}^T \dot{p}$ with a small ultimate bound, and ultimate boundedness of the solutions of the joint dynamics and all controller states.

The above problem formulation relies on the observation that if $\theta_N = \theta_{\text{ref}}(p)$, then making $\dot{p} \rightarrow \mu(p)$ is equivalent to making $(v_t, v_n) \rightarrow (v_{\text{ref}}(p), 0)$.

Path Following Problem (PFP): Given a desired continuously differentiable planar curve $\gamma \subset \mathbb{R}^2$ with implicit representation $\{p \in \mathbb{R}^2 : h(p) = 0\}$ with $dh_p \neq 0$ on γ , design a smooth feedback controller achieving the following specifications:

- (i) Path stabilization: make $p(t) \rightarrow \gamma$.
- (ii) Velocity control: make $\|\dot{p}\| = v$ on γ , where v is the desired velocity on the path γ .

The first control specification, i.e., the VCP, will be used to achieve the second control specification, i.e., the PFP.

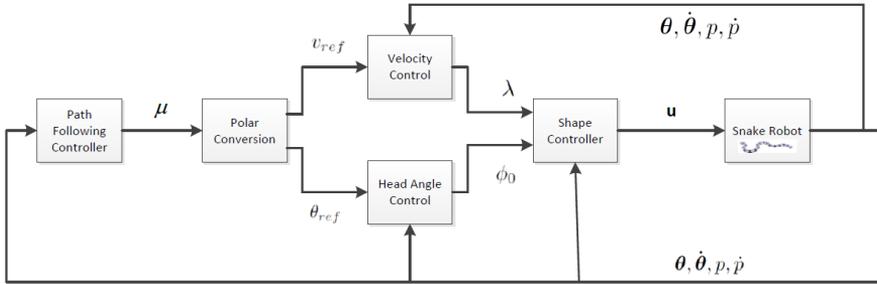


Figure 5.2: The structure of the path following controller.

Solution Methodology:

In order to solve VCP and PFP, we create a hierarchy of three control specifications, and therefore our control design unfolds in three stages.

Stage 1: Body Shape Control. We use the controls u in (5.1a) to stabilize VHC encoding a lateral undulatory gait similar to the well-known reference joint trajectories (5.4), in which (ωt) is replaced by a state λ , and ϕ_o affects only the head angle θ_N . The evolution of λ, ϕ_o is governed by two dynamic compensators, $\ddot{\phi}_o = u_{\phi_o}$ and $\ddot{\lambda} = u_\lambda$, which will be used to control the orientation and position of the robot, respectively.

Stage 2: Velocity Control. This stage unfolds in two substages:

- **Head Angle Control.** Inspired by the biological observation that snakes keep their head pointed towards a target while their body undulates behind the head, we design u_{ϕ_o} to practically stabilize $\theta_N \rightarrow \theta_{ref}(p)$ while guaranteeing that $(\phi_o, \dot{\phi}_o)$ is uniformly ultimately bounded.
- **Speed Control.** We design u_λ to practically stabilize $v_t \rightarrow v_{ref}(p)$ while guaranteeing that v_n converges into a small neighborhood of the origin and $\dot{\lambda}$ is uniformly ultimately bounded.

Stage 3: Path Following Control. We design $\mu(p)$ such that making $|\dot{p} - \mu(p)|$ sufficiently small solves the PFP.

Remark 5.7 *Achieving shape control, head angle control, and velocity regulation solves the DFP, making $|\dot{p} - \mu(p)|$ sufficiently small.*

Figure 5.2 depicts the structure of the proposed path following controller.

Remark 5.8 *As discussed at the beginning of this chapter, snake robots move forwards by tracing out a periodic curve. Because of this oscillatory motion, the head angle and velocity tangential and normal to the snake motion will not be constant, but rather oscillate around their steady state values. This is the reason why practical stability is sought, as opposed to asymptotic stability of constant values which is not a feasible control objective for the snake robot locomotion.*

5.2.2 Body Shape Control

As we discussed above, the first stage of our path following control design approach is to control the body shape of the snake robot by controlling the joint angles to given references given by the following VHC

$$\theta_i - \theta_{i+1} = \alpha \sin(\lambda + (i - 1)\delta), \quad i = 1, \dots, N - 2, \quad (5.49a)$$

$$\theta_{N-1} - \theta_N = \alpha \sin(\lambda + (N - 2)\delta) + \phi_o \quad (5.49b)$$

Associated with the given VHC is the constraint manifold

$$\Gamma_3 = \left\{ (\theta, \dot{\theta}, p, \dot{p}, \lambda, \dot{\lambda}, \phi_o, \dot{\phi}_o) \in \mathbb{R}^{2N+8} : D\theta = \Phi(\lambda) + b\phi_o, D\dot{\theta} = \Phi'(\lambda)\dot{\lambda} + b\dot{\phi}_o \right\}. \quad (5.50)$$

Stabilizing the VHC (5.49a)–(5.49b) corresponds to stabilizing Γ_3 . To this end, we use the input-output linearizing control law (5.12), which asymptotically stabilizes the constraint manifold Γ_3 for the dynamical system (5.1a)–(5.1b).

Reduced Dynamics on the Constraint Manifold

After asymptotically stabilizing Γ_3 , we are left with two control inputs, (u_λ, u_{ϕ_o}) to solve the VCP. As described in Section 5.1, we will use the dynamic compensators to regulate the head angle and the velocity of the robot to desired values. To this end, we first derive the reduced dynamics of the robot, i.e. we reduce the system to the invariant manifold Γ_3 . By left multiplying both sides of (5.1a) by e^T , which is a left annihilator of the control input matrix D^T , and evaluating the result on the virtual constraint manifold Γ_3 , the dynamics of the snake robot on the virtual constraint manifold Γ_3 read as

$$\ddot{\theta}_N = \Psi_1(\theta_N, \dot{\theta}_N, \lambda, \dot{\lambda}, \phi_o, \dot{\phi}_o, p, \dot{p}) + \Psi_2(\theta_N, \lambda, \phi_o) u_\lambda + \Psi_3(\theta_N, \lambda, \phi_o) u_{\phi_o}, \quad (5.51a)$$

$$\ddot{p} = \Psi_4(\theta_N, \lambda, \phi_o) \dot{p} + \Psi_5(\theta_N, \lambda, \phi_o) \dot{\theta}_N + \Psi_6(\theta_N, \lambda, \phi_o) \dot{\lambda} + \Psi_7(\theta_N, \lambda, \phi_o) \dot{\phi}_o, \quad (5.51b)$$

$$\ddot{\phi}_o = u_{\phi_o}, \quad (5.51c)$$

$$\ddot{\lambda} = u_\lambda, \quad (5.51d)$$

where the scalar-valued functions $\Psi_i(\cdot)$ for all $i \in \{1, \dots, 7\}$ were defined in (5.14a)–(5.14f). Note that each $\Psi_i(\cdot)$ is evaluated on the constraint manifold Γ_3 . In the process of developing controllers for the reduced dynamics system, we use the knowledge of each $\Psi_i(\cdot)$ that we described in Remark 5.1. The equations in (5.51a)–(5.51d) describe a control system with two inputs, (u_{ϕ_o}, u_λ) . This system completely describes the motion of the snake once the VHC (5.49a)–(5.49b) have been enforced. For system (5.51a)–(5.51d), the control objective is to stabilize θ_N to an arbitrarily small neighborhood of θ_{ref} ; to stabilize $v_t = u_{\theta_N}^T \dot{p}$ to an arbitrarily small neighborhood of v_{ref} ; and finally, to guarantee that $v_n = v_{\theta_N}^T \dot{p}$ converges to a neighborhood of the origin. Meanwhile, we also require that the solutions of the dynamic compensators $(\dot{\lambda}, \dot{\phi}_o, \dot{\phi}_o)$ remain bounded.

5.2.3 Velocity Control

According to the solution methodology given above, we solve the velocity control problem in two steps. In the first step, we control the head angle of the robot to a reference head angle which points towards the target point on the path. In the second step, we use the frequency of the periodic body motions, i.e. the gait pattern, as an additional control term in order to control the velocity of the robot to a reference velocity.

Head Angle Control

In this section, we consider the head angle control for the snake robot. Using the control input u_{ϕ_o} , we control the head angle of the snake robot by controlling the states $(\theta_N, \dot{\theta}_N, \phi_o, \dot{\phi}_o)$ of the constrained system (5.51a)–(5.51d). To this end, we design a high-gain feedback $u_{\phi_o}(\theta_N, \dot{\theta}_N, \phi_o, \dot{\phi}_o)$ that makes $(\theta_N - \theta_{\text{ref}}(p), \dot{\theta}_N - \dot{\theta}_{\text{ref}}(p))$ converge to an arbitrarily small neighborhood of the origin and $(\phi_o, \dot{\phi}_o)$ uniformly ultimately bounded. This analysis is made independent of the choice of u_λ , using time scale separation.

By (5.51a) and (5.51c), the dynamic equations governing the states $(\theta_N, \dot{\theta}_N, \phi_o, \dot{\phi}_o)$ of the constrained system can be written as

$$\begin{aligned}\ddot{\theta}_N &= f_1 \left(\theta_N, \dot{\theta}_N, \lambda, \dot{\lambda}, \phi_o, \dot{\phi}_o, u_\lambda \right) + \Psi_3(\cdot) u_{\phi_o}, \\ \ddot{\phi}_o &= u_{\phi_o}.\end{aligned}\tag{5.52}$$

In the next proposition, we consider the head angle control for the robot using a high-gain feedback control law.

Proposition 5.3 Consider the head angle control law for the system (5.52) defined as

$$u_{\phi_o} = \frac{1}{\Psi_3(\cdot)} \left\{ \frac{1}{\epsilon} \left(\dot{\tilde{\theta}}_N + k_N \tilde{\theta}_N \right) \right\} - k_1 \phi_o - k_2 \dot{\phi}_o. \quad (5.53)$$

where $\tilde{\theta}_N = \theta_N - \theta_{\text{ref}}(p)$ denotes the head angle error. Also, assume that $u_\lambda(t), \lambda(t), \dot{\lambda}(t)$ are defined for all $t \geq 0$. Then for any $k_1, k_2, \epsilon_1 > 0$, there exist $\epsilon, k_N, \epsilon_2 > 0$ and a positive definite function $V(\phi_o, \dot{\phi}_o)$ such that the set

$$\Omega^* = \left\{ \left(\theta_N, \dot{\theta}_N, \phi_o, \dot{\phi}_o \right) \mid \left\| \left(\tilde{\theta}_N, \dot{\tilde{\theta}}_N + k_N \tilde{\theta}_N \right) \right\| \leq \epsilon_1, V(\phi_o, \dot{\phi}_o) \leq \epsilon_2 \right\} \quad (5.54)$$

is asymptotically stable. Then for any $k_N, k_1, k_2 > 0$, there exist $\epsilon^*, k > 0$ such that for all $\epsilon \in (0, \epsilon^*)$

$$\limsup_{t \rightarrow +\infty} \left| \tilde{\theta}_N(t) \right| = k\epsilon \quad (5.55)$$

$$\limsup_{t \rightarrow +\infty} \left| \dot{\tilde{\theta}}_N(t) \right| = k\epsilon. \quad (5.56)$$

Moreover, the states of the dynamic compensator $(\phi_o, \dot{\phi}_o)$ are uniformly ultimately bounded.

Remark 5.9 In the next section we define a feedback controller u_λ guaranteeing that for any initial condition, the closed-loop system has no finite escape time (see Remark 5.11). This will guarantee that the above proposition is applicable.

Remark 5.10 The result of Proposition 5.3 can be interpreted as follows. Under (5.53), the head angle error can be made arbitrarily small provided that ϵ is chosen to be sufficiently small. Also, ϕ_o and $\dot{\phi}_o$ remain uniformly ultimately bounded.

Proof: Viewing the states $\lambda(t), \dot{\lambda}(t)$, and the input $u_\lambda(t)$ as exogenous signals, the control system (5.52) can be viewed as a time-varying system with states $(\theta_N, \dot{\theta}_N, \phi_o, \dot{\phi}_o)$. Under the control input (5.53), the closed-loop dynamics of the system (5.52) in the standard singular perturbation form become

$$\dot{\tilde{\theta}}_N = \tilde{\omega}_N, \quad (5.57a)$$

$$\epsilon \dot{\tilde{\omega}}_N = \epsilon \left[\ddot{\theta}_{\text{ref}} + g_1 \left(t, \phi_o, \dot{\phi}_o, \theta_N, \dot{\theta}_N \right) + \Psi_3(\cdot) \left(k_1 \phi_o + k_2 \dot{\phi}_o \right) \right] - \left(\tilde{\omega}_N + k_N \tilde{\theta}_N \right), \quad (5.57b)$$

where

$$g_1 \left(t, \phi_o, \dot{\phi}_o, \theta_N, \dot{\theta}_N \right) = f_1 \left(\theta_N, \dot{\theta}_N, \lambda(t), \dot{\lambda}(t), \phi_o, \dot{\phi}_o, u_\lambda(t) \right). \quad (5.58)$$

Here we use time-scale separation to make the analysis independent of the choice of u_λ . Note that (5.57a)–(5.57b) is a singularly perturbed system with reduced dynamics

$$\dot{\tilde{\theta}}_N = -k_N \tilde{\theta}_N \quad (5.59)$$

and boundary-layer dynamics

$$\frac{d\hat{y}}{d\tau} = -\hat{y} \quad (5.60)$$

where $\hat{y} = \tilde{\omega}_N + k_N \tilde{\theta}_N$. The origin is an exponentially stable equilibrium point of the reduced system. Also, the origin is an exponentially stable equilibrium point of the boundary-layer system. According to the singular perturbation theorem on an infinite interval (see Theorem 11.2 in [99]), for all initial conditions $\tilde{\theta}_N(t_0), \hat{y}(t_0) \in \mathbb{R}$ and $t_0 \geq 0$, the singularly perturbed system (5.57a)–(5.57b) has a unique solution $(\tilde{\theta}_N(t, \epsilon), \tilde{\omega}_N(t, \epsilon))$ such that

$$\begin{aligned} & \tilde{\theta}_N(t, \epsilon) - \exp(-k_N(t - t_0)) \tilde{\theta}_N(t_0) \\ & = O(\epsilon), \end{aligned} \quad (5.61a)$$

$$\begin{aligned} & \tilde{\omega}_N(t, \epsilon) + k_N \exp(-k_N(t - t_0)) \tilde{\theta}_N(t_0) \\ & - \exp\left(-\frac{t}{\epsilon}\right) \hat{y}(t_0) = O(\epsilon), \end{aligned} \quad (5.61b)$$

for all $t \in [t_0, \infty)$. Note that the closed-loop dynamics governing the states $(\phi_o, \dot{\phi}_o)$ become

$$\ddot{\phi}_o + k_2 \dot{\phi}_o + k_1 \phi_o = \underbrace{\frac{1}{\Psi_3(\cdot)} \left\{ \frac{1}{\epsilon} \left(\dot{\tilde{\theta}}_N + k_N \tilde{\theta}_N \right) \right\}}_{f_N(t, \epsilon)}. \quad (5.62)$$

From (5.61a)–(5.61b), it can be seen that $f_N(t, \epsilon)$ is uniformly bounded and of order $O(1)$. Since the unforced system $\ddot{\phi}_o + k_2 \dot{\phi}_o + k_1 \phi_o = 0$ is an LTI system and has a globally exponentially stable equilibrium point at the origin $(\phi_o, \dot{\phi}_o) = (0, 0)$, the system (5.62) is input-to-state stable. Therefore, there exists an ISS Lyapunov function $V(\cdot)$ and ϵ_2 such that the set $\{V(\phi_o, \dot{\phi}_o) \leq \epsilon_2\}$ is asymptotically stable. Now, we consider the change of variable $\hat{y} = \tilde{\omega}_N + k_N \tilde{\theta}_N$. The closed-loop dynamics become

$$\dot{\tilde{\theta}}_N = \hat{y} - k_N \tilde{\theta}_N, \quad (5.63a)$$

$$\epsilon \dot{\hat{y}} = \epsilon \left[\ddot{\theta}_{\text{ref}} + g_1 \left(t, \phi_o, \dot{\phi}_o, \theta_N, \dot{\theta}_N \right) + \Psi_3(\cdot) \left(k_1 \phi_o + k_2 \dot{\phi}_o \right) + k_N \left(\hat{y} - k_N \tilde{\theta}_N \right) \right] - \hat{y}, \quad (5.63b)$$

Next, we consider the Lyapunov function candidate

$$V_1 = \frac{1}{2}\tilde{\theta}_N^2 + \frac{1}{2}\hat{y}^2 \quad (5.64)$$

Taking the time-derivative of V_1 along the trajectories of (5.63a)–(5.63b) yields

$$\dot{V}_1 = \tilde{\theta}_N \dot{\hat{y}} - k_N \tilde{\theta}_N^2 + \hat{y} \dot{\hat{y}} \quad (5.65)$$

It can be shown that there exists $L_3 > 0$ such that $\hat{y} \dot{\hat{y}} \leq -(1/2\epsilon)\hat{y}^2 + L_3\hat{y}$ (see proof of Th. 11.1 in [99]). We have

$$\dot{V}_1 \leq \tilde{\theta}_N \hat{y} - k_N \tilde{\theta}_N^2 - \frac{1}{2\epsilon}\hat{y}^2 + L_3\hat{y} \quad (5.66)$$

Completing the square, we get

$$\dot{V}_1 \leq -\left(k_N - \frac{1}{2}\right)\tilde{\theta}_N^2 - \left(\frac{1}{2\epsilon} - 1\right)\hat{y}^2 + \frac{1}{2}L_3^2 \quad (5.67)$$

For $k_N > (1/2)(L_3^2/4\epsilon_1^2 + 1)$ and $\epsilon < 1/(L_3^2/4\epsilon_1^2 + 2)$ we have

$$\dot{V}_1 \leq -\frac{L_3^2}{4\epsilon_1^2}V + \frac{1}{2}L_3^2 \quad (5.68)$$

By the comparison lemma, we get

$$V_1(t) \leq V_1(0) \exp\left(-\frac{L_3^2}{4\epsilon_1^2}t\right) + 2\epsilon_1^2 \quad (5.69)$$

This implies that $\|[\tilde{\theta}_N, \hat{y}]^T\|$ converges to a neighborhood of the origin given by ϵ_1 . Therefore, the set $\{\|[\tilde{\theta}_N, \hat{y}]^T\| \leq \epsilon_1\}$ is asymptotically stable. Note that ϵ_1 is a design parameter that we can choose arbitrarily. This completes the proof. ■

Speed Control

Consider the reduced dynamics (5.51a)–(5.51d). In the previous section, we controlled the states $(\theta_N, \dot{\theta}_N, \phi_o, \dot{\phi}_o)$. Now, we are left with the states $(p, \dot{p}, \lambda, \dot{\lambda})$. The map $\dot{p} \mapsto (v_t, v_n)$ is a diffeomorphism so for velocity control we may consider the subsystem with states $(\tilde{v}_t, v_n, \lambda, \dot{\lambda})$, with $\tilde{v}_t = v_t - v_{\text{ref}}(p)$ denotes the tangential velocity error. In order to obtain the tangential and normal velocity dynamics evaluated on the constraint manifold, we take the time derivatives of Equations (5.3a)–(5.3b), which using (5.51b) yields

$$\begin{aligned} \dot{v}_t = & u_{\theta_N}^T \Psi_4(\cdot) u_{\theta_N} v_t + u_{\theta_N}^T \Psi_4(\cdot) v_{\theta_N} v_n + \dot{\theta}_N v_n + \\ & u_{\theta_N}^T \Psi_5(\cdot) \dot{\theta}_N + u_{\theta_N}^T \Psi_6(\cdot) \dot{\lambda} + u_{\theta_N}^T \Psi_7(\cdot) \dot{\phi}_o \end{aligned} \quad (5.70a)$$

$$\begin{aligned} \dot{v}_n = & v_{\theta_N}^T \Psi_4(\cdot) u_{\theta_N} v_t + v_{\theta_N}^T \Psi_4(\cdot) v_{\theta_N} v_n - \dot{\theta}_N v_t + \\ & v_{\theta_N}^T \Psi_5(\cdot) \dot{\theta}_N + v_{\theta_N}^T \Psi_6(\cdot) \dot{\lambda} + v_{\theta_N}^T \Psi_7(\cdot) \dot{\phi}_o. \end{aligned} \quad (5.70b)$$

Thus, the velocity error dynamics have the form

$$\begin{aligned} \dot{\tilde{v}}_t = & f_2 \left(\theta_N, \dot{\theta}_N, \lambda, \phi_o, \dot{\phi}_o, \tilde{v}_t, v_n \right) + \\ & u_{\theta_N}^T \Psi_6(\cdot) \dot{\lambda} - (dv_{\text{ref}})_p \dot{p}, \end{aligned} \quad (5.71a)$$

$$\begin{aligned} \dot{v}_n = & f_3 \left(\theta_N, \dot{\theta}_N, \lambda, \dot{\lambda}, \phi_o, \dot{\phi}_o, \tilde{v}_t, v_n \right) + \\ & v_{\theta_N}^T \Psi_4(\cdot) v_{\theta_N} v_n, \end{aligned} \quad (5.71b)$$

$$\ddot{\lambda} = u_\lambda. \quad (5.71c)$$

In order to stabilize the solutions of (5.71a)–(5.71b) to a neighborhood of the origin, we use the following control input

$$\begin{aligned} u_\lambda = & -K_z \left(\dot{\lambda} + K_\lambda \tilde{v}_t \right) \\ & - K_\lambda \left[f_2(\cdot) + u_{\theta_N}^T \Psi_6(\cdot) \dot{\lambda} - (dv_{\text{ref}})_p \dot{p} \right]. \end{aligned} \quad (5.72)$$

where $K_\lambda > 0$ and $K_z > 0$ are positive constant controller gains. Note that $u_{\theta_N}^T \Psi_6(\cdot)$ is bounded away from zero by part (c) of Remark 5.1 provided that the ultimate bound on ϕ_o from Proposition 5.1 is small enough.

Remark 5.11 Consider the state vector $x = [v_t, v_n, \lambda, \dot{\lambda}, \phi_o, \dot{\phi}_o]^T$. Under the control laws (5.53) and (5.72), we have $\dot{x} = f(x)$ for the closed-loop system. Because of the uniform bounds on $\Psi_i, i = 2, \dots, 7$, it can be seen that

$$\|f(x)\| \leq B(1 + \|x\|) \quad (5.73)$$

for some constant B . Because of this linear growth condition, there is no finite escape time and the signals $\dot{\lambda}(t), u_\lambda(t)$ are defined for all $t \geq 0$ as required by Proposition 5.3.

We have the following proposition regarding the forward velocity control system.

Proposition 5.4 Consider the control system (5.71a)–(5.71c) under the controller (5.72) with $c_n > c_t$. If the ultimate bound on ϕ_o from Proposition 5.3 is small enough that $u_{\theta_N}^T \Psi_6(\cdot)$ is bounded away from zero, then for all $\epsilon_3 > 0$ and for sufficiently large controller gain $K_\lambda > 0$, there exists $\epsilon_4 > 0$ such that the compact set

$$\Lambda_1 = \left\{ \left(\lambda, \dot{\lambda}, \tilde{v}_t, v_n \right) : |\tilde{v}_t| \leq \epsilon_3, \dot{\lambda} = -K_\lambda \tilde{v}_t, |v_n| \leq \epsilon_4 \right\} \quad (5.74)$$

is asymptotically stable.

Remark 5.12 *The result of Proposition 5.4 can be interpreted as follows. Under (5.72), the velocity error \tilde{v}_t can be made arbitrarily small provided that the gain K_λ is chosen sufficiently large. Also, the normal velocity v_n remains uniformly ultimately bounded.*

Proof: The control law (5.72) is a feedback linearizing controller for system (5.71a) with output $z = \dot{\lambda} + K_\lambda \tilde{v}_t$, and it makes the set

$$\Lambda_3 = \left\{ \left(\lambda, \dot{\lambda}, \tilde{v}_t, v_n \right) : \dot{\lambda} = -K_\lambda \tilde{v}_t \right\} \quad (5.75)$$

asymptotically stable. On the set Λ_3 , the subsystem (5.71a) becomes

$$\dot{\tilde{v}}_t = f_2(\cdot) - K_\lambda u_{\theta_N}^T \Psi_6(\cdot) \tilde{v}_t - (dv_{\text{ref}})_p \dot{p} \quad (5.76)$$

Now, we find a positively invariant set

$$\Omega = \left\{ \left(\lambda, \dot{\lambda}, \tilde{v}_t, v_n \right) : |\tilde{v}_t| \leq \bar{V}_1, |v_n| \leq \bar{V}_2 \right\}, \quad (5.77)$$

such that $|f_2(\theta_N, \dot{\theta}_N, \lambda, \phi_o, \dot{\phi}_o, \tilde{v}_t, v_n)|$ is uniformly bounded on Ω . Note that $(\phi_o, \dot{\phi}_o)$ have been proven to be uniformly ultimately bounded in Proposition 5.3. Therefore we need to show boundedness of \tilde{v}_t, v_n . We pick \bar{V}_1 arbitrary and determine K_3 such that $|f_3(\cdot)| \leq K_3$. Note that K_3 depends on \bar{V}_1 . Next, we pick $\bar{V}_2 > K_3/K_n$. Finally, we choose

$$K_\lambda > \frac{K_1 + K_2 \bar{V}_2}{\gamma_6 \bar{V}_1}. \quad (5.78)$$

We claim that Ω is positively invariant. Note that

$$-K_\lambda \gamma_6 \tilde{v}_t - K_1 - K_2 |v_n| \leq \dot{\tilde{v}}_t \leq -K_\lambda \gamma_6 \tilde{v}_t - K_1 + K_2 |v_n|, \quad (5.79)$$

and

$$-K_n v_n - K_3 \leq \dot{v}_n \leq -K_n v_n + K_3. \quad (5.80)$$

On $\tilde{v}_t = \bar{V}_1$, we have

$$\dot{\tilde{v}}_t \leq -K_\lambda \gamma_6 \bar{V}_1 + K_1 + K_2 |v_n| \leq -K_\lambda \gamma_6 \bar{V}_1 + K_1 + K_2 \bar{V}_2 \leq 0 \quad (5.81)$$

On $\tilde{v}_t = -\bar{V}_1$, we have

$$\dot{\tilde{v}}_t \geq K_\lambda \gamma_6 \bar{V}_1 - K_1 - K_2 |v_n| \geq K_\lambda \gamma_6 \bar{V}_1 - K_1 - K_2 \bar{V}_2 \geq 0 \quad (5.82)$$

On $v_n = \bar{V}_2$, we have

$$\dot{v}_n \leq -K_n \bar{V}_2 + K_3 \leq 0. \quad (5.83)$$

On $v_n = -\bar{V}_2$, we have

$$\dot{v}_n \geq K_n \bar{V}_2 - K_3 \geq 0 \quad (5.84)$$

The inequalities above prove that on $\partial\Omega$, the vector field given by (5.71a)–(5.71b) points inside Ω . Therefore, by Nagumo's theorem [111], the set Ω is positively invariant. For all initial conditions inside the set Ω , we have $|f_2(\cdot)| \leq \gamma_2 = K_1 + K_2 \bar{V}_2$, i.e. the function $f_2(\cdot)$ is bounded. Now, we employ the Lyapunov function candidate

$$V_1 = \frac{1}{2} \tilde{v}_t^2, \quad (5.85)$$

for which we have $\dot{V}_1 < -K_\lambda \gamma_6 \tilde{v}_t^2 + \gamma_2 \tilde{v}_t$. Therefore we have

$$\dot{V}_1 < -\left(K_\lambda \gamma_6 - \frac{1}{2}\right) \tilde{v}_t^2 + \frac{1}{2} \gamma_2^2. \quad (5.86)$$

Using the Comparison Lemma, we have, for all $t \geq 0$

$$V_1(t) \leq \exp\left(-\left(K_\lambda \gamma_6 - \frac{1}{2}\right)t\right) V_1(0) + \frac{1}{2\left(K_\lambda \gamma_6 - \frac{1}{2}\right)} \gamma_2^2 \quad (5.87)$$

Therefore, \tilde{v}_t converges to a ball of radius

$$r_t = \sqrt{\frac{\gamma_2^2}{\left(K_\lambda \gamma_6 - \frac{1}{2}\right)}} \quad (5.88)$$

Choosing K_λ large enough makes the ultimate bound of \tilde{v}_t less than ϵ_3 for any desired $\epsilon_3 > 0$. Letting $\epsilon_3 = \sqrt{\gamma_2^2 / (K_\lambda \gamma_6 - \frac{1}{2})}$, the set

$$\Lambda_2 = \left\{ \left(\lambda, \dot{\lambda}, \tilde{v}_t, v_n \right) \in \Lambda_3 : |\tilde{v}_t| \leq \epsilon_3 \right\} \quad (5.89)$$

is asymptotically stable relative to Λ_3 . On the set Λ_2 , the dynamics are described by subsystem . The function $f_3(\cdot)$ is uniformly bounded on Λ_2 , namely, there exists $\gamma_3 > 0$ such that $\|f_3(\cdot)\| \leq \gamma_3$ on Λ_2 . Employing the Lyapunov function candidate

$$V_2 = \frac{1}{2} v_n^2 \quad (5.90)$$

and using part (b) of Remark 5.1 yields

$$\dot{V}_2 \leq \frac{-c_n}{m} v_n^2 + \gamma_3 v_n \leq -\frac{c_n}{m} v_n^2 + \frac{\gamma}{2} v_n^2 + \frac{1}{2\gamma} \gamma_3^2, \quad (5.91)$$

where γ is some positive constant and we have used Young's inequality, $ab \leq (\gamma/2)a^2 + (1/2\gamma)b^2$. We conclude that there exists a sufficiently small positive constant β such that

$$\dot{V}_2 \leq -\beta V_2 + \frac{1}{2\gamma} \gamma_3^2. \quad (5.92)$$

Using the Comparison Lemma, we have, for all $t \geq 0$,

$$V_2(t) \leq \exp(-\beta t) V_2(0) + \frac{1}{2\gamma\beta} \gamma_3^2. \quad (5.93)$$

Therefore, v_n converges to a ball of radius

$$r_n = \sqrt{\frac{\gamma_3^2}{\gamma\beta}} \quad (5.94)$$

Letting $\epsilon_4 = \sqrt{\gamma_3^2/(\gamma\beta)}$, the set

$$\Lambda_1 = \left\{ (\lambda, \dot{\lambda}, \tilde{v}_t, v_n) \in \Lambda_2 : |v_n| \leq \epsilon_4 \right\} \quad (5.95)$$

is asymptotically stable relative to Λ_2 . This set is compact because $\lambda \in S^1$, which is a compact set and on Λ_1 , $|\tilde{v}_t| \leq \epsilon_3$, and $\dot{\lambda} = -K_\lambda \tilde{v}_t$. In the above analysis, $\Lambda_1 \subset \Lambda_2 \subset \Lambda_3$. Also, Λ_i is asymptotically stable relative to Λ_{i+1} for the closed-loop system for $i = 1, 2$. On the other hand, Λ_1 is a compact set. Using Proposition 4.1, we conclude that the set Λ_1 is asymptotically stable. ■

5.2.4 Path Following Control of Snake Robots

In this section we study the path following control of snake robots. Thus far we have developed a velocity controller that asymptotically stabilizes the **direction following manifold**

$$\begin{aligned} \Gamma_2 = \{ & (\theta, \dot{\theta}, p, \dot{p}, \lambda, \dot{\lambda}, \phi_o, \dot{\phi}_o) \in \Gamma_3 : \\ & \|(\tilde{\theta}_N, \dot{\tilde{\theta}}_N + k_N \tilde{\theta}_N)\| \leq \epsilon_1, V(\phi_o, \dot{\phi}_o) \leq \epsilon_2 \\ & |v_t - v_{\text{ref}}(p)| \leq \epsilon_3, |v_n| \leq \epsilon_4, \dot{\lambda} = -K_\lambda \tilde{v}_t \} \end{aligned} \quad (5.96)$$

We also define the **path following manifold** as follows

$$\Gamma_1 = \left\{ (\theta, \dot{\theta}, p, \dot{p}, \lambda, \dot{\lambda}, \phi_o, \dot{\phi}_o) \in \Gamma_2 : |h(p)| \leq \epsilon_5 \right\} \quad (5.97)$$

The following remark investigates the compactness of the path following manifold. According to Proposition 4.1, this compactness property is a necessary

condition for the reduction theory that we are using to prove the asymptotic stability of the path following manifold in this section.

Remark 5.13 *The set Γ_1 is compact. The reason is that the inequality $|h(p)| \leq \epsilon_5$ implies that p is bounded because $h(\cdot)$ is a continuous function. Since $\theta_{\text{ref}}(\cdot)$ and $v_{\text{ref}}(\cdot)$ are continuous functions, $\theta_{\text{ref}}(p)$ and $v_{\text{ref}}(p)$ are bounded. Therefore, θ_N and v_t are bounded. Since v_t and v_n are bounded, \dot{p} is bounded. Since θ_N and ϕ_o are bounded and $\theta = e\theta_N + H\Phi(\lambda) + Hb\phi_o$ on the set Γ_1 , θ is bounded. Since $\dot{\theta} = e\dot{\theta}_N + H\Phi'(\lambda)\dot{\lambda} + Hb\dot{\phi}_o$, and $\dot{\theta}_N$, $\dot{\lambda}$, and $\dot{\phi}_o$ are bounded, $\dot{\theta}$ is bounded.*

We let $\mu = [v_{\text{ref}} \cos(\theta_{\text{ref}}), v_{\text{ref}} \sin(\theta_{\text{ref}})]^T$. It can be shown that

$$\dot{p} = R_{\Delta_1} \mu + d(v_t, v_n, \theta_N) \quad (5.98)$$

where R_{Δ_1} is the rotation matrix with angle $\Delta_1 = \theta_N - \theta_{\text{ref}}(p)$ and $\|d(\cdot)\| \leq \epsilon_3 + \epsilon_4$. If we let $y = h(p)$, we want $y \rightarrow 0$ to meet specification (i) of the PFP. On the direction following manifold Γ_2 , we have

$$\dot{y} = dh_p \dot{p} = dh_p R_{\Delta_1} \mu + dh_p d(\cdot) \quad (5.99)$$

We propose to use the following control law

$$\mu = -\frac{dh_p^T}{\|dh_p\|^2} K_{\text{tran}} h(p) + \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} dh_p^T \frac{v}{\|dh_p\|} \quad (5.100)$$

where K_{tran} is a positive constant. We have the following proposition regarding the path following controller.

Proposition 5.5 *Consider system (5.99) where $|\Delta_1| < \epsilon_1$. For sufficiently small ϵ_1 the following property holds: for any $\epsilon_5 > 0$ there exists K_{tran} such that the set $\{|h(p)| \leq \epsilon_5\}$ is asymptotically stable for (5.99). Moreover, the velocity control specification, i.e., $\|\dot{p}\| = v$, is approximately met on γ : $|\|\dot{p}\| - v| \leq \epsilon_3 + \epsilon_4$.*

Proof: Inserting the control input (5.100) into (5.99), the closed-loop equation is obtained as follows

$$\dot{y} = -\frac{dh_p R_{\Delta_1} dh_p^T}{\|dh_p\|^2} K_{\text{tran}} y + dh_p R_{\Delta_1} \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} dh_p^T \frac{v}{\|dh_p\|} + dh_p d(\cdot) \quad (5.101)$$

Now, we consider the Lyapunov function candidate

$$V = \frac{1}{2} y^2 \quad (5.102)$$

We pick $c > 0$ and define $\Omega_c = \{|y| \leq c\}$. By assumption, on $\{p : h(p) = 0\}$ it holds that $dh_p \neq [0 \ 0]$. Therefore, there exists $c > 0$ such that $dh_p \neq 0$ for all $p \in \{p : |h(p)| \leq c\}$. Let $\Omega_c = \{p : |h(p)| \leq c\}$. We will now show that for sufficiently large K_{tran} , Ω_c is positively invariant. To this end, it is enough to show that there exists $K^* > 0$ such that for all $K_{\text{tran}} \geq K^*$, $\dot{V} \leq 0$ for all $p \in \partial\Omega_c$. On $\partial\Omega_c$, dh_p is bounded. Therefore, $|dh_p d(\cdot)| \leq K$. By continuity, for small enough ϵ_1 (note that ϵ_1 can be made arbitrarily small by Proposition 5.3), there exist $a_1, a_2 > 0$ such that

$$-\frac{dh_p R_{\Delta_1} dh_p^T}{\|dh_p\|^2} \leq -a_1, \quad (5.103)$$

and

$$\left| dh_p R_{\Delta_1} \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} dh_p^T \frac{v}{\|dh_p\|} \right| \leq a_2 \quad (5.104)$$

We have

$$\dot{V} = y\dot{y} \leq -K_{\text{tran}} a_1 y^2 + a_2 y + dh_p d(\cdot) y \leq -K_{\text{tran}} c^2 + a_2 |c| + K |c| \quad (5.105)$$

Thus, if $K_{\text{tran}} \geq \frac{a_2 + K}{|c|}$, we get $\dot{V} \leq 0$ on $\partial\Omega_c$. This means that Ω_c is positively invariant. On Ω_c , we have $|dh_p d(\cdot)| \leq K_c$ because Ω_c is compact. Thus, we get

$$\begin{aligned} \dot{V} &\leq -K_{\text{tran}} a_1 y^2 + (K_c + a_2) |y| \leq \\ &-K_{\text{tran}} y^2 + \frac{1}{2} y^2 + \frac{1}{2} (K_c + a_2)^2 \implies \\ \dot{V} &\leq -\left(K_{\text{tran}} - \frac{1}{2}\right) y^2 + \frac{1}{2} (K_c + a_2)^2 \end{aligned} \quad (5.106)$$

Therefore, by the Comparison Lemma, for $K_{\text{tran}} \geq \frac{1}{2}$, we have

$$V(t) \leq \exp\left(-\left(K_{\text{tran}} - \frac{1}{2}\right)t\right) V(0) + \frac{\frac{1}{2}(K_c + a_2)^2}{K_{\text{tran}} - \frac{1}{2}} \quad (5.107)$$

Therefore, y converges to a ball of radius

$$r_y = \sqrt{\frac{\frac{1}{2}(K_c + a_2)^2}{K_{\text{tran}} - \frac{1}{2}}} \quad (5.108)$$

Choosing K_{tran} large enough makes the ultimate bound of y less than ϵ_5 for any desired $\epsilon_5 > 0$. Therefore, the path γ is practically stable with domain of attraction containing Ω_c . On the path γ , $h(p) = 0$, and we have

$$\dot{p} = R_{\Delta_1} \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} dh_p^T \frac{v}{\|dh_p\|} + d \quad (5.109)$$

Consequently, we have that

$$v - \|d\| \leq \|\dot{p}\| \leq v + \|d\| \implies | \|\dot{p}\| - v | \leq \|d\| \leq \epsilon_3 + \epsilon_4 \quad (5.110)$$

Thus, we have approximate velocity control on γ . This completes the proof. ■

Main Result

For the snake robot model (5.1a)–(5.1b), we proposed the following control law

$$\begin{aligned} u = & (DM_\theta^{-1}D^T)^{-1} \{ DM_\theta^{-1}W_\theta\dot{\theta}^2 \\ & - lDM_\theta^{-1}SC_\theta^T f_R + \Phi''(\lambda)\dot{\lambda}^2 + \Phi'(\lambda)u_\lambda \\ & + bu_{\phi_o} - K_P [D\theta - \Phi(\lambda) - b\phi_o] \\ & - K_D [D\dot{\theta} - \Phi'(\lambda)\dot{\lambda} - b\dot{\phi}_o] \}, \end{aligned} \quad (5.111)$$

where ϕ_o , $\dot{\phi}_o$, λ , and $\dot{\lambda}$ are the states of the following dynamic compensators

$$\ddot{\lambda} = u_\lambda, \quad \ddot{\phi}_o = u_{\phi_o}, \quad (5.112)$$

and the control input u_{ϕ_o} is given by

$$u_{\phi_o} = \frac{1}{\Psi_3(\cdot)} \left\{ \frac{1}{\epsilon} \left(\dot{\tilde{\theta}}_N + k_N \tilde{\theta}_N \right) \right\} - k_1 \phi_o - k_2 \dot{\phi}_o, \quad (5.113)$$

where $\tilde{\theta}_N = \theta_N - \theta_{\text{ref}}(p)$. Also, the control input u_λ is given by

$$u_\lambda = -K_z \left(\dot{\lambda} + K_\lambda \tilde{v}_t \right) - K_\lambda \left[f_2(\cdot) + u_{\theta_N}^T \Psi_6(\cdot) \dot{\lambda} - (dv_{\text{ref}})_p \dot{p} \right], \quad (5.114)$$

where $K_\lambda > 0$ and $K_z > 0$ are constants. The reference signals $\theta_{\text{ref}}(p)$ and $v_{\text{ref}}(p)$ in (5.113) and (5.114) are defined by the following path following control law

$$\mu = -\frac{dh_p^T}{\|dh_p\|^2} K_{\text{tran}} h(p) + \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} dh_p^T \frac{v}{\|dh_p\|} \quad (5.115)$$

where K_{tran} is a positive constant. Note that

$$\mu(p) = [v_{\text{ref}}(p) \cos(\theta_{\text{ref}}(p)), v_{\text{ref}}(p) \sin(\theta_{\text{ref}}(p))] \quad (5.116)$$

We have the following theorem regarding the snake robot control system.

Theorem 5.1 [Main Result] *Consider the snake robot model (5.1a)–(5.1b) with feedback (5.111), (5.113), (5.114), and (5.115). Suppose that the ultimate bound on ϕ_o from Proposition 5.3 is small enough such that $u_{\theta_N}^T \Psi_6(\cdot)$ is bounded away from zero. For any $\epsilon_5 > 0$, there exist a sufficiently small ϵ in (5.113), a sufficiently large K_λ in (5.114) and K_{tran} in (5.115) such that the path following manifold Γ_1 in (5.97) is asymptotically stable and the velocity of the snake robot satisfies the asymptotic bound $\limsup \|\dot{p}\| - v \leq \epsilon_3 + \epsilon_4$.*

Proof: Consider the sets $\Gamma_1, \Gamma_2, \Gamma_3$ defined in (5.97), (5.96), (5.50) and note that $\Gamma_1 \subset \Gamma_2 \subset \Gamma_3$. Also, by Proposition 5.5, Γ_1 is asymptotically stable relative to Γ_2 and by Propositions 5.4 and 5.3, Γ_2 is asymptotically stable relative to Γ_3 for the closed-loop system. On the other hand, Γ_1 is a compact set (see Remark 5.13). Using Proposition 4.1, we conclude that the set Γ_1 is asymptotically stable.

5.3 Simulation Results

Simulation Results for the Direction Following Controller

In this part, we present the simulation results which illustrate the performance of the proposed direction following controller. We considered a snake robot with $N = 10$ links with $2l = 0.14$ m, $m = 1$ kg, and $J = 0.0016$ kg.m². The friction coefficients were $c_t = 0.1$ and $c_n = 1$. The parameters of the VHC were chosen to be $\alpha = 30\pi/180$ rad, and $\delta = 72\pi/180$ rad. We would like to regulate the velocity of the CM of the robot to $[-0.0354 \quad -0.0354]^T$, i.e., the reference head angle is taken to be $-\pi/4$ rad and $v_{ref} = 5$ cm/s. The controller parameters were chosen to be $K_p = 100, K_d = 10$ in (5.12), $\epsilon = 10^{-4}, k_N = 10, k_1 = 1, k_2 = 1$ in (5.16), and $k_\lambda = 1000, K_z = 1000$ in (5.34). Note that ϵ determines the ultimate bound on heading angle error. Also, k_N determines the rate of convergence of θ_N to θ_{ref} . The gains k_1 and k_2 have influence on the ultimate bound of ϕ_o . Finally, k_λ and K_z determine the rate of convergence and ultimate bound of \tilde{v}_t . The simulation results show that the snake robot follows the desired direction while the forward and normal velocities converge to small neighbourhoods of the desired values. Figure 5.3 depicts the snake robot at $t = 0, 30, 60$ seconds, respectively. Figure 5.4 depicts the dynamic variable ϕ_o . Figure 5.5 depicts the dynamic variable $\dot{\lambda}$. Figure 5.6 depicts the shape variable error. Figure 5.7 depicts the tangential and normal velocities. Finally, Figure 5.8 depicts the head angle of the snake robot. Note that the VHC error in Figure 5.6 and the head angle in Figure 5.8 converge faster to their steady state values because they are the first control specifications that we enforce.

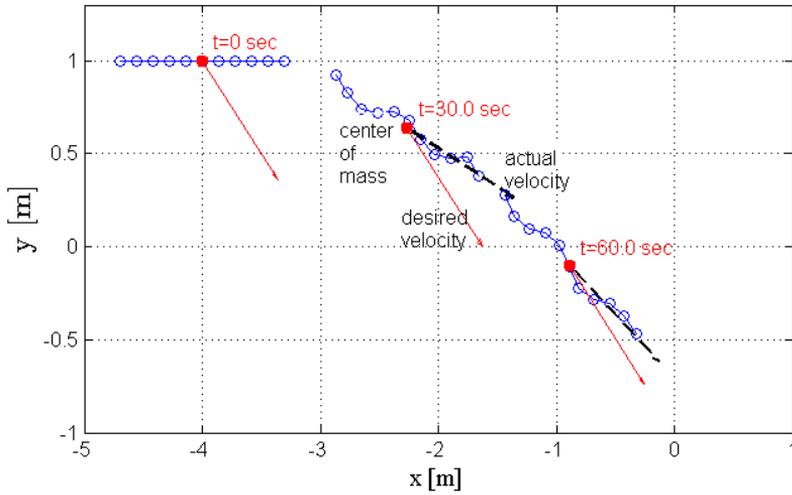


Figure 5.3: Plots of the snake robot with 10-links.

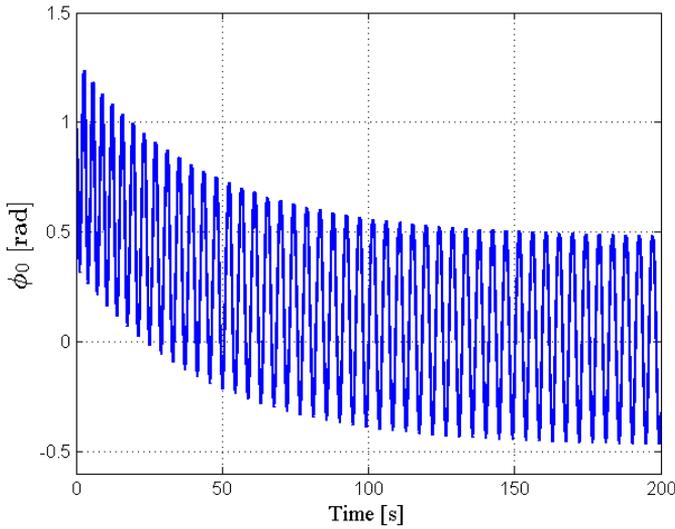


Figure 5.4: The dynamic variable ϕ_o which controls the head angle of the robot remains uniformly ultimately bounded.

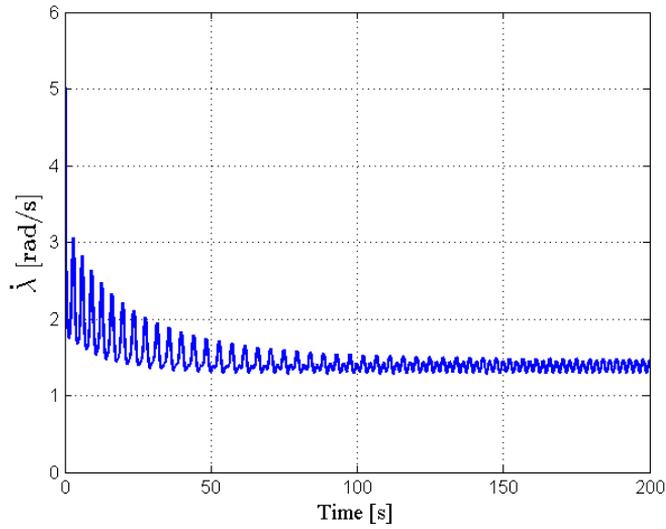


Figure 5.5: The dynamic variable $\dot{\lambda}$ remains uniformly ultimately bounded.

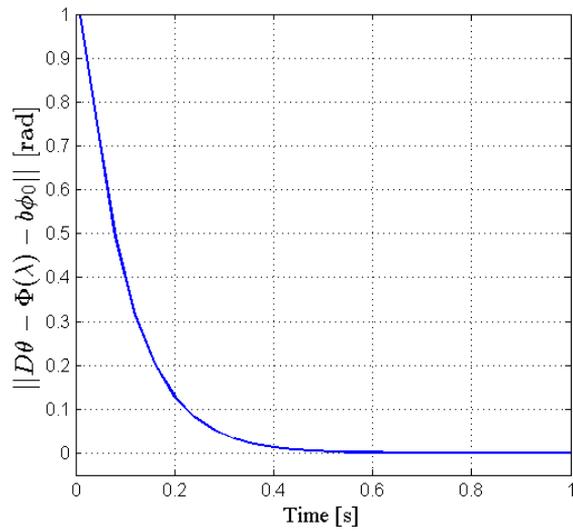


Figure 5.6: The lateral undulatory gait (5.5a)–(5.5b) is stabilized among the shape variables of the snake robot.

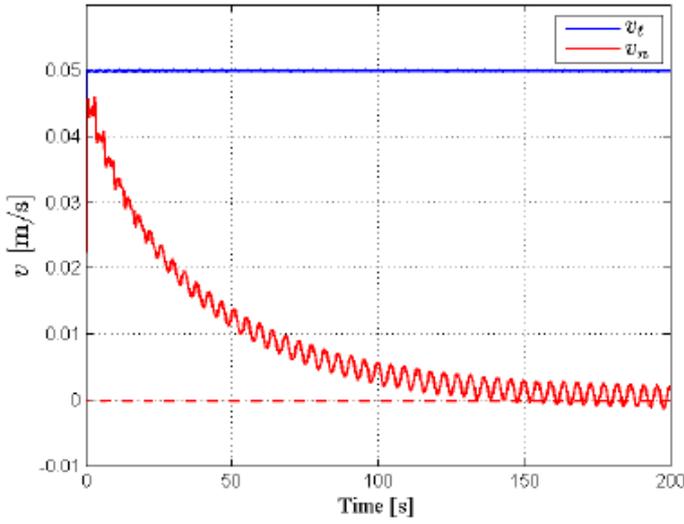


Figure 5.7: Tangential and normal velocities.

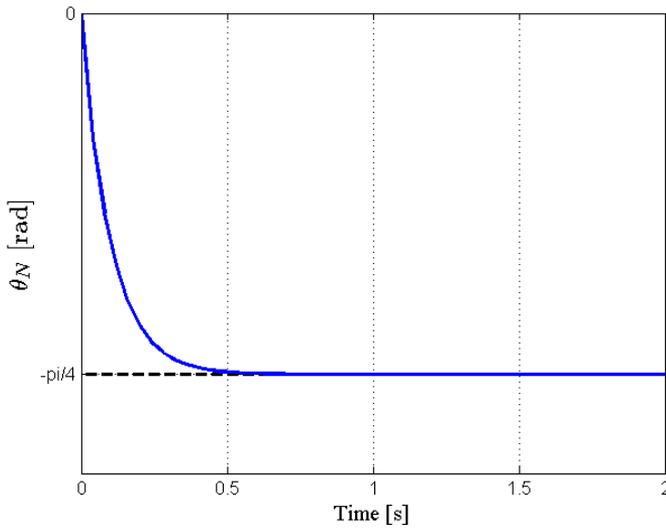


Figure 5.8: The head angle of the robot.

Simulation Results for the Maneuvering Controller

In this part, we present the simulation results which illustrate the performance of the proposed maneuvering controller. For the simulations, we considered a snake robot with $N = 10$ links with length $2l = 0.14$ m, mass $m = 1$ kg, and moment of inertia $J = 0.0016$ kg.m². The ground friction coefficients were $c_t = 0.5$ and $c_n = 3$. The parameters of the VHC (5.49a)–(5.49b) were chosen to be $\alpha = 30\pi/180$ rad, and $\delta = 72\pi/180$ rad. We would like to follow a circular path with radius 2m. The controller parameters were chosen to be $k_p = 10, k_d = 10$ in (5.111), $\epsilon = 10^{-1}, k_N = 10, k_1 = 1, k_2 = 1$ in (5.113), $K_z = 50$ and $K_\lambda = 50$ in (5.114), and $K_{\text{tran}} = 0.6, v = 0.05$ in (5.115). We ran the simulation for 600 seconds.

As it can be seen from (5.115), when the path following error $\|h(p)\|$ is large, the reference speed $\|\mu\| = v_{\text{ref}}$ will be large. Tracking such a large reference speed will require very fast oscillations of the snake robot and large control torques. In order to avoid such large initial oscillations and joint control torques, we set $v_{\text{ref}} = 0.05$ while the path following error $\|h(p)\|$ was greater than 0.3. For $\|h(p)\| < 0.3$, we let the reference speed v_{ref} to be determined by (5.115). Note that ϵ determines the ultimate bound on head angle error. Also, k_N determines the rate of convergence of θ_N to θ_{ref} . The gains k_1 and k_2 have influence on the ultimate bound of ϕ_o . The gains K_λ and K_z determine the rate of convergence and ultimate bound of \tilde{v}_t . Finally, K_{tran} controls the path following error. In order to show the performance of the proposed control scheme in the presence of angular position measurement noise, we subjected every i -th link angle θ_i to an additive noise by using Matlab function `randn()` which generates normally distributed pseudorandom numbers that can be considered as measurement noise for the joint angles.

The simulation results show that the snake robot follows the desired path in the presence of measurement noise, while the states of the compensators in (5.112) remain uniformly ultimately bounded. Figure 5.9 depicts the snake robot and the trajectory of the CM of the robot. Figure 5.10 depicts the path following error. Figure 5.11 shows the time evolution of the dynamic variable ϕ_o . Figure 5.12 depicts the time evolution of the dynamic variable $\dot{\lambda}$, and thus gives the frequency of the undulatory motion. Figure 5.13 depicts the shape variable error. Figure 5.14 depicts the actual and the reference tangential velocities. Figure 5.15 depicts the head angle tracking error. Finally, Figure 5.16 depicts the norm of the control torque vector, from which we can see that the control torques are within the physical limitations/saturation values of the existing snake robots at the NTNU snake robotics laboratory.

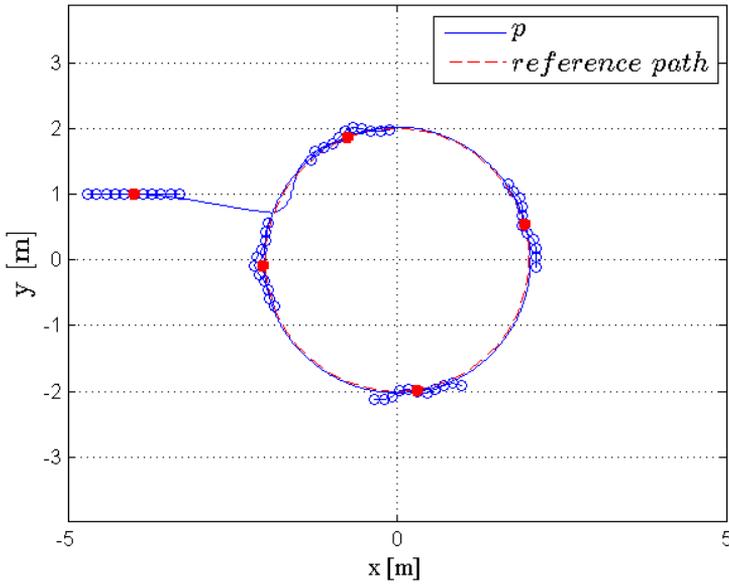


Figure 5.9: Plots of the snake robot with 10 links.

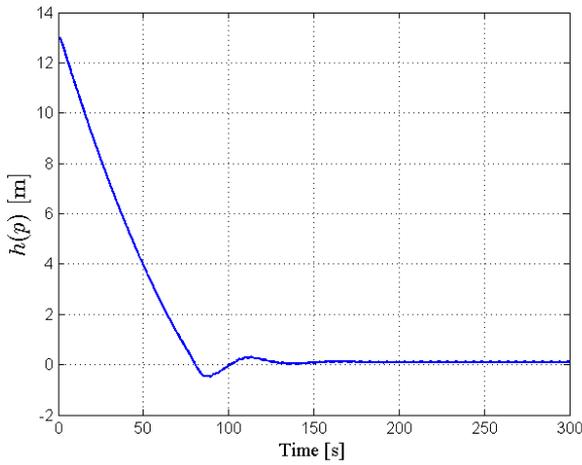


Figure 5.10: The path following error.

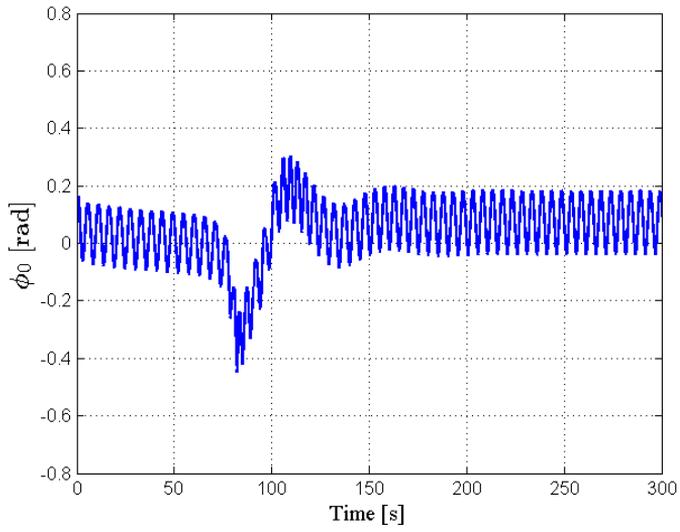


Figure 5.11: The dynamic variable ϕ_0 remains uniformly ultimately bounded.

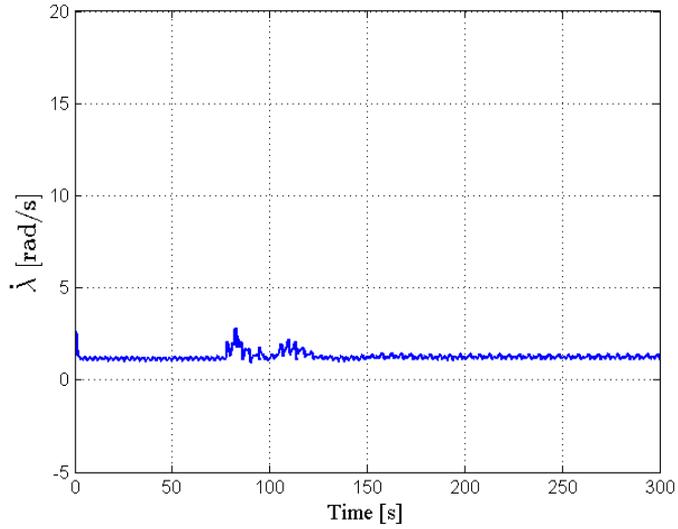


Figure 5.12: The dynamic variable $\dot{\lambda}$ remains uniformly ultimately bounded.

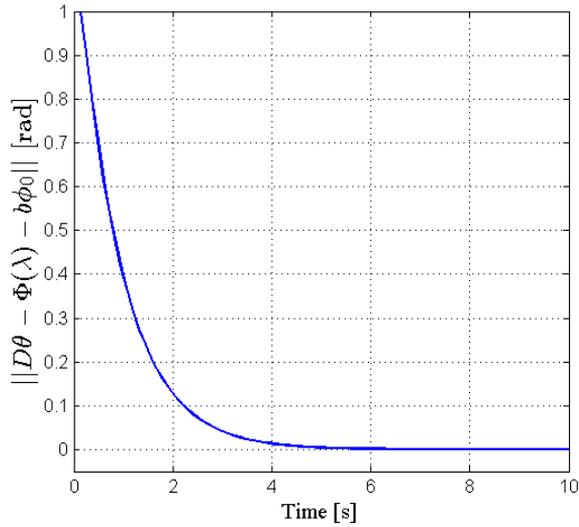


Figure 5.13: The lateral undulatory gait (5.49a)–(5.49b) is stabilized among the shape variables of the snake robot.

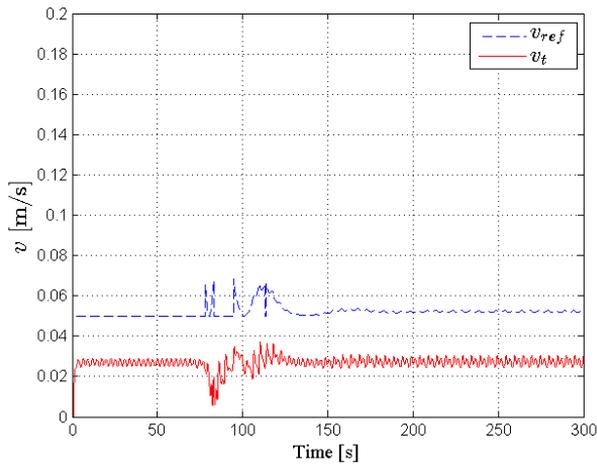


Figure 5.14: Reference and actual tangential velocities.

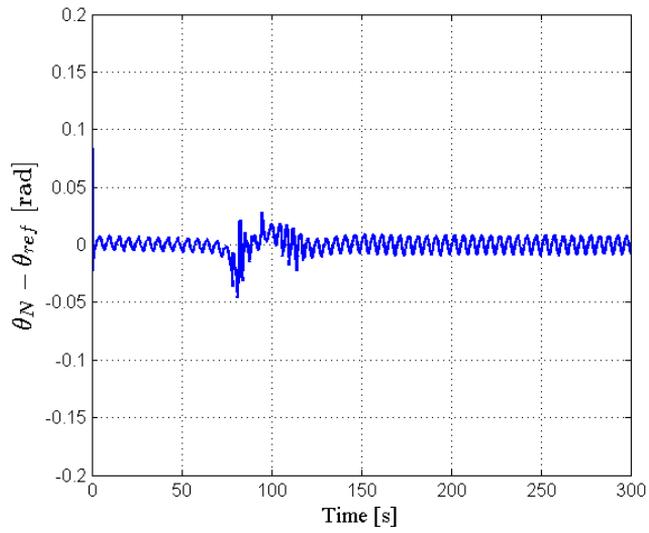


Figure 5.15: The head angle tracking error.

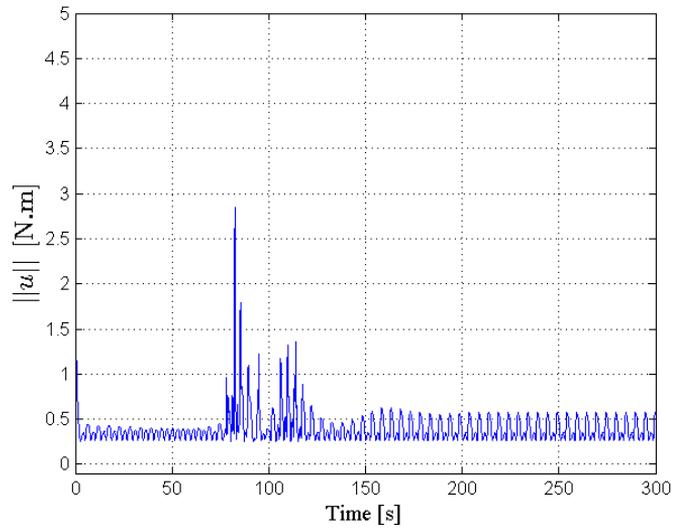


Figure 5.16: Norm of the control torque vector.

Chapter Summary

- We addressed the problems of direction following and maneuvering control for a planar snake robot. We defined $N - 1$ constraint functions for directly actuated shape variables of the robot. These constraint functions were dependent on the variations of the states of dynamic compensators which were used to control the head angle and the forward velocity of the robot on the constraint manifold.
- Extensive simulations results were presented which showed the performance of the proposed direction following and maneuvering control design approaches.

Conclusions and Future Challenges

Motivated by the lack of analytical feedback control approaches relying on formal stability proofs, in this thesis, we proposed various model-based feedback control strategies for the snake robot. In particular, the main distinguished characteristic of this work with respect to the previous literature on snake robots, is the derivation of control laws based on the kinematic and dynamic models along with presenting formal stability proofs for the closed-loop systems. Due to complexities in the dynamic behaviour of snake robots, and due to the underactuation which is characterized by the lack of direct and independent control inputs for some degrees of freedom of the system, model-based control design for these robots is a challenging task. We hope that the analytical approaches to snake robot control which were carried throughout this thesis, can contribute to further developments of control theory of snake robots.

Our approach to locomotion control of snake robots was primarily based on some biological observations, along with mathematical descriptions of biological snake locomotion. In particular, in this thesis we used the most common gait pattern (i.e. fixed periodic body motions) used by biological snakes, known as lateral undulation. In this type of gait pattern, a wave-like motion travels backward along the body of snakes, from head to tail. As a result of this motion, the snake body traces out a periodic curve on the plane, which Hirose [1] mathematically represented as a serpenoid. Hirose [1] along with some other researchers (e.g. [105]) figured out that the serpenoid curve can be well-approximated by imposing the sinusoidal reference trajectory for the i -th joint angle of the snake robot as

$$\phi_{\text{ref},i}(t) = \alpha \sin(\omega t + (i - 1)\delta) + \phi_o \quad (6.1)$$

where α denotes the amplitude of the sinusoid, ω denotes the frequency of the joint oscillations, δ denotes the phase shift between two consecutive joints, and

ϕ_o is a joint offset used to control the direction of locomotion.

The dynamic model of the snake robot includes strong nonlinear terms which make the control design challenging. Furthermore, the configuration of a snake robot can be completely described by the shape variables which define the internal configuration of the robot, along with the position variables which define the orientation and planar position of the robot with respect to some inertial frame. The shape variables are directly and independently actuated by motors located in the joints of the robot. However, there is no direct and independent control input for the position variables of a snake robot. In addition, the angle of the N -th link, or equivalently the orientation of the snake robot, has no direct control input. This makes the robot underactuated. The motion of the fully-actuated shape variables can be shaped arbitrarily, as long as the dynamical constraints of the system are not violated. In turn, the position and orientation should be indirectly controlled to given references. In particular, the motion of the joints of the robot can only affect the position of the CM of the robot through the anisotropic friction forces. This means that in the absence of anisotropic friction forces, the joints of the snake robot cannot accelerate the CM and the robot is not controllable. The above challenges are among the main reasons that make locomotion control of snake robots an interesting and open area of research.

Throughout this thesis, we have dealt with these challenges by employing various techniques from nonlinear control theory. To this end, we utilize the dynamic model of the snake robot in order to derive a dynamic feedback control law which controls the motion of the joints of the snake robot to the following reference joint angle

$$\phi_{\text{ref},i}(\lambda, \phi_o) = \alpha \sin(\lambda + (i - 1)\delta) + \phi_o \quad (6.2)$$

The major difference between the reference joint angles (6.1) and (6.2) is that in the latter one rather than using a constant frequency ω for the periodic body oscillations, we put a feedback on $\dot{\lambda}$ which employs this frequency in order to control the forward velocity of the robot to a given reference forward velocity.

In particular, the relation (6.2) is a virtual holonomic constraint which we enforce for the joint angles of the robot through the actions of an input-output linearizing feedback controller. Associated with the virtual constraints defined by (6.2), is a constraint manifold which we asymptotically stabilize for the system. Subsequently, we reduce the dynamics of the system to the globally invariant constraint manifold, where we employ the offset term ϕ_o along with the frequency of the gait pattern $\dot{\lambda}$ as two additional control inputs which we use to control the orientation and velocity of the robot, respectively. In particular, we employ the following dynamic compensators

$$u_{\phi_o} = \ddot{\phi}_o, \quad u_{\lambda} = \ddot{\lambda} \quad (6.3)$$

and based on the dynamic model of the snake robot we derive feedback laws which control the orientation and velocity to given references. In the last stage of our hierarchical control approach for the snake robot locomotion, we use the reference orientation and velocity as additional control terms such that the robot converges to and follows any given continuously differentiable path. Moreover, by using Lyapunov functions, we show that all the solutions of the controlled system including the states of the dynamic compensators (6.3) remain uniformly ultimately bounded.

Along with presenting rigorous mathematical proofs, we also have performed extensive simulation studies and real-time experiments in the snake robots lab at NTNU, which show the effectiveness of the proposed control strategies.

6.1 Summary of the Chapters

In this part, we present the conclusive remarks on each chapter of the thesis.

Summary of Chapter 1

In Chapter 1, we stated the main objective of the thesis as bridging the gap between control theory and practice for snake robots. This objective was followed throughout the remaining chapters of this thesis. Moreover, we briefly discussed the locomotion mechanism in biological snakes, and we investigated various challenges arising in locomotion control of snake robots. Furthermore, we presented a literature review for snake robot control. In addition, some preliminary mathematical discussions on the constraints of the snake robot were presented in this chapter.

Summary of Chapter 2

In Chapter 2, we presented several kinematic and dynamic models for snake robot locomotion. Since the major contribution of this thesis is to present model-based control approaches for snake robots, then deriving accurate mathematical descriptions of the behaviour of the system is an important task. In this chapter, we presented three different models of a snake robot without nonholonomic velocity constraints. The first model was derived in a Lagrangian framework, and the effects of anisotropic friction forces were integrated into the dynamic model by using the Jacobian matrices of the CM of the links. The second model that we presented in this chapter was derived based on the techniques of differential geometry. Furthermore, we introduced nonlinear terms arising due to parametric modelling uncertainties in the dynamic equations of motion, that modelled the

effects of varying friction properties on different surfaces. This model was later used for robust locomotion control of the snake robot. The third model that we presented in Chapter 2 was a simplified model of the snake robot locomotion which was previously derived in [11]. In this simplified model, the motion of the links were approximated as translational motion rather than rotational joint motion. Since in general translational motion is less complex than rotational motion, the resulting model was simpler and more amenable to model-based control design and analysis.

Summary of Chapter 3

In Chapter 3, we considered model-based body shape and orientation control for locomotion of snake robots. First, using the method of virtual holonomic constraints, we defined reference joint angle and head angle trajectories for the snake robot. In particular, for an N -link snake robot we defined N constraint functions in the following form

$$\Phi_i = \alpha \sin(\lambda + (i - 1)\delta) + \phi_o, \quad i = 1, \dots, N - 1 \quad (6.4)$$

$$\Phi_N = -\text{atan2}\left(\frac{p_y}{\Delta}\right) \quad (6.5)$$

The first $N - 1$ constraint functions were inspired by the reference joint trajectories (6.1) which induce lateral undulatory locomotion on snake robots. These virtual constraints were stabilized for the joint angles of the snake robot using the control torques provided by motors mounted on each joint. The last constraint function which was stabilized for the head link was inspired by the Line-of-Sight (LOS) path following guidance law, where p_y denotes the shortest normal distance from the CM of the robot to the desired path, and Δ denotes the look-ahead-distance. This constraint function was stabilized using a dynamic compensator which was defined through $\ddot{\phi}_o$. Using numerical simulations and real-time experiments we showed that stabilizing the constraint functions (6.4)–(6.5) induces path following on the snake robot.

In the second part of Chapter 3 we proposed a robust generalization for the controller derived in the first part. In particular, we assumed that the robot moves on different surfaces with different friction properties. This makes the robot subject to parametric modelling uncertainty. Furthermore, we used sliding mode techniques in order to derive robust body shape and orientation control laws which could make the robot converge to and follow the desired path in the presence of these modelling uncertainties. The approach for deriving this robust control laws were similar to the previous case. In particular we defined the i -th joint angle control law ϑ_i in the form

$$\vartheta_i = \vartheta_{i,\text{nom}} + \vartheta_{i,\text{add}} \quad (6.6)$$

where the nominal part $\vartheta_{i,\text{nom}}$ was defined in an input-output feedback linearizing form, and the additional part $\vartheta_{i,\text{add}}$ was defined in a discontinuous form which was used to deal with the model uncertainties. Through the robust control law (6.6), we stabilized the solutions of the joint angle dynamics to an appropriately defined sliding manifold. The same approach applies to the dynamic compensator $\ddot{\phi}_o$ which controls the head angle of the robot to an sliding manifold in the presence of modelling uncertainties.

Both of the foregoing body shape and orientation control approaches share a common theoretical gap which was bridged in the third part of this chapter. In particular, in the third part of Chapter 3 we used an input-output stability analysis in order to prove the boundedness of the solutions of the dynamic compensator $\ddot{\phi}_o$ which controls the orientation of the robot. Furthermore, we presented a Lyapunov based stability proof which showed the practical stability of the origin of the orientation error dynamics.

In the last part of this chapter, we presented simulation results along with the results of real-time experiments on mechanical snakes which showed the effectiveness of the proposed guidance-based path following control strategies in this chapter.

Summary of Chapter 4

In Chapter 4, we employed the simplified model of the snake robot locomotion, in order to derive model-based direction following and maneuvering control laws for the robot. In particular, we introduced (6.2) as the reference joint angle for the snake robot. The direction following control objective was defined as regulating the orientation and forward velocity of the robot to constant references, while guaranteeing the boundedness of the solutions of the controlled system. To achieve these objectives, we followed a hierarchical control design approach given in the following steps:

1. In the first step, we used the control torques in the joints of the robot to stabilize the solutions of the joint angle dynamics to the constraint manifold. This stabilizes a lateral undulatory gait pattern among the fully-actuated shape variables of the robot. Consequently, this induces a forward motion based on the gait pattern lateral undulation for the robot.
2. In the second step, we evaluated the dynamics of the system on the invariant constraint manifold, where we used λ and ϕ_o as two additional control terms, which were employed to control the velocity and orientation of the robot, respectively.

In the second part of this chapter, we tackled the maneuvering control problem in which the control objective is to stabilize a desired path for the CM of the snake robot and to control the robot along the path to a desired dynamic profile. In order to solve the maneuvering control problem we used a similar hierarchical control approach as in the direction following problem. However, there were two major differences between these approaches. The first difference was that we used the dynamic compensator $\ddot{\phi}_o$ to control the orientation of the robot to a time-varying reference orientation defined by a path following guidance law, rather than a constant orientation angle. This guidance law allowed the robot to converge to and follow a planar path. The second difference was that we derived a more complex dynamic compensator $\ddot{\phi}_o$ which controlled the position of the robot along with its forward velocity as in the direction following control design.

In the third part of this chapter, we considered the maneuvering control along general curved paths.

In the last part of Chapter 4, we presented the simulation results for the controllers derived in this chapter which showed the performance of the proposed control approaches.

Summary of Chapter 5

In Chapter 5, we developed model-based feedback control approaches for the snake robot based on the complex model of the snake robot dynamics. In particular, we solved the direction following and maneuvering control problems for the robot using the method of virtual holonomic constraints. The main idea for locomotion control of snake robot in this chapter is same as the approach given in Chapter 4; we first stabilize the motion of the joints to the constraint functions similar to (6.2) which induce lateral undulation motion on the robot, and then we control the constrained dynamics of the system by using the two dynamic compensators similar to (6.3).

In order to solve the direction following problem, we first stabilize a constraint manifold which is associated with the virtual holonomic constraints (6.2). Then we use the constrained dynamics of the system in order to control the dynamics of the underactuated degrees of freedom of the robot on the invariant constraint manifold. In particular, we design a high-gain feedback law for $\ddot{\phi}_o$ which regulates the head angle to a constant reference, and we use backstepping in order to design a feedback control law for $\ddot{\lambda}$ which regulates the tangential velocity of the robot to a constant reference $\dot{\lambda}$ through regulating $\dot{\lambda}$ to a required frequency of the body oscillations which provides this velocity for the robot.

For solving the maneuvering control problem in Chapter 5, we used similar design steps, however, rather than regulating the tangential velocity and head angle to constant references, we controlled these variables to time-varying references. In particular, we used these references as additional control terms which were defined such that the convergence of the path following error to an arbitrarily small neighbourhood of the origin was guaranteed.

6.2 Future Challenges in Snake Robot Control

In this section, we investigate the future challenges in motion control of snake robots. Throughout this thesis, we tried to contribute to some of these challenges by proposing model-based feedback control approaches for these complex robotic structures. However, complexity and variety of snake robot locomotion implies that there still remain many interesting open challenges for these robots that can be topics of future work. In the following, we list some of these open challenges:

Experimental Verification of the Presented Control Strategies: Throughout this thesis, we presented various feedback control approaches which were derived based on different dynamic models for the snake robot. In particular, we validated all the presented approaches through extensive numerical simulations which were performed based on realistic parameters for the snake robot. However, for complex robotic structures such as snake robots, it is very important to validate theories through experiments. We have already presented experimental verification for the body shape and orientation controller that we derived in Chapter 3. Experimental verification of the rest of the proposed control strategies in this thesis, remains as a topic of future work.

From 2D to 3D: All the methods that we presented in this thesis considered a snake robot moving in 2-dimensional Euclidean space. However, many real-time applications of snake robots need motion in 3-dimensional space. Consequently, it will be very interesting, and useful, to generalize the proposed strategies to the 3D Euclidean space. A rough explanation of the idea of generalizing these approaches to 3D space can be given as follows. In order to build a snake robot which is able to move in 3D space, we need to build two set of modules for the robots. In one set of these modules, the links oscillate in the $x - y$ plane, and in the other set, the links oscillate in the $x - z$ plane. Suppose that we stabilize the relations (6.2) for both these sets. Then we can use the corresponding offset term ϕ_o and the frequency λ of each of these sets in order to stabilize a

given orientation and velocity for the robot.

Motion in Unstructured Environments: In this thesis, we considered snake robots which move on a horizontal and flat surface. However, the long term goal for the snake robotic community is to use the benefits of these interesting robotic structures for industrial, medical, search and rescue, and other relevant real-time applications. In the majority of these applications, snake robots are subject to unstructured environments which are characterized by non-flat and cluttered surfaces. Consequently, it will be very interesting to generalize the proposed control strategies to unstructured environments, for which the equations of motion of the robot should be presented as a hybrid dynamical system. This is due to the presence of impacts with the obstacles in the workspace of the robot which make the states of the system discontinuous. However, with a careful analysis of the stability of the hybrid controlled system, it is possible to generalize the given approaches to hybrid dynamic models of the snake robot.

Swimming Snake Robots: Swimming snake robots are a class of snake robots which move inside a fluid, see e.g. [106]–[110]. These robots share many similarities, and few differences with on land snake robots. The similarities lie in a common physical structure, identical motion patterns, and analogous underactuation in the orientation and planar position of the robot. However, since the robot moves inside a fluid, then there are differences in environmental forces acting on the robot, e.g. fluid drag rather than friction forces. We believe that the similarities in the physical structure and also the motion of these robots with on land snake robots, makes the application of the proposed control strategies for these robots promising. In particular, since in the proposed control strategies in this thesis, we cancelled the effects of the environmental forces through the actions of feedback controllers, then if the hydrodynamic coefficients for swimming snake robots can be precisely determined, then it should be possible to extend our results to swimming snake robots which are characterized by similar physical structure and different external forces acting on them.

Bibliography

- [1] S. Hirose, "Biologically Inspired Robots: Snake-Like Locomotors and Manipulators". Oxford University Press, 1993.
- [2] D. Rollinson, "Control and Design of Snake Robots", PhD dissertation, Carnegie Mellon University, Pittsburgh, PA, 2014.
- [3] Z. Wang, and E. Appleton, "The concept and research of a pipe crawling rescue robot". *Advanced Robotics*, 17.4, pp:339-358, 2003.
- [4] S. A. Fjerdingen, P. Liljebäck, and A. A. Transeth, "A snake-like robot for internal inspection of complex pipe structures (PIKo)". in Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems, St. Louis, MO, USA, 2009.
- [5] D. Rollinson, and Howie Choset, "Pipe Network Locomotion with a Snake Robot", *Journal of Field Robotics*, doi: 10.1002/rob.21549, 2014.
- [6] T. Ota, A. Degani, D. Schwartzman, B. Zubiato, J. McGarvey, H. Choset, and M. Zenati, "A Highly Articulated Robotic Surgical System for Minimally Invasive Surgery", *The Annals of Thoracic Surgery*, Vol. 87, No. 4, pp:1253-1256, April, 2009.
- [7] P. Dupont, A. Gosline, N. Vasilyev, J. Lock, E. Butler, C. Folk, A. Cohen, R. Chen, G. Schmitz, H. Ren, and P. del Nido, "Concentric Tube Robots for Minimally Invasive Surgery", Hamlyn Symposium on Medical Robotics, 7(8), 2012.
- [8] K. Xu, and N. Simaan, "Actuation compensation for flexible surgical snake-like robots with redundant remote actuation" - IEEE Int. Conf. on Robotics and Automation, 2006.
- [9] Y. Zhou, H. Ren, M. Q.-H. Meng, Z. T. H. Tse, and H. Yu, "Robotics in Natural Orifice Transluminal Endoscopic Surgery *Journal of Mechanics in Medicine and Biology*, 13, 2013.
- [10] D. Berenson, J. Kuffner, and H. Choset, "An Optimization Approach to Planning for Mobile Manipulation", IEEE Int. Conf. on Robotics and Automation, Pasadena, CA, USA, 2008.

- [11] P. Liljebäck, K. Y. Pettersen, Ø. Stavdahl, and J. T. Gravdahl, "Snake Robots - Modelling, Mechatronics, and Control", *Advances in Industrial Control*, Springer, 2013.
- [12] C. Mattison, "The Encyclopaedia of Snakes", Cassell Paperbacks, London, 2002.
- [13] J. Gray, "The mechanism of locomotion in snakes", *Journal of Experimental Biology*, 23(2), pp:101–120, 1946.
- [14] A. De Luca, and G. Oriolo, "Modelling and control of nonholonomic mechanical systems", pp:277–342. Springer, Vienna, 1995.
- [15] K. M. Lynch, "Underactuated Robots", *Encyclopaedia of Systems and Control*, Springer-Verlag, London, 2014.
- [16] R. Olfati-Saber, "Nonlinear Control of Underactuated Mechanical. Systems with Application to Robotics and Aerospace Vehicles", PhD dissertation, Massachusetts Institute of Technology, 2001.
- [17] M. Mori, and S. Hirose, "Locomotion of 3d snake-like robots-shifting and rolling control of active cord mechanism acm-r3. *Journal of Robotics and Mechatronics*, 18(5), 521, 2006.
- [18] H. Yamada, S. Chigisaki, M. Mori, K. Takita, K. Ogami, and S. Hirose, "Development of amphibious snake-like robot ACM-R5", in Proc. ISR2005, 2005.
- [19] F. Matsuno, and H. Sato, "Trajectory tracking control of snake robots based on dynamic model", in Proc. IEEE Int. Conf. on Robotics and Automation. pp:3029–3034, 2005.
- [20] H. Date, Y. Hoshi, and M. Sampei, "Locomotion control of a snake-like robot based on dynamic manipulability", in Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems, Takamatsu, Japan, 2000.
- [21] M. Tanaka, and F. Matsuno, "Control of 3-dimensional snake robots by using redundancy", in Proc. IEEE Int. Conf. Robotics and Automation, pp:1156-1161, Pasadena, CA, 2008.
- [22] S. Kelly, and R. M. Murray, "Geometric phases and robotic locomotion", *Journal of Robotic Systems*, 12(6), pp:417–431, 1995.
- [23] J. P. Ostrowski, "The mechanics and control of undulatory robotic locomotion", PhD Dissertation, California Institute of Technology, 1996.

- [24] J. P. Ostrowski, and J. Burdick, "The geometric mechanics of undulatory robotic locomotion", *International Journal of Robotics Research*, 17(7), pp:683–701, 1998.
- [25] S. Ma S, Y. Ohmameuda, K. Inoue, and B. Li, "Control of a 3-dimensional snake-like robot", in Proc. IEEE Int. Conf. on Robotics and Automation, pp:2067-2072, Taipei, Taiwan, 2003.
- [26] M. Tanaka, and F. Matsuno, "A study on sinus-lifting motion of a snake robot with switching constraints", in Proc. IEEE Int. Conf. on Robotics and Automation. pp:2270-2275, 2009.
- [27] P. Prautsch, T. Mita, and T. Iwasaki, "Analysis and control of a gait of snake robot". *Trans. IEE J. Ind. Appl. Soc.*, pp:372–381, 2000.
- [28] D. Hu, J. Nirody, T. Scott, and M. Shelley, "The mechanics of slithering locomotion", in Proc. of the National Academy of Sciences USA, 106, pp:10081–10085, 2009.
- [29] H. Date, and Y. Takita, "Control of 3D snake-like locomotive mechanism based on continuum modeling", in Proc. International Design Engineering Technical Conferences, No. DETC2005-85130, 2005.
- [30] H. Date, Y. Hoshi, and M. Sampei, "Locomotion control of a snake-like robot based on dynamic manipulability", in Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems, 2000.
- [31] H. Date, Y. Hoshi, M. Sampei, and N. Shigeki, "Locomotion control of a snake robot with constraint force attenuation", in Proc. American Control Conference, pp:113–118, 2001.
- [32] H. Date, M. Sampei, and S. Nakaura, "Control of a snake robot in consideration of constraint force", in Proc. IEEE Int. Conf. on Control Applications, pp:966–971, 2001.
- [33] F. Matsuno, and K. Mogi, "Redundancy controllable system and control of snake robots based on kinematic model", in Proc. IEEE Int. Conf. Decision and Control, vol. 5, pp:4791–4796, 2000.
- [34] M. Ishikawa, K. Owaki, M. Shinagawa, and T. Sugie, "Control of snake-like robot based on nonlinear controllability analysis", in Proc. IEEE Int. Conf. Control Applications, pp:1134–1139, 2010.

- [35] M. Ishikawa, "Iterative feedback control of snake-like robot based on principal fiber bundle modeling", *International Journal of Advanced Mechatronic Systems*, 1(3), pp:175–182, 2009.
- [36] X. Wu, and S. Ma, "CPG-based control of serpentine locomotion of a snake-like robot", *Mechatronics*, Vol. 20, Issue 2, pp: 326–334, 2010.
- [37] P. Liljebäck, K. Y. Pettersen, Ø. Stavdahl, J. T. Gravdahl, "A review on modelling, implementation, and control of snake robots", *Robotics and Autonomous Systems*, vol.60, no.1, pp:29-40, 2012.
- [38] P. Liljebäck, Ø. Stavdahl, K. Y. Pettersen, and J. T. Gravdahl, "Mamba – A waterproof snake robot with tactile sensing", *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp:294–301, 2014.
- [39] S. Ma, Y. Ohmameuda, and K. Inoue, "Dynamic analysis of 3-dimensional snake robots", in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pp:767-772, 2004.
- [40] R. L. Hatton, and H. Choset, "Generating gaits for snake robots: annealed chain fitting and keyframe wave extraction." *Autonomous Robots*, 28.3, pp: 271–281, 2010.
- [41] K. McIsaac, and J. Ostrowski, "Motion planning for anguilliform locomotion", *IEEE Transactions on Robotics and Automation*, (19):637–652, 2003.
- [42] G. Hicks, and K. Ito, "A method for determination of optimal gaits with application to a snake-like serial-link structure", *IEEE Transactions on Automatic Control*, (50):1291–1306, 2005.
- [43] S. Ma, "Analysis of creeping locomotion of a snake-like robot", *Advanced Robotics*, 15(2):205–224, 2001.
- [44] 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*, 20:111-126, 2012.
- [45] P. Liljebäck, K. Y. Pettersen, Ø. Stavdahl, and J. T. Gravdahl, "Controllability and Stability Analysis of Planar Snake Robot Locomotion", *IEEE Transactions on Automatic Control*, 56.6:1365-1380, 2013.
- [46] 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*, Vol. 24, No. 1, 2008.

- [47] A. A. Transeth, S. A. Fjerdings, P. Liljebäck, "Snake robot obstacle-aided locomotion on inclined and vertical planes: Modeling, control strategies and simulation", 2013 IEEE Int. Conf. on Mechatronics (ICM), pp:321–328, 2013.
- [48] P. Liljebäck, K. Y. Pettersen, Ø. Stavdahl, and J. T. Gravdahl, "Hybrid modelling and control of obstacle-aided snake robot locomotion", *IEEE Transactions on Robotics*, vol. 26, no. 5, pp:781–799, 2010.
- [49] P. Liljebäck, K. Y. Pettersen, Ø. Stavdahl, and J. T. Gravdahl, "Experimental Investigation of Obstacle-Aided Locomotion With a Snake Robot", *IEEE Transactions on Robotics*, vol. 27, no. 4, pp:792–800, 2011.
- [50] R. Goebel, R. G. Sanfelice, and A. R. Teel, "Hybrid dynamical systems", *IEEE Control Systems Magazine*, 29(2), pp:28–93, 2009.
- [51] Z. Y. Bayraktaroglu, "Snake-like locomotion: experimentations with a biologically inspired wheel-less snake robot", *Mechanism and Machine Theory*, 44(3), pp:591–602, 2008.
- [52] D. Rollinson, K. V. Alwala, N. Zevallos, and H. Choset, "Torque control strategies for snake robots", IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp:1093–1099, 2014.
- [53] P. Liljebäck, Ø. Stavdahl, and A. Beitnes, "SnakeFighter - Development of a water hydraulic fire fighting snake robot", in Proc. IEEE Int. Conf. on Control, Automation, Robotics, and Vision (ICARCV), Singapore, 2006.
- [54] H. Kimura and S. Hirose, "Development of Genbu: Active wheel passive joint articulated mobile robot", in Proc. IEEE Int. Conf. Intelligent Robots and System, vol.1, pp:823-828, 2002.
- [55] Z. Wang, and E. Appleton, "The concept and research of a pipe crawling rescue robot", *Advanced Robotics*, vol. 17, No. 4, pp:339–358, 2003.
- [56] D. Palmer, S. Cobos-Guzman, and D. Axinte, "Real-time method for tip following navigation of continuum snake arm robots", *Robotics and Autonomous Systems* vol. 62(10), pp:1478–1485, 2014.
- [57] 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*, vol. 19, no. 3, pp:221–248, 2005.

- [58] Jiuguang Wang, <http://162.13.14.158/uknews/uknews-uknews/30450/robot-snakes-attack-humanoid-footballer-talking-shoes-robot-news/>
- [59] 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. on Intelligent Robots and Systems, San Francisco, CA, USA, 2011.
- [60] M. A. Zenati, T. Yokota, D. Schwartzman, T. Ota, B. Zubiante, C. Wright, C. Nikou, and H. Choset "Model-Guided Single Port Epicardial Interventions Using a Highly Articulated Robotic System ", in Proc. of the Minimally Invasive Robotic Association Fifth International Congress, San Diego, CA, January, 2010.
- [61] M. A. Zenati, T. Yokota, D. Schwartzman, T. Ota, B. Zubiante, C. Wright, C. Nikou, and H. Choset. "Model-Guided Epicardial Interventions using a Novel Snake Robot (Cardio-Arm) through a Subxiphoid Port Access", in Proc. The 14th IASTED Int. Conf. on Robotics Applications, Cambridge, MA, November, 2009.
- [62] T. Yokota, T. Ota, D. Schwartzman, C. Nikou, B. Jaramaz, B. Zubiante, H. Choset, and M. A. Zenati. "Model-Guided Single Port Interventions using a Novel Highly Articulated Mechatronic System", in Proc. the BMES (Biomedical Engineering Society) 2009 Annual Fall Scientific Meeting, Pittsburgh, PA, October, 2009.
- [63] C. Riviere, N. Patronik , and M. Zenati, "Prototype epicardial crawling device for intrapericardial intervention on the beating heart", Heart Surgery Forum, Vol. 7, no. 6, pp. 639-643, 2004.
- [64] T. Ota, A. Degani, D. Schwartzman, B. Zubiante, J. McGarvey, H. Choset, and M. Zenati, "A Highly Articulated Robotic Surgical System for Minimally Invasive Surgery", *The Annals of Thoracic Surgery*, vol. 87, no. 4, pp:1253–1256, 2009.
- [65] A. Degani, H. Choset. M. A. Zenati, T. Ota and B. Zubiante "Highly Articulated Robotic Probe for Minimally Invasive Surgery", in Proc. the Int. Conf. of the IEEE Engineering in Medicine and Biology Society, Vancouver, British Columbia, Canada, August, 2008.
- [66] A. Rangaprasad, M. Travers and H. Choset, "Using Lie Algebra to estimate the shape of medical snake robots", in Proc. 27th IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), Chicago, USA, September 2014.

- [67] A. Wolf, H. Ben Brown Jr, R. Casciola, A. Costa, M. Schwerin, E. Shammass and H. Choset, "A Mobile Hyper Redundant Mechanism for Search and Rescue Tasks", in Proc. 2003 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, Las Vegas, Nevada, USA, vol. 3, pp:2889-2895, 2003.
- [68] A. Ferworn, C. Wright, J. Tran, C. Li, H. Choset, "Dog and snake marsupial cooperation for urban search and rescue deployment", 2012 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), 2012.
- [69] A. A. Transeth, P. Liljebäck, S. Fjerdings, E. Kyrkjebø, and T. Mugaas, "New possibilities - Next generation robotic systems for inspection and maintenance operations", in Proc. EuroMaintenance, pp:229-232, Fiera di Verona, Italy, 2010.
- [70] S. A. Fjerdings, P. Liljebäck, and A. A. Transeth, "A snake-like robot for internal inspection of complex pipe structures (PIKo)", in Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems, pp:5665-5671, St. Louis, MO, USA, 2009.
- [71] E. Kyrkjebø, P. Liljebäck, and A. A. Transeth, "A robotic concept for remote inspection and maintenance on oil platforms", in Proc. ASME 28th Int. Conf. Ocean, Offshore and Arctic Engineering (OMAE), Honolulu, HI, USA, 2009.
- [72] F. Enner, D. Rollinson and H. Choset, "Motion Estimation of Snake Robots in Straight Pipes", 2013 IEEE Int. Conf. on Robotics and Automation (ICRA), Karlsruhe, Germany, 2013.
- [73] A. Kuwada, S. Wakimoto, K. Suzumori, and Y. Adomi, "Automatic pipe negotiation control for snake-like robot", IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics, pp:558-563, 2008.
- [74] S. Wakimoto, J. Nakajim, M. Takata, T. Kanda, and K. Suzwori, "A Micro Snake-like Robot for Small Pipe Inspection", in Proc. 2003 Int. Symposium on Micromechatronics and Human Science, pp:303-308, 2003.
- [75] Z. Wang, Q. Cao, N. Luan, and L. Zhang, "Development of an autonomous in-pipe robot for offshore pipeline maintenance", *Industrial Robot: An International Journal*, vol. 37 issue: 2, pp:177-184, 2010.
- [76] D. Rollinson, and H. Choset, "Gait-based compliant control for snake robots", in Proc. IEEE Int. Conf. on Robotics and Automation (ICRA), pp:5123-5128, Karlsruhe, Germany, 2013.

- [77] E. Rezapour, P. Liljebäck, "Path Following Control of a Planar Snake Robot with an Exponentially Stabilizing Joint Control Law", in Proc. IAV 2013 - IFAC Symposium on Intelligent Autonomous Vehicles, Vol. 8, Part 1, pp. 28-35, Gold Coast, Australia, June 2013.
- [78] E. Rezapour, K. Y. Pettersen, P. Liljebäck, and J. T. Gravdahl, "Path Following Control of Planar Snake Robots Using Virtual Holonomic Constraints", in Proc. 2013 IEEE Int. Conf. on Robotics and Biomimetics, pp. 530-537, Shenzhen, China, Dec. 12-14, 2013.
- [79] E. Rezapour, K. Y. Pettersen, P. Liljebäck, and J. T. Gravdahl, "Differential Geometric Modelling and Robust Path Following Control of Snake Robots Using Sliding Mode Techniques", in Proc. IEEE International Conference on Robotics and Automation (ICRA 2014), Hong Kong, China, 31 May-5 June, 2014.
- [80] E. Rezapour, K. Y. Pettersen, J. T. Gravdahl, and P. Liljebäck, "Body Shape and Orientation Control for Locomotion of Biologically-Inspired Snake Robots", in Proc. 5th IEEE RAS/EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob 2014), São Paulo, Brazil, Aug. 12-15, 2014.
- [81] E. Rezapour, A. Hofmann, K. Y. Pettersen, A. Mohammadi, and M. Maggiore, "Virtual Holonomic Constraints Based Direction Following Control of Snake Robots Described by a Simplified Model", in Proc. 2014 IEEE Multi-Conference on Systems and Control (IEEE MSC 2014), Antibes, France, Oct. 8-10, 2014.
- [82] E. Rezapour, A. Hofmann, and K. Y. Pettersen, "Maneuvering Control of Planar Snake Robots Based on a Simplified Model", in Proc. 2014 IEEE Int. Conf. on Robotics and Biomimetics (ROBIO 2014), Bali, Indonesia, Dec. 5-10, 2014.
- [83] 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. on Decision and Control (CDC 2014), Los Angeles, CA, Dec. 15 - 17, 2014.
- [84] E. Rezapour, K. Y. Pettersen, P. Liljebäck, J. T. Gravdahl, and E. Kela-sidi, "Path Following Control of Planar Snake Robots Using Virtual Holonomic Constraints: Theory and Experiments", *Robotics and Biomimetics*, SpringerOpen, 1:3, 2014.

- [85] 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*, 2014. Submitted.
- [86] M. W. Spong, S. Hutchinson, and M. Vidyasagar, "Robot modeling and control", Vol. 3. New York: Wiley, 2006.
- [87] F. Bullo, and A. Lewis, "Geometric control of mechanical systems", Springer, 2005.
- [88] Z. Wang, S. Ma, B. Li, and Y. Wang, "A unified dynamic model for locomotion and manipulation of a snake-like robot based on differential geometry", *Science China Information Sciences*, 54, no. 2, pp: 318-333, 2011.
- [89] T. I. Fossen, "Marine control systems: Guidance, navigation and control of ships, rigs and underwater vehicles", Trondheim, Norway: Marine Cybernetics, 2002.
- [90] G. Chirikjian, and J. Burdick, "The kinematics of hyper-redundant robot locomotion", *IEEE Transactions on Robotics and Automation*, 11(6), pp:781-793, 1995.
- [91] P. Liljebäck, K. Y. Pettersen, Ø. Stavdahl, and J. T. Gravdahl, "Stability analysis of snake robot locomotion based on averaging theory", 49th IEEE Conf. on Decision and Control, Atlanta, GA, USA, 2010.
- [92] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. H. Choi, and B. Morris, "Feedback control of dynamic bipedal robot locomotion", Boca Raton: CRC press, 2007.
- [93] M. Maggiore, and L. Consolini, "Virtual Holonomic Constraints for Euler-Lagrange Systems", *IEEE Transactions on Automatic Control*, 58(4):1001-1008, 2013.
- [94] L. Consolini, and M. Maggiore, "Control of a bicycle using virtual holonomic constraints", *Automatica* 49.9, pp. 2831-2839, 2013.
- [95] 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.
- [96] L. Freidovich, A. Robertsson, A. Shiriaev, and R. Johansson, "Periodic motions of the Pendubot via virtual holonomic constraints: Theory and experiments", *Automatica*, 44(3):785-791, 2008.

- [97] J. J. Slotine, and W. Li, "Applied nonlinear control", Prentice-Hall, NJ, 1991.
- [98] M. I. El-Hawwary, and M. Maggiore. "Reduction theorems for stability of closed sets with application to backstepping control design", *Automatica*, 49.1 pp:214–222, 2013.
- [99] H. Khalil, "Nonlinear Systems", Third edition. Prentice Hall, 2000.
- [100] X. Yang, "Practical stability in dynamical systems", *Chaos, Solitons and Fractals*, vol.11 no.7, pp:1087–1092, 2000.
- [101] V. I. Arnol'd, "Mathematical methods of classical mechanics", Vol. 60. Springer, 1989.
- [102] D. B. Dacic, D. Nesic, A. R. Teel, and W. Wang, "Path following for nonlinear systems with unstable zero dynamics: an averaging solution", *IEEE Transactions on Automatic Control*. 56:880-886, 2011.
- [103] A. De Luca and W. Book, "Robots with Flexible Elements," in *Springer Handbook of Robotics*, Berlin/Heidelberg: Springer-Verlag, pp. 287–319, 2008.
- [104] R. Skjetne, "The maneuvering problem", PhD-thesis, NTNU, 2005.
- [105] M. Sato, M. Fukaya, and T. Iwasaki, "Serpentine locomotion with robotic snakes," *IEEE Control Systems Magazine*, vol. 22, no. 1, pp. 64–81, 2002.
- [106] E. Kelasidi, K. Y. Pettersen, and J. T. Gravdahl, "A waypoint guidance strategy for underwater snake robots," in Proc. IEEE Mediterranean Conference of Control and Automation, pp. 1512–1519, June 2014.
- [107] E. Kelasidi, K. Y. Pettersen, and J. T. Gravdahl, "Modeling of underwater snake robots moving in a vertical plane in 3D," in Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 266–273, Chicago, USA, September 2014.
- [108] E. Kelasidi, K. Y. Pettersen, J. T. Gravdahl, and P. Liljebäck, "Modeling of underwater snake robots", in Proc IEEE Int. Conf. on Robotics and Automation, pp. 4540–4547, 31 May – 7 June, Hong Kong, China.
- [109] E. Kelasidi, K. Y. Pettersen, P. Liljebäck, and J. T. Gravdahl, "Integral Line-of-Sight for path following of underwater snake robots", in Proc. the IEEE Multi-Conference on Systems and Control (IEEE MSC 2014), Antibes, France, October 2014.

-
- [110] A. Crespi, A. Badertscher, A. Guignard, and A. J. Ijspeert, "Swimming and Crawling with an Amphibious Snake Robot," in Proc. of the 2005 IEEE Int. Conf. Robotics and Automation, pp. 3024–3028, 18–22 April 2005.
- [111] M. Nagumo, "Über die lage der integralkurven gewöhnlicher differentialgleichungen," Proc. Phys. Math. Soc. Japan (3), vol. 24, pp. 551–559, 1942.