Dynamics and Control of Snake Robot in Uncertain Underwater Environment

A Thesis Submitted
in Partial Fulfilment of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

By

Patel Bhavik Maheshkumar
(Roll No: 206103018)

Under the Supervision

Prof. S. K. Dwivedy

Department of Mechanical Engineering
Indian Institute of Technology Guwahati
Guwahati - 781 039, INDIA.
February, 2025

Certificate

This is to certify that the thesis entitled "**Dynamics and Control of the Snake Robot in Uncertain Underwater Environment**", submitted by **Patel Bhavik Maheshkumar** (206103018), a research scholar in the *Department of Mechanical Engineering, Indian Institute of Technology Guwahati*, for the award of the degree of **Doctor of Philosophy**, is a record of an original research work carried out by him under my supervision and guidance. The thesis has fulfilled all requirements as per the regulations of the Institute and in my opinion has reached the standard needed for submission. The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

Dated: 27.02.2025
Guwahati, Assam, India.

Prof. Santosha Kumar Dwivedy
Department of Mechanical Engineering,
Indian Institute of Technology Guwahati
Guwahati - 781039, India.

Declaration

I, Bhavik M. Patel (Roll no: 206103018) declare that the present written submission is my thoughts in my own words. I have adequately cited and referenced the original sources, where other's ideas have been involved. I also declare that I have adhered to all principles of academic honesty and integrity and have neither fabricated nor falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Dated: 27.02.2025
Guwahati, Assam, India.

Bhavik M. Patel
Department of Mechanical Engineering,
Indian Institute of Technology Guwahati
Guwahati - 781039, India.

This thesis is dedicated to Browniee

Acknowledgement

I would like to express my deep and sincere thanks to my supervisor Prof. Santosha K. Dwivedy for providing technical guidance, enthusiastic encouragement, and helpful discussions throughout the course of this research. Without his proper guidance, it would have been much more difficult to complete my research work and thesis on time. I am especially grateful for his patience in carefully reviewing and correcting all my manuscripts. Beyond academics, he has supported me in several other difficult situations whenever I needed help. I am also very thankful to my doctoral committee members, Prof. Shyamanta M. Hazarika, Prof. Deepak Sharma, and Prof. Arijit Sur, for taking the time to evaluate the progress of my work and their valuable suggestions have been crucial throughout my PhD journey. I would also like to thank other faculty members of the Mechanical Engineering Department for their kind help during my academic work. I would thank to IITG-TIDF for 3D printing facilities to develop model. I am sincerely grateful to Prof. Shyamal Chatterjee, my post-graduate supervisor at the Indian Institute of Engineering Science and Technology, Shibpur, for sparking my interest in research at a very early stage.

Fortunately, I had the chance to work with great colleagues during this journey at IIT Guwahati. I am honored to mention Dr. Viwek Kumar, whom I first met in IIT Guwahati and who has been a senior and mentor to me from day one. In the early days of joining the Mechatronics and Robotics Lab, I met Dr. Mohmmed Abbas, Dr. Jyotindra Narayan, and Rahul Kumar. The collaborating work with them was a joy and fun-filled experience. I would also like to express my gratitude to Dr. Anshul Garg and Dr. Sibananda Mohanty for sharing their research experience and helping me in various aspects.

I am blessed to have a group of close friends from Umiam Hostel: Anirban Basumatary, Akshay Daydar, Hemanga Bhattacharya, and Gunanka Hazarika, who stood by me in every situation. I also acknowledge Dr. Pankaj, Dr. Utpal, Sachit Narayan Pandey, Rajan Gupta, Ranit Roy, Atul Gautam, and Pankaj Kumar, with whom I frequently discussed matters beyond work in their lab. In the middle of my PhD journey, I worked in RIC'23 in which I met several great fellows, including Kalpana Kumari, Hitesh Ramrakhiyani, Prangan and Imran. After the event, these individuals became an integral part of my life. I also appreciate the technical team members of RIC'23 for their cooperation during the event. I was part of the RSF,ME and it was a joy to work with Pantha, Sumit, Abir, Suman and Dhanjita. In the final year of my PhD, I met Santosh Kumar, Subhash Pratap, Nikhil Mishra, Navneet Kumar, and Vivek Banjeet and had many cheerful moments occurred with these people in the last year. I also had the pleasure of working with undergraduate students Sanket, Chandrahas, Paarth, and Shreyash, as well as postgraduate students Shubham and Ankit. Working with them was a fun and enriching experience.

During my PhD, I also engaged with the DISHA club, which works for the education of underprivileged students. I acknowledge Mohit Kumar and Prof. Sriparna Bandopadhyay for their support and working with these amazing people enhanced my knowledge and prepared me for teaching. Here, I deeply connected with students Rashmina, Ritika, Biswa, and Manav and learned a lot from their lives and wishing them great success in the future.

I had the opportunity to work with Dr. Monalisa Pal and Dr. Roberto Valenti from MathWorks and their support gave me an edge in my research work. I also acknowledge the hostel cleaner - Mr. Navjeet, the canteen worker - Mr. Gautam, and the juice center worker - Mr. Jatin, the conversations I had with them were always enlightening. I am also thankful to the unknown reviewers who criticized and recognized my work that helped me to improve it. I am fortunate to have friends Chintan Pithadiya, Maunil Shah, Swapnil Dhobale, Sayan Das and Dr. Pradhyuman Sahoo. Their conversations and support have motivated me greatly.

While it may not be customary to formally thank family, I feel compelled to express my deepest gratitude. I am truly blessed to have my father, Mr. Maheshkumar, my mother, Mrs. Premilaben, my sister, Ms. Vaishaliben, my brother-in-law, Mr. Jigarkumar, and my nephew, Prish. They have always stood by me in every situation, offering unconditional support. My parents have consistently encouraged me with the words, "Follow your heart and pursue your dreams as long as you don't hurt anyone, and we will always be there to support you." I eagerly await the moment when I can see the tears of joy in their eyes as I return home after completing my PhD.

Finally, I dedicate this thesis to Browniee for her love and kindness. When I first arrived on campus, I accidentally stepped on her paw, but we became friends from that day onward. Whenever she couldn't find me, she would get upset and wait for me in the cycle parking area. Her pure love made this PhD journey more beautiful.

Bhavik M. Patel

Abstract

In recent years, researchers are interested in understanding living creatures and replicating their movement, structure, and cognition to develop biomimetic robots. Extracting these features, the biomimetic robots are developed for suitable environmental conditions and their applications. The snake is a versatile creature that can be found on land and in water environments with similar motion. Therefore, the mimicking of a snake can be used for wide applications in both environments. The motion of a snake is very sensitive to environmental conditions because of its crawling nature and that makes it a challenging problem to understand dynamics and control perspectives in a robotic system. The snake robot consists of the multi-links and joints that make it a complex mechanical system and controlling this system in an adaptive environment is a challenge. Therefore, the design of an appropriate control scheme for the complex dynamical system and modification in the mechanical system (that leads to dynamics) to get more suitable solutions become important. The main objective of the present work is to study the motion dynamics and design of a control scheme for a snake robot in uncertain underwater environment. Hence, it is proposed to study the motion dynamics of a snake robot and its control strategies in uncertain underwater conditions. The snake robot, consisting of 9 links and 8 joints, is considered under the assumption of planar motion with negligible added mass effect.

The thesis is focus on designing control strategies with existing mathematical model of the snake robot. The virtual holonomic constraints (VHCs) are used to derive the reduced dynamics of a mathematical model for a snake robot. The benefit of this method is that it determines the control objective for the head link of the snake robot with the entire robot following it. The sliding mode control (SMC) approach is used where the sliding surfaces are designed based on the angular position and tangential velocity tracking of head-link of the snake robot. Here, the chattering effect is reduced by improving the reaching law of the SMC scheme. In first study, the super twisting sliding mode control (STSMC) scheme is proposed that has a minimum chattering effect. The results are compared qualitatively and quantitatively with SMC and adaptive sliding mode control (ASMC) scheme. The dynamical uncertainties in the drag forces and fluidic torque acting on the snake robot during the motion are considered and estimated lumped value in

control algorithm. The control scheme is improved in the second study by introducing the uncertainties estimation using the neural network approach. In this, the radial basis function neural network (RBF-NN) is used for the estimation of the uncertain part of a dynamical system and that is inputted to ASMC scheme. So, ASMC scheme with RBF-NN is used for the motion control of a snake robot. Additionally in this study, the generation of the reference trajectory for the path following of a snake robot operating in the presence of water currents along with dynamical uncertainties is addressed. The stability of a proposed scheme is demonstrated through Lyapunov stability analysis.

The modification in the mathematical model of the snake robot by introducing different mechanical components such as pumps and thruster and then designing the control scheme for it. In third study, the snake robot dynamics are modified to 3D motion in the underwater environment along with the added mass effect. In this, the buoyancy variation technique is proposed for the vertical motion of a snake robot which is found in many underwater creatures. The horizontal motion of a snake robot is the serpentine motion as considered earlier. The combination of these two motions results in the 3D motion of a snake robot. The external disturbances are considered along with dynamical uncertainties in this work. The horizontal motion is controlled using existing STSMC and the vertical motion is achieved using the pump operations. For that, the event-based control scheme and PD control scheme are proposed. Moreover, the proposed dynamics are verified using the simscape multibody environment where the equation of motion is not required explicitly. In fourth study, the snake robot is considered with a thruster on the tail. The effect of thrust force on each link of a snake robot is studied in this work and the actor-critic neural network with the STSMC scheme is proposed for the motion control of a snake robot. The neural network estimates the dynamical system that is inputted to the STSMC scheme. The proposed control approach is verified with the existing STSMC scheme and the effectiveness of this scheme is compared in terms of chattering indicators and control effort.

In addition to the above studies, various actuator failure cases are studied. As snake robot has many actuators, there may be a chance of failure. Therefore, the control scheme is designed such that the robot motion can be sustained in the presence of the actuator failure. In this, the snake robot is constructed in the simscape multibody environment. The RBF-STSMC scheme is proposed to deal with various uncertainties and actuator failure. The proposed control scheme is compared with the existing control scheme and that performance of snake robot. These studies will help to develop the experimental set-up of underwater snake robot and its control applications.

Contents

List of Figures

List of Tables

List of Algorithms

List of Acronyms

3D	Three Dimensional
ACSTSMC	Actor critic super twisting sliding mode control
ASMC	Adaptive sliding mode control
AUV	Autonomous underwater vehicle
CE	Control effort
CI	Chattering indicator
CM	centre of mass
CPG	Central pattern generator
DOF	Degree of freedom
PD	Proportional derivative
PID	Proportional integral derivative
RBF-NN	Radial basis function neural network
RBF-NNSMC	Radial basis function neural network sliding mode control
RBF-STSMC	Radial basis function neural network sliding mode control
RMS	Root mean square
RL	Reinforcement learning

ROV	Remotely operated vehicle
SMC	Sliding mode control
STSMC	Super twisting sliding mode control
UUV	Underwater unmanned vehicle
UVM	Underwater vehicle manipulator
VHCs	Virtual holonomic constraints

List of Symbols

$\in$	Belong to
$\mathbb{R}$	The space of real numbers
$\mathbb{R}^n$	The space of real valued vectors of dimension n
$\mathbb{R}^{n\times m}$	The space of real valued $n \times m$ matrices
$\lvert\cdot\rvert$	Absolute value of a scalar argument
$\lVert\cdot\rVert$	Euclidian norm for vectors and spectral norm for matrices
$(\cdot)^T$	Transpose of the argument (vector or matrix)
$\max(\cdot)$	The maximum of the arguments
$\min(\cdot)$	The minimum of the arguments
$\dot{(\cdot)},\ \ddot{(\cdot)}$	First and second derivative of variable with respect to time
$\tilde{(\cdot)}$	represent the error of actual and reference variable
$\hat{(\cdot)}$	Uncertain variables/ parameters
I_n	Identity matrix of dimension n
$V,\ V_1,\ldots,V_n$	Lyapunov function
α	The amplitude of the joint angle
β	The phase angle between adjacent joint
ω	Angular velocity of joint
t	Time
$\phi_i(t)$	Joint angle between to two adjacent links
ϕ_0	offset angle
γ	input controlled joint angle
u_γ	Evaluation compensator of γ
u_{ϕ_0}	Evaluation compensator of ϕ_0
θ_i	Angle between link i and global X-axis

θ_n	Angle between head-link of snake robot and global X-axis
τ_i	Fluidic torque on link i
$\sigma_1,\ \sigma_2$	Sliding surface
(x_i, y_i)	Global coordinates of the center of mass of link i
(p_x, p_y)	Global coordinates of the center of mass of robot
u_i	Actuator torque of the joint between link i and link $i+1$
u_{i-1}	Actuator torque of the joint between link i and link $i-1$
$(f_{x,i}, f_{y,i}, f_{z,i})$	Fluid force on link i
$(h_{x,i}, h_{y,i})$	Joint constraint force on link i from link $i+1$
$(h_{x,i-1}, h_{y,i-1})$	Joint constraint force on link i from link $i-1$
v_t	Tangential velocity of head link
v_n	Normal velocity of head link
$\mathbf{R}_{\theta_n}$	Rotational matrix
l	Half length of the link
m	Mass of link
$\mathbf{u}_{eq}$	Equivalent control law
$\mathbf{u}_{sw}$	Reaching/ Switching law

1

Introduction

Contents

1.1 Introduction

Earth's surface is surrounded by 70% water. However, most of them are yet to be explored. This triggers many researchers to work on underwater vehicles as onboard human intervention or unmanned underwater vehicles (UUVs). A simple example of a human intervention vehicle is the submarine. Various applications of the submarines are surveillance, underwater operation tasks, and defence. There are various kinds of unmanned underwater vehicles such as Remotely Operated Vehicles (ROVs), Autonomous Underwater Vehicles (AUVs), Bio-inspired Underwater Robot and Underwater Vehicle Manipulators (UVMs) to explore the underwater environment and operations.

In the last two decades, there has been a growing demand for research using UUVs, especially biomimetic UUVs. The advantage of biomimetic robotic study is that it helps to understand the physics of the creatures, and those concepts would be used to develop adaptive robotic systems with many applications in different environments. A brief study of biomimetic robots is discussed in subsequent sections.

1.1.1 Biomimetic Underwater Robots

Various underwater biomimetic robots are available based on fish, octopus, snake, turtle, etc. These biomimetic robots are classified by structure, motion, navigation, and their capability for the tasks. Some of the available biomimetic underwater robots are SOFi [1], Octopus [2], U-CAT [3], Mamba [4], turtle [5], jellyfish [6], etc. and few of them are shown in Fig. 1.1.

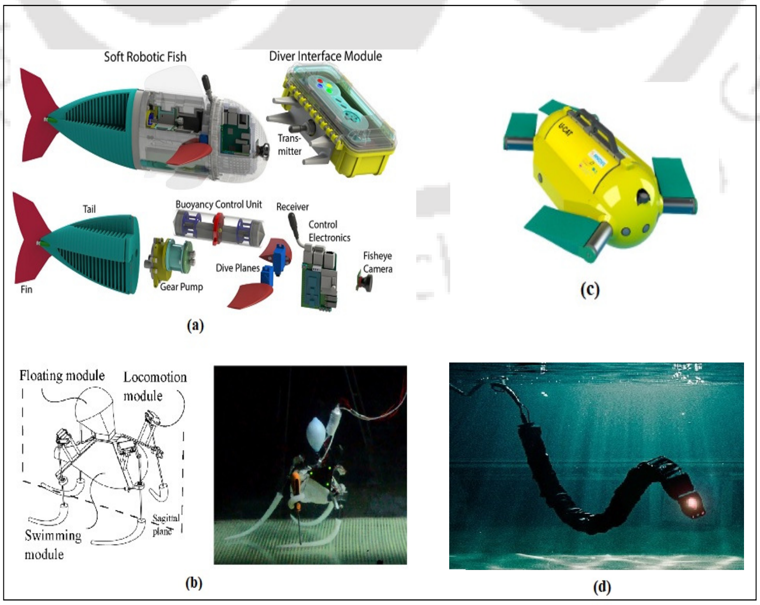

Figure 1.1: Biomimetic robots (a) Soft Robotic Fish [1] (b) Octopus [2] (c) U-CAT [3] (d) Mamba [4]

In the underwater environment, bioinspired robots have morphological similarities with aquatic creatures which can help them for the propulsion in underwater conditions [7]. Due to the slender structure of biomimetic robots, they can travel easily in a narrow area in the underwater environment for maintenance and surveillance of the offshore structure [8]. Mimic the biology of the creature and make it worthwhile for the surveillance, maintenance and underwater operations. These capabilities are observed in the snake robot. Therefore, in this thesis work, the study of modification in motion dynamics and control of the snake robot is focused.

1.1.2 Application of Snake Robot in the Underwater Environment

Snake-like robots are very useful in underwater applications. It requires low operational energy because most of the robot works on a servo motor [8]. It has a modular type structure that will adapt very well to any constraint environment. Some application of the snake robot is described below.

The snake robot is very thin in size. So, it can travel in a narrow area of the underwater environment. Due to a high number of degree-of-freedom, it can enter into some complex areas of the water. Now a day, the oil level goes very below in the ocean. To get the raw oil, the company established plant equipment below the water surface. Inspection of this plant is a very difficult task. A single leakage in a pipe is hazardous to sea life. This snake robot can wrap up the pipe and it can inspect the pipe leakage. Biomimetic design is helpful for the surveillance of sea life and checking whether it can influence life. It can be useful in the surveillance of underwater farming.

Manipulator operation is very challenging in an underwater environment. Usually, ROV with manipulators are used for dexterous work but it is very costly in the sense of manufacturing as well as operation. Due to its large size, it is not desirable to work in a complex environment. The underwater snake robot is the substitute robot for the manipulator operation by securing one end in a fixed position and attaching an end-effector to the other end. This configuration will function as a manipulator. The required control algorithm can be updated with it. Most of the snake robot is used for surveillance purposes, but it can be utilised as a manipulator. AUV equipped with a robotic arm is known as intervention-AUV (I-AUV). ALIVE [9], SAUVIM [10], RAUVI/ Tridentare [11–13] are the example of autonomous intervention AUV and these are shown in Fig. 1.2. In 2016, ECA Group officially announced H-ROV Ariane [14, 15] as a commercial intervention underwater robot. As per the design, the snake robot is made with several links. If we set one link as ground or with a thruster, it will work as a low-cost swimming manipulator. Therefore, the use of snake robots in the field of hovering, inspection and light intervention capabilities increases the attention of researchers [15]. The literature review on the snake robot is given in the next section, from the physics behind snake motion to various control approaches for motion control of a snake robot in different environmental conditions.

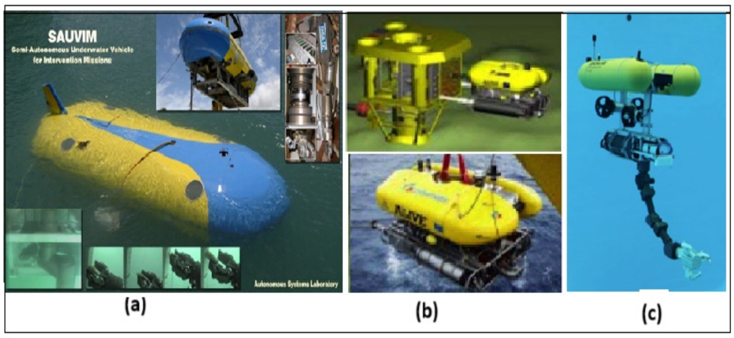

Figure 1.2: Intervention Robots (a) SAUVIM [10] (b) ALIVE [9] (c) RAUVI/ Tridentare[11]

1.2 Literature Review

To start work in the field of dynamics and control of snake robots, let's first be aware of the physics behind real snake motion that is given in the next subsection.

1.2.1 Physics behind the Motion of a Snake

In this section, the structure and biomechanical behaviour of snakes are explained.

1.2.1.1 Snake Structure

Liljeback et al. [16] explained the snake structure as follows. Normally, a biological snake consists of vertebrae of 130 to 400 maximum. Here, the joints can have a maximum of 10° side-to-side rotation. Despite this small angle movement, it is possible to get big curvature in the snake body because of the large number of vertebrae as shown in Fig. 1.3. These vertebrae are covered with skin which is direct contact with the environment (surface) and causes the locomotion due to friction [17]. Due to the rib type structure, the friction is less during the forward motion and higher during the backward motion as shown in Fig. 1.4. In the next subsection, the biomechanical behaviour of the snake is described.

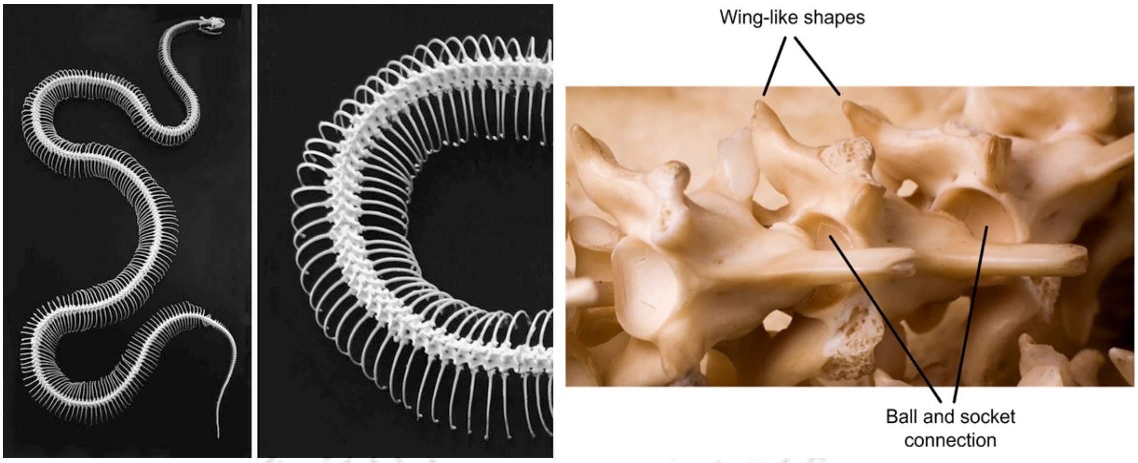

Figure 1.3: Skeleton, vertebrae, joint of snake [16]

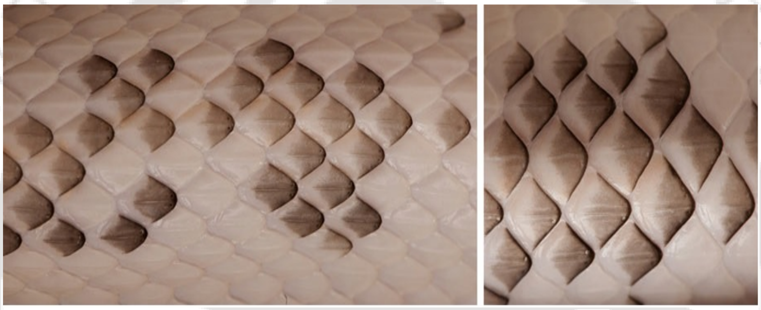

Figure 1.4: Skin of snake [16]

1.2.1.2 Biomechanical Behaviour of the Snake

Gray [18] described four locomotion types of a biological snake in 1946 on land *viz.* *'serpentine motion'*, *'concertina motion'*, *'rectilinear motion'*, *'crotaline motion'*. A brief description of these motions is given below [16]:

- ***Serpentine motion***: It is also called *'lateral undulation motion'*. It is the most common motion of the snake. In this motion, the entire body of the snake forms continuous sinuous movement as shown in Fig. 1.5(a). This motion can be described using three contact points in two for force generation and one for balancing the snake. Using this gait pattern does not only avoid the obstacle but also utilities the counter force to help forward propagation. This gait pattern is not suitable for narrow and very low friction surfaces. Every point along the body passes the same point on the ground, and there is never any static contact between the ground and any point along the body [19, 20]. During swimming, the same wave motion is produced, but the body then pushes against the resistance of the water.

- ***Concertina motion***: In this locomotion, part of the body is stationary and the other part moves in a forward direction as shown in Fig. 1.5(b). It is a very slow motion compared to serpentine motion. Snake switches this motion when it passes through a narrow area. This motion can be observed along pipes, branches, wires and cables. The motion pattern is not very efficient in terms of energy consumption but is often needed to traverse narrow spaces.

- ***Rectilinear motion***: In rectilinear locomotion, several parts of the body are in contact with the ground at any moment, and the gait uses a symmetrical form rather than a staggered contraction wave, shown in Fig. 1.5(c). This motion requires small vertical movement of the body. Heavy-body snakes use rectilinear motion.

- ***Crotaline motion***: It is also called side-winding locomotion. Instead of moving in the forward direction, the snake head travels in a transverse direction as shown in Fig. 1.5(d). The entire body travels in a forward direction and it looks like a transverse bending oscillation. This snake motion is associated with the desert and loose soil area.

- ***Other motion***: Other motion of the snake is considered as a part of the major four patterns. That means it has some variation in the parent pattern. Various motions are observed in a special situation such as rectilinear crawling, burrowing, jumping, sinus lifting, skidding, swimming and climbing [21]. Skidding which is also known as side-pushing, is observed in the low-friction area. Snake makes this motion energy efficient by down the head toward the land and passing the bending waves to the body. Snake climbs the tree using vertical, lateral undulation by taking branches support. Ariizumi and Matsuno [22] observed that sinus lifting and side-winding are more efficient than lateral undulation. Koopaee et al. [23] observed the eel-like motion of swimming snakes to move in a forward direction which is similar to serpentine motion.

It is observed from the above discussion that the locomotion of the snake depends on the size and the environment in which motion is considered. The further discussion on how the locomotion is different and control by following [16, 24]. The main difference between legged animals and serpentine is that the gait pattern's repeatability differs from the animal's speed that means faster repeating of gait increases the speed of the legged animal. In contrast, snake gait repetition does not affect the speed, the propulsion of the snake depends on the concerned environment. On a slippery surface, its forward velocity is low even though the high speed of the gait pattern. A large number of the muscles are involved in the snake locomotion and all the muscles are connected with each nervous

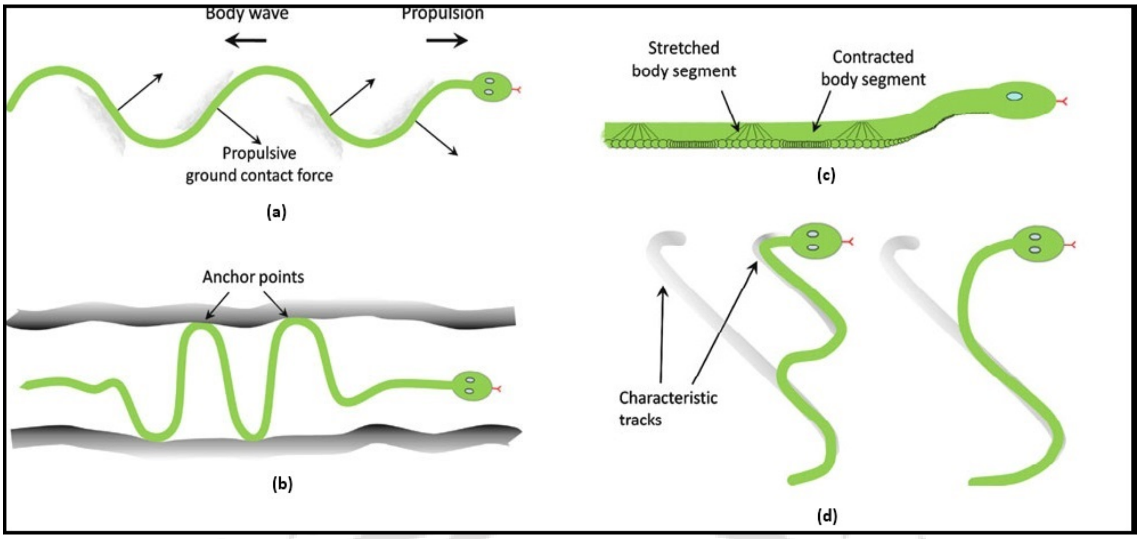

Figure 1.5: Snake locomotion (a) Serpentine motion (b) Concertina motion (c) Rectilinear motion (d) Crotaline motion. [16]

system. The nervous system works as an electrical wire connection that transmits the position signal of the muscles through neurons. The entire gait is controlled by the brain of a snake using this sensory motion. Based on this, many researchers have attempted to develop the snake-like robots and their control strategies for various applications that is explained in the next subsection.

1.2.2 Classification of Snake-like Robots

Snake robots are categorized on some basic properties [25] such as actuator types, degree of freedom (DOF) of each module, locomotion types and operating environment. The first snake robot was developed by Hirose [26] in 1972. It was used 20 revolute joints of 2-m length in *'Active Cord Mechanism'*. Further, design-based classification is *'Passive wheel Snake robot, Active wheel snake robot, Snake robot with active treads, Undulation using linear expansion (wheel-less snake robot)'*. In **Passive wheel snake robot**, the wheels of the robot are not directly connected to the actuator. These wheels are for the smooth forward motion of the snake robot. These wheels are not directly responsible for the motion of the robot. Because of the undulation motion of the robot, the force is exerted on the wheel that reduces the friction in the tangential direction. Some examples passive wheel snake robot are ACM R3 [27], ACM R5 [28], AmphiBot-I [29] and AmphiBot-2 [30] and shown in Fig. 1.6. In **Active wheel snake robot**, the wheel of the snake robot is directly connected to the actuator. So, the motion of the wheel is depended on the applied torque on the wheel. It is also called a train wheel structure. Examples of active wheel snake robots are GMD-Snake2 [31], Koryu-II [32] and ACM R4 [33] and shown in Fig. 1.8. The motion of **Active treads snake robot** is like a treadmill. Continuous belt contact with the ground surface is responsible for the motion of a robot as observed from the work of Omani Tread OT-4 [34] and JL -I [35] and shown in Fig. 1.7. **Wheel-less snake robot** are the similar to snake. Examples of these robots can be found in the work of Modsnake [36], PolyBot [37], M-Tran [38], CONRO [39], GMD snake, Uncle Sam [40], Mamba [4] and Kulko [16]. Some of wheel-less snake robots are shown in Fig. 1.9. The wheel-less snake robots can also be used for underwater environments [4] by making them waterproof. The locomotion of a wheel-less snake robot is '*lateral undulation motion*' in most of the cases and that makes the possible motion of snake robot in underwater environment [41].

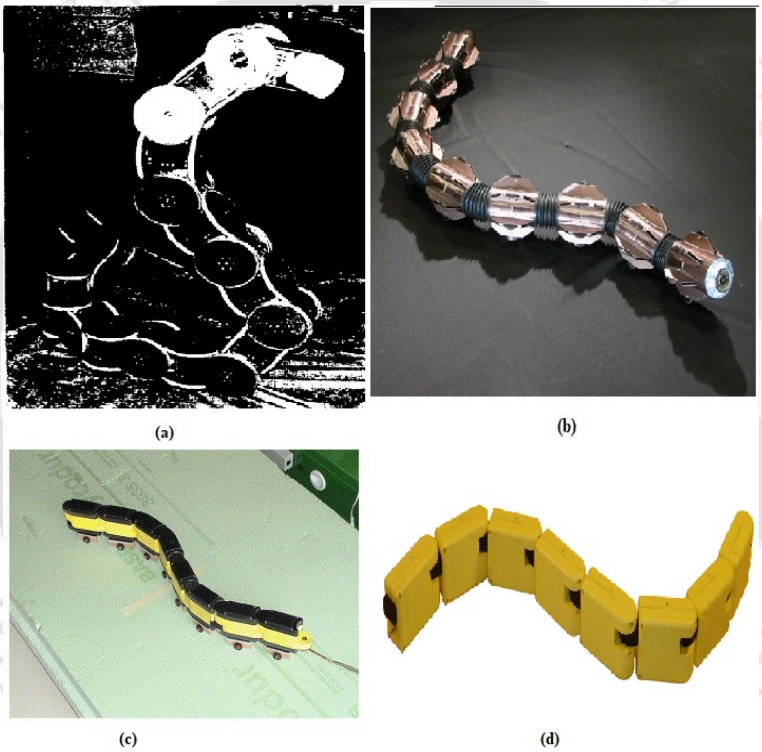

Figure 1.6: Passive Wheel Snake Robots (a)ACM R3[27],(b) ACM R5 [28], (c) Amphibot-I[29], (d) Amphibot-II [30]

The control of these robotic systems is an essential part of motion and to design control scheme the mathematical model of the snake robot is required to be known. So, in the next section, the dynamics and control approaches of the snake robot in various environments are explained in detail.

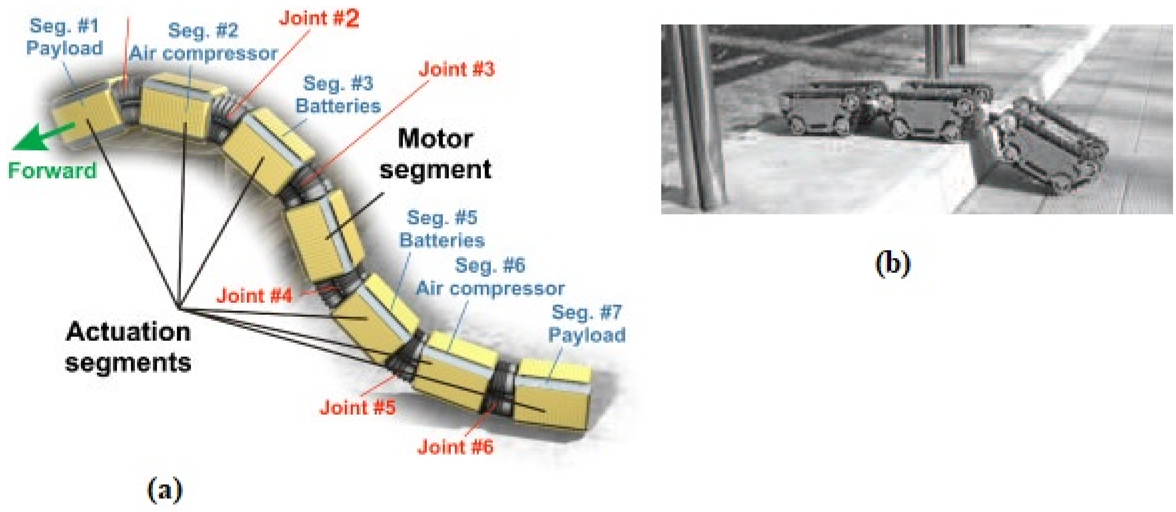

Figure 1.7: Active tread Snake robots (a)OT-4[34], Jl-I[35]

1.2.3 Background of the Dynamics and Control Approaches for the Snake Robots in Various Environments

The literature review of dynamics and control approaches for the snake robot is given in this subsection. The planar motion of the underwater snake robot has been considered in most of the literature [41–43]. However, the vertical motion is possible in the underwater environment. Another way of getting vertical motion in an underwater environment is the buoyancy variation technique. The vertical motion along with the horizontal motion of a snake robot can be generated by setting each actuator orthogonal [44–46] in a land based environment. The combination of vertical and horizontal motion reflects the 3D motion of a snake robot. Li et al. [47] designed a two controller for the land-based snake robot with desired pitch and yaw motions. However, it makes difficult to synchronize vertical and horizontal motions due to instability in pitch motion in the underwater environment. As mentioned earlier the motion of a snake robot happens in underwater conditions because of the drag forces and added mass effect. Increment in the speed of the snake robot can be possible using the external thrust force. Kelasidi et al. [48] used a quasi-static foil model to thrust the tail. The thruster tail-based snake robot [49] was experimentally verified by considering the serpentine and eel-like motions of the snake robot for the different gait parameters. In this, the speed of the snake robot which has a thruster at the tail, was found to be better than without a thruster. Snake robots equipped with different numbers of thrusters [50, 51] are considered for energy efficiency optimization. However, the detailed mathematical model of snake robot during the thrust force along with serpentine motion, its effect on each link and controller design are not

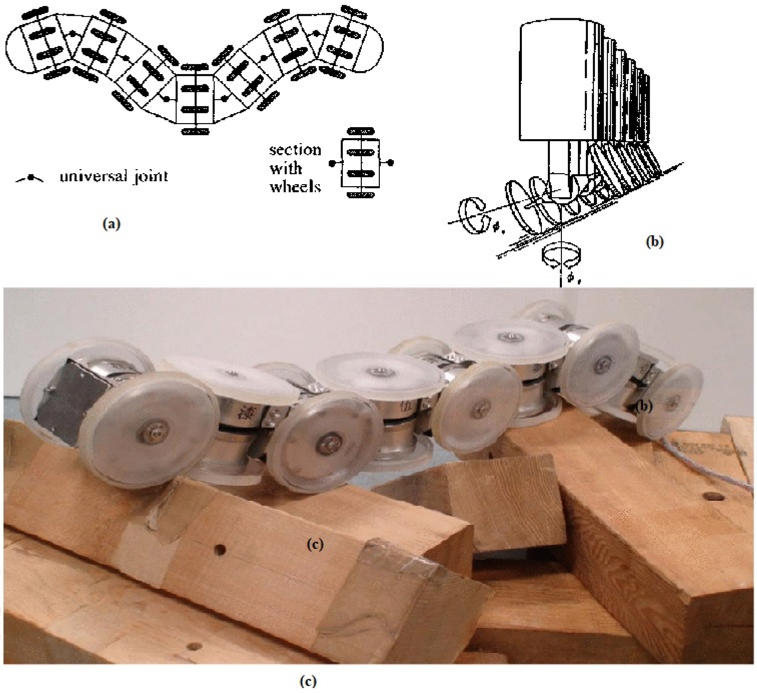

Figure 1.8: Active Wheel Snake robots (a)GMD snake[31], (b)Koryu-II[32], (c)ACM R4[33]

given in the literature.

The controller is an essential aspect of defining the position of the robot in the desired position. So, this is the most important research interest for the researchers. The robot's position along the direction of motion is required to be tracked to a predefined value or function [52]. There are various control techniques available for better system performance such as adaptive locomotion control [53], adaptive creeping locomotion control [54], sliding mode control [55] and neural network control [56].

Initially, the design and control of the snake robot have been investigated for land-based applications. Sato et al. [57] described the motion propagation of the multi-link snake robot in a land environment. Onal and Rus [42] proposed a fluidic soft robot's autonomous undulatory motion. Xiao and Murphy [58] described different types of practical applications of a snake robot and their test bed. Lim et al. [59] validated the serpentine locomotion of the robot in the various sloped environmental conditions. Liljeback et al. [60] proposed a cascaded approach for the path following of a snake robot. The gait pattern generation of a snake robot varies with the environmental conditions and requirements. The controller has to follow the stable limit cycle which consists of time-varying body shape. There are various techniques to generate the serpentine motion of the snake robot. The autonomous serpentine motion was generated using compressor threshold valve [61]. Dowling [62] described multiple gait motions for limbless locomotion. Ludeke

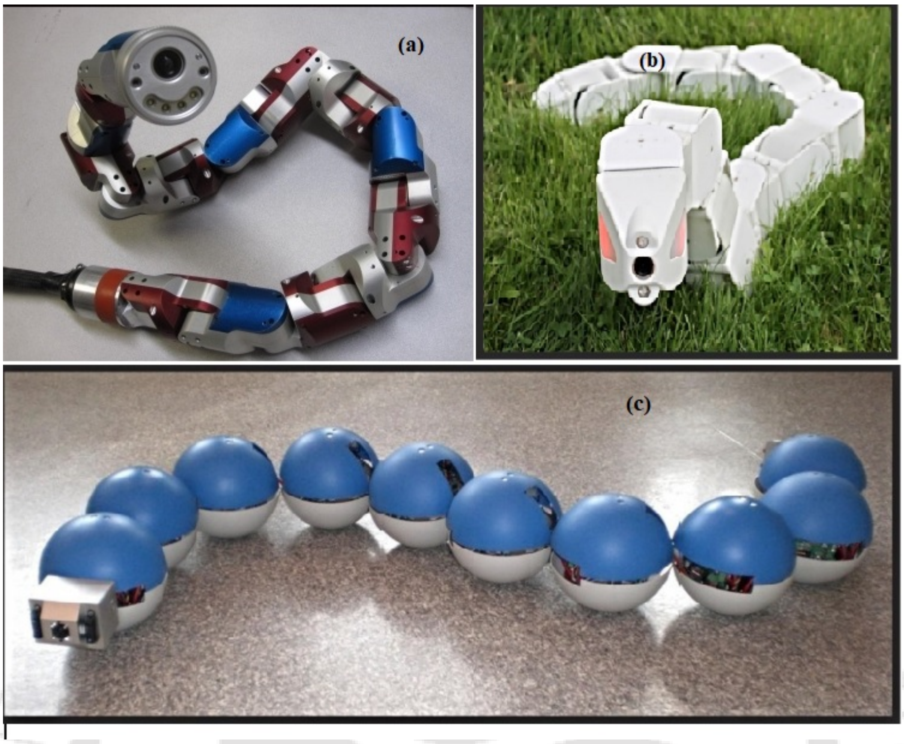

Figure 1.9: Wheel-less snake robots (a)Uncle Sam [40],(b)Mamba [4],(c)Kulko [16]

and Iwasaki [63] identified essential dynamics and defined the mode-shape of the natural oscillation of a snake robot such as lateral rolling locomotion in various environments [64, 65]. Park et al. [66] proposed RL-based controller that is trained using a proximal policy optimisation algorithm to generate gait motion and the inverse-RL controller is trained using an adversarial inverse reinforcement learning algorithm for damage recovery of the gait pattern. The another approach for gait pattern generation is a central pattern generator (CPG) which works on rhythmic neural network approach. Bing et al. [67] proposed data-driven inversion of CPG adoption law based on backpropagation method for robot environment interaction and robust motion control. Ijspeert and Crespi [68, 69] studied that CPG-based controllers produce rhythmic patterns without any sensory feedback. Prautsch and Mita [70] controlled of the gait pattern with respect to the number of links. Prautsch et al. [71] assumed nonholonomic velocity and nonslip wheel condition to propose gait control of the snake robot. Sato et al. [72] described the detailed dynamic model of a wheel-less snake robot with a gait pattern following using a PD controller and verified it with an experimental setup. Gray [73] studied that the snake can travel in the underwater conditions with the same serpentine gait pattern as land due to the morphological structure. This motivation was considered to develop an underwater snake robot [74]. Kohl et al. [43] proposed a control-oriented model for an underwater snake robot. Motion resistance of the snake robot in the underwater condition is higher than land due to nonlinear drag forces, fluidic torque and added mass effect [41].

Continuing with conventional control approaches for the cluttered environmental conditions is difficult. Therefore, alternative approaches are investigated for the locomotion control of the snake robot. In a land-based snake robot, locomotion control was addressed using virtual holonomic constraints (VHCs) [75, 76]. The VHCs enforce time-independent (autonomous) relations for biological gait patterns on the configured robot. Rezapour et al. [75] proposed a linear control-oriented model using VHCs. Similar VHCs approach has been applied to biped walking robots [77, 78], which consists of a multi-link joint mechanism. Mohammadi et al. [76] described using the singular perturbation-based controller that make the controller design of the snake robot in the hierarchical synthesis in which gait pattern generation is the lowest level and path planning is the highest level of hierarchy. The VHCs approach was used for the underwater snake robot [79] and designed a velocity observer for the measurement of the unknown ocean current velocity. The benefit of the VHCs approach over another path-following linearised control approach [80–83] is that controlling the head angle and velocity of the snake robot can control the entire robot such as rail-like motion where the engine at the front and maintain the motion of the entire train. Therefore, VHCs-based approach is used to design the motion control of the snake robot in this thesis work. Here, the robust controller is designed for the head link and the snake robot follows accordingly.

The sliding mode control (SMC) approach is one of the robust control approaches widely used for mechanical systems because of its robustness toward uncertain conditions [84, 85]. Mukherjee et al. [55, 86] used VHCs-based SMC and adaptive sliding mode control (ASMC) to deal with friction uncertainty in land-based snake robots where qualitatively chattering has been observed. Orucevic et al. [87] controlled snake robot automatically in vortex environment.

1.2.4 Robust Control Scheme and its Applications

As it is observed from the previous section, the less literature are available in the application of the underwater snake robot. There are very less control scheme explored in the field of snake robot. There is scope of research to explore the various control scheme that has been applied to different robotic application other than snake robot that can be used in the motion control of snake robots. Therefore, in this subsection, the various application of robust control schemes are explored. The applications of SMC scheme are found in underactuated bio-inspired robot (U-CAT) [88] and soft robot [89]. The higher chattering might increase the malfunctioning of the actuators. Therefore, it is necessary to reduce chattering for smooth trajectory tracking to complete the operation. The super twisting sliding mode control (STSMC) can be helpful in reducing the chattering effect [90]. The limitation of these robust control approaches is that the designer assumes uncertainties during the controller design [91] which means the prior idea of the uncertainties is required. However, it is difficult to know the uncertainties if the robot moves in unknown environments and the controller designer uses lumped values of the uncertainties. Therefore, it is necessary to estimate the uncertainties from the feedback responses of the robot. Real-time estimation is possible using the neural network approach. Liu et al. [92] used a CPG-based neural network approach to learn the locomotion of the soft snake-like robot. The neural network requires a lot of data to estimate the exact value of the uncertainties, which might not be possible during real-time applications due to computational limitations. The estimation, learning and motion of the robot with a neural network are difficult. Therefore, it is necessary to use such an algorithm that can estimate uncertainties as fast as possible for real-time operations. The real-time estimation with limited feedback data can be possible using the radial basis function (RBF) method [91]. The RBF neural network is faster than other methods with better approximation capabilities [93]. The hidden layer can make the entire problem be solved parallel by operating each neuron separately. The improvement in the neural network approach can be possible using the optimal control approach. The online optimization of the neural network can be referred to as 'Reinforcement Learning' (RL) [94].

The applications of RL-based control techniques are found in the following literature. Elkenawy et al. [95] proposed an actor-critic neural network-based control method and experimentally verified it with servo motor rotation. Guo and Zhao [96] proposed an online adaptive optimal algorithm using a self-adaptive actor-critic neural network approach. He et al. [97] used the actor-critic neural network control scheme for two-link flexible manipulators. Sousy et al. [98] tuned STSMC control gain parameters using the actor-critic neural network approach for permanent magnet synchronized motor drive. Li et al. [99] used supervised RL for controlling the disturbance rejection of a high dynamic quadrotor. Song et al. [100] used Q-Learning-based RL control for speed control of the permanent magnet synchronous motor. The recent applications of neural network-based SMC are approximation of uncertain robot dynamic model [101], slipping parameters estimation of wheeled mobile robot [102], robotic manipulator [103], permanent magnet linear synchronous motor [104] and maglev system [105]. In the case of an actuator failure, the application of a neural network with SMC is found in fault-tolerant aircraft control [106] and ship dynamic position stabilization [107].

To get the vertical motion in an underwater environment buoyancy variation method is studied. In the case of the robotic systems, Woods et al. [108] used a ballast tank (containing water) to change depth using mass variation and simulated vertical motion. Tiwari and Sharma [109] experimentally studied the depth variation of the spherical object by taking the water from outside. This can be helpful for the snake robot in underwater vertical motion using the buoyancy variation technique. Snake robot also has horizontal motion which is serpentine motion. Hence, with the serpentine horizontal motion and buoyancy variation based vertical motion, the snake robot can achieve the 3D motion. The 3D motion can be controlled in two parts, i.e. horizontal motion control and vertical motion control. The event-based control technique is helpful for vertical motion control because it is easy to control the pumps. The event-based techniques have been used in the field of robotics various applications [110–112]. Lunze and Lehmann [110] used an event-based state feedback approach to generate the control signal. Abbas et al. [111] used event-triggered adaptive control for upper-limb rehabilitation. The control input is generated if an event condition is triggered in the event-based control method. Nath and Bera [112] used event-based control with integral sliding mode control to redesign control gains for the robotic manipulator. There is a need to observe the negative/ neutral/ positive buoyancy for the vertical motion of the snake robot. Therefore, the event-based control method can be helpful for vertical motion control. PD controller can also be used to control the water supply of the pump.

Similarly the recent studies on the thrust force allocation on other underwater vehicles are given [113–115]. However, a detailed explanation of the force allocation due to tail thrust on each link has not been reported in the literature. Therefore, in the present thesis work an attempt has been made to fill this gap.

Another approach to analyze the proposed dynamics and control scheme is to verify on the virtual platform. In which, the proposed mathematical model of dynamics and control scheme can be verified. In the recent literature, the use of this virtual platform is found that is discussed further. Zhang et al. [116] performed motion control of a snake using reinforcement learning method in the coppeliasim environment. Le and Vo [117] analysed RUU Delta robot using the SimScape environment. Modak and Krishna [118] simulated industrial manipulator in simscape for kinematics and singularity analysis. Cosenza et al. [119] analysed rocker-bogie suspension for robotic rover using the simscape environment.

1.3 Potential Research Gap

From the above literature studies on the controller design of the snake robot following technical gaps are observed.

Motion control in underwater uncertainties: In land-based snake robots, friction force uncertainty is considered in the literature. However, in underwater conditions, in place of friction uncertainty, one may encounter uncertainty due to drag force, fluidic torque and force due to added mass effect. It is observed from the extensive literature review no work has been carried out considering these uncertainties in underwater applications. Hence, it is necessary to implement a robust controller for underwater applications by controlling the body shape, head angle and forward velocity. The estimation of the uncertain dynamics using a neural network approach is also possible for the motion control of a snake robot. This work can be extended by considering external disturbances.

Buoyancy variation techniques for the modular snake robot: Buoyancy variation can make the robot go up and down in the water body. This type of work can be seen in AUVs and ROVs. One may extend this technique to underwater snake robots.

Snake robot with thrust force and control scheme for it: In literature, a snake robot equipped with a thruster is given. However, the detailed dynamics and its effect on each link is not explained in the literature. One may extend its detailed dynamics and its control scheme for motion control of a snake robot in the underwater environment.

Tracking head angle during the actuator failure condition in the underwater application: Arizumi et al. [22] reported the head angle tracking but for the land-based snake robot. So, the actuator failure of the snake robot can be extended with underwater uncertainties by designing the controller for the head of it.

1.4 Objective of the Present Work

The dynamics and control of a snake robot with nine links and eight joints actuated by servo motors in uncertain underwater environmental conditions is attempted in this thesis. The novelty of this work to develop mathematical model of the snake robot (leads to model based design (MBD)) for the different motion in an uncertain underwater environment and investigate the different control scheme which can improve the performance of it. To achieve this objective following five studies are contributed in this work.

- The STSMC scheme is proposed for the orientation and velocity control of the head link of a snake robot using the VHCs.
- Estimation of the uncertain dynamics using the neural network approach is proposed with motion control using the ASMC scheme.
- The derivation of the mathematical model for the 3D motion of a snake robot in an underwater environment is proposed.
- The dynamic equation of a snake robot equipped with a thruster at the tail is derived, and the VHC-based RL-robust control scheme is proposed.

- Motion control of a snake robot in an underwater environment with various cases of failed actuators is proposed.

1.5 Organization of the Thesis

This thesis is organized into two main parts, each chapter addressing a particular research problem as elaborated in the previous section. The first part of the thesis covers the robust control techniques and deals with the uncertainties.

- **Chapter 2:** The super twisting sliding mode control scheme is proposed to deal with uncertainties in the drag force and fluidic torque with minimum chattering effect.

- **Chapter 3:** The adaptive RBF neural network sliding mode control is proposed for the path following of a snake robot in the presence of the water current and environmental uncertainties.

In the second part of the thesis, the dynamics of a snake robot is modified and the controller is designed to deal with various environmental uncertainties.

- **Chapter 4:** In this chapter, the mathematical model of 3D motion of a snake robot is proposed in the combination of the horizontal and vertical motion. In this, the vertical and horizontal motion are controlled simultaneously to get the 3D motion of a snake robot.

- **Chapter 5:** In this chapter, the mathematical model of a snake robot equipped with the tail-thrust is derived. The reinforcement learning-based control scheme is proposed for the motion control of a snake robot.

- **Chapter 6:** In this chapter, the path following of a snake robot with various actuator failure cases are considered. The performance of a snake robot is evaluated.

- **Chapter 7:** This chapter concludes the research work and outlines the scope for further research.

2

Motion Control of the Planar Snake Robot using the VHCs based STSMC

Contents

2.1 Introduction

In this chapter, the serpentine-gait motion of multi-link snake robot is studied in the underwater environment. The motion of a snake robot is possible due to nonlinear drag forces and fluidic torque acting on the robot in underwater environment. The water environments have many challenges. The major challenges are considered as uncertainties in drag forces and fluidic torque during the motion of a snake robot. To deal with these uncertainties, there is a need for controller design. In this work, the sliding mode controller is designed on the reduced order dynamical system. The reduced dynamics is derived by using the virtual holonomic constraints (VHCs). The parameters of the gait pattern are obtained using the sliding surface which is formulated for head link angle and tangential velocity tracking. The super twisting sliding mode control (STSMC) approach is proposed to deal with uncertainties and results are compared with sliding mode control (SMC) and adaptive sliding mode control (ASMC). The results are compared in terms of chattering indicators to check the effectiveness of the proposed control scheme.

This chapter is organized as follows. In section 2.2, the mathematical modelling of the snake robot is studied. The proposed VHCs-based STSMC scheme and implemented in the section 2.3. Finally, the results of the implemented control scheme are discussed in section 2.4 and the work is summarized in section 2.5.

2.2 Mathematical Modelling of the Underwater Snake Robot

Figure 2.1 shows the schematic diagram of a snake robot with n links and $(n-1)$ joints. It has $(n+2)$ degree-of-freedom which can be described as *n-link angles* and *centre of mass (CM) position* (p_x, p_y). Here, *the neutral buoyant snake robot motion in a hypothetical horizontal plane for the underwater condition* is assumed. In Fig. 2.1, angle θ_i is the orientation of link 'i' with respect to global X- axis, angle ϕ_i is the joint angle between two adjacent links and the CM position of a link 'i' describes as (x_i, y_i). The rotation matrix of i^{th} - link body frame with respect to the global frame can be expressed as follows.

$$\mathbf{R}_{i^{th}-link}^{global} = \begin{bmatrix} \cos\theta_i & -\sin\theta_i \\ \sin\theta_i & \cos\theta_i \end{bmatrix}. \tag{2.1}$$

[1]This chapter is part of the publication: B. M. Patel and S. K. Dwivedy (2023). Virtual holonomic constraints based super twisting sliding mode control for motion control of planar snake robot in the uncertain underwater environment. Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering, 237(8), 1480-1491.

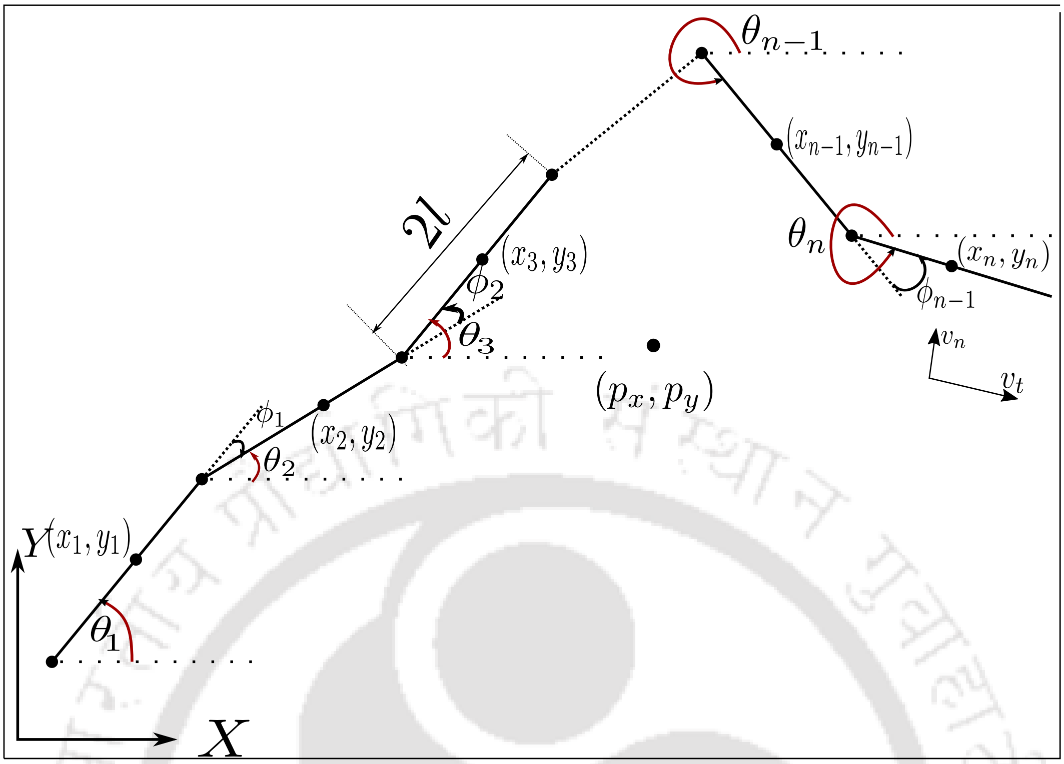

Figure 2.1: Schematic view of the snake robot

The CM position (p_x, p_y) of the snake robot can be expressed as

$$\mathbf{p_{xy}} = \begin{bmatrix} p_x \\ p_y \end{bmatrix} = \frac{1}{n} \begin{bmatrix} \mathbf{e}^T\mathbf{X} \\ \mathbf{e}^T\mathbf{Y} \end{bmatrix}, \tag{2.2}$$

where, $\mathbf{X} = [x_1, \cdots\cdots, x_n]^T \in \mathbb{R}^n$, $\quad \mathbf{Y} = [y_1, \cdots \ \cdots, y_n]^T \in \mathbb{R}^n$ and $\mathbf{e} = [1 \cdots\cdots 1]^T, \in \mathbb{R}^n$. The geometric constraint of the snake robot can be described as

$$x_{i+1} - x_i = l\cos\theta_i + l\cos\theta_{i+1}, \tag{2.3a}$$

$$y_{i+1} - y_i = l\sin\theta_i + l\sin\theta_{i+1}, \tag{2.3b}$$

$$\mathbf{DX} + l\mathbf{A}\cos\boldsymbol{\theta} = 0, \tag{2.3c}$$

$$\mathbf{DY} + l\mathbf{A}\sin\boldsymbol{\theta} = 0, \tag{2.3d}$$

where, $\mathbf{A} = [\mathbf{I}_{(n-1)}, \mathbf{0}_{(n-1,1)}] + [\mathbf{0}_{(n-1,1)}, \mathbf{I}_{(n-1)}] \in \mathbb{R}^{(n-1)\times n}$ is addition matrix, $\sin\boldsymbol{\theta} = [\sin\theta_1, \cdots \ \cdots, \sin\theta_n]^T \in \mathbb{R}^n$ is sine vector, $\mathbf{D} = [\mathbf{I}_{(n-1)}, \mathbf{0}_{(n-1,1)}]$ - $[\mathbf{0}_{(n-1,1)}, \mathbf{I}_{(n-1)}] \in \mathbb{R}^{(n-1)\times n}$ is subtracting matrix and $\cos\boldsymbol{\theta} = [\cos\theta_1, \cdots \ \cdots, \cos\theta_n]^T \in \mathbb{R}^n$ is cosine vector. By using Eq. 2.2 and Eq. 2.3, the final position of all the links can be written as

$$\mathbf{X} = -l\mathbf{K}^T\cos\boldsymbol{\theta} + \mathbf{e}p_x, \tag{2.4a}$$

$$\mathbf{Y} = -l\mathbf{K}^T \sin \boldsymbol{\theta} + \mathbf{e}p_y, \tag{2.4b}$$

where, $\mathbf{K} = \mathbf{A}^T(\mathbf{D}\mathbf{D}^T)^{-1}\mathbf{D}, \in \mathbb{R}^{n\times n}$. The first derivative of Eq. 2.4 gives velocity and the second derivative of Eq. 2.4 gives acceleration of the links those are given below.

$$\dot{\mathbf{X}} = l\mathbf{K}^T\mathbf{S}_\theta\dot{\boldsymbol{\theta}} + \mathbf{e}\dot{p}_x, \tag{2.5a}$$

$$\dot{\mathbf{Y}} = -l\mathbf{K}^T\mathbf{C}_\theta\dot{\boldsymbol{\theta}} + \mathbf{e}\dot{p}_y, \tag{2.5b}$$

$$\ddot{\mathbf{X}} = l\mathbf{K}^T\mathbf{C}_\theta diag(\dot{\boldsymbol{\theta}})\dot{\boldsymbol{\theta}} + l\mathbf{K}^T\mathbf{S}_\theta\ddot{\boldsymbol{\theta}} + \mathbf{e}\ddot{p}_x, \tag{2.5c}$$

$$\ddot{\mathbf{Y}} = l\mathbf{K}^T\mathbf{S}_\theta diag(\dot{\boldsymbol{\theta}})\dot{\boldsymbol{\theta}} - l\mathbf{K}^T\mathbf{C}_\theta\ddot{\boldsymbol{\theta}} + \mathbf{e}\ddot{p}_y, \tag{2.5d}$$

where, $\boldsymbol{S_\theta} = diag(\sin \boldsymbol{\theta}) \in \mathbb{R}^{n\times n}, \boldsymbol{C_\theta} = diag(\cos \boldsymbol{\theta}) \in \mathbb{R}^{n\times n}$ and $\boldsymbol{\theta} = [\theta_1........\theta_n]^T \in \mathbb{R}^n$. The relative tangential velocity $(v_{t,r})$ and normal velocity $(v_{n,r})$ of the snake robot head link can be given as

$$\begin{bmatrix} v_{t,r} \\ v_{n,r} \end{bmatrix} = \mathbf{R}_{\theta_n}^T \left(\dot{\mathbf{p}}_{xy} - \mathbf{v}_c\right). \tag{2.6}$$

where, $\mathbf{v}_c = [v_{cx}, v_{cy}]^T$ and $\mathbf{R}_{\theta_n}$ is head link rotational matrix. v_{cx} and v_{cy} are the water current velocity in X and Y direction, respectively and $\dot{\mathbf{p}}_{\mathbf{xy}}$ is the velocity of the center of mass. It may be noted that in the global X-Y frame the velocity of CG of each link should be same as CG of snake robot to avoid the separation of joints.

The free-body diagram of the link 'i' is shown in Fig. 2.2. The snake robot has an undulatory movement of the links with phase angle delay. Due to the undulatory motion of the links, they interact with the water environment which creates linear and nonlinear drag forces, fluid torque and added mass effect to generate propulsion. *The assumption has been taken in this chapter for deriving the dynamical equation is that added mass effect is negligible for slowly moving underwater vehicle [120] and especially for the bio-inspired robot [121].* The force balance equation is given for a link 'i' as

$$\begin{aligned} m\ddot{x}_i &= f_{x,i} + h_{x,i} - h_{x,i-1}, \\ m\ddot{y}_i &= f_{y,i} + h_{y,i} - h_{y,i-1}, \end{aligned} \tag{2.7}$$

where, $(f_{x,i}, f_{y,i})$ are the drag forces acting on the link 'i', $(h_{x,i}, h_{y,i})$ are the joint constraints between link 'i' and '$i+1$', $(h_{x,i-1}, h_{y,i-1})$ is the joint constraint between link '$i-1$' and 'i' as shown in Fig. 2.2. The moment balance equation of link 'i' can be given below.

$$J\ddot{\theta}_i = u_i - u_{i-1} - l\sin\theta_i(h_{x,i} + h_{x,i-1}) + l\cos\theta_i(h_{y,i} + h_{y,i-1}) + \tau_i. \tag{2.8}$$

where J is the mass moment of inertia of link 'i', u_i is the actuator torque between link 'i' and '$i+1$', u_{i-1} is the actuator torque between link '$i-1$' and 'i', $\tau_i = -\lambda_1\dot{\theta}_i - \lambda_2\dot{\theta}_i^2$, in

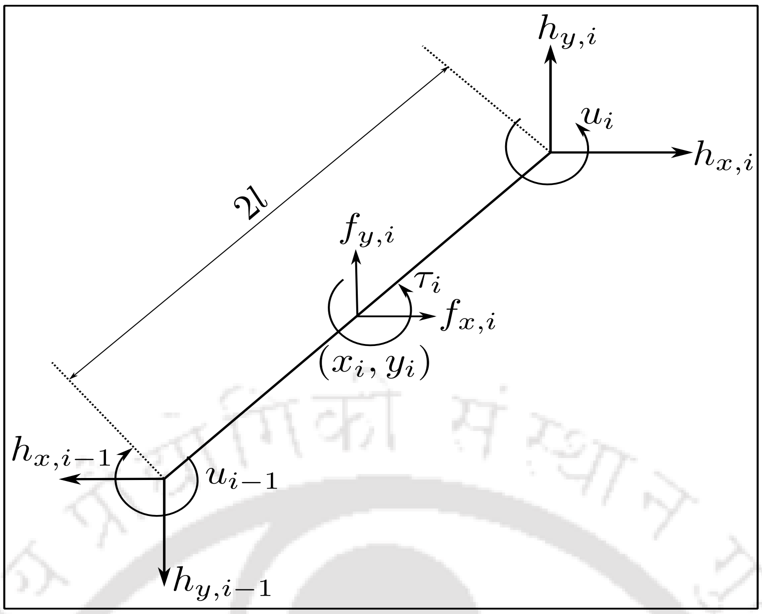

Figure 2.2: Forces on a link 'i'

which λ_1 and λ_2 are the linear and nonlinear rotational damping parameters, respectively [41]. The Eqs. 2.7- 2.8 can be extended for the whole body of the snake robot as follows.

$$m\ddot{\mathbf{X}} = \mathbf{f}_{d,x} + \mathbf{D}^T\mathbf{h}_x, \tag{2.9a}$$

$$m\ddot{\mathbf{Y}} = \mathbf{f}_{d,y} + \mathbf{D}^T\mathbf{h}_y, \tag{2.9b}$$

$$J\mathbf{I}_n\ddot{\boldsymbol{\theta}} = \mathbf{D}^T\mathbf{u} - l\mathbf{S}_\theta\mathbf{A}^T\mathbf{h}_x + l\mathbf{C}_\theta\mathbf{A}^T\mathbf{h}_y + \boldsymbol{\tau}, \tag{2.9c}$$

where, $\mathbf{u} = [u_1, \cdots\cdots, u_{n-1}]^T$, $\mathbf{f}_{d,x} = [f_{x,1}, \cdots\cdots, f_{x,n}]^T$, $\mathbf{f}_{d,y} = [f_{y,1}, \cdots\cdots, f_{y,n}]^T$, $\mathbf{h}_x = [h_{x,1}, \cdots\cdots, h_{x,n}]^T$, $\mathbf{h}_y = [h_{y,1}, \cdots\cdots, h_{y,n}]^T$, $\boldsymbol{\tau} = [\tau_1, \cdots\cdots, \tau_n]^T$. The robot changes position during its motion. So, the acceleration of CM position can be obtained using Eq. 2.9a- 2.9b as follows.

$$\ddot{\mathbf{p}}_{xy} = \frac{1}{n}\begin{bmatrix}\mathbf{e}^T\ddot{\mathbf{X}} \\ \mathbf{e}^T\ddot{\mathbf{Y}}\end{bmatrix} = \frac{1}{nm}\mathbf{E}^T\mathbf{f}_d, \tag{2.10}$$

where, $\mathbf{e}^T\mathbf{D}^T = 0, \mathbf{f}_d = [\mathbf{f}_{d,x}, \mathbf{f}_{d,y}]^T$, $\mathbf{E} = \begin{bmatrix}\mathbf{e} & \mathbf{0}_{(n,1)} \\ \mathbf{0}_{(n,1)} & \mathbf{e}\end{bmatrix} \in \mathbb{R}^{2n\times n}$. The equations of motion of the multi-link snake robot are given by substituting Eq. 2.5 in Eqs. 2.9- 2.10 as follow.

$$\mathbf{M}_\theta\ddot{\boldsymbol{\theta}} + \mathbf{W}_\theta\dot{\boldsymbol{\theta}}^2 + \boldsymbol{\Lambda}_1\dot{\boldsymbol{\theta}} + \boldsymbol{\Lambda}_2 diag(|\dot{\boldsymbol{\theta}}|)\dot{\boldsymbol{\theta}} - l\mathbf{S}\mathbf{C}_\theta^T\mathbf{f}_d = \mathbf{D}^T\mathbf{u}, \tag{2.11a}$$

$$\ddot{\mathbf{p}}_{\mathbf{xy}} = \frac{1}{nm}\mathbf{E}^T\mathbf{f}_d. \tag{2.11b}$$

where,

$$\begin{aligned}
&\mathbf{M}_\theta = J\mathbf{I_n} + ml^2\mathbf{S}_\theta\mathbf{V}\mathbf{S}_\theta + ml^2\mathbf{C}_\theta\mathbf{V}\mathbf{C}_\theta,\\
&\mathbf{W}_\theta = ml^2\mathbf{S}_\theta\mathbf{V}\mathbf{C}_\theta - ml^2\mathbf{C}_\theta\mathbf{V}\mathbf{S}_\theta,\\
&\mathbf{V} = \mathbf{A}^T(\mathbf{D}\mathbf{D}^T)^{-1}\mathbf{A}, \in \mathbb{R}^{n\times n},\\
&\boldsymbol{\Lambda_1} = \lambda_1\mathbf{I}_n, \quad \boldsymbol{\Lambda_2} = \lambda_2\mathbf{I}_n,\\
&\mathbf{f}_d = \mathbf{f}_{Ld} + \mathbf{f}_{NLd},\\
&\mathbf{f}_{Ld} = l\mathbf{Q}_\theta\mathbf{S}\mathbf{C}_\theta\dot{\boldsymbol{\theta}} + \mathbf{Q}_\theta\mathbf{E}\mathbf{v}_r, \quad where \quad \mathbf{v}_r = \dot{\mathbf{p}}_x y - \mathbf{v}_c,\\
&\mathbf{f}_{NLd} = -\begin{bmatrix} c_t\mathbf{C}_\theta & -c_n\mathbf{S}_\theta \\ c_t\mathbf{S}_\theta & c_n\mathbf{C}_\theta \end{bmatrix} diag\left(abs\begin{bmatrix}\mathbf{V}_{rx}\\ \mathbf{V}_{ry}\end{bmatrix}\right)\begin{bmatrix}\mathbf{V}_{rx}\\ \mathbf{V}_{ry}\end{bmatrix},\\
&\begin{bmatrix}\mathbf{V}_{rx}\\ \mathbf{V}_{ry}\end{bmatrix} = \begin{bmatrix}\mathbf{C}_\theta & \mathbf{S}_\theta\\ -\mathbf{S}_\theta & \mathbf{C}_\theta\end{bmatrix}\begin{bmatrix}\dot{\mathbf{X}} - \mathbf{V_x}\\ \dot{\mathbf{Y}} - \mathbf{V_y}\end{bmatrix},\\
&\mathbf{S}\mathbf{C}_\theta = \begin{bmatrix}\mathbf{K}^T\boldsymbol{S_\theta}\\ -\mathbf{K}^T\boldsymbol{C_\theta}\end{bmatrix}, \in \mathbb{R}^{2n\times n}\\
&\mathbf{Q}_\theta = -\begin{bmatrix} c_t(\mathbf{C}_\theta)^2 + c_n(\mathbf{S}_\theta)^2 & (c_t - c_n)\mathbf{S}_\theta\mathbf{C}_\theta\\ (c_t - c_n)\mathbf{S}_\theta\mathbf{C}_\theta & c_t(\mathbf{S}_\theta)^2 + c_n(\mathbf{C}_\theta)^2\end{bmatrix},\\
&\mathbf{V_x} = \mathbf{e}v_{cx}, \quad \mathbf{V_y} = \mathbf{e}v_{cy},\\
&\mathbf{I}_n = \begin{bmatrix}1 & & & \\ & \ddots & & \\ & & \ddots & \\ & & & 1\end{bmatrix}, \in \mathbb{R}^{n\times n}.
\end{aligned}$$

where, c_t and c_n are the tangential and normal drag force parameters, respectively. The nonlinear drag force ($\mathbf{f}_{NLd}$) and fluidic torque ($\boldsymbol{\tau}$) terms differentiate the nature of Eq. 2.11 from the dynamical equation of land-based snake robot as given in [55, 79].

2.3 Controller Design

In this work, the closed-loop control scheme is applied by using the virtual holonomic constraints (VHCs). The VHCs enforce time-independent (autonomous) relations for biological gait patterns on the configured robot. The control objective is to achieve a serpentine body shape to generate propulsive motion on the desired trajectory. It is achieved by using the following two tasks.

- The virtual holonomic constraints (VHCs) are made to regulate body shape of the snake robot.
- The proposed control scheme is design to track the head angle (θ_n) and relative tangential velocity ($v_{t,r}$) in the global frame with minimum chattering of a snake robot.

2.3.1 Biological Gait Pattern Generation using VHCs

Lateral undulation motion is the most common gait pattern of the snake robot for both on land and in the water environments. The equation of gait pattern generation can be given as follows [41].

$$\phi_{i,ref}(t) = \alpha \sin(\omega t + (i-1)\beta) + \phi_0(t), \tag{2.12}$$

In this, the parameter ω is joint frequency and t is time. The parameters ωt is replaced by a variable γ. In present case, the offset angle ϕ_0 affects all the joints unlike in [55, 86] where it affects only to the head link of a snake robot and the joint equation can be written as follows [57].

$$\phi_{i,ref}(t) = \alpha \sin(\gamma(t) + (i-1)\beta) + \phi_0(t), \tag{2.13}$$

where $i \in [1, \cdots, n-1]$, α is the amplitude of the joint angle (rad), $\gamma(t)$ is the input angle (rad), β (rad) is the phase angle between adjacent joint, ϕ_0 (rad) is the offset angle which makes possible to turn of the robot. The orientation and velocity control of the robot can be performed by adjusting these parameters. In this work, α and δ are kept constant positive values. The evaluation compensator of the state (γ and ϕ_0) is described below.

$$\ddot{\gamma}(t) = u_\gamma(t), \qquad \ddot{\phi}_0(t) = u_{\phi_0}(t). \tag{2.14}$$

The VHCs enforce an autonomous (time-independent) relation with involving configuration variables (γ, ϕ_0) which does not exist in a mechanical system but enforce via feedback. So, VHCs can be defined as "a relation specifying $n-1$ configuration variables in terms of a single angular configuration variable" [122]. So, the stabilize relation of VHCs can be written for n generalized coordinates and $n-1$ inputs as follows from Eq.2.13 by considering relation of $\phi_i = \theta_i - \theta_{i+1}$.

$$\theta_i - \theta_{i+1} = \alpha \sin(\gamma + (i-1)\beta) + \phi_0. \tag{2.15}$$

Let $\Phi_i(\gamma) = \alpha \sin(\gamma + (i-1)\beta)$. Therefore, the $\mathbf{\Phi}(\gamma) = [\Phi_1, \Phi_2, \cdots, \Phi_{n-1}]^T$. Using the Eq. 2.15 $\boldsymbol{\theta}$ can be expressed as

$$\boldsymbol{\theta} = \mathbf{e}\theta_n + \mathbf{H}\mathbf{\Phi}(\gamma) + \mathbf{H}\mathbf{b}\phi_0, \tag{2.16}$$

where, $\mathbf{b} = [1, \cdots\cdots, n-1]^T \in \mathbb{R}^{n-1}$, $\mathbf{H} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 0 & 1 & \cdots & 1 \\ & \cdots & \cdots & \\ 0 & 0 & \cdots & 1 \\ 0 & 0 & \cdots & 0 \end{bmatrix} \in \mathbb{R}^{n\times(n-1)}$. The VHCs from the above can be expressed by following [76] as

$$\mathbf{g}(\gamma, \phi_0, \boldsymbol{\theta}) = \mathbf{D}\boldsymbol{\theta} - \Phi(\gamma) - \mathbf{b}\phi_0 = 0. \tag{2.17}$$

$\mathbf{g}(\gamma, \phi_0, \boldsymbol{\theta})$ is an error output function for the system Eq. 2.11 with compensator Eq. 2.14 and it is needed to be zero. In most of the snake robot, servo motors are used which has in built PID controller. Therefore, the the torque equation can be written as follows.

$$\mathbf{u} = -k_p\mathbf{g}(\gamma, \phi_0, \boldsymbol{\theta}) - k_d\dot{\mathbf{g}}(\gamma, \phi_0, \boldsymbol{\theta})\dot{\gamma}\dot{\phi}_0\dot{\boldsymbol{\theta}} - k_i\int \mathbf{g}(\gamma, \phi_0, \boldsymbol{\theta})dt, \tag{2.18}$$

where, k_p, k_d, k_i are positive gain parameters. On the satisfaction of the VHCs, the reduced dynamical system corresponding to manifold can be written by substituting Eq.2.16 in Eq.2.11.

$$\ddot{\theta}_n = \Upsilon_1 + \Upsilon_2 u_\gamma + \Upsilon_3 u_{\phi_0}, \tag{2.19a}$$

$$\ddot{\mathbf{p}}_{xy} = \Upsilon_4\mathbf{v}_r + \Upsilon_5\dot{\theta}_n + \Upsilon_6\dot{\gamma} + \Upsilon_7\dot{\phi}_0 + \Upsilon_8. \tag{2.19b}$$

where,

$$\Upsilon_1 = -\frac{\mathbf{e}^T\mathbf{M}_\theta\mathbf{H}\ddot{\Phi}(\gamma)\dot{\gamma}^2}{\mathbf{e}^T\mathbf{M}_\theta\mathbf{e}} - \frac{\mathbf{e}^T}{\mathbf{e}^T\mathbf{M}_\theta\mathbf{e}}\{\mathbf{W}_\theta\dot{\boldsymbol{\theta}}^2 + \mathbf{\Lambda}_1\dot{\theta} + \mathbf{\Lambda}_2|\dot{\boldsymbol{\theta}}|\dot{\boldsymbol{\theta}} - l\mathbf{S}\mathbf{C}_\theta^T\mathbf{f}_d\},$$

$$\Upsilon_2 = -\frac{\mathbf{e}^T\mathbf{M}_\theta\mathbf{H}\dot{\Phi}(\gamma)}{\mathbf{e}^T\mathbf{M}_\theta\mathbf{e}},$$

$$\Upsilon_3 = -\frac{\mathbf{e}^T\mathbf{M}_\theta\mathbf{H}\mathbf{b}}{\mathbf{e}^T\mathbf{M}_\theta\mathbf{e}},$$

$$\Upsilon_4 = \frac{1}{nm}\mathbf{E}^T\mathbf{Q}_\theta\mathbf{E},$$

$$\Upsilon_5 = \frac{l}{nm}\mathbf{E}^T\mathbf{Q}_\theta\mathbf{S}\mathbf{C}_\theta\mathbf{e},$$

$$\Upsilon_6 = \frac{l}{nm}\mathbf{E}^T\mathbf{Q}_\theta\mathbf{S}\mathbf{C}_\theta\mathbf{H}\dot{\mathbf{\Phi}}(\gamma),$$

$$\Upsilon_7 = \frac{l}{nm}\mathbf{E}^T\mathbf{Q}_\theta\mathbf{S}\mathbf{C}_\theta\mathbf{H}\mathbf{b},$$
$$\Upsilon_8 = \frac{1}{nm}\mathbf{E}^T\mathbf{F}_{NLd}.$$

It is noted that reduced dynamical equation does not have control input term. This form is used for the designing of the control law. In the next subsection, the STSMC scheme is proposed for the motion control of a snake robot.

2.3.2 Super Twisting Sliding Mode Control

The nominal and uncertain part (ˆ over the cap) of Eq. 2.19 can be separated and written as follows.

$$\ddot{\theta}_n = \hat{\Upsilon}_1 + \Upsilon_2 u_\gamma + \Upsilon_3 u_{\phi 0} + \tilde{\Upsilon}_{\theta_n}, \tag{2.20a}$$
$$\ddot{\mathbf{p}}_{xy} = \hat{\Upsilon}_4\mathbf{v}_r + \hat{\Upsilon}_5\dot{\theta}_n + \hat{\Upsilon}_6\dot{\gamma} + \hat{\Upsilon}_7\dot{\phi}_0 + \hat{\Upsilon}_8 + \tilde{\Upsilon}_p, \tag{2.20b}$$

where, $\tilde{\Upsilon}_{\theta_n}$ and $\tilde{\Upsilon}_p$ represents the lumped uncertainties and disturbances in the head angle and position due different environmental condition. Mathematical representation of the head-link orientation (θ_n) and tangential velocity (v_t) tracking are given below.

$$\theta_n \to \theta_{ref}, \quad v_{t,r} \to v_{ref}.$$

The error function of the head angle and tangential velocity can be written as

$$\tilde{\theta}_n = \theta_n - \theta_{ref}, \quad \tilde{v}_{t,r} = v_{t,r} - v_{ref}$$

and error dynamics can be written as

$$\ddot{\tilde{\theta}}_n = \hat{\Upsilon}_1 - \ddot{\theta}_{ref} + \Upsilon_2 u_\gamma + \Upsilon_3 u_{\phi_0} + \tilde{\Upsilon}_{\theta_n}, \tag{2.21a}$$
$$\begin{aligned}\dot{\tilde{v}}_{t,r} = {} & u_{\theta_n}^T\hat{\Upsilon}_4 u_{\theta_n} v_{t,r} + u_{\theta_n}^T\hat{\Upsilon}_4 v_{\theta_n} v_{n,r} + \dot{\theta}_n v_{n,r} + u_{\theta_n}^T\hat{\Upsilon}_5\dot{\theta}_n + u_{\theta_n}^T\hat{\Upsilon}_6\dot{\gamma} \\ & + u_{\theta_n}^T\hat{\Upsilon}_7\dot{\phi}_0 + u_{\theta_n}^T\hat{\Upsilon}_8 + u_{\theta_n}^T\tilde{\Upsilon}_p - \dot{v}_{ref},\end{aligned} \tag{2.21b}$$
$$\begin{aligned}\dot{v}_{n,r} = {} & v_{\theta_n}^T\hat{\Upsilon}_4 u_{\theta_n} v_{t,r} + v_{\theta_n}^T\hat{\Upsilon}_4 v_{\theta_n} v_{n,r} - \dot{\theta}_n v_{n,r} + v_{\theta_n}^T\hat{\Upsilon}_5\dot{\theta}_n + v_{\theta_n}^T\hat{\Upsilon}_6\dot{\gamma} \\ & + v_{\theta_n}^T\hat{\Upsilon}_7\dot{\phi}_0 + v_{\theta_n}^T\hat{\Upsilon}_8 + v_{\theta_n}^T\tilde{\Upsilon}_p,\end{aligned} \tag{2.21c}$$
$$\ddot{\gamma} = u_\gamma, \tag{2.21d}$$
$$\ddot{\phi}_0 = u_{\phi_0}, \tag{2.21e}$$

where, $u_\theta = [\cos\theta_n, \sin\theta_n]^T$ and $v_\theta = [-\sin\theta_n, \cos\theta_n]^T$. The state vector can be written as $\bar{\xi} = [\tilde{\theta}_n, \dot{\tilde{\theta}}_n, \tilde{v}_{t,r}, v_{n,r}, \gamma, \dot{\gamma}, \phi_0, \dot{\phi}_0]^T$.

The sliding surface is selected using reduced ordered dynamical system [123] as follows.

$$\boldsymbol{\sigma}(\bar{\xi}) = \begin{bmatrix} \sigma_1(\bar{\xi}) \\ \sigma_2(\bar{\xi}) \end{bmatrix} = \begin{bmatrix} \dot{\tilde{\theta}}_n + k_n \tilde{\theta}_n \\ \dot{\gamma} + k_v \tilde{v}_{t,r} \end{bmatrix} \tag{2.22}$$

where, $k_n, k_v > 0$ are the design parameters. The derivative of the sliding surface is given in the following equation.

$$\dot{\sigma}_1(\bar{\xi}) = \ddot{\tilde{\theta}}_n + k_n \dot{\tilde{\theta}}_n, \tag{2.23a}$$
$$\dot{\sigma}_2(\bar{\xi}) = \ddot{\gamma} + k_v \dot{\tilde{v}}_{t,r}. \tag{2.23b}$$

Therefore, Eq.2.23 can be rewritten by substituting Eqs. 2.21a-2.21b as follows.

$$\dot{\sigma}_1(\bar{\xi}) = \hat{\Upsilon}_1 - \ddot{\theta}_{ref} + \ddot{\theta}_{ref} + \Upsilon_2 u_\gamma + \Upsilon_3 u_{\phi_0} + k_n \dot{\tilde{\theta}}_n + \tilde{\Upsilon}_{\theta_n}, \tag{2.24a}$$
$$\begin{aligned}\dot{\sigma}_2(\bar{\xi}) = u_\gamma + k_v (& u_{\theta_n}^T \hat{\Upsilon}_4 u_{\theta_n} v_{t,r} + u_{\theta_n}^T \hat{\Upsilon}_4 v_{\theta_n} v_{n,r} + \dot{\theta}_n v_{n,r} + u_{\theta_n}^T \hat{\Upsilon}_5 \dot{\theta}_n \\ & + u_{\theta_n}^T \hat{\Upsilon}_6 \dot{\gamma} + u_{\theta_n}^T \hat{\Upsilon}_7 \dot{\phi}_0 + u_{\theta_n}^T \hat{\Upsilon}_8 - \dot{v}_{ref}) + u_{\theta_n}^T \tilde{\Upsilon}_p.\end{aligned} \tag{2.24b}$$

Now, Eq. 2.24 can be written in the simplified form as follows.

$$\dot{\boldsymbol{\sigma}}(\bar{\xi}) = \mathbf{f}_\sigma + \mathbf{g}_\sigma \mathbf{u}_\sigma + \bar{\mathbf{f}}_\sigma, \tag{2.25}$$

where,

$$\mathbf{f}_\sigma = \begin{bmatrix} R_1 \\ R_2 \end{bmatrix}, \mathbf{g}_\sigma = \begin{bmatrix} \Upsilon_2 & \Upsilon_3 \\ 1 & 0 \end{bmatrix}, \mathbf{u}_\sigma = \begin{bmatrix} u_\gamma \\ u_{\phi_0} \end{bmatrix}, \bar{\mathbf{f}}_\sigma = \begin{bmatrix} \tilde{\Upsilon}_{\theta_n} \\ u_{\theta_n}^T \tilde{\Upsilon}_p \end{bmatrix}.$$

$$\begin{aligned} R_1 &= \hat{\Upsilon}_1 - \ddot{\theta}_{ref} + k_n \dot{\tilde{\theta}}_n, \\ R_2 &= k_v (u_{\theta_n}^T \hat{\Upsilon}_4 u_{\theta_n} v_{t,r} + u_{\theta_n}^T \hat{\Upsilon}_4 v_{\theta_n} v_{n,r} + \dot{\theta}_n v_{n,r} + u_{\theta_n}^T \hat{\Upsilon}_5 \dot{\theta}_n + u_{\theta_n}^T \hat{\Upsilon}_6 \dot{\gamma} + u_{\theta_n}^T \hat{\Upsilon}_7 \dot{\phi}_0 \\ &\quad + u_{\theta_n}^T \hat{\Upsilon}_8 - \dot{v}_{ref}). \end{aligned}$$

The assumption is considered that the upper bound on the uncertainty in a system are known priori and represented as $\| \bar{\mathbf{f}}_\sigma \| \leq \Omega$. The control law can be described in two parts as mentioned in [55]. The first part is *'equivalent control'* law which stabilizes system on sliding surface in the absence of external disturbances [85]. The second part ensures the finite time reachability [124]. It is called '*switching control*' law. In most of the control approaches, the switching control law is responsible for chattering in the system. Therefore, it is important to choose suitable switching law. The equivalent control law for the given sliding surface can be derived by equating $\dot{\boldsymbol{\sigma}}(\bar{\xi}) = \bar{\mathbf{f}}_\sigma = 0$ and represented as

$$\mathbf{u}_{eq} = -\mathbf{g}_\sigma^{-1}\mathbf{f}_\sigma. \tag{2.26}$$

The switching control law is given as below.

$$\mathbf{u}_{sw} = -\,\mathbf{g}_\sigma^{-1}\Big(k_1 sgn(\boldsymbol{\sigma})\|\boldsymbol{\sigma}\|^{\frac{1}{2}} + k_2\boldsymbol{\sigma}\|\boldsymbol{\sigma}\|^{\frac{3}{2}} + \int_0^t k_3 sgn(\boldsymbol{\sigma})\ dt\Big) \tag{2.27}$$

where, k_1, k_2 and k_3 are the controller design parameters. The final $\mathbf{u}_\sigma$ would be using the equivalent control law ($\mathbf{u}_{eq}$) and switching control law ($\mathbf{u}_{sw}$) as follows.

$$\mathbf{u}_\sigma = \mathbf{u}_{eq} + \mathbf{u}_{sw} \tag{2.28}$$

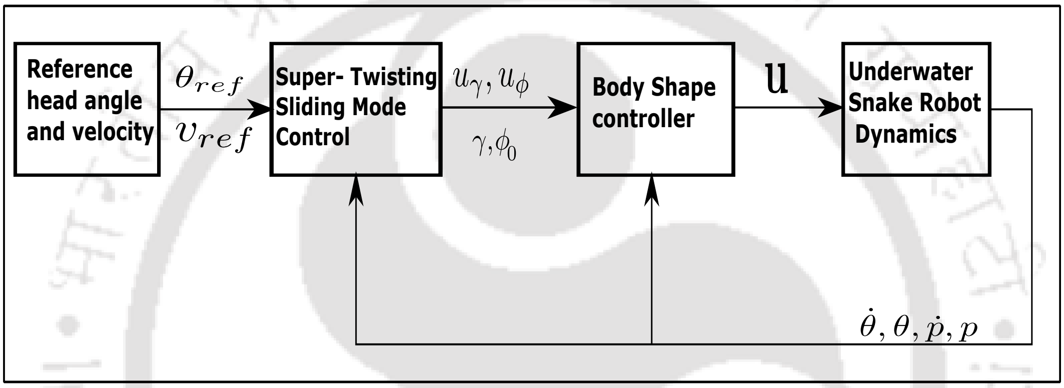

Figure 2.3: Block diagram of the control law

The block diagram of the controller implementation is shown in Fig. 2.3. Stability of the snake robot dynamics using proposed control scheme is given by the Lyapunov stability analysis.

2.3.3 Lyapunov Stability

The Lyapunov function is considered as below.

$$V = \frac{1}{2}\boldsymbol{\sigma}^T\boldsymbol{\sigma} \tag{2.29}$$

The derivative of Eq. 2.29 and substitute Eq. 2.26-2.27 in that the Lyapunov function becomes as shown following.

$$\begin{aligned}
\dot{V} &= \boldsymbol{\sigma}^T\dot{\boldsymbol{\sigma}} \\
&= \boldsymbol{\sigma}\left(\mathbf{f}_\sigma + \mathbf{g}_\sigma\mathbf{u}_\sigma + \bar{\mathbf{f}}_\sigma\right) \\
&= \boldsymbol{\sigma}\Big(-k_1 sgn(\boldsymbol{\sigma})\|\boldsymbol{\sigma}\|^{\frac{1}{2}} - k_2\boldsymbol{\sigma}\|\boldsymbol{\sigma}\|^{\frac{3}{2}} - \int_0^t k_3 sgn(\boldsymbol{\sigma})dt + \bar{\mathbf{f}}_\sigma\Big)
\end{aligned}$$

$$\leq -k_1\|\boldsymbol{\sigma}\|^{\frac{3}{2}} - k_2\|\boldsymbol{\sigma}\|^{\frac{7}{2}} - \|\boldsymbol{\sigma}\| \int_0^t k_3 dt + \|\boldsymbol{\sigma}\|\|\bar{\mathbf{f}}_\sigma\|$$

By considering the assumption$\|\bar{\mathbf{f}}_\sigma\| \leq \Omega \leq \dot{\delta}$, the equation can be modified as

$$\leq -k_1\|\boldsymbol{\sigma}\|^{\frac{3}{2}} - k_2\|\boldsymbol{\sigma}\|^{\frac{7}{2}} - \|\boldsymbol{\sigma}\| \int_0^t k_3 dt + \|\boldsymbol{\sigma}\| \int_0^t \dot{\delta} dt$$

$$\leq -k_1\|\boldsymbol{\sigma}\|^{\frac{3}{2}} - k_2\|\boldsymbol{\sigma}\|^{\frac{7}{2}} - \left(\|\boldsymbol{\sigma}\| \int_0^t k_3 dt - \|\boldsymbol{\sigma}\| \int_0^t \dot{\delta} dt\right)$$

$$\leq 0 \quad (\because k_3 > \Omega)$$

Hence, Lyapunov stability has been proven by considered $k_1, k_2, k_3 > 0$.

2.3.4 Existing Control Schemes

The proposed control scheme is compared with the existing control schemes which were used in land based snake robot. Here, the existing control approaches are given. The switching law of the sliding mode control (SMC) can be expressed as per [85].

$$\mathbf{u}_{sw} = -\eta \mathbf{g}_\sigma^{-1} sgn(\boldsymbol{\sigma}), \tag{2.30}$$

where, design parameter $\eta > 0$.
The adaptive law for switching gain by following [125] for the ASMC is described as follows.

$$\dot{\eta} = \begin{cases} \bar{\eta} \parallel \boldsymbol{\sigma} \parallel sgn(\parallel \boldsymbol{\sigma} \parallel - \epsilon) & if \quad \eta > \mu \\ \mu \quad if \quad \eta \leq \mu \end{cases} \tag{2.31}$$

2.4 Results and Discussion

The mass $m = 0.9047$ kg and length $2l = 0.18$ m of each link with radius of $r = 0.04$ m of the snake robot are considered for the numerical simulation and this makes snake robot neutral buoyant in underwater environment. The CAD model of a snake robot is shown in Fig. 2.4.

The snake robot moves in various environments such as a lake, pond, river, etc. Many impurities are present in this water. It directly reflects in drag forces and fluidic torque while the motion of any object in these water bodies. It is difficult to find out exact nature of drag parameters (c_t, c_n) and rotational damping parameters (λ_1, λ_2) for the snake robot using conventional formulae [41]. Hence, the overall uncertainty in these parameters is considered a problem for the manoeuvring control of the snake robot in the water environment. The friction uncertainties were considered for the land based

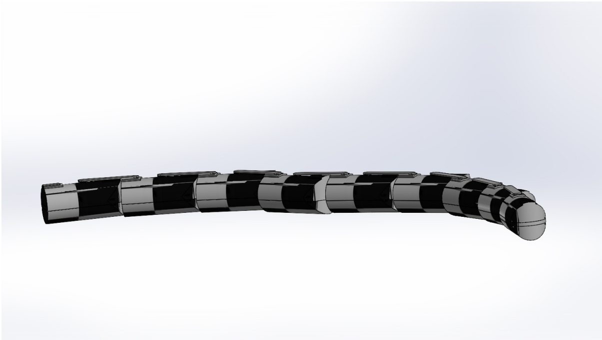

Figure 2.4: CAD model of the snake robot

snake robot similarly [55]. Here, the time-varying uncertainties with random variation are considered. The following equations are derived using curve fitting tool from the graphical representation (shown in Fig. 2.5) of the uncertainties.
For $t \leq t_1$,

$$\begin{aligned}
c_t &= 12.02\sin(0.0066t + 0.2329) + 22.8\sin(0.0114t + 2.339) \\
&\quad + 17\sin(0.0126t + 5.231) + 0.5rand(1), \\
c_n &= 14.554\sin(0.0018t + 1.28) + 0.863\sin(0.0170t - 1.833)) + 0.5rand(1), \\
\lambda_1 &= 0.0268\sin(0.0048t - 0.0678) + 0.012\sin(0.0076t + 2.115) + 0.5rand(1), \\
\lambda_2 &= 1.165 \times 10^{-3}\sin(0.0061t - 0.60) + 6.77 \times 10^{-4}\sin(0.0087t + 1.56) \\
&\quad + 1.13 \times 10^{-5}\sin(0.0276t + 3.09) + 0.25rand(1),
\end{aligned}$$

and for more than $t > t_1$,

$$c_t = 4.101, \quad c_n = 16.36, \quad \lambda_1 = 0.5, \quad \lambda_2 = 0.25.$$

These time depending uncertainties are considered for the snake robot dynamics in Eq. 2.11. However, in the controller formulation the nominal values $(\hat{c}_t, \hat{c}_n, \hat{\lambda}_1, \hat{\lambda}_2)$ of these uncertain parameters are considered. The graphical representation of the uncertain parameters is shown in Fig. 2.5.

The sliding surface parameters (k_n, k_v) and estimated values of the uncertainties $(\hat{c}_t, \hat{c}_n, \hat{\lambda}_1, \hat{\lambda}_2)$ are kept same for the comparative performance study as shown in Table- 2.1. The switching controller parameters are shown in Table- 2.2. As per the Lyapunov stability analysis, the η and k_3 look a like same term on the estimation lumped disturbance and uncertainties. Therefore, both are kept same for the simulation analysis. The other parameters of the PID controller and reference inputs are shown Table- 2.3.

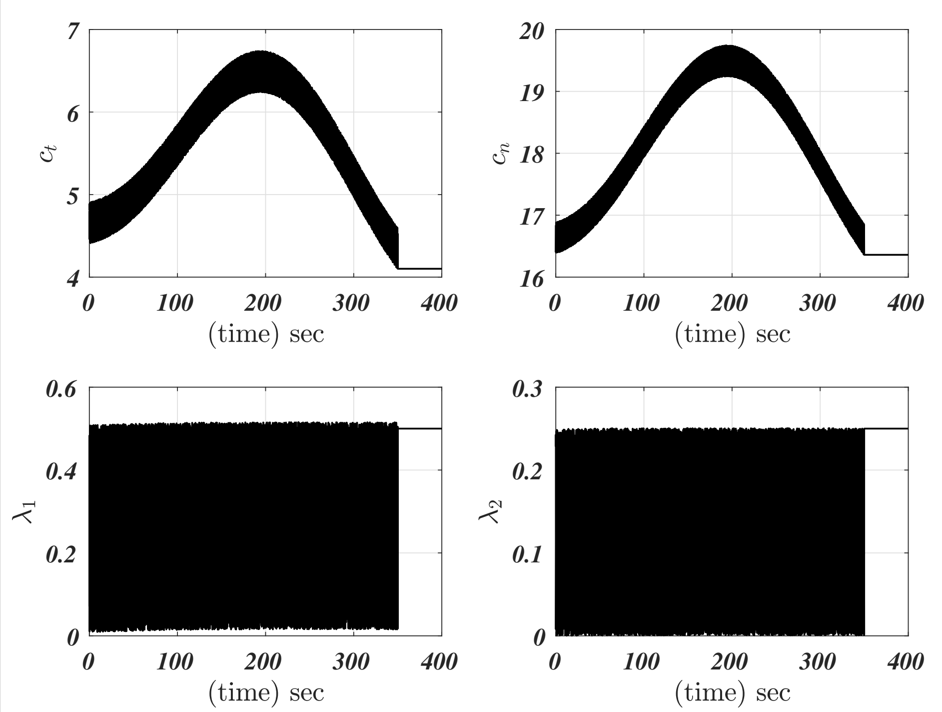

Figure 2.5: Uncertainty in the dynamical system

Table 2.1: Sliding surface and nominal parameters

Symbol	k_n	k_v	$\hat{c}_t$	$\hat{c}_n$	$\hat{\lambda}_1$	$\hat{\lambda}_2$	t_1
Value	10	12.5	9	22.5	2.5	2	350

Table 2.2: Switching controller parameters

Switching control law	Parameters
STSMC	$k_1 = 10, k_2 = 10, k_3 = 12$
SMC	$\eta = 12$
ASMC	$\mu = 0.5,\ \epsilon = 4\eta T_e,\ T_e = 0.0021$

Table 2.3: Parameters of Reference input and PID controller

k_p	k_d	k_i	v_{cx}	v_{cy}	θ_{ref}	v_{ref}	α	β
20	5	10	0	0	$-\frac{\pi}{4}$	0.05	$\frac{30\pi}{180}$	$\frac{50\pi}{180}$

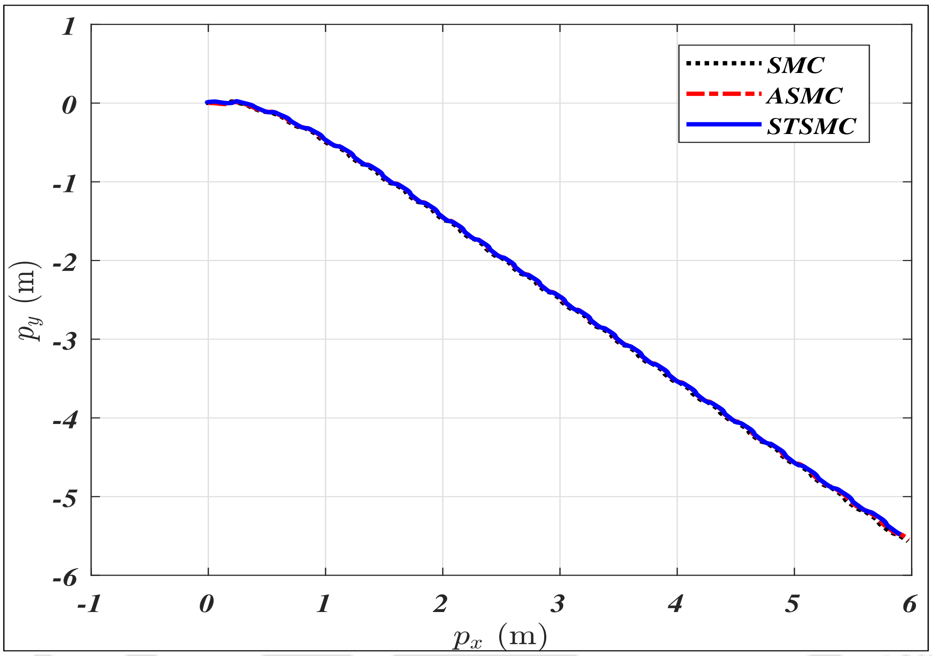

Figure 2.6: Centre of mass position of the robot

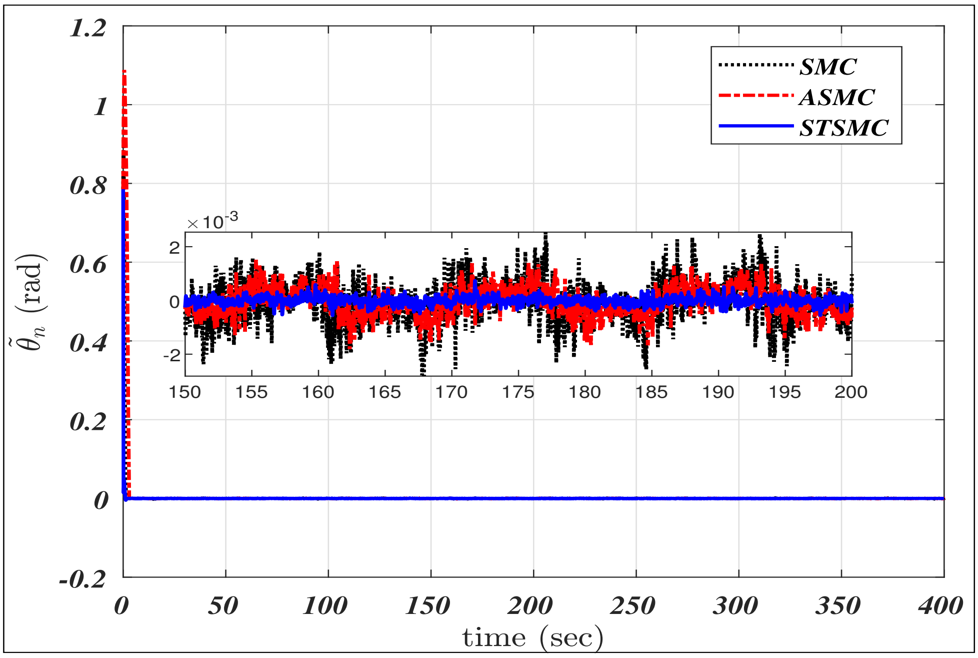

Figure 2.7: Error in head angle tracking

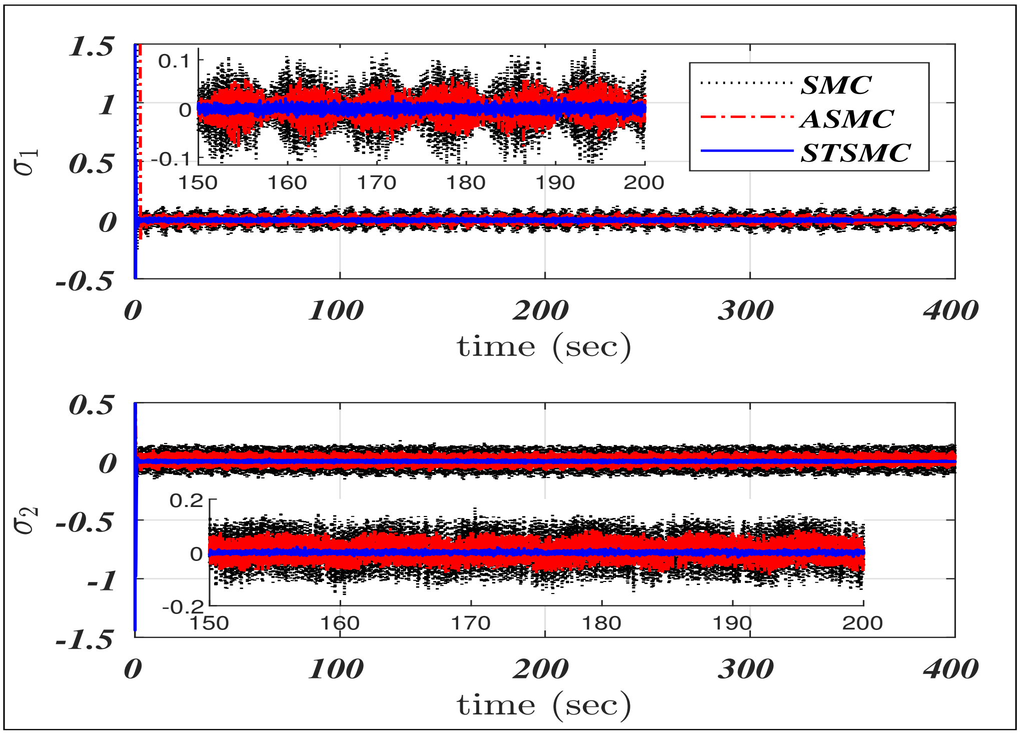

Figure 2.8: Sliding Surface

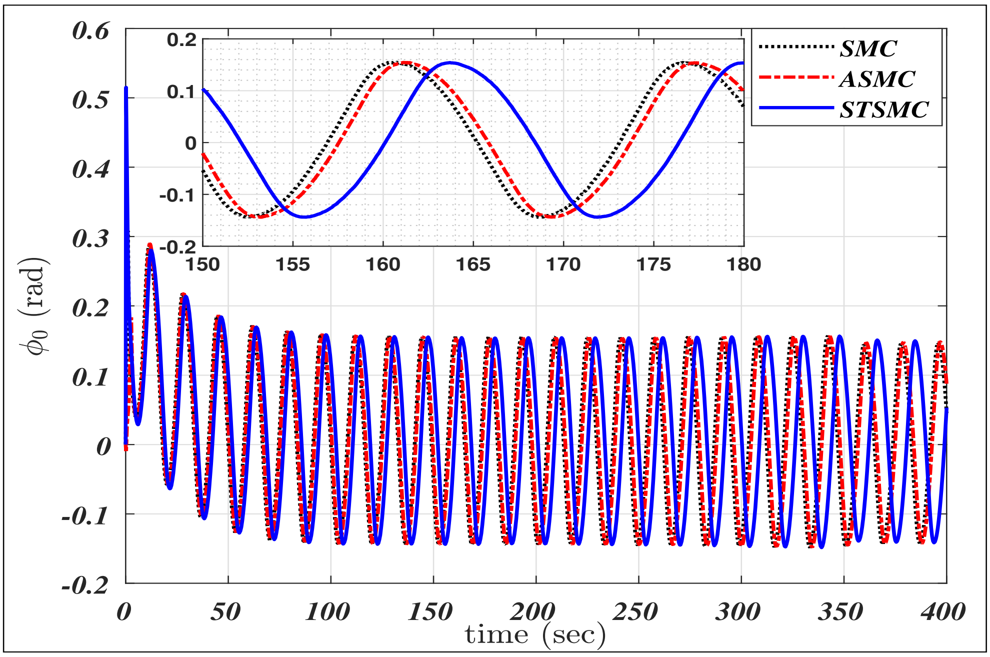

Figure 2.9: Offset angle

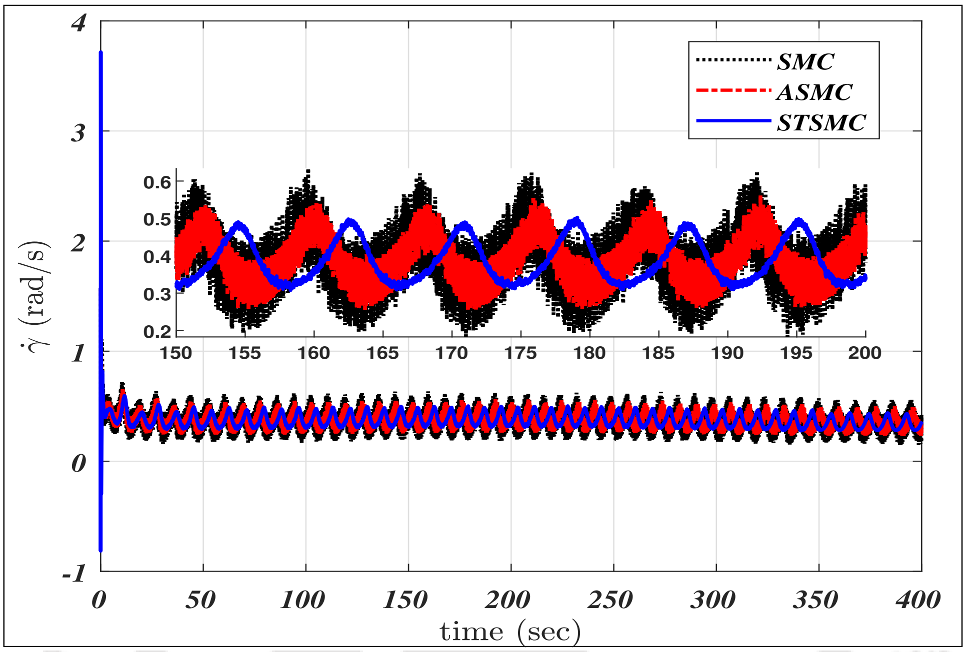

Figure 2.10: Input Frequency

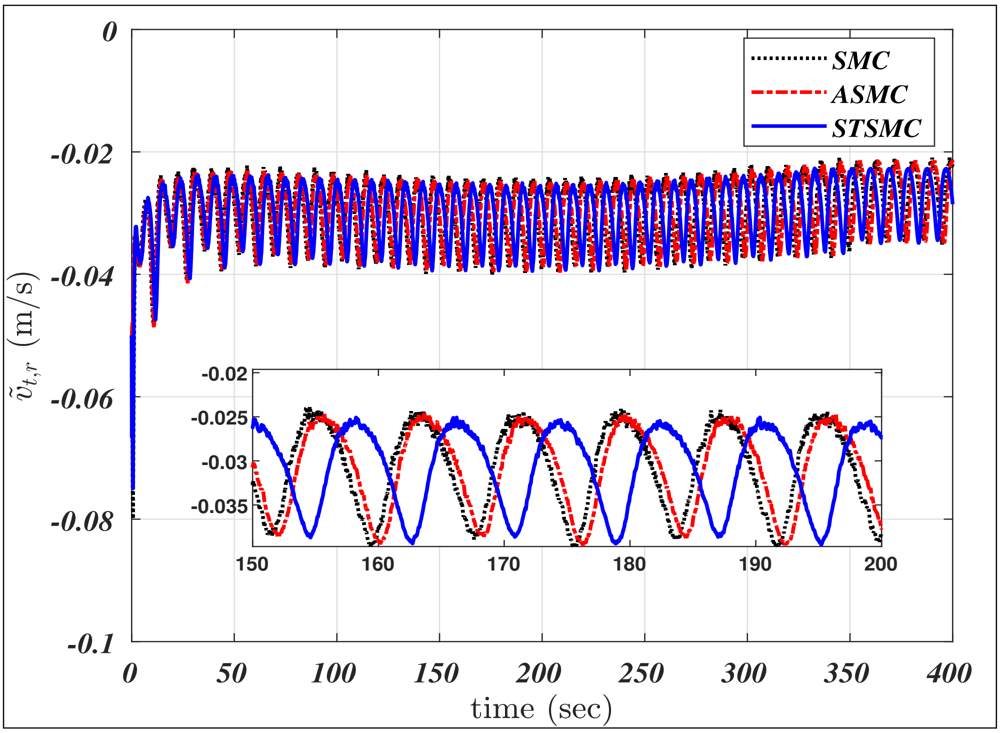

Figure 2.11: Error in tangential velocity

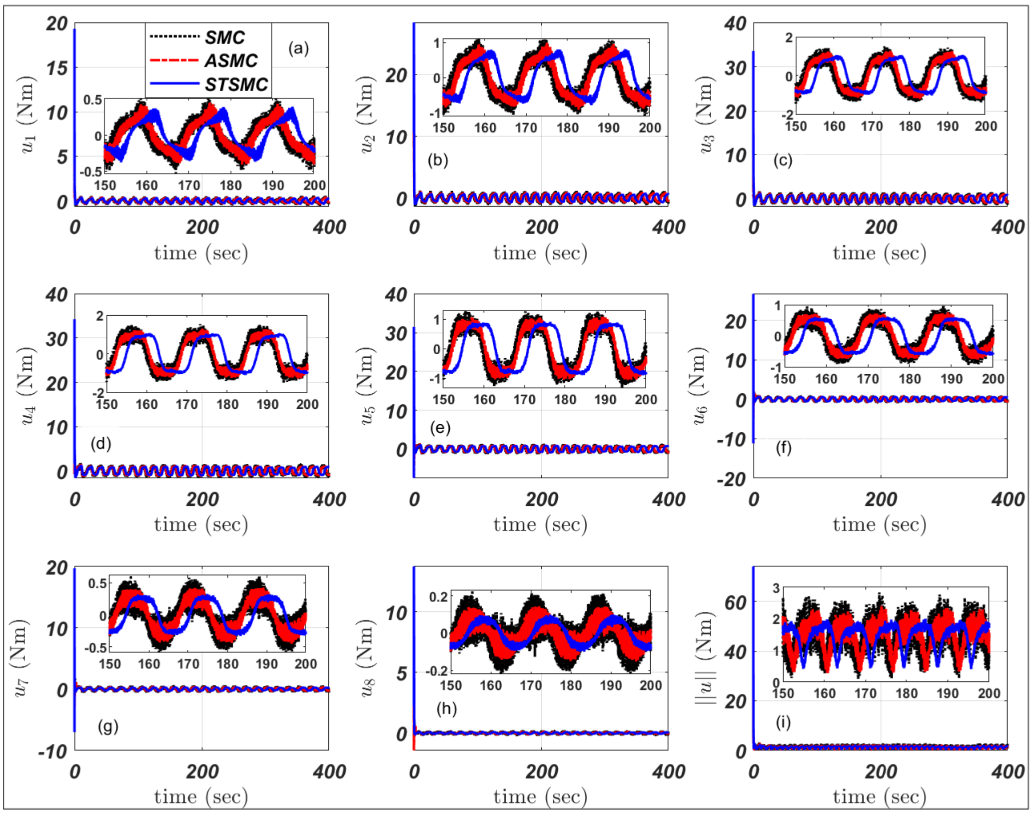

Figure 2.12: (a to h) Torque on each joint, (i) Norm of the torque

The marginal increment of error in velocity has been observed due to this the robot covered marginal less distance using proposed controller as shown in Fig. 2.6. The head angle tracking error with time is shown in Fig. 2.7 for SMC, ASMC and STSMC. Here, the error and chattering in response seem improved using the STSMC. The chattering effect of sliding surfaces are reduced by using STSMC compare to SMC and ASMC that can be seen in the Fig. 2.8. The variation in offset angle ϕ_0 with time is important parameter to guide the direction of robot which is also output function of the sliding surface that is shown in Fig. 2.9. In this, using three different SMC schemes, though the amplitude is found to be same, some phase difference is observed. Another output of the robust control scheme is the input frequency that decide the joint speed of the robot. The significant reduction in chattering is observed using the STSMC as shown in Fig. 2.10. In Fig. 2.11, the error in tangential velocity is described. The significant improvement in chattering effect is observed which is more important. Figure 2.12 (a:h) shows variation of torque with time for each joint torque and Fig. 2.12i shows the norm of all torques. From this figure, the significant reduction in chattering is observed by using proposed

STSMC.

The quantification of chattering indicator is explained in the next subsection. This quantification helps to identify the comparatively better controller for the snake robot motion in uncertain underwater environment.

2.4.1 Chattering Indicator

The chattering indicator (CI) can be defined [126] for variable of S for given time T as follows.

$$\mathrm{CI} = \frac{1}{\sqrt{T}}\sqrt{\sum_{i=1}^{T}(\dot{S}_i)^2}. \tag{2.32}$$

Table 2.4: Chattering Indicator

Parameters (S)	SMC	ASMC	STSMC
$\tilde{\theta}_n$	0.0327	0.0178	0.0053
$\tilde{v}_{t,r}$	0.0120	0.0084	0.0073
$\dot{\gamma}$	12.1390	6.0243	0.7090
σ_1	4.9728	2.6044	0.4786
σ_2	12.1056	6.0073	0.7104
u_1	13.3560	7.4766	4.0772
u_2	27.9723	14.3942	4.8397
u_3	31.3792	15.6058	3.5364
u_4	26.7821	12.9759	2.3956
u_5	19.4693	9.6271	2.3602
u_6	17.2201	8.7977	1.9966
u_7	18.1239	9.0399	1.6692
u_8	9.9172	4.9351	1.1970
$\|\|\mathbf{u}\|\|$	46.0283	22.9916	5.2887

This quantification of the CI rationalise the effectiveness of proposed control method. The significant performance improvement in chattering effect has been observed by using STSMC as shown in Table- 2.4. A very less chattering effect can be seen on sliding surface using proposed STSMC algorithm which makes system performance very robust and helps to get high accuracy operation tasks in uncertain environment.

2.5 Summary

In this chapter, the super twisting sliding mode control is used for tracking the head link angle and the tangential velocity of a snake robot in the uncertain underwater environment for given reference input. The uncertainties are considered in terms of drag force and fluidic torque. The body shape is stabilized using VHCs feedback law. The STSMC is compared with SMC and ASMC. The effectiveness of proposed control method has been calculated by chattering indicator. Almost chattering free response has been observed using the STSMC as per quantitative results of the proposed controller. The performance of STSMC is observed to be better than SMC and ASMC.

3

Motion Control of Planar Snake Robot using Adaptive Neural Network Sliding Mode Control

Contents

3.1 Introduction

In this chapter, the adaptive radial basis function neural network sliding mode control (RBF-NN SMC) is proposed for the motion control of the underwater snake robot by estimating uncertain environments. Generally, a SMC scheme is designed with lumped uncertainties and disturbances to deal with uncertain environmental conditions as mentioned in the chapter 2. In this chapter, the environmental uncertainties are estimated using RBF-NN. It estimates the system dynamics by using the state feedback of the system without prior knowledge of the uncertainties. The Lyapunov stability analysis is performed to identify the controller design parameters that stabilise the dynamical system. The effectiveness of the proposed control method results in a reduction in the chattering effect and error response compared to the existing adaptive SMC method. Moreover, the water current velocity and its estimator is considered in this work where the snake robot follows the reference path. The motion dynamics equations of the snake robot is same as mention in the chapter 2. Therefore, it is not explained in this chapter.

This chapter is organized as follows. The designing of adptive RBF sliding mode control scheme is explained in the section 3.2. The results are discussed in the section 3.3 and chapter is summarized in the section 3.4.

3.2 Controller Design

The controller is designed same as VHCs based approach based on the reduced ordered dynamics. The error function is same as in the chapter 2. Here, the same sliding surface is designed with same uncertainties considered in the previous chapter. The application of the RBF-NN with SMC and its results differentiate this chapter from the chapter 2. The procedure of a controller design is described as follows.

3.2.1 Simplified Sliding Surface

The sliding surface consists of the error in terms of states. The sliding surface is selected using a reduced ordered dynamical system [123] as follows.

$$\boldsymbol{\sigma}(\bar{\xi}) = \begin{bmatrix} \sigma_1(\bar{\xi}) \\ \sigma_2(\bar{\xi}) \end{bmatrix} = \begin{bmatrix} \dot{\tilde{\theta}}_n + k_n \tilde{\theta}_n \\ \dot{\gamma} + k_v \tilde{v}_{t,r} \end{bmatrix} \tag{3.1}$$

[2]This chapter is part of publication: B. M. Patel, J. Narayan & S. K. Dwivedy (2024). Manoeuvring planar snake robot in uncertain underwater condition using adaptive neural network sliding mode control. International Journal of Dynamics and Control, 1-19.

where, $k_n, k_v > 0$ are the design parameters. The derivative of the Eq: 3.1 is used to obtain the reaching law for sliding surface as following equation.

$$\dot{\sigma}_1(\bar{\xi}) = \ddot{\bar{\theta}}_n + k_n\dot{\bar{\theta}}_n, \tag{3.2a}$$

$$\dot{\sigma}_2(\bar{\xi}) = \ddot{\gamma} + k_v\dot{\bar{v}}_{t,r}. \tag{3.2b}$$

Therefore, Eq. 3.2 can be written as

$$\dot{\sigma}_1(\bar{\xi}) = \hat{\Upsilon}_1 - \ddot{\theta}_{ref} + \Upsilon_2 u_\gamma + \Upsilon_3 u_{\phi_0} + k_n\dot{\bar{\theta}}_n + \tilde{\Upsilon}_{\theta_n}, \tag{3.3a}$$

$$\begin{aligned}\dot{\sigma}_2(\bar{\xi}) &= u_\gamma + k_v(u_{\theta_n}^T\hat{\Upsilon}_4 u_{\theta_n} v_{t,r} + u_{\theta_n}^T\hat{\Upsilon}_4 v_{\theta_n} v_{n,r} + \dot{\theta}_n v_{n,r} + u_{\theta_n}^T\hat{\Upsilon}_5\dot{\theta}_n \\ &\quad + u_{\theta_n}^T\hat{\Upsilon}_6\dot{\gamma} + u_{\theta_n}^T\hat{\Upsilon}_7\dot{\phi}_0 + u_{\theta_n}^T\hat{\Upsilon}_8 - \dot{v}_{ref}) + u_{\theta_n}^T\tilde{\Upsilon}_p.\end{aligned} \tag{3.3b}$$

Now, Eq. 3.3 can be represented in the simplified form as

$$\dot{\boldsymbol{\sigma}}(\bar{\xi}) = \hat{\mathbf{f}}_\sigma + \mathbf{f}_e + \mathbf{g}_\sigma\mathbf{u}_\sigma + \bar{\mathbf{f}}_\sigma, \tag{3.4}$$

where,

$$\hat{\mathbf{f}}_\sigma = \begin{bmatrix} R_1 \\ R_2 \end{bmatrix}, \mathbf{f}_e = \begin{bmatrix} -\ddot{\theta}_{ref} + k_n\dot{\bar{\theta}}_n \\ -k_v\dot{v}_{ref} \end{bmatrix}, \mathbf{g}_\sigma = \begin{bmatrix} \Upsilon_2 & \Upsilon_3 \\ 1 & 0 \end{bmatrix},$$

$$\mathbf{u}_\sigma = \begin{bmatrix} u_\gamma \\ u_{\phi_0} \end{bmatrix}, \bar{\mathbf{f}}_\sigma = \begin{bmatrix} \tilde{\Upsilon}_{\theta_n} \\ u_{\theta_n}^T\tilde{\Upsilon}_p \end{bmatrix}.$$

$$\begin{aligned} R_1 &= \hat{\Upsilon}_1, \\ R_2 &= k_v(u_{\theta_n}^T\hat{\Upsilon}_4 u_{\theta_n} v_{t,r} + u_{\theta_n}^T\hat{\Upsilon}_4 v_{\theta_n} v_{n,r} + \dot{\theta}_n v_{n,r} + u_{\theta_n}^T\hat{\Upsilon}_5\dot{\theta}_n + u_{\theta_n}^T\hat{\Upsilon}_6\dot{\gamma} \\ &\quad + u_{\theta_n}^T\hat{\Upsilon}_7\dot{\phi}_0 + u_{\theta_n}^T\hat{\Upsilon}_8).\end{aligned}$$

As shown in Eq. 3.4, the uncertain value $\hat{\mathbf{f}}_\sigma$ is to be estimated using RBF neural network method.

3.2.1.1 Adaptive RBF Neural Network

The uncertainties in model dynamics are estimated using the radial basis function neural network (RBF-NN). The neural network inputs are decided based on the link-angle angular position and velocity, CM position and velocity of the snake robot. The feedback parameters are measured by the sensors. The modification in the RBF neural network algorithm is done based on the coupled dynamical sliding surface which make the weights of the neural network adaptive in nature. The mathematical expression of the Gaussian based activation function (h_j) RBF method for input $\boldsymbol{\zeta} = [\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}, \mathbf{p}_{xy}, \dot{\mathbf{p}}_{xy}]$ is given in Eq. 3.5. The neural network is designed such that the information required to estimate uncertain parameters could be sufficient.

$$h_j = exp\left(\frac{-\|(\zeta_j^T - \underline{\mathbf{C}}(:,j))\|^2}{2\underline{\mathrm{b}}^2}\right) \tag{3.5}$$

The uncertain parameters are obtained by the following equation.

$$\hat{\mathbf{f}}_\sigma = [\underline{\mathbf{W}}^1, \underline{\mathbf{W}}^2]^T \mathbf{h} \tag{3.6}$$

where, $\underline{\mathbf{C}}(:,j)$, $\underline{\mathrm{b}}$ and $[\underline{\mathbf{W}}^1, \underline{\mathbf{W}}^2]^T$ are centre vector, width and weights, respectively. These parameters are made adaptive based on the sliding surface and the input. The adaption law is decided based on the sliding surface of the coupled dynamical system as shown in Eq. 3.4. The neural network block diagram is shown in Fig. 3.1. This adaption

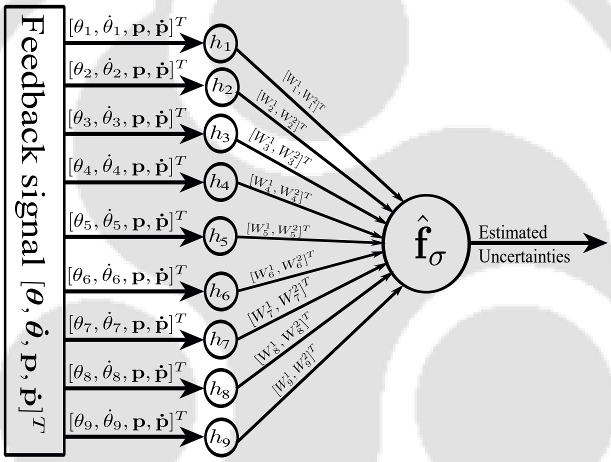

Figure 3.1: RBF neural network diagram

algorithm for the given parameters is explained in the Algorithm 3.1. The reason behind

Algorithm 3.1 Neural Network Algorithm

1: **for** j = 1:9 **do**
2: $\quad n = j + 9$
3: $\quad \zeta_{new} = \left[\boldsymbol{\zeta}([j,n], \mathbf{p}_{xy}, \dot{\mathbf{p}}_{xy})\right]^T$ ▷ Input of the neural network
4: $\quad \underline{\mathbf{C}}(:,j)_{new} = \underline{\mathbf{C}}(:,j)_{old} + \|\boldsymbol{\sigma}\|(\zeta_{new} - \underline{\mathbf{C}}(:,j)_{old})$ ▷ Center-line adaption
5: $\quad \underline{\mathrm{b}} = \frac{max(abs(\zeta_{new} - \underline{\mathbf{C}}(:,j)_{new}))}{\sqrt{6}}$ ▷ Width of the distribution
6: $\quad h(j) = exp\left(\frac{-\|\zeta_{new} - \underline{\mathbf{C}}(:,j)_{new}\|^2}{2\underline{\mathrm{b}}^2}\right)$ ▷ Gaussian activation function
7: **end for**
8: $\underline{\mathbf{C}}_{old} = \underline{\mathbf{C}}_{new}$

the adaptation of the centre vector $\underline{\mathrm{C}}(:,j)$ and width (b_j) of the Gaussian function is that one can not define precisely the centre vector for the feedback data in a mobile robot. The feedback results may vary due to environmental uncertainties, which may change the width of the feedback data. Therefore, the continuous update in the centre vector and width of data may give better results. The adaption law for the weight matrix for the coupled equation is proposed in the following equation.

$$\begin{bmatrix} \underline{\dot{\mathbf{W}}}^{\mathbf{1}} \\ \underline{\dot{\mathbf{W}}}^{\mathbf{2}} \end{bmatrix} = P_1 \boldsymbol{\sigma} \mathbf{h}^T + P_2 \|\boldsymbol{\sigma}\|^{\frac{4}{3}} \boldsymbol{\sigma} \mathbf{h}^T, \tag{3.7}$$

where, P_1 and P_2 are the controller parameters. The proposed neural network parameters and weight adaption method help better estimate uncertain parameters. The weight adaption law is designed based on the sliding surface which consists of the robot dynamics in terms of the error. As the sliding surface goes to zero, the system becomes stable and weights become constant by integrating the weight adaption law. The estimated ($\hat{\mathbf{f}}_\sigma$) is inputted to the ASMC scheme.

3.2.1.2 Adaptive Sliding Mode Control

As discussed in the chapter 2, the SMC scheme can be expressed in two parts i.e. equivalent control law ($\mathbf{u}_{eq}$) and switching control law ($\mathbf{u}_{sw}$) as per [85]. The equivalent control law for the given sliding surface can be derived by equating $\dot{\boldsymbol{\sigma}}(\bar{\xi}) = \bar{\mathbf{f}}_\sigma = 0$ and represented as

$$\mathbf{u}_{eq} = -\mathbf{g}_\sigma^{-1}(\hat{\mathbf{f}}_\sigma + \mathbf{f}_e). \tag{3.8}$$

The switch control law for the ASMC scheme is given below.

$$\mathbf{u}_{sw} = -\eta \mathbf{g}_\sigma^{-1} sgn(\boldsymbol{\sigma}), \tag{3.9}$$

the adaptive law for switching gain [86] is

$$\dot{\eta} = \begin{cases} \eta_1 \parallel \boldsymbol{\sigma} \parallel sgn(\parallel \boldsymbol{\sigma} \parallel - \epsilon) \quad if \quad \eta > \bar{\mu} \\ \bar{\mu} \quad if \quad \eta \leq \bar{\mu} \end{cases} \tag{3.10}$$

η_1, ϵ are the adaptive gain and estimated error in the adaption of the gain respectively. Therefore, the output of SMC is expressed as follows.

$$\mathbf{u}_\sigma = \mathbf{u}_{eq} + \mathbf{u}_{sw} \tag{3.11}$$

The results of the proposed control scheme (RBF-NN SMC) is compared with the ASMC scheme where the uncertainties are considered in as the lumped value in the controller design. The snake robot follows the reference trajectory that online reference trajectory generation during water current is given in further.

3.2.2 Reference Trajectory Generation

The reference trajectory can be generated by observing the current velocity of the water condition. The current water observer is briefly described in Eq. 3.12. More detail can be found in [79].

$$\mathbf{R}_\theta = [u_\theta, v_\theta] \tag{3.12a}$$

$$\mathbf{v}_{rel} = [v_{t,r}, v_{n,r}]^T \tag{3.12b}$$

$$\dot{\mathbf{q}} = -k\mathbf{q} - k^2\mathbf{p}_{xy} - k\mathbf{R}_\theta\mathbf{v}_{rel} \tag{3.12c}$$

$$\bar{\mathbf{v}}_c = \mathbf{q} + k\mathbf{p}_{xy} \tag{3.12d}$$

where $\bar{\mathbf{v}}_\mathbf{c}$ is an estimated current velocity of the water and k is a parameter. Using the Eq. 3.12 the reference head angle (θ_{ref}) and reference velocity (v_{ref}) are to be calculated for the different geometry of the path following.

$$\mu_0 = -\frac{\nabla h(p)^T}{\|\nabla h(p)\|^2}(k_t h(p) + k_h\rho) + \frac{v}{\|\nabla h(p)\|}\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}\nabla h(p)' \tag{3.13a}$$

$$\mu = \frac{v\mu_0}{(\|\mu_0\|} - \bar{v}_c \tag{3.13b}$$

$$\dot{\rho} = h(p) \tag{3.13c}$$

$$\theta_{ref} = arctan2(\mu_y, \mu_x) \tag{3.13d}$$

$$v_{ref} = \|\mu\|, \tag{3.13e}$$

where, $v, h(p)$ denote the desired velocity and path respectively. k_t, k_h are the design parameters.

3.2.3 Control Law Implementation

The block diagram of the RBF neural network sliding mode control implementation is shown in Fig. 3.2. The reference head angle is generated by the current observer and uncertainties are estimated using the RBF neural network method. This block diagram explains how the adaptive RBF-NN SMC is implemented on the underwater snake robot. After design of the control scheme there is need to check the stability of the dynamic system. Therefore, the Lyapunov stability analysis of the proposed RBF-NN SMC is described in the next subsection.

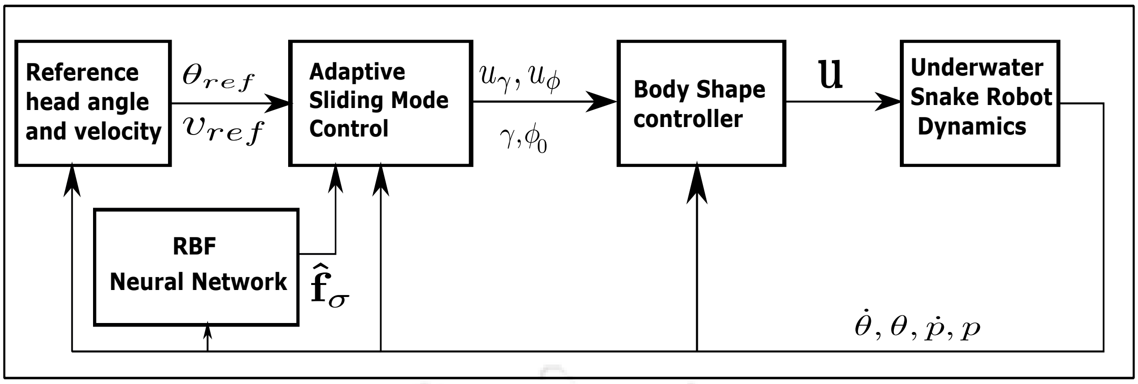

Figure 3.2: Block diagram of the control law

3.2.4 Lyapunov Stability Analysis

Consider the ideal neural network weight $\mathbf{W}^*$ for actual $\mathbf{f}_\sigma$ such as

$$\mathbf{f}_\sigma = \mathbf{W}^{*T}\mathbf{h}.$$

Therefore, the error function is formulated as $\tilde{\mathbf{f}} = \mathbf{f}_\sigma - \hat{\mathbf{f}}_\sigma$ and $\tilde{\mathbf{f}} = \tilde{\mathbf{W}}^T\mathbf{h}$, where, $\tilde{\mathbf{W}} = \mathbf{W}^* - \hat{\mathbf{W}}$ and $\mathbf{W}^* > \hat{\mathbf{W}}$. Substituting this error function in Eq. 3.4, the equation becomes

$$\dot{\boldsymbol{\sigma}}(\bar{\xi}) = \mathbf{f}_\sigma - \hat{\mathbf{f}}_\sigma + \mathbf{f}_e + \mathbf{g}_\sigma\mathbf{u}_\sigma + \bar{\mathbf{f}}_\sigma \tag{3.14}$$

The $\tilde{\eta} = \eta^* - \eta$ is chosen such that $\eta^* > \eta$ for the adaptive sliding mode control. The Lyapunov candidate is chosen as described below.

$$\begin{aligned}
V &= \frac{1}{2}\boldsymbol{\sigma}^T\boldsymbol{\sigma} + \frac{1}{2P_1}\tilde{\mathbf{W}}^T\tilde{\mathbf{W}} + \frac{1}{2}\tilde{\eta}^2 \\
\dot{V} &= \boldsymbol{\sigma}^T\dot{\boldsymbol{\sigma}} - \tilde{\mathbf{W}}^T\dot{\hat{\mathbf{W}}} - \tilde{\eta}\dot{\eta} \\
&\quad \text{substitute Eqs. 3.7, 3.8, 3.9, 3.14} \\
&\quad = \boldsymbol{\sigma}^T\left(\tilde{\mathbf{f}} - \eta sgn(\boldsymbol{\sigma}) + \bar{\mathbf{f}}_\sigma\right) - \frac{1}{P_1}\tilde{\mathbf{W}}^T\left(P_1\boldsymbol{\sigma}\mathbf{h}^T + P_2\|\boldsymbol{\sigma}\|^{\frac{4}{3}}\sigma\mathbf{h}^T\right) \\
&\quad - \tilde{\eta}\eta_1\|\boldsymbol{\sigma}\|sgn\left(\|\boldsymbol{\sigma}\| - \epsilon\right) \\
&\quad = \boldsymbol{\sigma}^T\left(\tilde{\mathbf{W}}^T\mathbf{h} - \eta sgn(\boldsymbol{\sigma}) + \bar{\mathbf{f}}_\sigma\right) - \frac{1}{P_1}\tilde{\mathbf{W}}^T\left(P_1\boldsymbol{\sigma}\mathbf{h}^T + P_2\|\boldsymbol{\sigma}\|^{\frac{4}{3}}\sigma\mathbf{h}^T\right) \\
&\quad - \tilde{\eta}\eta_1\|\boldsymbol{\sigma}\|sgn\left(\|\boldsymbol{\sigma}\| - \epsilon\right) \\
&\quad \leq \|\boldsymbol{\sigma}^T\|\left(\tilde{\mathbf{W}}^T\mathbf{h} - \eta sgn(\boldsymbol{\sigma}) + \|\bar{\mathbf{f}}_\sigma\|\right) - \frac{1}{P_1}\tilde{\mathbf{W}}^T\left(P_1\|\boldsymbol{\sigma}\|\mathbf{h}^T + P_2\|\boldsymbol{\sigma}\|^{\frac{7}{3}}\mathbf{h}^T\right) \\
&\quad - \tilde{\eta}\eta_1\|\boldsymbol{\sigma}\|sgn\left(\|\boldsymbol{\sigma}\| - \epsilon\right)
\end{aligned}$$

adding and subtracting $\eta^*\|\boldsymbol{\sigma}\|$

$$\leq -\left(\eta^* - \|\bar{\mathbf{f}}_\sigma\|\right)\|\boldsymbol{\sigma}\| - \frac{P_2}{P_1}\tilde{\mathbf{W}}^T\|\boldsymbol{\sigma}\|^{\frac{7}{3}}\mathbf{h}^T + (\eta^* - \eta)\|\boldsymbol{\sigma}\| - \tilde{\eta}\eta_1\|\boldsymbol{\sigma}\|sgn\left(\|\boldsymbol{\sigma}\| - \epsilon\right)$$
$$\leq -\left(\eta^* - \|\bar{\mathbf{f}}_\sigma\|\right)\|\boldsymbol{\sigma}\| - \frac{P_2}{P_1}\tilde{\mathbf{W}}^T\|\boldsymbol{\sigma}\|^{\frac{7}{3}}\mathbf{h}^T + \tilde{\eta}\|\boldsymbol{\sigma}\|\left(1 - \eta_1 sgn\left(\|\boldsymbol{\sigma}\| - \epsilon\right)\right).$$

For $\eta^* > \|\bar{\mathbf{f}}_\sigma\|, P_1, P_2 > 0$, the $\dot{V} < 0$ and as per the adaptive RBF algorithm $\mathbf{h} > 0$. This Lyapunov stability decides the parameter η_1 such that it must be $\eta_1 \geq 1$ otherwise the system might be unstable. In the term $sgn\left(\|\boldsymbol{\sigma}\| - \epsilon\right)$, $\|\boldsymbol{\sigma}\| > \epsilon$. If $\|\boldsymbol{\sigma}\| < \epsilon$, there might be a chance of instability in the dynamical system. If $\|\boldsymbol{\sigma}\| >> \epsilon$ satisfies the Lyapunov stability condition, this may lose the robustness of the controller. Therefore, ϵ is chosen as $\epsilon = 4\eta T_e$ (where T_e is the maximum step size of the solver) by following [125]. So, the dynamical system of the snake robot becomes stable for the proposed control method. Therefore, the parameters (P_1, P_2, η_1) are to be chosen such that the Lyapunov stability condition could be satisfied.

3.3 Results and Discussion

The snake robot having mass ($m = 0.9047kg$) and length ($2l = 0.18m$) of each link are considered for the numerical simulation. The drag coefficients and fluidic torque parameters of the water are determined by its impurities, which can result in uncertainties in these values. These uncertainties are accounted for in the parameters of the drag forces (c_t, c_n) and fluidic torque (λ_1, λ_2), same as chapter 2. Moreover, the water current $v_{c,x} = -0.01$ m/s and $v_{c,y} = -0.01$ m/s are considered in X-Y direction, respectively.

The controller parameters are given in Table 3.1. The parameters for the reference trajectory generation during the water current velocity are given in Table 3.2. The sine wave path-following of the snake robot is considered to verify the proposed control. The desired path function is decided as $h(p) = p_y - 0.8\sin(\frac{\pi p_x}{3})$.

Table 3.1: RBF neural network parameters

P_1	P_2	k_n	k_v	η_1	k_p	k_d	k_i	$\bar{\mu}$	T_e
5	2	10	16	1.2	20	5	10	3	0.0021

Table 3.2: Reference trajectory generation parameters

k	k_t	k_h	α	β
1	0.25	0.002	$\frac{30\pi}{180}rad$	$\frac{50\pi}{180}rad$

The results of proposed adaptive RBF neural network sliding mode control (RBF-NN SMC) are compared with the existing adaptive sliding mode control (ASMC) [86] to check the robustness of the proposed control scheme. The nominal parameters for the controller design of ASMC are taken ($\hat{c}_t = 6.5, \hat{c}_n = 17.5, \hat{\lambda}_1 = 1.5, \hat{\lambda}_2 = 0.5$). It is noted that these nominal parameters are not used for the RBF-NN SMC. Figure 3.3 shows the error in head angle tracking of reference head angle generated by the Eq. 3.13. The performance of the proposed control method is found to be better than the ASMC method to track the head angle. This can be observed in the sliding surface shown in Fig. 3.4.

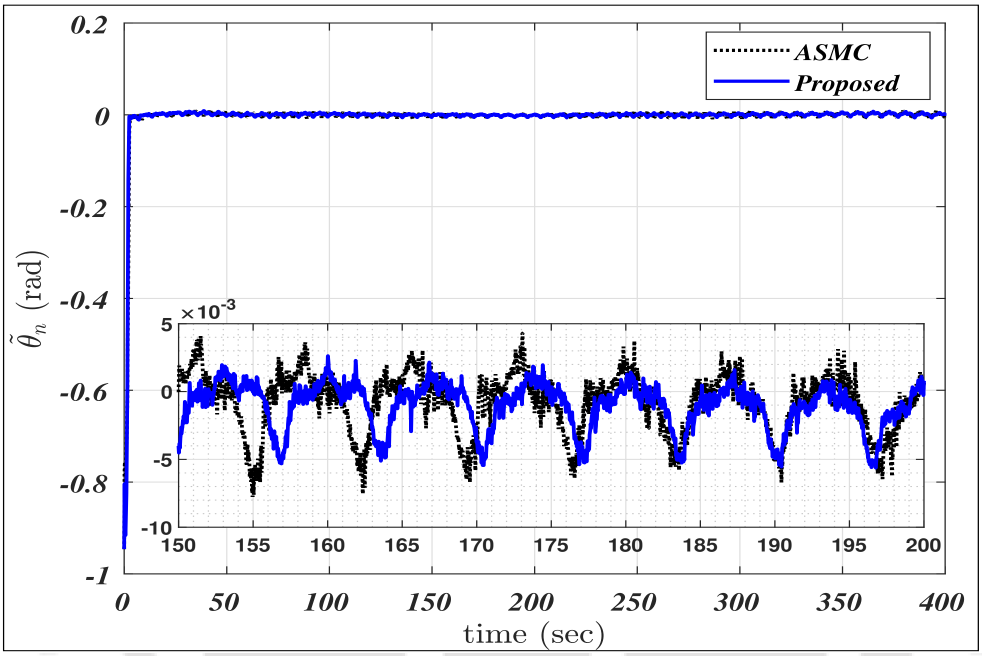

Figure 3.3: Error in head angle

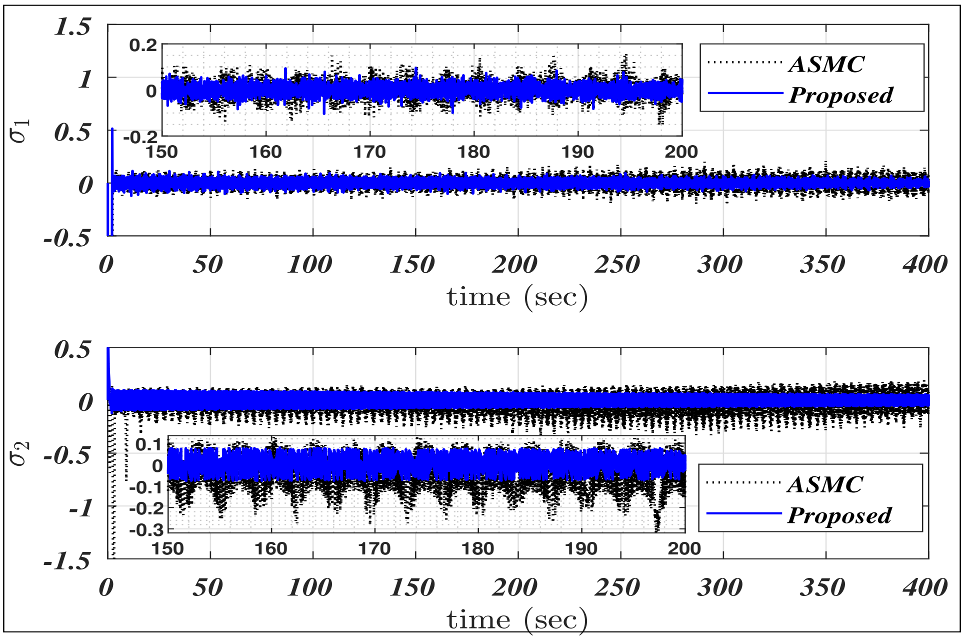

Figure 3.4: Sliding surface

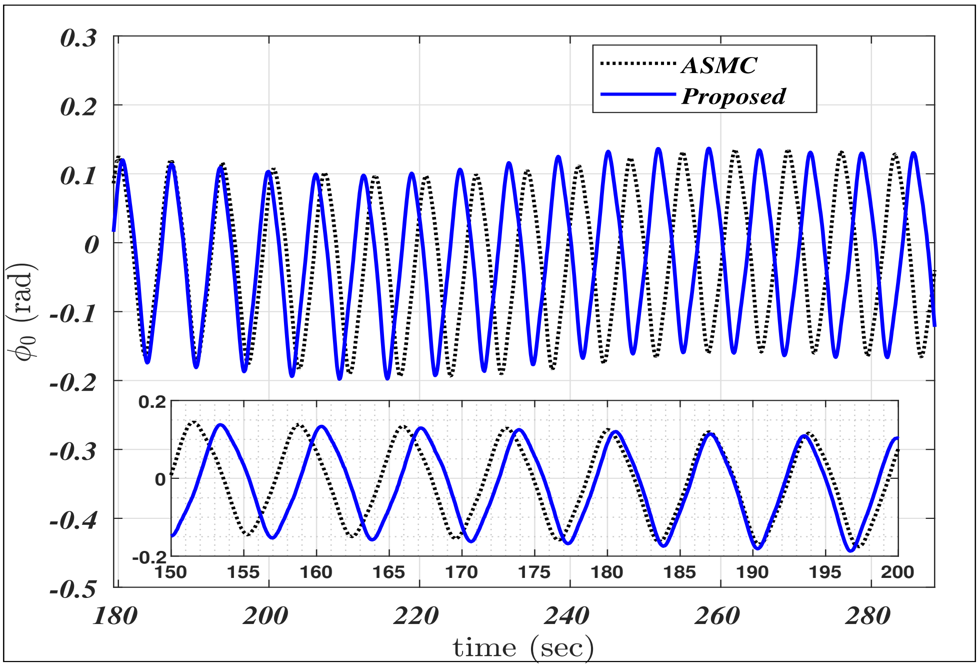

Figure 3.5: Offset angle

The chattering is observed to be less than ASMC by the proposed control method as shown in the zoom figure. These results show that the RBF-NN approach can estimate the uncertainties in a better way. The offset angle of each joint is shown in Fig. 3.5.

Only the phase difference is observed due to the autonomous control method. However,an amplitude of the offset angle seems to be same for control scheme. The reason for controlling the offset angle that is responsible for taking turns of the snake robot according to the path. This offset angle sets all joints as mentioned in Eq. 2.15. Figure 3.6 shows the input frequency for the joint angle. The joint frequency of the proposed controller is higher than the ASMC which helps to get the more tangential velocity of the snake robot. This result can be seen in Fig. 3.7 where the error of the tangential velocity is less than using the RBF-NN SMC. The centre of mass position of the snake robot is traced in Fig. 3.8 that verify the increment of the tangential velocity. Using the proposed control technique, the snake robot has covered more distance than ASMC which reflects that the robot's tangential velocity is increased using the adaptive RBF-NN SMC scheme. The animation video of the sinewave path follows can be found on this weblink.

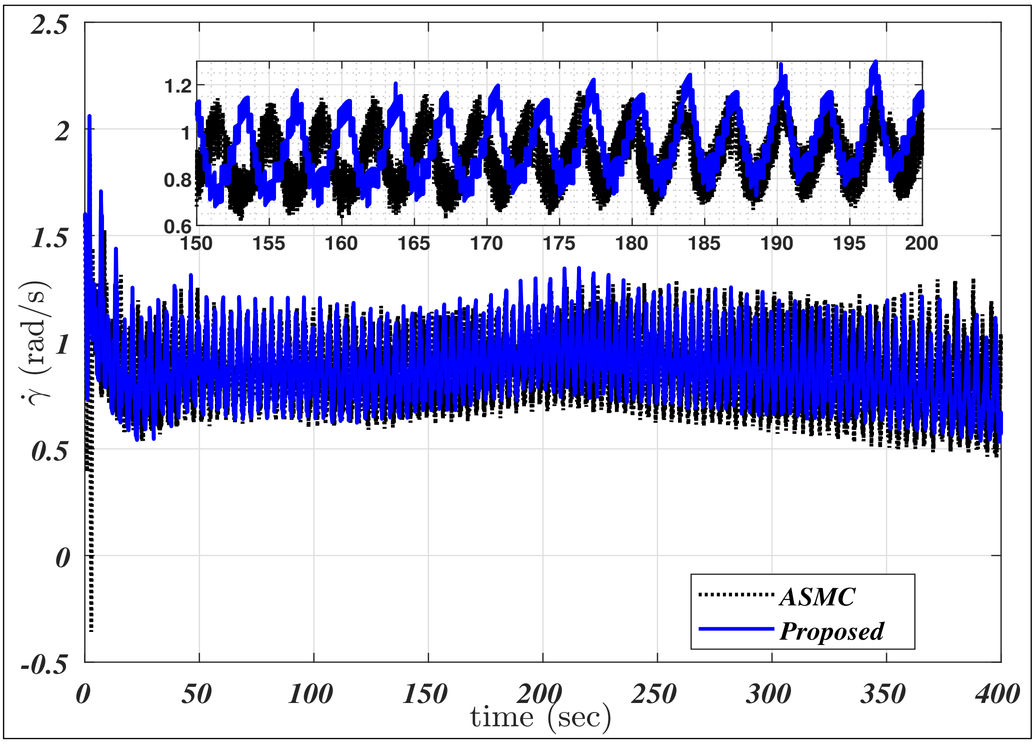

Figure 3.6: Input frequency

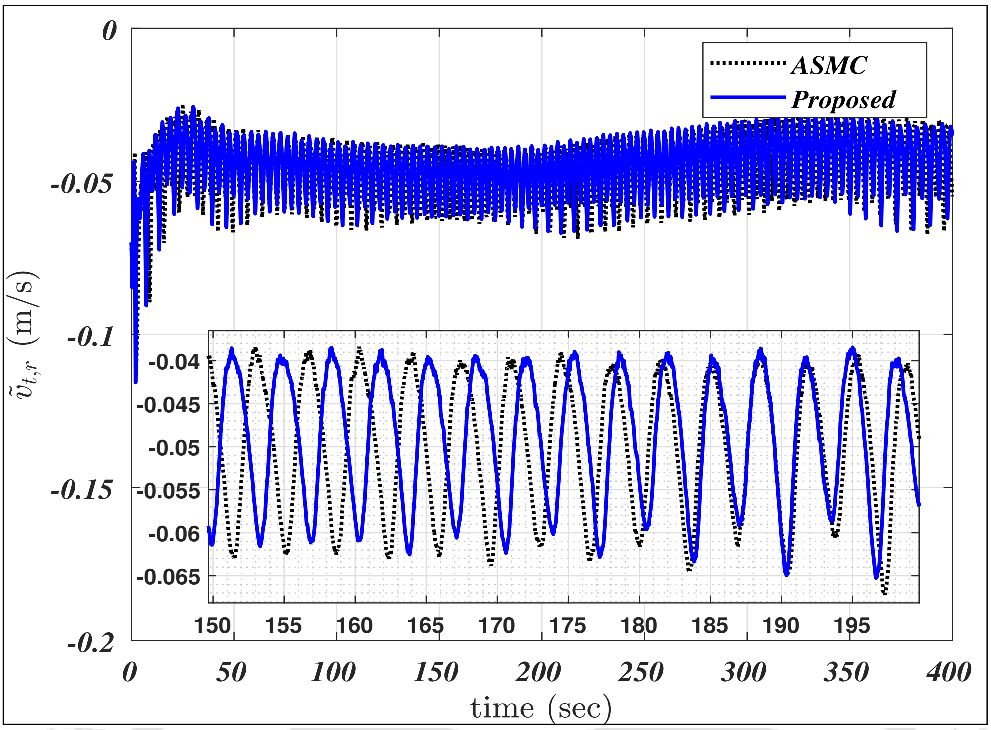

Figure 3.7: Error in tangential velocity

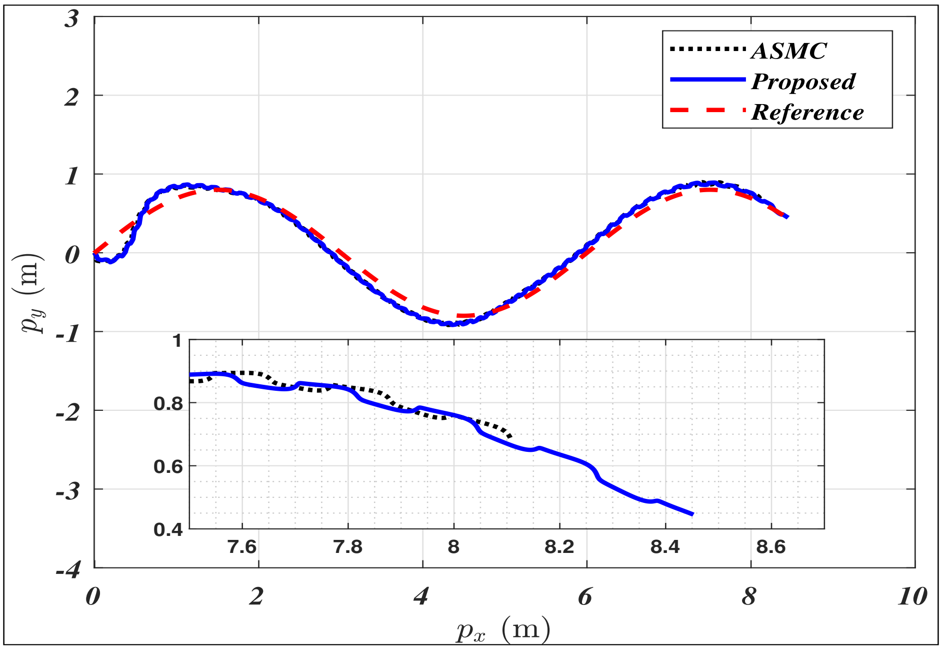

Figure 3.8: CG position trace of snake robot

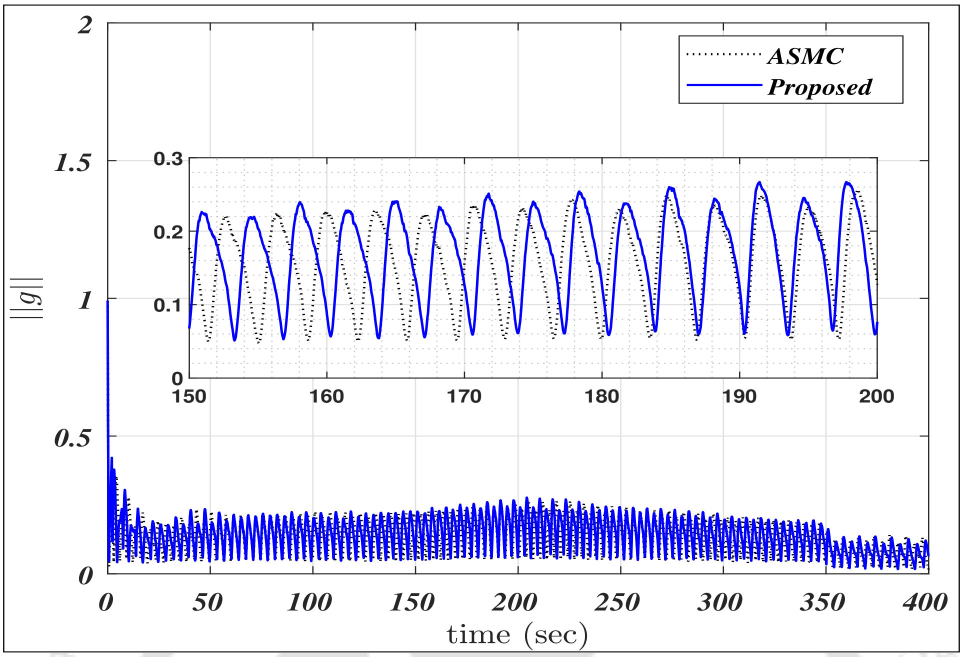

Figure 3.9: VHCs in norm

The norm of the VHCs is shown in Fig. 3.9. It shows the error of the norm which tends to be zero. The adaptive switching gain (η) is shown in Fig. 3.10. The switching gain of the proposed controller is observed to be less than the ASMC. The adaptive RBF-NN weights in norms are shown in Fig. 3.11. The weights are adapted according to uncertainties in the dynamical system. These weights are distributed on the output of the activation function of each node. This process is adaptive and weights change on each calculation according to the system response as mentioned in the adaptation algorithm of the neural network, if required.

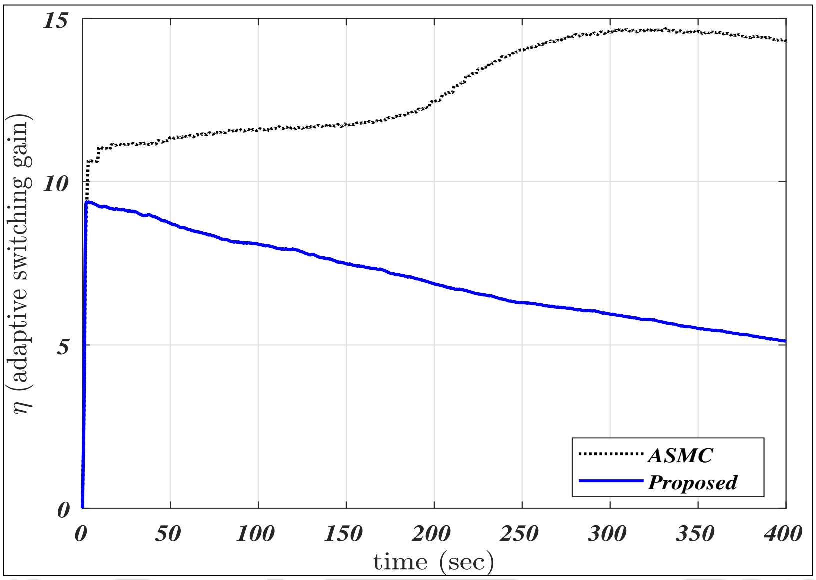

Figure 3.10: Adaptive switching gain

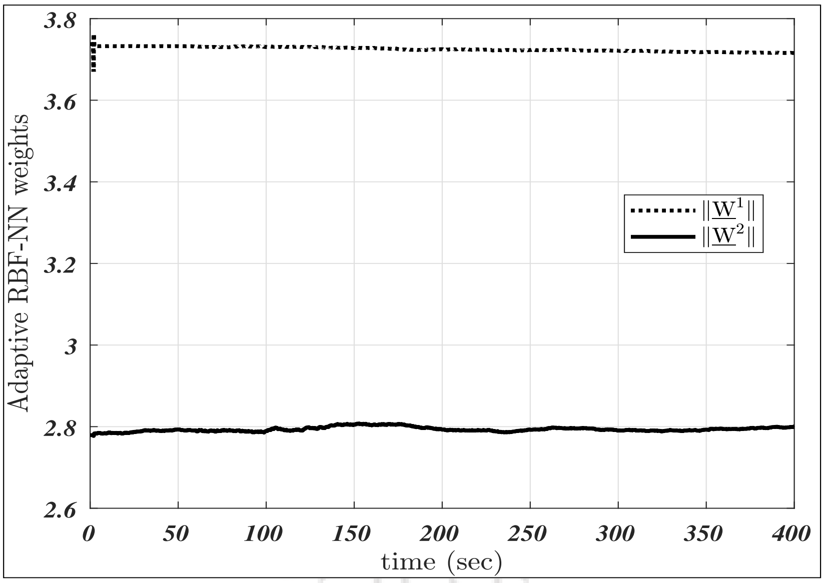

Figure 3.11: Radial basis function adapting weight in norm

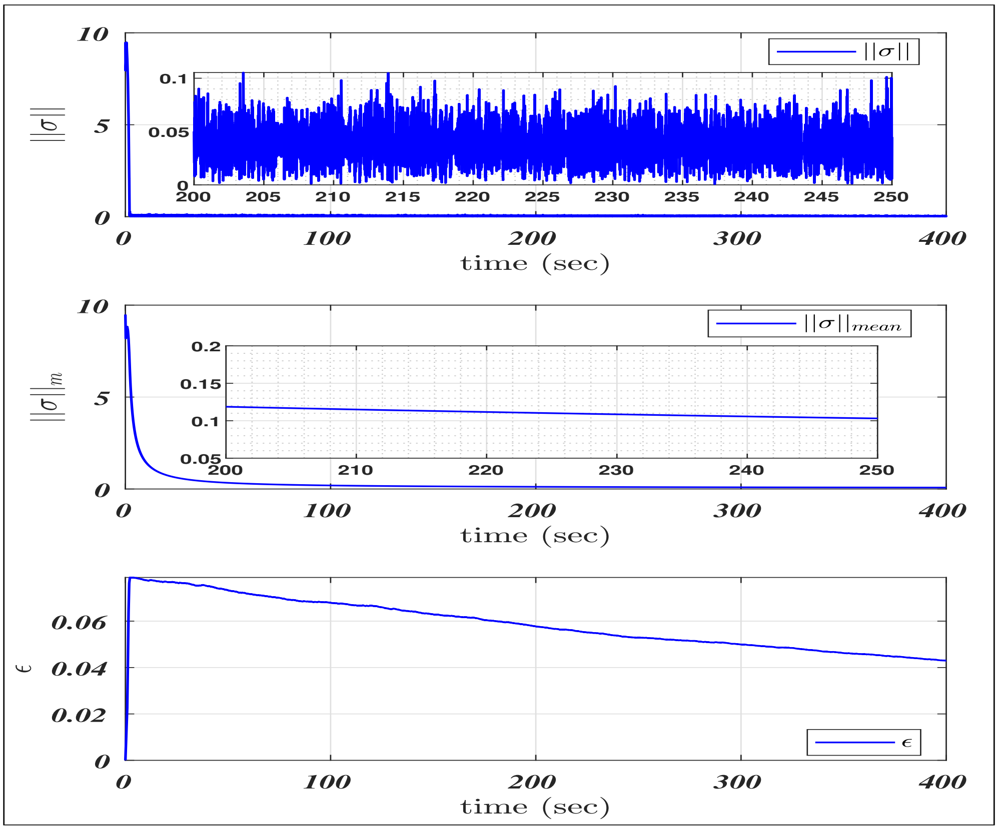

Figure 3.12: Condition $(\|\boldsymbol{\sigma}\| - \epsilon) > 0$ of the Lyapunov stability

Figure 3.12 shows the condition of the Lyapunov stability $sgn(\|\boldsymbol{\sigma}\| - \epsilon)$ to be satisfied. The mean value of sliding surface $(\|\boldsymbol{\sigma}\|_m)$ is used to depict that no over-estimation has been taken to satisfy the Lyapunov stability condition. The torque generated by each joint is shown in Fig. 3.13. The magnitude of each joint torque varies as per the requirement. The lowest torque is required for the head joint and the highest for joint-4 as shown in Fig. 3.13. The chattering in torque is reduced by using adaptive RBF-NN SMC. The chattering effect and root mean square error of the design variable $(\tilde{\theta}_n, \tilde{v}_{t,r})$ of the snake robot is explained further.

The root-mean-square error (RMSE) of the responses is compared as shown in Table 3.3. The improvement in the error is observed using the proposed control method. The chattering values are given in Table 3.4. It is noted that a significant reduction in the chattering effect is observed by using the proposed control method. The chattering effect reduces the smooth operation of the snake robot's motion in the underwater environment. The chattering reduction in the torque increases the motor life and decreases the chattering effect in the responses, as shown in Table 3.4.

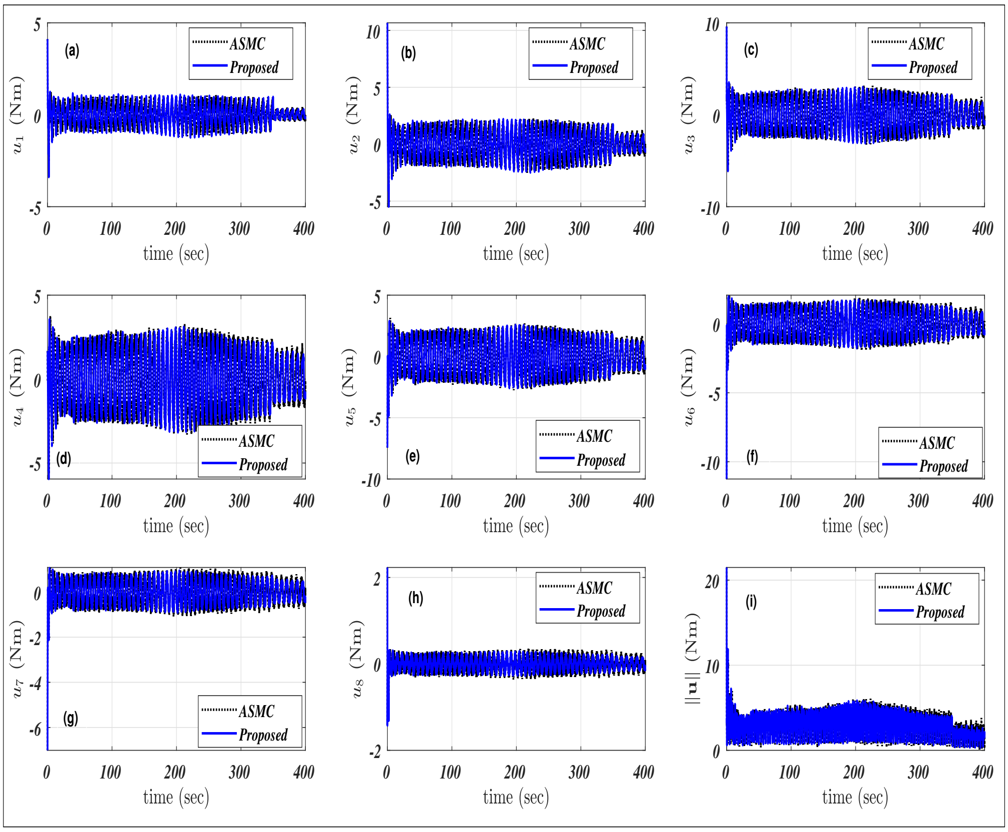

Figure 3.13: (a to h) Torque of each joint, (i) Torque in norm

Table 3.3: Root-mean-square error

Parameters	ASMC	Adaptive RBF-NN SMC
$\tilde{v}_{t,r}$	0.0466 m/s	0.0457 m/s
$\tilde{\theta}_n$	0.0031 rad	0.0027 rad

The overall good performance of the snake robot is observed using the proposed RBF-NN SMC method. The advantage of this method is that there is no requirement for prior knowledge of uncertainties and the uncertain parameters which are estimated based on the feedback of the system. Therefore, the problem of overestimating uncertain parameters can be eliminated using the proposed control scheme in the case of the snake robot.

Table 3.4: Chattering Indicator

Parameters	ASMC	Adaptive RBF-NN SMC
$\tilde{\theta}_n$	0.0392	0.0196
$\tilde{v}_{t,r}$	0.0217	0.0213
$\dot{\gamma}$	12.3834	7.1054
σ_1	5.3501	2.8561
σ_2	12.3483	7.0859
u_1	15.6581	13.2222
u_2	28.7877	20.4865
u_3	32.8202	20.1423
u_4	28.9991	16.6174
u_5	20.9115	12.7748
u_6	19.6462	11.0401
u_7	20.1963	11.1739
u_8	10.4673	6.3642
$\|\mathbf{u}\|$	48.4038	30.2263

3.4 Summary

In this chapter, the adaptive radial basis function neural network sliding mode control (RBF-NN SMC) is used for the motion of the snake robot in the uncertain underwater environmental conditions. The environmental uncertainties are estimated using the RBF neural network by using the feedback of the system response without prior knowledge of the uncertainties. The results are compared with adaptive sliding mode control (ASMC). The proposed controller design parameters are selected based on the Lyapunov stability analysis. The adaptive gains graph convergence reflects the system stability during the uncertainties. The results show that the proposed control method can significantly reduce the chattering effect. The snake robot covered more distance for a given time using the proposed control scheme which means the improvement in tangential velocity is found.

4

3D Dynamics and Control of a Snake Robot in Uncertain Underwater Environment

Contents

4.1 Introduction

In this chapter, 3D motion dynamics of a snake robot for the underwater environment is proposed with vertical motion using the buoyancy variation technique and horizontal motion using lateral undulation. "The neutral buoyant snake robot motion in hypothetical plane and added mass effect are considered" in this chapter. Two different control algorithms are designed for horizontal and vertical motions. The existing super twisting sliding mode control (STSMC) is used for the horizontal serpentine motion of the snake robot. The control law is designed on a reduced ordered dynamic system based on virtual holonomic constraints (VHCs) similar to previous chapters. The vertical motion is achieved by controlling the mass variation using a pump. The water pumps are controlled using the event-based controller/ PD controller. The results of the proposed control technique is verified with various external environmental disturbances and uncertainties to check the robustness of the control approach for various path following cases. In addition, the saturation function is used to limit the input torque to the robot dynamics. The practical implementation of the work is also performed using Simscape-Multibody environment where the designed control algorithm is deployed on the virtual snake robot.

The chapter is organized as follows. The mathematical model of the 3D motion dynamics of a snake robot is derived in section 4.2. The controller design methodology for the proposed dynamics is explained in the section 4.3. The obtained results are discussed in the section 4.4 and finally chapter is summarized in the section 4.5.

4.2 Mathematical Model of the Underwater Snake Robot

In this work, the previous assumptions, "the added mass effect is negligible [79] and the planar motion of neutrally buoyant snake robot in hypothetical horizontal plane [41]" both are removed and 3D motion dynamics of the snake robot is proposed. The robot can change the horizontal plane (up and down) to get vertical motion; as shown in Fig. 4.1. This vertical motion can be achieved by the buoyancy variation technique found in underwater creatures. The proposed technique uses water from the environment to change the mass of each link. The event-based/ PD controller is used for the pump operation to inlet and outlet of the water in the link. The 3D motion of a snake robot can be described as the horizontal motion of the snake robot in the X-Y plane and the

[3]This chapter is part of the publication: B. M. Patel and S. K. Dwivedy (2024). 3D dynamics and control of a snake robot in uncertain underwater environment. Robotica, 1-28.

vertical motion on the Z-axis as shown in Fig. 4.1. Therefore, the motion in the Z-direction can be considered independent from the X-Y plane motion. However, the effect due to variation in mass for the vertical motion would result in horizontal motion during the dynamic condition. Therefore, two different control algorithms are required. These can be explained using the dynamics of the snake robot. Hence, the kinematics and dynamics of the snake robot are explained in the following subsections.

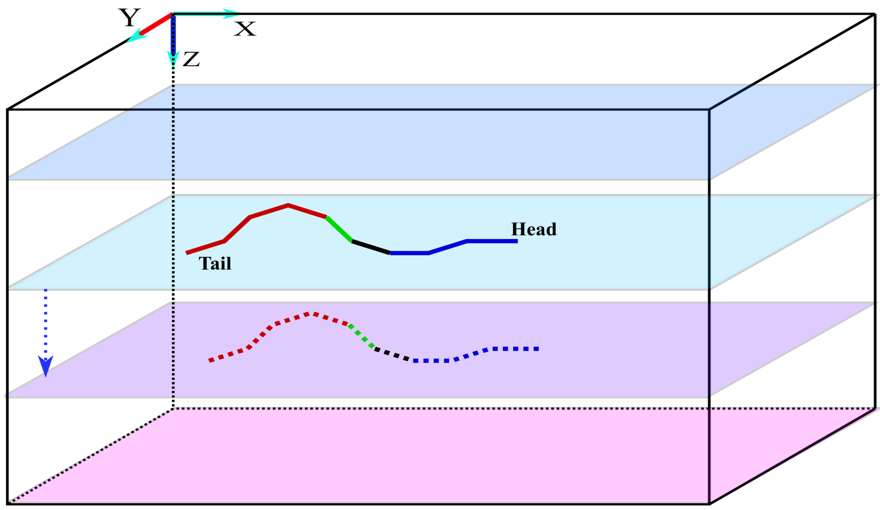

Figure 4.1: Schematic view of 3D motion of a snake robot

4.2.1 Kinematics of the Underwater Snake Robot

In this subsection, the kinematics of the 3D motion of a snake robot is discussed which consists of (horizontal)planar motion in X-Y plane and vertical motion on Z-axis. The planar kinematics of a snake robot has already mentioned in the chapter 2. The snake robot motion in the X-Y plane can be considered as shown in Fig. 2.1. Therefore, it is not explained in detail in this chapter. The relative tangential velocity ($v_{t,r}$) and normal velocity ($v_{n,r}$) of the head link of a snake robot in the X-Y direction can be given as follows.

$$\begin{bmatrix} v_{t,r} \\ v_{n,r} \end{bmatrix} = \mathbf{R}_{\theta_n}^T \left(\dot{\mathbf{p}}_{xy} - \mathbf{v}_c \right), \tag{4.1}$$

where, $\mathbf{v}_c = [v_{cx}, v_{cy}]^T$. v_{cx} and v_{cy} are the water current velocity in X and Y direction, respectively and $\dot{\mathbf{p}}_{xy}$ is the velocity of center-of-mass. The rotation matrix $R_{\theta_n} = \begin{bmatrix} \cos\theta_n & -\sin\theta_n \\ \sin\theta_n & \cos\theta_n \end{bmatrix}$. The relative velocity of each link in the X-Y plane can be given as follows.

$$\begin{bmatrix} \mathbf{V}_{rx} \\ \mathbf{V}_{ry} \end{bmatrix} = \begin{bmatrix} \mathbf{C}_\theta & \mathbf{S}_\theta \\ -\mathbf{S}_\theta & \mathbf{C}_\theta \end{bmatrix} \begin{bmatrix} \dot{\mathbf{X}} - \mathbf{V}_{cx} \\ \dot{\mathbf{Y}} - \mathbf{V}_{cy} \end{bmatrix}, \tag{4.2}$$

where, the relative velocity of each link can be described as $\mathbf{V}_{rx} = [v_{rx,1}, \dots, v_{rx,n}] \in \mathbb{R}^n$, $\mathbf{V}_{ry} = [v_{ry,1}, \dots, v_{ry,n}] \in \mathbb{R}^n$, the water current velocity in X-Y direction of each link can be expressed as $\mathbf{V}_{cx} = [v_{cx,1}, \dots, v_{cx,n}] \in \mathbb{R}^n$, $\mathbf{V}_{cy} = [v_{cy,1}, \dots, v_{cy,n}] \in \mathbb{R}^n$. The relative acceleration can be obtained by differentiating of Eq. 4.2. This relation can be used for the added mass effect dynamics.

$$\dot{\mathbf{V}}_r = \mathbf{K}_1\mathbf{N}_2 + \mathbf{K}_1\mathbf{SC}_\theta\ddot{\boldsymbol{\theta}} + \mathbf{K}_1\mathbf{E}\ddot{\mathbf{p}}_{xy} - \mathbf{K}_1 + \mathbf{K}_2\mathbf{N}_1, \tag{4.3}$$

where,

$$\begin{aligned}
\dot{\mathbf{V}}_r &= [\dot{\mathbf{V}}_{rx}, \dot{\mathbf{V}}_{ry}]^T, \\
\mathbf{K}_1 &= \begin{bmatrix} \mathbf{C}_\theta & \mathbf{S}_\theta \\ -\mathbf{S}_\theta & \mathbf{C}_\theta \end{bmatrix} \begin{bmatrix} \dot{\mathbf{V}}_x \\ \dot{\mathbf{V}}_y \end{bmatrix}, \in \mathbb{R}^{2n\times 2n} \\
\mathbf{K}_2 &= \begin{bmatrix} -\mathbf{S}_\theta & \mathbf{C}_\theta \\ -\mathbf{C}_\theta & -\mathbf{S}_\theta \end{bmatrix} \begin{bmatrix} diag(\dot{\boldsymbol{\theta}}) & \mathbf{0} \\ \mathbf{0} & diag(\dot{\boldsymbol{\theta}}) \end{bmatrix} \in \mathbb{R}^{2n\times 2n}, \\
\mathbf{N}_1 &= \begin{bmatrix} l\mathbf{K}^T\mathbf{S}_\theta\dot{\boldsymbol{\theta}} + \mathbf{e}\dot{p}_x - \mathbf{V}_x \\ -l\mathbf{K}^T\mathbf{C}_\theta\dot{\boldsymbol{\theta}} + \mathbf{e}\dot{p}_y - \mathbf{V}_y \end{bmatrix} \in \mathbb{R}^{2n\times 1}, \\
\mathbf{N}_2 &= \begin{bmatrix} l\mathbf{K}^T\mathbf{C}_\theta diag(\dot{\theta})\dot{\theta} \\ l\mathbf{K}^T\mathbf{S}_\theta diag(\dot{\theta})\dot{\theta} \end{bmatrix} \in \mathbb{R}^{2n\times 1}, \\
\mathbf{SC}_\theta &= \begin{bmatrix} l\mathbf{K}^T\boldsymbol{S_\theta} \\ -l\mathbf{K}^T\boldsymbol{C_\theta} \end{bmatrix} \in \mathbb{R}^{2n\times n}.
\end{aligned}$$

The kinematic relation for the Z-direction motion is as follows.

$$p_z = \frac{1}{n}\mathbf{e}^T\mathbf{Z}, \tag{4.4}$$

where, $Z = [z_1, \dots, z_n] \in \mathbb{R}^n$ is the position vector in the Z-direction.The velocity ($\dot{p}_z$) and acceleration ($\ddot{p}_z$) in Z-direction can be found by differentiating and double differentiating p_z with time respectively. The 3D motion dynamics of the underwater snake robot is given in the following subsection using above kinematic expressions.

4.2.2 3D Motion Dynamics of the Underwater Snake Robot

The free body diagram of the 'i^{th}' link of a snake robot is shown in Fig. 4.2. Figure 4.2a shows the forces and moments acting in the X-Y plane (horizontal) and Figure 4.2b shows the force acting in Z-direction (vertical). The equation of motion can be obtained using Newton's 2^{nd} law for a link 'i' as given below.

$$m\ddot{x}_i = f_{x,i} + h_{x,i} - h_{x,i-1}, \tag{4.5a}$$

$$m\ddot{y}_i = f_{y,i} + h_{y,i} - h_{y,i-1}, \tag{4.5b}$$

$$m\ddot{z}_i = m_b g - f_{z,i}, \tag{4.5c}$$

where, $(f_{x,i}, f_{y,i})$ are the forces acting due to added mass effect and linear-nonlinear drag forces on the link 'i', $(h_{x,i}, h_{y,i})$ are the joint constraint forces between link 'i' and '$i+1$', $(h_{x,i-1}, h_{y,i-1})$ is the joint constraint forces between link '$i-1$' and 'i' as shown in Fig. 4.2a. 'g' is acceleration due to gravity and '$f_{z,i}$' is the force acting in the Z-direction during vertical motion. It may be noted that the total mass 'm' can be calculated as $m = m_s + m_w \pm m_b$ in which m_s is the mass of the snake robot, and m_w is the mass of water. Some amount of the water (m_w) is initially considered inside the link to make the robot neutral buoyant. When the robot gets down, the m_b amount of water is pumped inside the link. To get some position in the 'Z'-direction, the m_b amount of water is discharged from the link using 2^{nd} pump and the robot becomes neutral buoyant at that position level. When the robot moves upward, the m_b amount of the water is discharged from the initial m_w water, becomes a positive buoyant and it lifts due to buoyancy force. A robot can become neutral buoyant by taking water inside the link as the exact amount of water discharged earlier. That means the Eq. 4.5c works when the robot moves in the vertical direction. The schematic view of this discussion is shown in Fig. 4.3. This way, the snake robot would get 3D motion in underwater conditions.

The moment balance equation of link 'i' in the X-Y plane is given below.

$$J\ddot{\theta}_i = u_i - u_{i-1} - l\sin\theta_i(h_{x,i} + h_{x,i-1}) + l\cos\theta_i(h_{y,i} + h_{y,i-1}) + \tau_i. \tag{4.6}$$

where J is the mass moment-of-inertia of link 'i', u_i is the actuator torque between link 'i' and '$i+1$', u_{i-1} is the actuator torque between link '$i-1$' and 'i', $\tau_i = -\lambda_1\dot{\theta}_i - \lambda_2\dot{\theta}_i^2 - \lambda_3\ddot{\theta}$, in which λ_1 and λ_2 are the linear and nonlinear rotational damping parameters, respectively. λ_3 is a rotational added mass effect parameter. No moment in the Z-direction occurs because all links are symmetric and have equal mass. The Eqs. 4.5-4.6 can be extended for the whole body of the snake robot as follows.

$$m\ddot{\mathbf{X}} = \mathbf{F}_x + \mathbf{D}^T\mathbf{h}_x, \tag{4.7a}$$

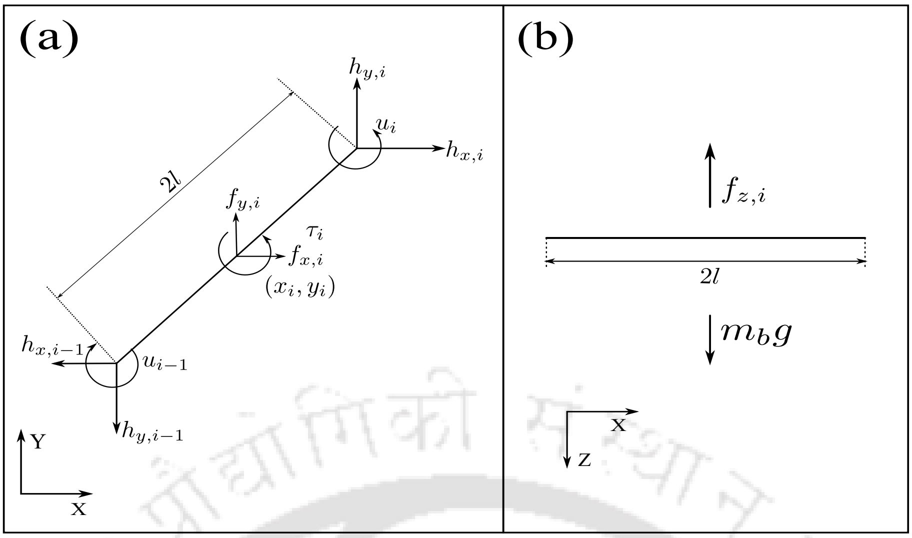

Figure 4.2: Free body diagram of link 'i' (a) forces and moments in XY plane, (b) force in Z-direction

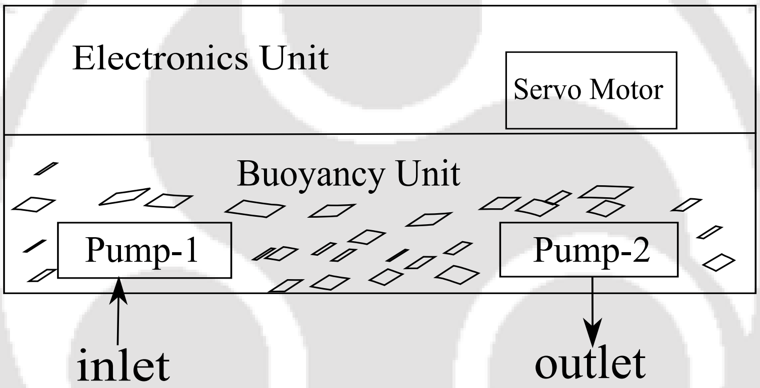

Figure 4.3: Internal components of a link

$$m\ddot{\mathbf{Y}} = \mathbf{F}_y + \mathbf{D}^T\mathbf{h}_y, \tag{4.7b}$$

$$m\ddot{\mathbf{Z}} = m_b g\mathbf{e} - \mathbf{F}_z, \tag{4.7c}$$

$$J\mathbf{I}_n\ddot{\boldsymbol{\theta}} = \mathbf{D}^T\mathbf{u} - l\mathbf{S}_\theta\mathbf{A}^T\mathbf{h}_x + l\mathbf{C}_\theta\mathbf{A}^T\mathbf{h}_y + \boldsymbol{\tau}. \tag{4.7d}$$

where, $\mathbf{u} = [u_1, \cdots\cdots, u_{n-1}]^T$, $\mathbf{F}_x = [f_{x,1}, \cdots\cdots, f_{x,n}]^T$, $\mathbf{F}_y = [f_{y,1}, \cdots\cdots, f_{y,n}]^T$, $\mathbf{F}_z = [f_{z,1}, \cdots\cdots, f_{z,n}]^T$, $\mathbf{h}_x = [h_{x,1}, \cdots\cdots, h_{x,n}]^T$, $\mathbf{h}_y = [h_{y,1}, \cdots\cdots, h_{y,n}]^T$, $\boldsymbol{\tau} = [\tau_1, \cdots\cdots, \tau_n]^T$, $\mathbf{I}_n = \begin{bmatrix} 1 & & \\ & \ddots & \\ & & 1 \end{bmatrix}, \in \mathbb{R}^{n\times n}$. The total force in X-Y direction $\mathbf{F} = [\mathbf{F}_x, \mathbf{F}_y]^T = \mathbf{F_a} + \mathbf{F_d} \in \mathbb{R}^{2n}$, where $\mathbf{F_d} \in \mathbb{R}^{2n}$ is drag force and $\mathbf{F_a} \in \mathbb{R}^{2n}$ is force due to added mass effect. The force $\mathbf{F}_a$ acting on the snake robot due to the added mass effect can be written below.

$$\mathbf{F}_a = \mathbf{D}_1\mathbf{K}_1\mathbf{S}\mathbf{C}_\theta\ddot{\boldsymbol{\theta}} + \mathbf{D}_1\mathbf{K}_1\mathbf{E}\ddot{\mathbf{p}}_{xy} + \mathbf{D}_1\mathbf{K}_3, \tag{4.8}$$

where,

$$\mathbf{E} = \begin{bmatrix} \mathbf{e} & \mathbf{0}_{(n,1)} \\ \mathbf{0}_{(n,1)} & \mathbf{e} \end{bmatrix} \in \mathbb{R}^{2n\times 2},$$
$$\mathbf{D}_1 = -\begin{bmatrix} \mathbf{C}_\theta & -\mathbf{S}_\theta \\ \mathbf{S}_\theta & \mathbf{C}_\theta \end{bmatrix}\begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\mu} \end{bmatrix} \in \mathbb{R}^{2n\times 2n},$$
$$\boldsymbol{\mu} = \mu\mathbf{I}_n,$$
$$\mathbf{K}_3 = \mathbf{K}_1\mathbf{N}_2 - \mathbf{K}_1\dot{\mathbf{V}}_c + \mathbf{K}_2\mathbf{N}_1 \in \mathbb{R}^{2n\times 1},$$

and μ is added mass coefficient, The force due to the drag effect ($\mathbf{F}_d$) can be expressed in terms of linear drag force ($\mathbf{F}_{Ld}$) and nonlinear drag force ($\mathbf{F}_{NLd}$) such as $\mathbf{F}_d = \mathbf{F}_{Ld} + \mathbf{F}_{NLd}$. The linear drag force ($\mathbf{F}_{Ld}$) can be expressed as

$$\mathbf{F}_{Ld} = l\mathbf{Q}_\theta\mathbf{S}\mathbf{C}_\theta\dot{\boldsymbol{\theta}} + \mathbf{Q}_\theta\mathbf{E}\mathbf{v}_r, \tag{4.9}$$

where, $\mathbf{v}_r = \dot{\mathbf{p}}_{xy} - \mathbf{v}_c$, $\mathbf{Q}_\theta = -\begin{bmatrix} c_t(\mathbf{C}_\theta)^2 + c_n(\mathbf{S}_\theta)^2 & (c_t - c_n)\mathbf{S}_\theta\mathbf{C}_\theta \\ (c_t - c_n)\mathbf{S}_\theta\mathbf{C}_\theta & c_t(\mathbf{S}_\theta)^2 + c_n(\mathbf{C}_\theta)^2 \end{bmatrix}$. The nonlinear drag force ($\mathbf{F}_{NLd}$) can be expressed as

$$\mathbf{F}_{NLd} = -\begin{bmatrix} c_t\mathbf{C}_\theta & -c_n\mathbf{S}_\theta \\ c_t\mathbf{S}_\theta & c_n\mathbf{C}_\theta \end{bmatrix} diag\left(abs\begin{bmatrix} \mathbf{V}_{rx} \\ \mathbf{V}_{ry} \end{bmatrix}\right)\begin{bmatrix} \mathbf{V}_{rx} \\ \mathbf{V}_{ry} \end{bmatrix}, \tag{4.10}$$

The force acting on the robot during motion in Z-direction (vertical) can be explained as follows.

$$\mathbf{F}_z = \frac{1}{2m}\rho\mathbf{C}_\mathbf{D}\mathbf{A}_\mathbf{p}diag(|\,\dot{\mathbf{Z}}\,|)\dot{\mathbf{Z}}, \tag{4.11}$$

where, $\mathbf{A}_p = a_p\mathbf{I}_n$, $\mathbf{C}_\mathbf{D} = c_d\mathbf{I}_n$. a_p is projected area of link and c_d is coefficient of drag in Z-direction. The robot changes its position during motion. So, the acceleration of CM in the X-Y plane can be obtained using Eq. 4.7a-4.7b as follows.

$$\ddot{\mathbf{p}}_{xy} = \frac{1}{n}\begin{bmatrix} \mathbf{e}^T\ddot{\mathbf{X}} \\ \mathbf{e}^T\ddot{\mathbf{Y}} \end{bmatrix} = \frac{1}{nm}\mathbf{E}^T\mathbf{F}, \tag{4.12}$$

$\because$ $\mathbf{e}^T\mathbf{D}^T = 0$. The substitution of Eq. 4.11 in Eq. 4.7c gives the motion equation in

Z-direction as follows.

$$\ddot{\mathbf{Z}} = \frac{m_b g}{m}\mathbf{e} - \frac{1}{2m}\rho \mathbf{C_D}\mathbf{A_p} diag(|\ \dot{\mathbf{Z}}\ |)\dot{\mathbf{Z}}, \tag{4.13}$$

The equations of motion of the multi-link snake robot can be obtained by substituting Eq. 2.5 in Eqs. 4.7a,4.7b,4.7d. The acceleration in X-Y plane can be deduced after substituting Eqs. 4.8, 4.9, 4.10 in Eq. 4.12. Vertical CM-acceleration ($\ddot{(p)}_z$) of the snake robot can be derived using the Eq. 4.13 and Eq. 4.4. Therefore, the equation-of-motion for the snake robot in 3D by using the drag forces and added mass effect are as follows.

$$\mathbf{M}_\theta \ddot{\boldsymbol{\theta}} + \mathbf{W}_\theta diag(\dot{\boldsymbol{\theta}})\dot{\boldsymbol{\theta}} - \mathbf{S}\mathbf{C}_\theta^T(\mathbf{f}_d + \mathbf{A}_g) + \boldsymbol{\Lambda}_1\dot{\boldsymbol{\theta}} + \boldsymbol{\Lambda}_2 diag(|\ \dot{\boldsymbol{\theta}}\ |)\dot{\boldsymbol{\theta}} = \mathbf{D}^T\mathbf{u} + \boldsymbol{\Delta}, \tag{4.14a}$$

$$\ddot{\mathbf{p}}_{xy} = \mathbf{H}_x^{-1}\left\{\mathbf{E}^T\mathbf{f}_d + \mathbf{E}^T\mathbf{D}_1\mathbf{K}_1\mathbf{S}\mathbf{C}_\theta\ddot{\boldsymbol{\theta}} + \mathbf{E}^T\mathbf{D}_1\mathbf{K}_3\right\}, \tag{4.14b}$$

$$\ddot{p}_z = \frac{m_b g}{nm}\mathbf{e}^T\mathbf{e} - \frac{1}{2nm}\rho\mathbf{e}^T\mathbf{C_D}\mathbf{A_p} diag(|\ \dot{\mathbf{Z}}\ |)\dot{\mathbf{Z}}, \tag{4.14c}$$

where,

$$\begin{aligned}
\mathbf{M}_\theta &= J\mathbf{I_n} + ml^2\mathbf{S}_\theta\mathbf{V}\mathbf{S}_\theta + ml^2\mathbf{C}_\theta\mathbf{V}\mathbf{C}_\theta - \mathbf{S}\mathbf{C}_\theta^T\mathbf{A}_k + \boldsymbol{\Lambda}_3,\\
\mathbf{A}_k &= \mathbf{D}_1\mathbf{K}_1\mathbf{S}\mathbf{C}_\theta + \mathbf{D}_1\mathbf{K}_1\mathbf{E}\mathbf{H}_x^{-1}\mathbf{E}^T\mathbf{D}_1\mathbf{K}_1\mathbf{S}\mathbf{C}_\theta,\\
\mathbf{A}_g &= \mathbf{D}_1\mathbf{K}_1\mathbf{E}\mathbf{H}_x^{-1}\mathbf{E}^T\mathbf{f}_d + \mathbf{D}_1\mathbf{K}_1\mathbf{E}\mathbf{H}_x^{-1}\mathbf{E}^T\mathbf{D}_1\mathbf{K}_3 + \mathbf{D}_1\mathbf{K}_3,\\
\mathbf{H}_x &= nm\mathbf{I}_2 - \mathbf{E}^T\mathbf{D}_1\mathbf{K}_1\mathbf{E},\\
\mathbf{W}_\theta &= ml^2\mathbf{S}_\theta\mathbf{V}\mathbf{C}_\theta - ml^2\mathbf{C}_\theta\mathbf{V}\mathbf{S}_\theta,\\
\mathbf{V} &= \mathbf{A}^T(\mathbf{D}\mathbf{D}^T)^{-1}\mathbf{A}, \in \mathbb{R}^{n\times n},\\
\boldsymbol{\Lambda_1} &= \lambda_1\mathbf{I}_n, \quad \boldsymbol{\Lambda_2} = \lambda_2\mathbf{I}_n, \quad \boldsymbol{\Lambda_3} = \lambda_3\mathbf{I}_n,
\end{aligned}$$

Δ is the external disturbances. The equation of motion Eq. 4.14a and 4.14b shows the motion of a snake robot in the X-Y plane and Eq. 4.14c shows the motion of a snake robot along the Z-axis. This equations of motion express the snake robot moving in the 3D *viz.* X-Y-Z motion. Hence, two independent controllers must be designed for the snake robot, one for X-Y plane motion and other for Z-axis motion. Here, it is noted that the torque $\mathbf{u}$ is responsible for the horizontal motion and m_b is responsible for the vertical motion. The VHCs are designed for motion control in the X-Y plane and the event-based controller/ PD controller is designed for the motion in the Z-direction. The controller design for the motion of a snake robot for a modified dynamics is explained in the next section.

4.3 Controller Design

The planar motion control approach is similar to explain in the previous chapters. However, the planar motion dynamics equation is changed due to the added mass effect hence the updated reduced dynamics is written as follows.

$$\ddot{\theta}_n = \hat{\Upsilon}_1 + \hat{\Upsilon}_2 u_\gamma + \hat{\Upsilon}_3 u_{\phi_0} + \tilde{\Upsilon}_{\theta_n}, \tag{4.15a}$$

$$\ddot{\mathbf{p}}_{xy} = \hat{\Upsilon}_4 + \hat{\Upsilon}_5 u_\gamma + \hat{\Upsilon}_6 u_{\phi_0} + \tilde{\Upsilon}_p, \tag{4.15b}$$

where,

$$\hat{\Upsilon}_1 = -\frac{\mathbf{e}^T \mathbf{M}_\theta \mathbf{H} \ddot{\mathbf{\Phi}}(\gamma)\dot{\gamma}^2}{\mathbf{e}^T \mathbf{M}_\theta \mathbf{e}} - \frac{\mathbf{e}^T}{\mathbf{e}^T \mathbf{M}_\theta \mathbf{e}} \left(\mathbf{W}_\theta \dot{\theta}^2 - \mathbf{S}\mathbf{C}_\theta(\mathbf{f}_d + \mathbf{A}_g) + \mathbf{\Lambda_1}\dot{\boldsymbol{\theta}} + \mathbf{\Lambda_2} diag(|\,\dot{\boldsymbol{\theta}}\,|)\dot{\boldsymbol{\theta}} \right),$$
$$\hat{\Upsilon}_2 = -\frac{\mathbf{e}^T \mathbf{M}_\theta \mathbf{H} \dot{\mathbf{\Phi}}(\gamma)}{\mathbf{e}^T \mathbf{M}_\theta \mathbf{e}},$$
$$\hat{\Upsilon}_3 = -\frac{\mathbf{e}^T \mathbf{M}_\theta \mathbf{H}\mathbf{b}}{\mathbf{e}^T \mathbf{M}_\theta \mathbf{e}},$$
$$\hat{\Upsilon}_4 = \mathbf{H}_x^{-1}\mathbf{E}^T\mathbf{f}_d + \mathbf{H}_x^{-1}\mathbf{E}^T\mathbf{D}_1\mathbf{K}_1\mathbf{S}\mathbf{C}_\theta\mathbf{H}\ddot{\boldsymbol{\phi}}(\gamma)\dot{\gamma}^2 + \mathbf{H}_x^{-1}\mathbf{E}^T\mathbf{D}_1\mathbf{K}_3 + \mathbf{H}_x^{-1}\mathbf{E}^T\mathbf{D}_1\mathbf{K}_1\mathbf{S}\mathbf{C}_\theta\mathbf{e}\hat{\Upsilon}_1,$$
$$\hat{\Upsilon}_5 = \mathbf{H}_x^{-1}\mathbf{E}^T\mathbf{D}_1\mathbf{K}_1\mathbf{S}\mathbf{C}_\theta\mathbf{e}\hat{\Upsilon}_2 + \mathbf{H}_x^{-1}\mathbf{E}^T\mathbf{D}_1\mathbf{K}_1\mathbf{S}\mathbf{C}_\theta\mathbf{H}\dot{\boldsymbol{\phi}}(\gamma),$$
$$\hat{\Upsilon}_6 = \mathbf{H}_x^{-1}\mathbf{E}^T\mathbf{D}_1\mathbf{K}_1\mathbf{S}\mathbf{C}_\theta\mathbf{e}\hat{\Upsilon}_3 + \mathbf{H}_x^{-1}\mathbf{E}^T\mathbf{D}_1\mathbf{K}_1\mathbf{S}\mathbf{C}_\theta\mathbf{H}\mathbf{b}.$$

$\tilde{\Upsilon}_{\theta_n}, \tilde{\Upsilon}_p$ are the lumped disturbances in reduced dynamics of head angle and CM acceleration, respectively. Based on the reduced order dynamics, the controller would be designed. Here, the planar motion is controlled by the STSMC scheme as mentioned in the chapter 2 and an event-based/ PD control technique controls vertical motion.

The control objective is to *to track the desired path with the desired velocity*. As discussed earlier in section 4.2, the controller is designed in two parts in *viz.* horizontal motion control and vertical motion control as described in the following subsection.

4.3.1 Horizontal Motion Control

The reduced dynamics equation for a snake robot is given Eq. 4.15 in terms of head angle (θ_n) and tangential velocity (v_t). Considering (θ_{ref}, v_{ref}) reference head angle and tangential velocity, the error equation can be written as follows.

$$\ddot{\tilde{\theta}}_n = \hat{\Upsilon}_1 + \hat{\Upsilon}_2 u_\gamma + \hat{\Upsilon}_3 u_{\phi_0} - \ddot{\theta}_{ref}, \tag{4.16a}$$

$$\dot{\tilde{v}}_{t,r} = \mathbf{u}_{\theta_n}^T \hat{\Upsilon}_4 + \mathbf{u}_{\theta_n}^T \hat{\Upsilon}_5 u_\gamma + \mathbf{u}_{\theta_n}^T \hat{\Upsilon}_6 u_{\phi_0} + \dot{\theta}_n v_n - \dot{v}_{ref} \tag{4.16b}$$

These error equations for the state vector $\bar{\xi} = [\tilde{\theta}_n, \dot{\tilde{\theta}}_n, \tilde{v}_{t,r}, v_{n,r}, \gamma, \dot{\gamma}, \phi_0, \dot{\phi}_0]^T$ are used in the designing of a sliding surface that expression is given below.

$$\boldsymbol{\sigma}(\bar{\xi}) = \begin{bmatrix} \sigma_1(\bar{\xi}) \\ \sigma_2(\bar{\xi}) \end{bmatrix} = \begin{bmatrix} \dot{\tilde{\theta}}_n + k_n\tilde{\theta}_n \\ \dot{\gamma} + k_v\tilde{v}_{t,r} \end{bmatrix} \tag{4.17}$$

where, $k_n, k_v > 0$ are the design parameter. The derivative of sliding surface is given below.

$$\dot{\sigma}_1(\bar{\xi}) = \ddot{\tilde{\theta}}_n + k_n\dot{\tilde{\theta}}_n, \tag{4.18a}$$

$$\dot{\sigma}_2(\bar{\xi}) = \ddot{\gamma} + k_v\dot{\tilde{v}}_{t,r}. \tag{4.18b}$$

Therefore, Eq. 4.18 can be rewritten by substituting Eq. 4.16 as

$$\dot{\sigma}_1(\bar{\xi}) = \hat{\Upsilon}_1 + \hat{\Upsilon}_2 u_\gamma + \hat{\Upsilon}_3 u_{\phi_0} - \ddot{\theta}_{ref} + +k_n\dot{\tilde{\theta}}_n + \tilde{\Upsilon}_{\theta_n}, \tag{4.19a}$$

$$\dot{\sigma}_2(\bar{\xi}) = u_\gamma + k_v(\mathbf{u}_{\theta_n}^T\hat{\Upsilon}_4 + \mathbf{u}_{\theta_n}^T\hat{\Upsilon}_5 u_\gamma + \mathbf{u}_{\theta_n}^T\hat{\Upsilon}_6 u_{\phi_0} + \dot{\theta}_n v_n - \dot{v}_{ref}) + u_{\theta_n}^T\tilde{\Upsilon}_p. \tag{4.19b}$$

Now, Eq. 4.19 can represent in sliding vector form as

$$\dot{\boldsymbol{\sigma}}(\bar{\xi}) = \hat{\mathbf{f}}_\sigma + \mathbf{f}_e + \hat{\mathbf{g}}_\sigma\mathbf{u}_\sigma + \bar{\mathbf{f}}_\sigma, \tag{4.20}$$

where,

$$\hat{\mathbf{f}}_\sigma = \begin{bmatrix} \hat{\Upsilon}_1 \\ k_v(\mathbf{u}_{\theta_n}^T\hat{\Upsilon}_4 + \dot{\theta}_n v_n) \end{bmatrix}, \mathbf{f}_e = \begin{bmatrix} k_n\dot{\tilde{\theta}}_n - \ddot{\theta}_{ref} \\ -k_v\dot{v}_{ref} \end{bmatrix}, \hat{\mathbf{g}}_\sigma = \begin{bmatrix} \hat{\Upsilon}_2 & \hat{\Upsilon}_3 \\ k_v(1 + \mathbf{u}_{\theta_n}^T\hat{\Upsilon}_5) & k_v\mathbf{u}_{\theta_n}^T\hat{\Upsilon}_6 \end{bmatrix},$$
$$\mathbf{u}_\sigma = \begin{bmatrix} u_\gamma \\ u_{\phi_0} \end{bmatrix}, \bar{\mathbf{f}}_\sigma = \begin{bmatrix} \tilde{\Upsilon}_{\theta_n} \\ u_{\theta_n}^T\tilde{\Upsilon}_p \end{bmatrix}.$$

The STSMC (ref. chapter 2) scheme is used for the motion control in the XY plane. The equivalent control law can be written for ($\dot{\boldsymbol{\sigma}} = 0$) as follows.

$$\mathbf{u}_{eq} = -\hat{\mathbf{g}}_\sigma^{-1}(\hat{\mathbf{f}}_\sigma + \mathbf{f}_e). \tag{4.21}$$

The reaching law that helps to reach the initial error function on the sliding surface is same as in the chapter 2.

$$\mathbf{u}_{sw} = -\hat{\mathbf{g}}_\sigma^{-1}\Big(k_1 sgn(\boldsymbol{\sigma})\|\boldsymbol{\sigma}\|^{\frac{1}{2}} + k_2\boldsymbol{\sigma}\|\boldsymbol{\sigma}\|^{\frac{3}{2}} + \int_0^t k_3 sgn(\boldsymbol{\sigma})\, dt\Big). \tag{4.22}$$

Therefore, the control law for the horizontal motion is given as follows.

$$\mathbf{u}_{\sigma} = \mathbf{u}_{eq} + \mathbf{u}_{sw} \tag{4.23}$$

The vertical motion control technique is explained in the next section.

4.3.2 Motion Control in Vertical Direction

Z-direction motion is open-loop control as it is independent from the XY motion control. Z-direction motion is controlled using an event-based technique. As discussed earlier, the Z-direction motion can be controlled by using the inlet and outlet pumps. The schematic view of link 'i' is shown in Fig. 4.3. To operate these pumps, the following given logic control algorithm can be used. When one pump starts, the mass of the robot gets changes and the robot starts motion in the Z-direction. In this work, the 3V DC pump has a flow rate of 10 ml/s is considered. It is assumed that *the clean water after small filtering goes inside the pump which prevents the pump from blockages.* Let's consider that Pump-1 is used for taking water inside the link and Pump-2 for discharging water from the link as shown in Fig. 4.3. Triggering conditions for the pump based on the feedback system are explained in Algorithm 4.1. It is noted that at a particular instant either Pump-1 or Pump-2 operates. Both pumps do not work simultaneously to keep the balance the snake robot in underwater conditions.

Algorithm 4.1 Z-direction motion control algorithm

```
if (t ≥ t1)&&(t ≤ t2) then
    Pump-1 ON
else if (t ≥ t3)&&(t ≤ t4)&&(m ≥ mn) then
    Pump-2 ON
else if (px ≥ px1)&&(pz ≥ pz1) then
    Pump-2 ON
else if (pz ≤ pz2)&&(m ≤ mn) then
    Pump-1 ON
else
    No Pump operates
end if
```

This event-based control technique is triggered when the condition is fulfilled. This event-based controller is responsible for the motion in the vertical direction. Another way to control the Z-direction motion is by using a PD control which is described in the following subsection. In that, reference Z-position is required that robot would be reached.

4.3.3 Z-direction Control using PD Controller

It is clear from the previous subsection discussion that the Z-direction motion is possible using the event based algorithm. To reach the exact reference position, the PD based controller can be helpful. In this, for a given reference Z-position, the PD based control scheme is used for the controlling of the pump which is given below.

$$\mathbf{F_z} = k_{pz}(\mathbf{z}_{ref} - \mathbf{z}) + k_{dz}(\dot{\mathbf{z}}_{ref} - \dot{\mathbf{z}}), \tag{4.24}$$

where, k_{pz} and k_{dz} are the proportional and derivative gains, respectively. The pump is directly deal with buoyancy so the saturation of function is used. In which, generated control signal starts the pumps on changing the error sign or achieve the reference position and changing in mass of the link would help to achieve the reference position in Z-direction. The reference trajectory generation is same as given in the chapter 3 in subsection 3.2.2.

4.3.4 Control Law Implementation

There are two controllers that work simultaneously. This gives a clear idea about the implication of STSMC scheme and event-based controller/ PD controller simultaneously as shown in the block diagram given in Fig. 4.4. The saturation block limits the torque input to the robot because the motor has limited torque output in real-time. The demanding torque of the controller might not be fulfilled by the actuator. Therefore, the saturation function is used to meet the real-time condition of the actuator. The complete implementation gives the 3D motion of the snake robot in underwater conditions.

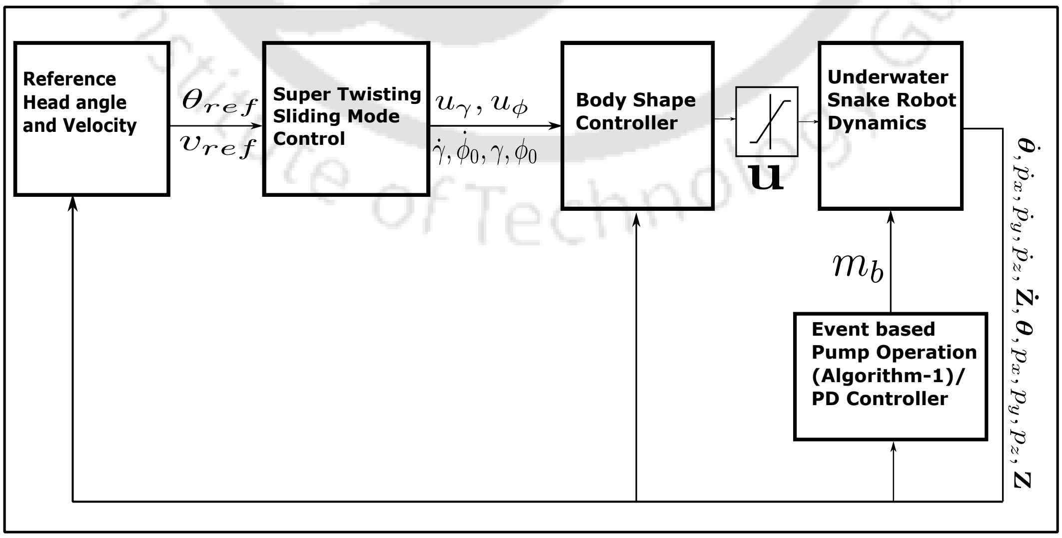

Figure 4.4: Block diagram of the control law implementation

4.4 Results and Discussion

In this section, the simulation results for the proposed dynamics and control of the snake robot are discussed. The parametric uncertainties are considered for the simulation which is graphically shown in Fig. 4.5. The details of given uncertainties are explained in the Eq. 4.25 by following [55].

$$\begin{aligned} c_t &= 12.0\sin(0.00658t + 0.2329) + 22.8\sin(0.01141t + 2.339) \\ &\quad + 17\sin(0.01264t + 5.231) + 0.5rand(1), \end{aligned} \tag{4.25a}$$

$$c_n = 16.371\sin(0.00175t + 1.28) + 0.9771\sin(0.01704t - 1.833) + 0.5rand(1), \tag{4.25b}$$

$$\mu = 2.683\sin(0.00480t - 0.06779) + 1.15\sin(0.00758t + 2.115), \tag{4.25c}$$

$$\begin{aligned} \lambda_1 &= 0.02683\sin(0.00480t - 0.06779) + 0.0115\sin(0.00758t + 2.115) \\ &\quad + 0.5rand(1), \end{aligned} \tag{4.25d}$$

$$\begin{aligned} \lambda_2 &= 0.0012\sin(0.00607t - 0.6005) + 0.00068\sin(0.00867t + 1.561) \\ &\quad + 0.25rand(1), \end{aligned} \tag{4.25e}$$

$$\lambda_3 = 0.0268\sin(0.00480t - 0.06779) + 0.0115\sin(0.00758t + 2.115). \tag{4.25f}$$

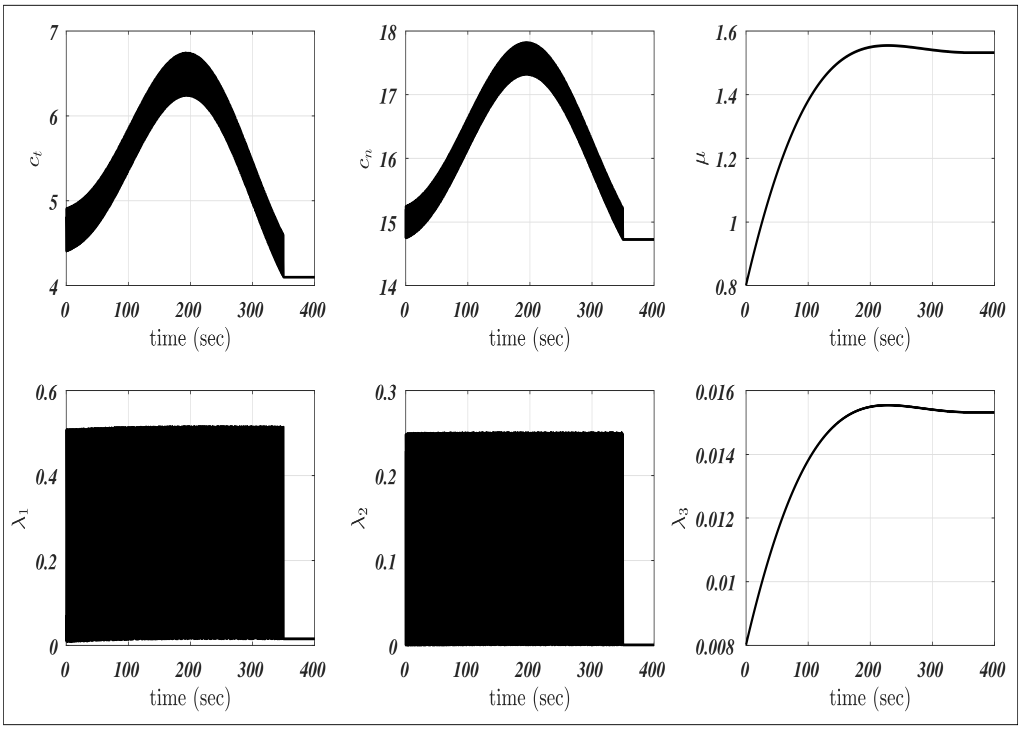

Figure 4.5: Uncertainties in dynamical system parameters

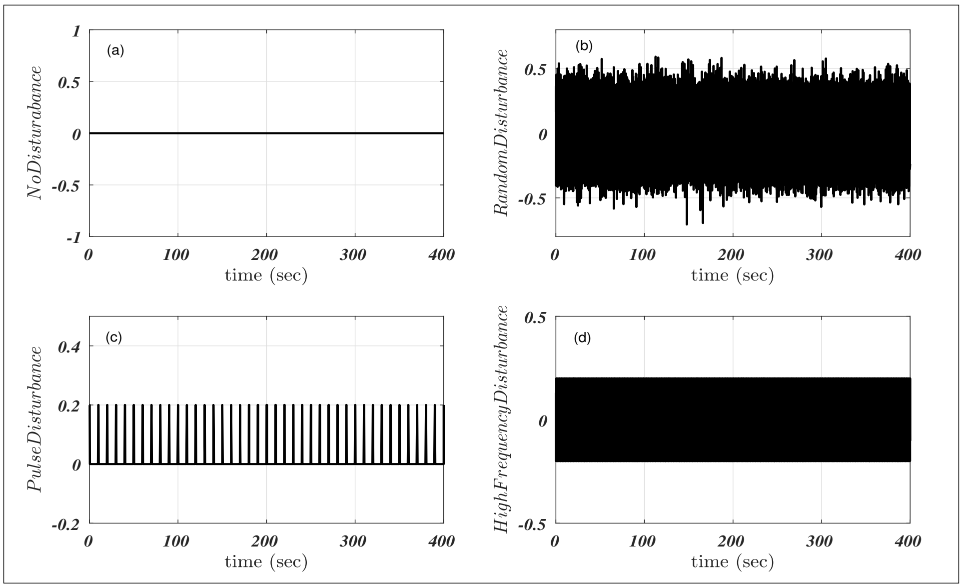

Figure 4.6: Environmental disturbances (a) no disturbance condition, (b) random disturbance condition, (c) pulse disturbance (d) high-frequency disturbance

An additional factor to consider is the presence of external disturbances (denoted as Δ) during the motion of the snake robot. Hence, it becomes important to account for various types of disturbances in order to assess the robustness of the considered control scheme. These disturbances are treated as extreme scenarios for evaluating the effectiveness of the STSMC. The different type of external disturbances acting on link of the snake robot is shown in Fig. 4.6. In this work, four different disturbance cases are considered. The first case is without any external disturbance which can be seen in Fig. 4.6a. The subsequent cases are random disturbance (Fig.4.6b), pulse disturbance (Fig. 4.6c), and high-frequency disturbance (Fig. 4.6d). The link length ($2l$) of a snake robot is considered to be 0.018 m. The mass of a link varies with time to get vertical motion. As given in the Eq. 4.14 for neutral buoyant condition. $m = m_s + m_w$. Here, the solid link mass $m_s = 0.310$ kg and to make the neutral buoyant, the required mass of water $m_w = 0.5947$ kg. Therefore, the neutral buoyant mass $m_n = 0.9047$ kg. The parameters of control Algorithm 4.1 is given in Table 4.1 and the controller parameters are given in Table 4.2.

Table 4.1: Algorithm parameters

Parameters	t_1	t_2	t_3	t_4	m_n	p_{x1}	p_{z1}	p_{z2}
Value	10	20	25	100	0.9047	4	1.7	1

Table 4.2: Parameters for the controlling snake robot

STSMC	$k_n = 10,\ k_v = 12.5,\ k_1 = 10,\ k_2 = 10,\ k_3 = 2$
Water current observer	$k = 1$
Path generation	$v = 0.08$ m/s, $k_t = 0.25,\ k_h = 0.002$
PID controller	$k_p = 20,\ k_d = 5,\ k_i = 10$
Water current	$v_{c,x} = -0.01$ m/s, $v_{c,y} = -0.01$ m/s
Reference Path	$h(p_{xy}) = p_y$ straight line path $h(p_{xy}) = p_y - 0.8\sin(\frac{2\pi p_x}{6})$ sinusoidal path

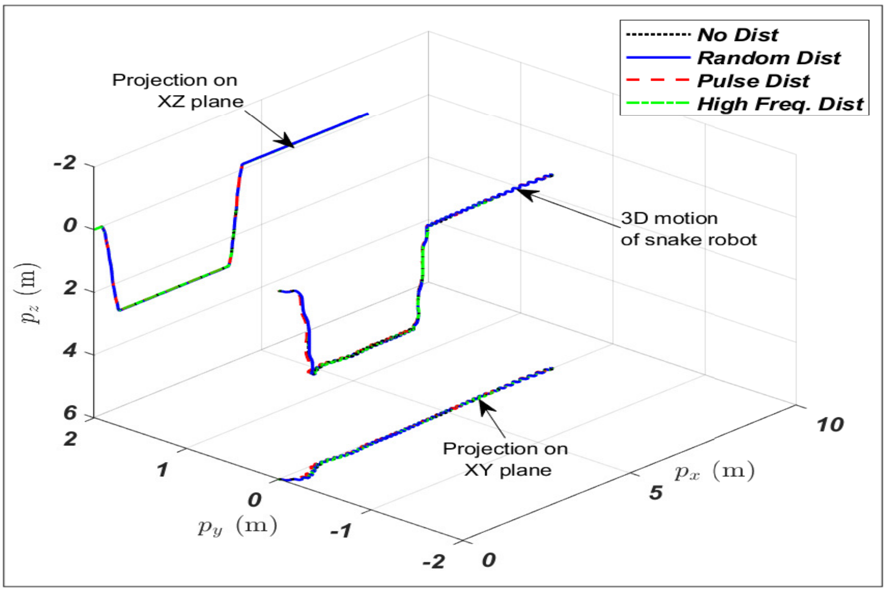

Figure 4.7: 3D motion of the snake robot in a horizontal straight line along with vertical motion

The lumped parameters of the uncertain parts are considered $\hat{\mathbf{f}}_\sigma = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$, $\hat{\mathbf{g}} = \begin{bmatrix} 1, -4.5 \\ 2, -1 \end{bmatrix}$ for the controller design. It is noted that the controller parameters are kept same for all the cases considered for the numerical simulation. The 3D motion of the snake robot using the considered control scheme can be seen in Figs. 4.7 and 4.8. In these figures, the projection on the X-Y plane reflects the motion of the snake robot in a straight line direction (Fig. 4.7) and sinusoidal-like path (Fig. 4.8) and the projection on X-Z plane shows the vertical motion of the robot due to buoyancy variation. The motion of the snake robot is like varying its path in a horizontal plane using the STSMC scheme and vertical motion using the buoyancy variation technique. Hence, it is verified that the proposed control technique achieves the 3D motion of the snake robot. Moreover, the study is not only about achieving the 3D motion of the snake robot but also checking the

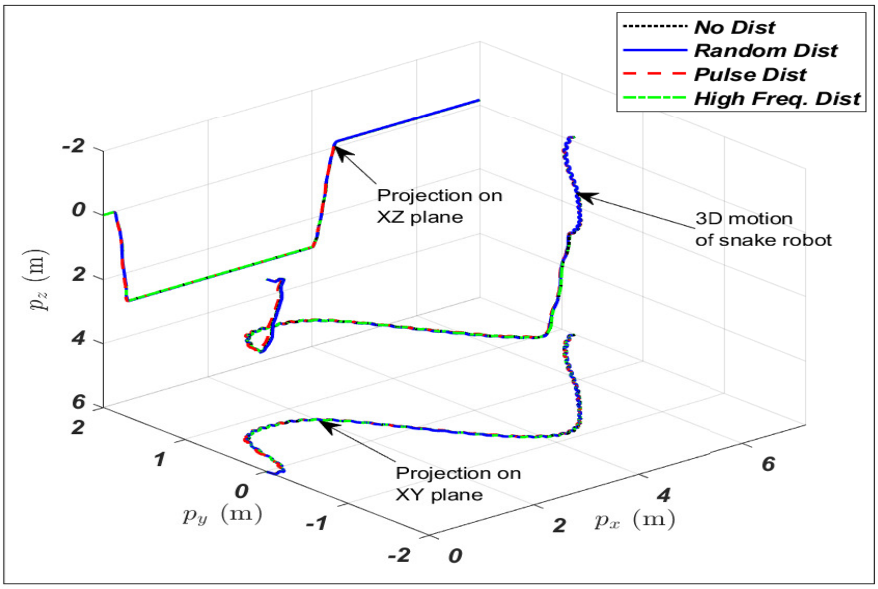

Figure 4.8: 3D motion of the snake robot in horizontal sinusoidal path along with vertical motion

robustness of the control scheme that is further explained.

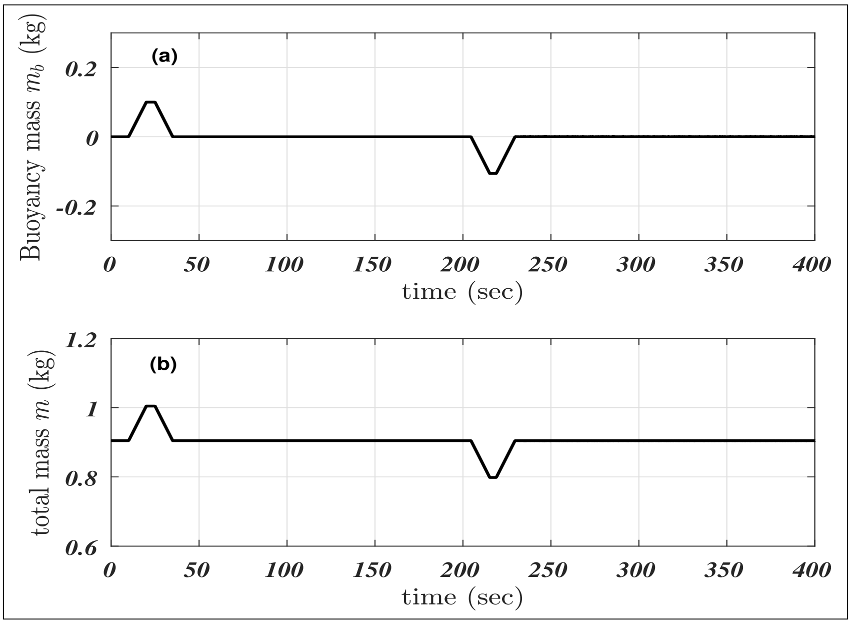

Figure 4.9: Variation in Buoyancy in a single link (a) buoyant mass (b) total mass variation in a link

The variation in buoyant and total mass with respect to time are shown in Fig. 4.9a and Fig. 4.9b, respectively. The pump is ON when the condition is triggered as mentioned in Algorithm 4.1.

4.4.1 Straight Line Path Tracking

Figure 4.10 shows the error in head angle tracking for different disturbances. It is found that the tracking performance is similar for the various disturbances. There is no deviation found in the head angle tracking. These can also be verified using a sliding surface that shown in Fig. 4.11. The sliding surface is meant to be zero to achieve the control objective. The quantified chattering value is discussed in subsection 4.4.3 which is found significantly less using the STSMC. The offset angle is used to turn the robot.

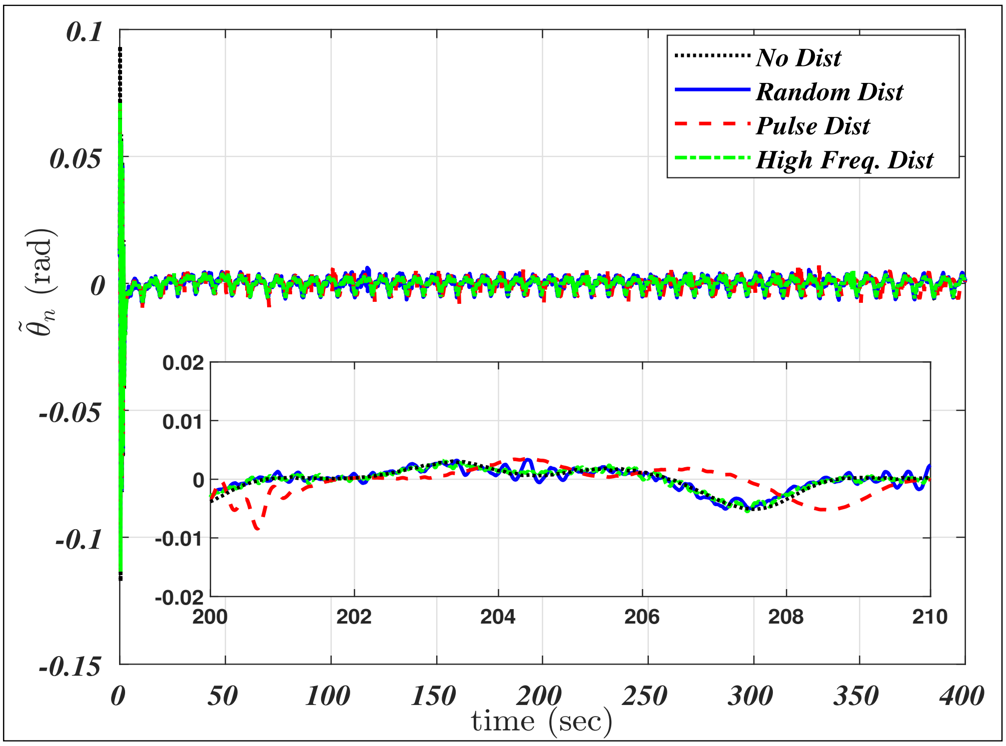

Figure 4.10: Error in head angle: Straight line path

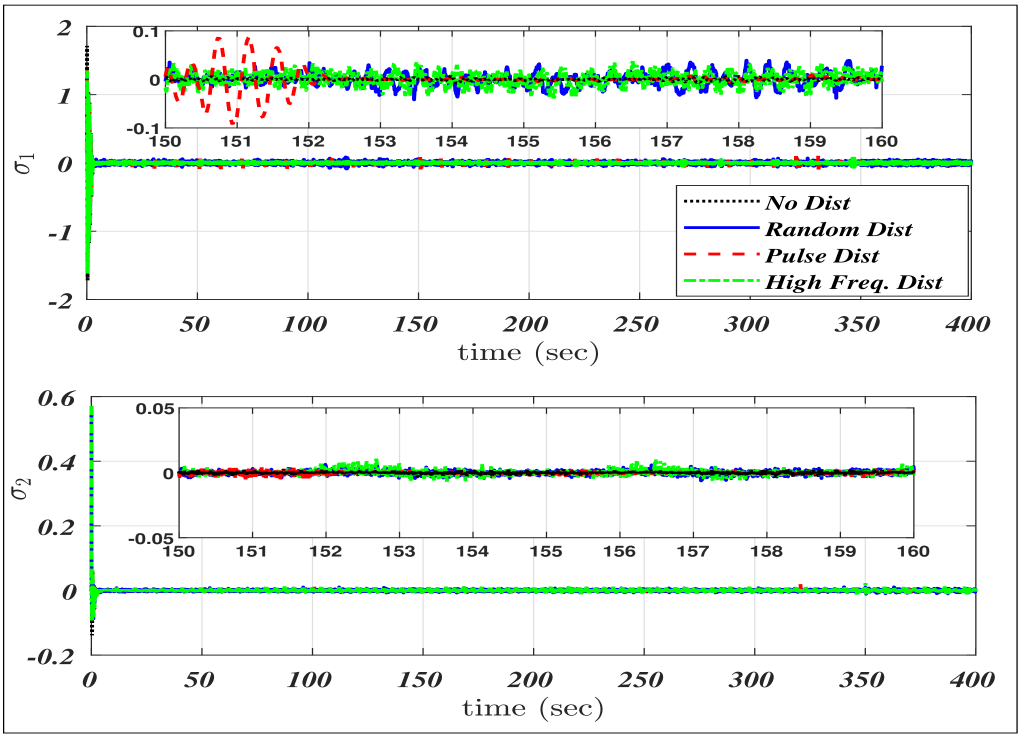

Figure 4.11: Sliding Surface: Straight line path

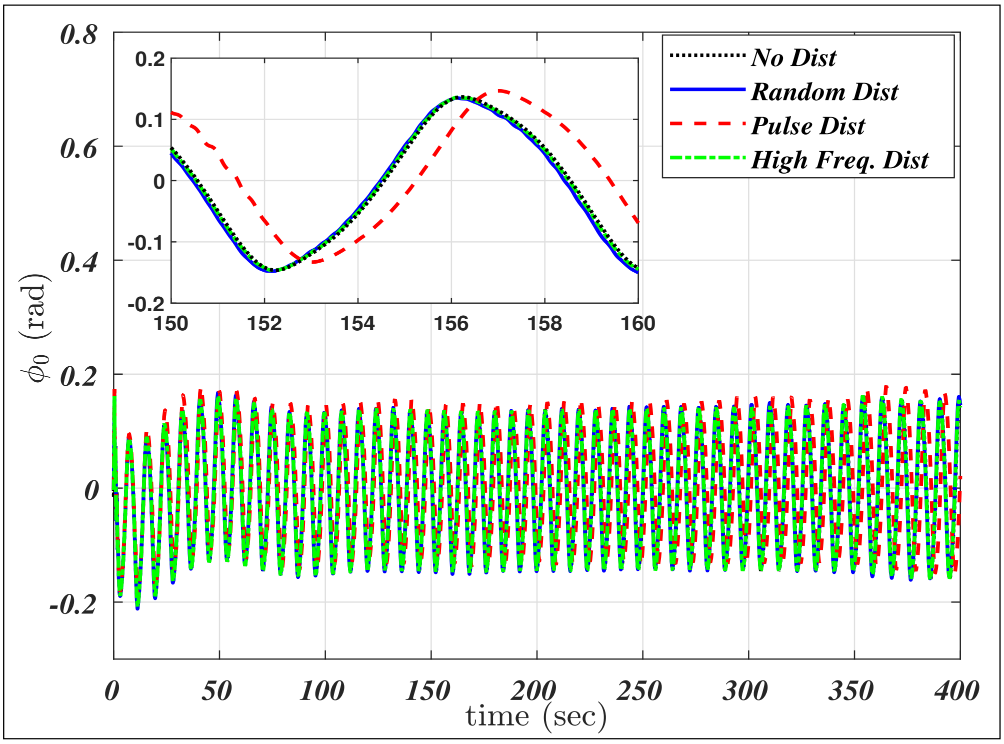

Figure 4.12: Offset angle: Straight line path

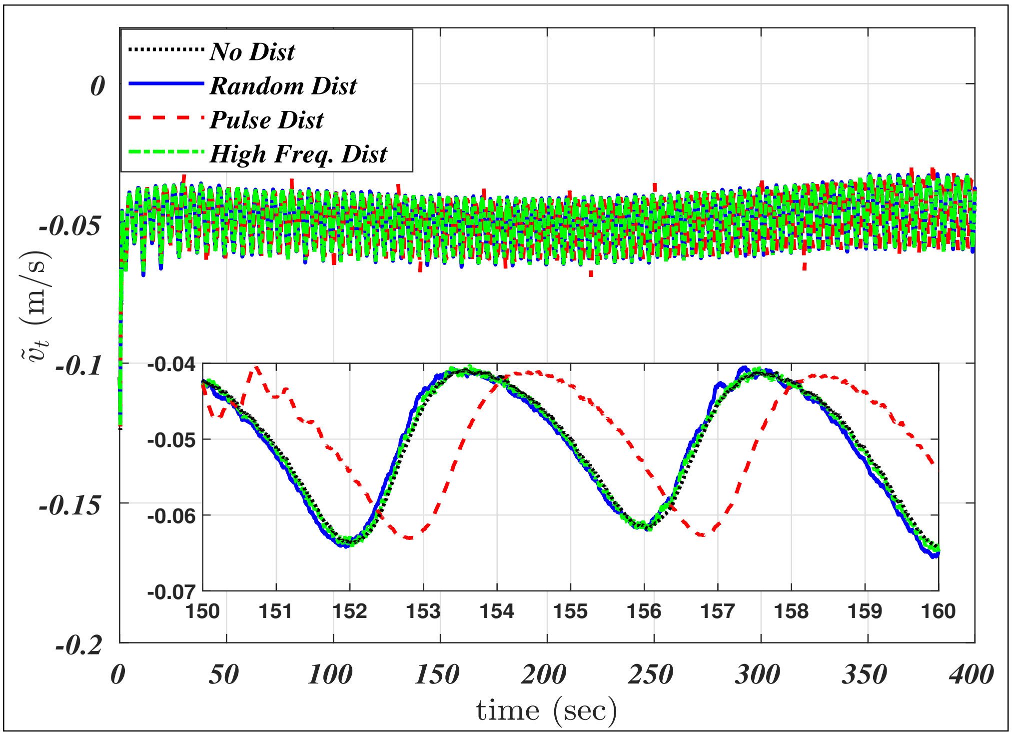

Figure 4.13: Velocity tracking of the snake robot: Straight line path

Figure 4.12 shows the offset angle during different disturbances. The phase variation has been observed in offset angle due to different disturbances and autonomous motion of the robot. Figure 4.13 shows the tangential velocity tracking of the snake robot. It is found that the speed of the snake robot has not changed for any disturbance condition.

4.4.2 Sinusoidal Path Tracking

Figure 4.14 shows the head angle tracking of the sinusoidal path tracking of the snake robot. Here, it observed that initially higher angle deviation compared to straight-line path tracking. Once the robot's motion tracking is converged, the tracking error is almost zero. The sliding surface (error function) of the control objective is shown in Fig. 4.15 for the sinusoidal path following. The high peak initially observed in the simulated sliding surface tracking result is due to fast reaching.

The offset angle which makes it possible to generate the sinusoidal-like path in joint space control is shown in Fig. 4.16. The fluctuation in offset angle is observed in Fig. 4.16 compared to Fig. 4.12 due to different path tracking. The tracking of velocity is shown in Fig. 4.17. The velocity tracking is found to be similar to Fig. 4.13. These sinusoidal and straight-line path following cases verify the robustness and application of the STSMC scheme. The above results are qualitatively good. However, the robustness of the control scheme can be done using quantified results explained in the following subsection.

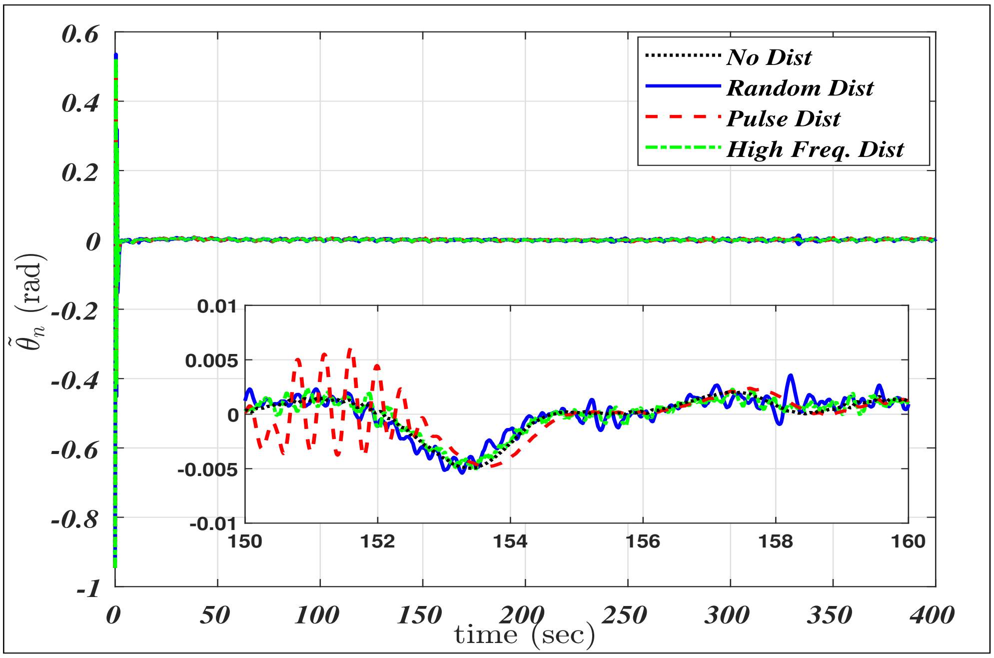

Figure 4.14: Error in head angle: sinusoidal path

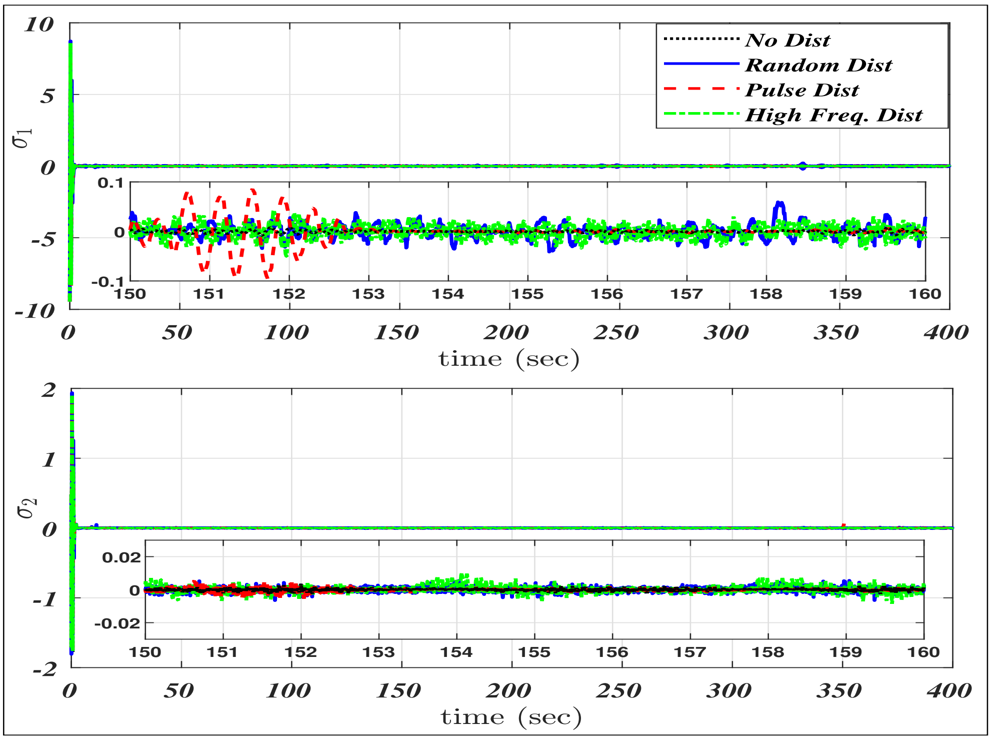

Figure 4.15: Sliding surface: sinusoidal path

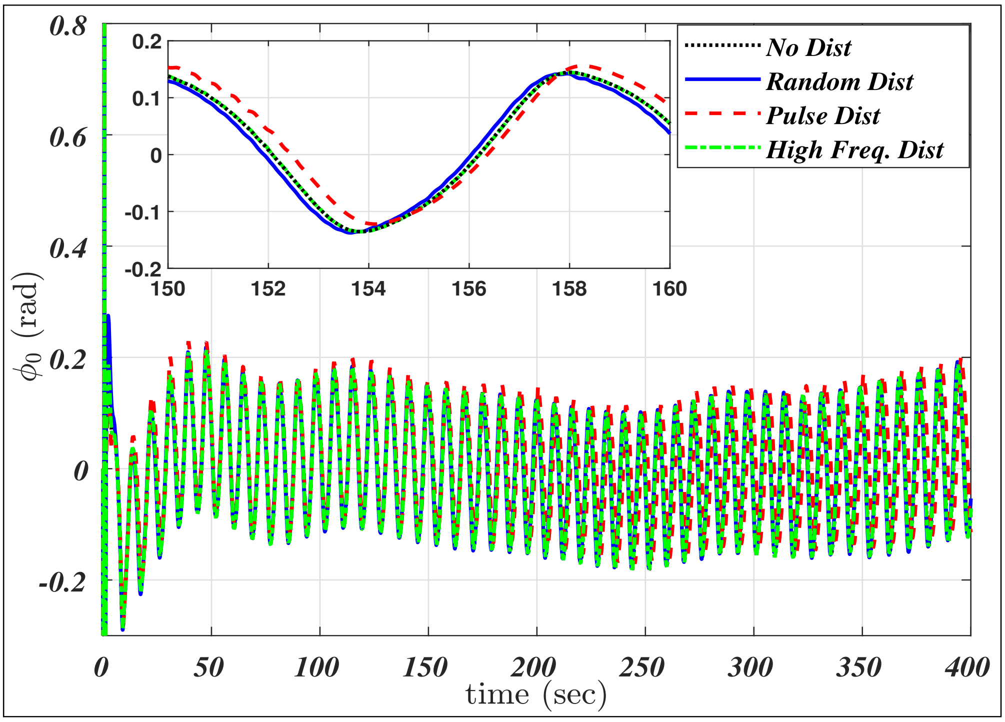

Figure 4.16: Offset angle: sinusoidal path

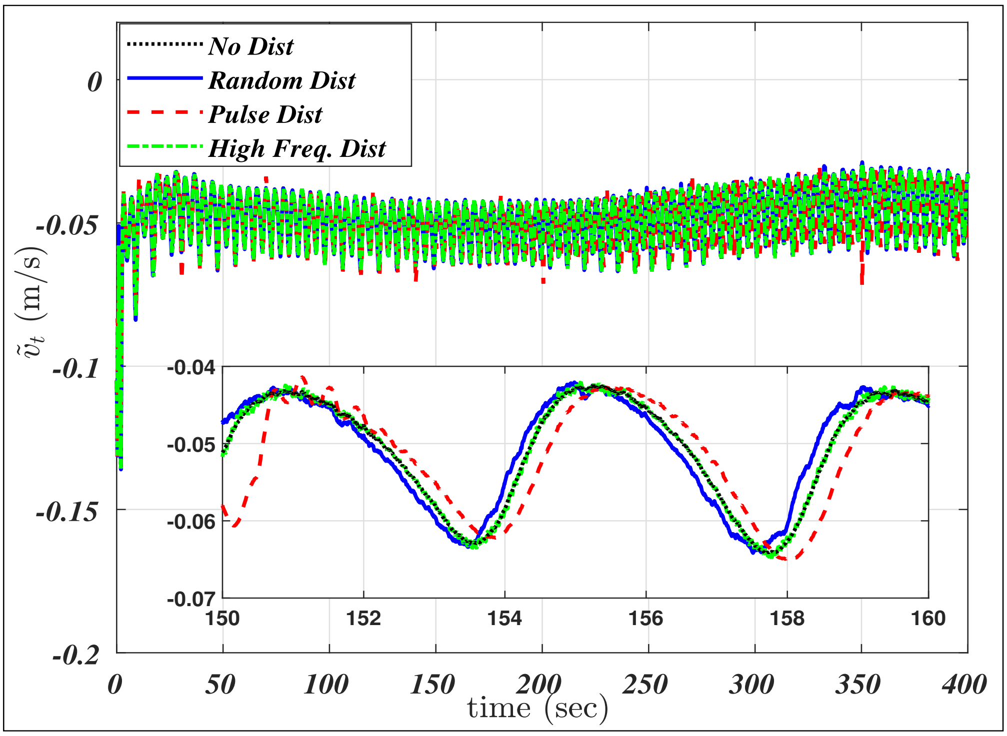

Figure 4.17: Velocity tracking of the snake robot: sinusoidal path

4.4.3 Quantified Results of Chattering and Tracking Error

The quantified chattering effect is shown in Table 4.3 for the straight line path following and in Table 4.4 for the sinusoidal path following. The chattering effect results show the effectiveness of the STSMC scheme. The chattering effect trend is similar for both path-following cases. During the random disturbances, higher chattering is observed for both cases. However, the chattering is nearer for the all cases of disturbances. The error tracking for both cases is shown in Table 4.5 and Table 4.6. Almost similar tracking is observed for all disturbances and path-following cases. That means, the proposed control technique is enough robust to deal with different disturbances and uncertainties.

Table 4.3: Chattering Indicator straight line path

	No disturbance	**Random disturbance**	**Pulse disturbance**	**High frequency disturbance**
u_1	4.7290	13.0287	4.8527	6.7173
u_2	2.3404	3.6868	2.5000	3.4362
u_3	1.6275	3.0024	1.9065	2.4745
u_4	1.5013	2.8225	1.8143	2.4151
u_5	1.4258	2.9857	1.7546	2.7418
u_6	1.2272	3.3077	1.6388	3.0174
u_7	1.0960	3.6733	1.5486	3.9068
u_8	0.9912	12.0914	1.4851	6.0304
$\|\mathbf{u}\|$	2.7468	4.7582	3.3426	4.6455

Table 4.4: Chattering Indicator sinusoidal path

	No disturbance	**Random disturbance**	**Pulse disturbance**	**High frequency disturbance**
u_1	4.5714	12.9811	4.6812	6.6343
u_2	2.2703	3.7301	2.4418	3.3934
u_3	1.5846	3.1296	1.9018	2.4591
u_4	1.4586	2.9658	1.8146	2.4187
u_5	1.3800	3.0970	1.7444	2.7586
u_6	1.1861	3.4107	1.6421	3.0455
u_7	1.0596	3.7932	1.5526	3.9444
u_8	0.9608	12.0917	1.4798	6.0362
$\|\mathbf{u}\|$	2.6622	5.1342	3.3358	4.6375

Table 4.5: Error Tracking: Straight line path

	No disturbance	**Random disturbance**	**Pulse disturbance**	**High frequency disturbance**
$\tilde{v}_{t,r}$ (m/s)	0.0485	0.0485	0.0486	0.0485
$\tilde{\theta}_n$ (rad)	0.0023	0.0022	0.0024	0.0021

Table 4.6: Error Tracking: sinusoidal path

	No disturbance	**Random disturbance**	**Pulse disturbance**	**High frequency disturbance**
$\tilde{v}_{t,r}$ (m/s)	0.0483	0.0484	0.0486	0.0483
$\tilde{\theta}_n$ (rad)	0.0023	0.0024	0.0024	0.0022

The robustness of STSMC control scheme is tested with different path following and external disturbances along with the vertical motion control of the snake robot. Finally, this synchronization of the control scheme produced the 3D motion of the snake robot. Motion control in Z-direction using PD controller is explained below.

4.4.4 PD Controller based Z-direction Motion

Motion control in Z-direction using PD controller is explained here. The reference Z-position (z_{ref}) is given by the slider/ user input in the Simulink. The saturation block is used for it to work with buoyancy control.

Figure 4.18 shows the motion of snake robot control with STSMC in XY-direction and PD controller in Z-direction. The robot motion on triggered condition in Z-direction is shown here. The tracking in Z-direction is shown in Fig.4.19 which which for a given triggered reference the tracking of Z-position is shown. It is noted that there is a delay in achieving the reference Z-position due to constant buoyancy force. The error tracking in the Z-direction is shown in Fig. 4.20. The slope indicates the how fast the small buoyancy force to achieve the zero error.

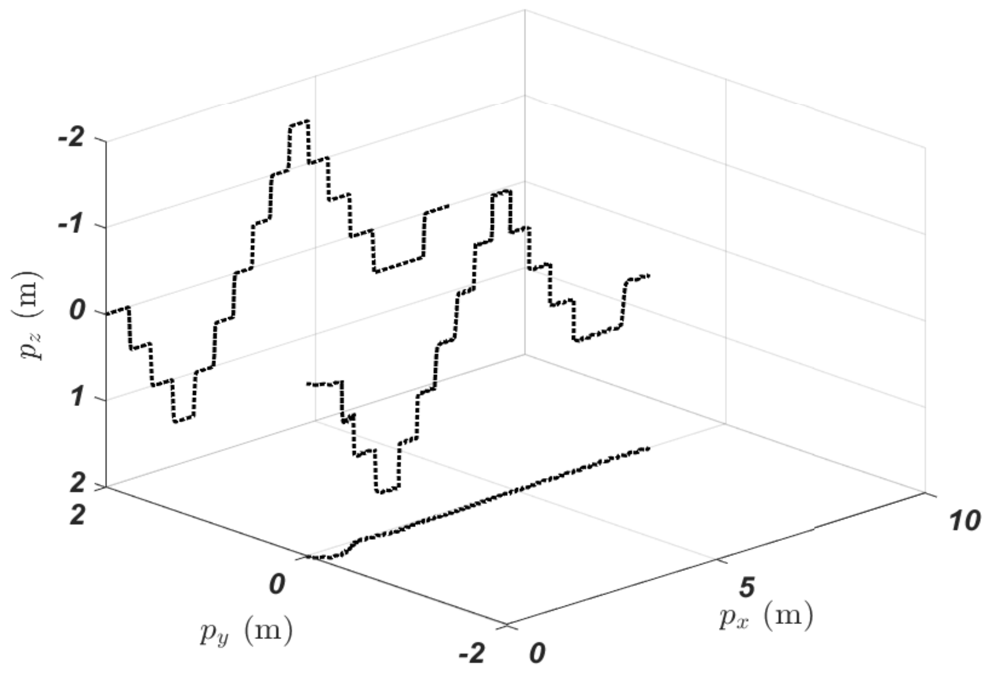

Figure 4.18: 3D motion using PD controller

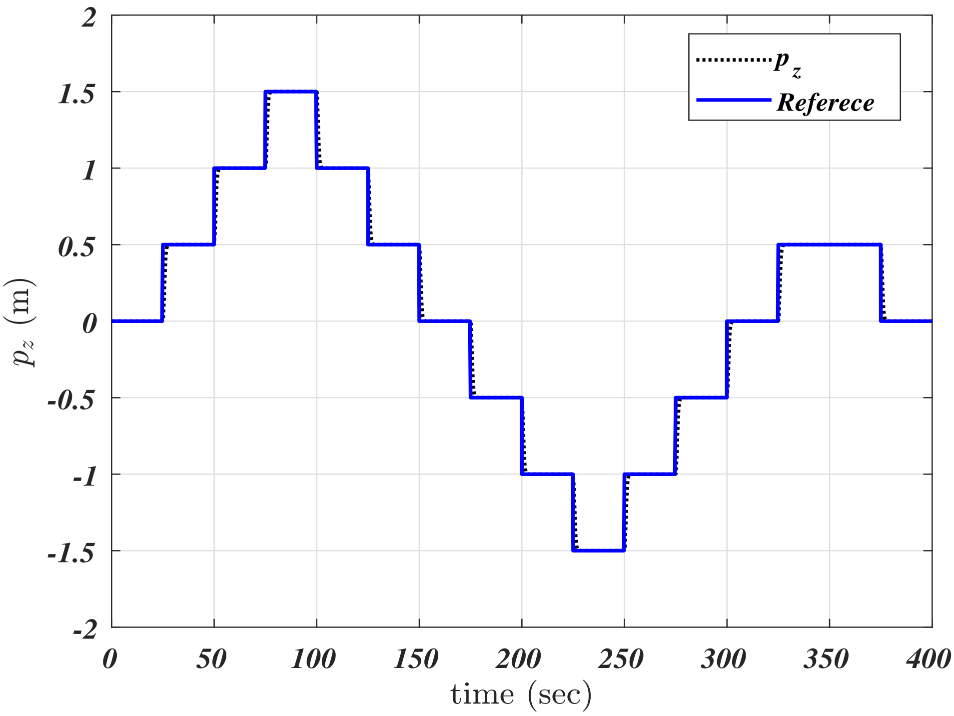

Figure 4.19: Z-position control

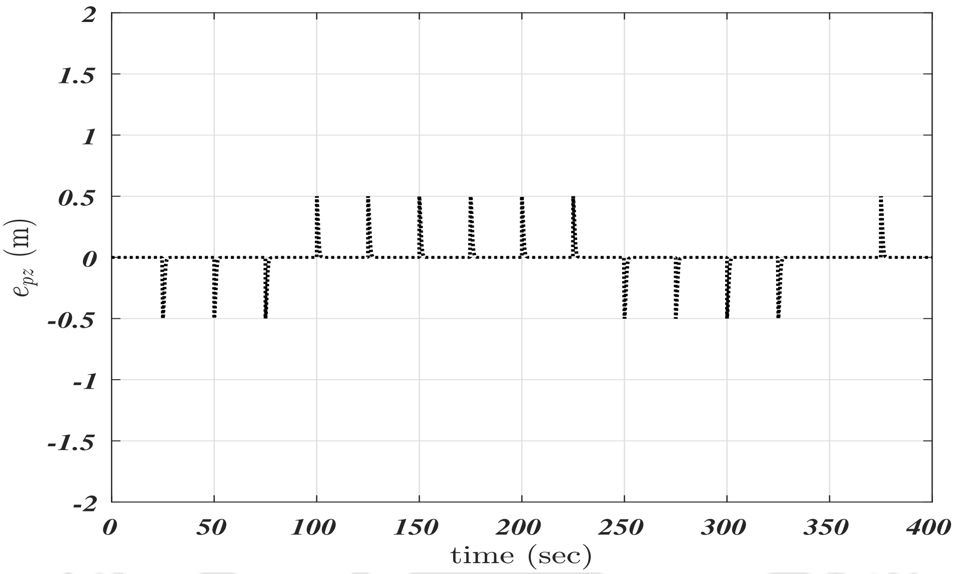

Figure 4.20: Error tracking in Z-direction

4.4.5 Validation in the Simscape-multibody Environment

In this subsection, further verification of the proposed dynamics is carried out using the Simscape multibody environment where there is no need of the mathematical model of the snake robot explicitly. In this, the joints and links are connected. The mathematical equation is not required for the construction of this model. This is a physics based solver. The designed control scheme is deployed by taking the feedback from the joint sensor block and position feedback from the Transform sensor block. The joint stiffness and friction are also considered for this experiment. It may be noted that the transform sensor is placed on the head link. The tracking results are discussed in the following paragraphs.

Figure 4.21 shows the 3D motion of the snake robot in the simscape multibody environment where the motion is tracked from the CG of the head link. The head link CG of the mathematical model is behind the CG of Simscape model because of the co-ordinate system. However, the results are similar in trend. Figure 4.22 shows the vertical motion of a snake robot using the buoyancy variation. The overshoot is observed in Z-direction motion control in the numerical experiment. However, the robot converge to the given reference position in vertical motion. That can also be verified by measuring the error in Z-direction tracking as shown in Fig. 4.23. The video of a motion can be found in the Web Link-1 and Web Link-2.

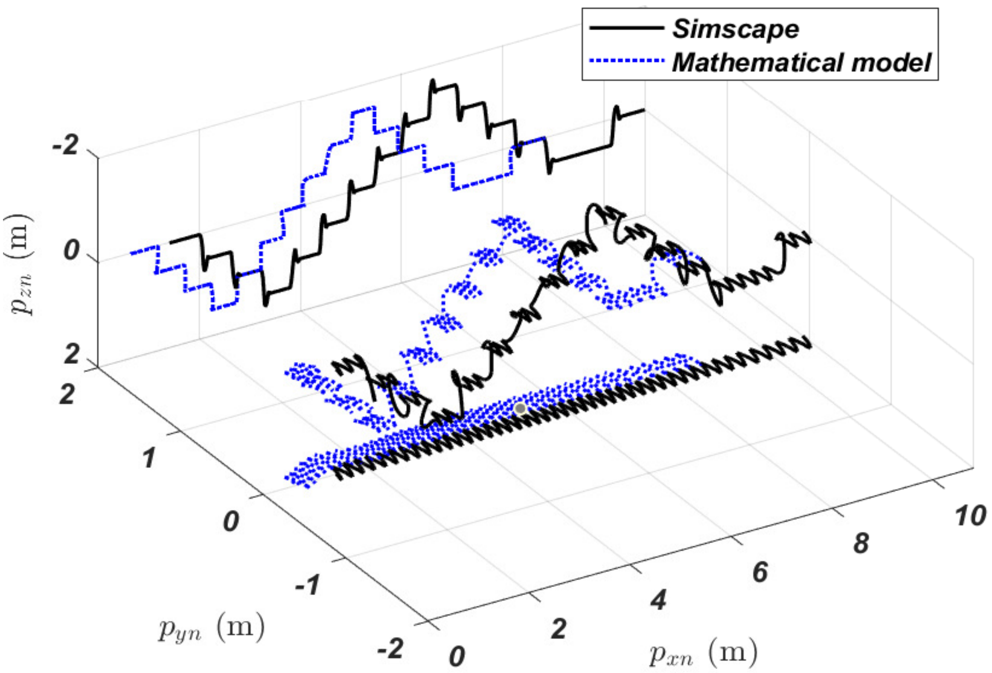

Figure 4.21: 3D motion of the snake robot

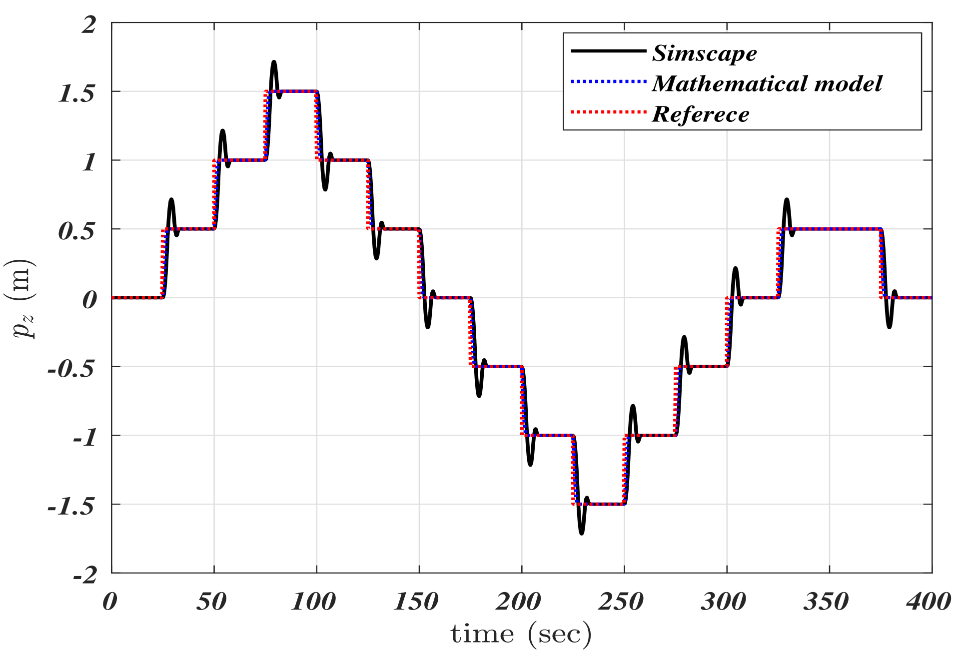

Figure 4.22: Position tracking of the vertical motion

These proposed buoyancy variation techniques and their dynamics will help to develop

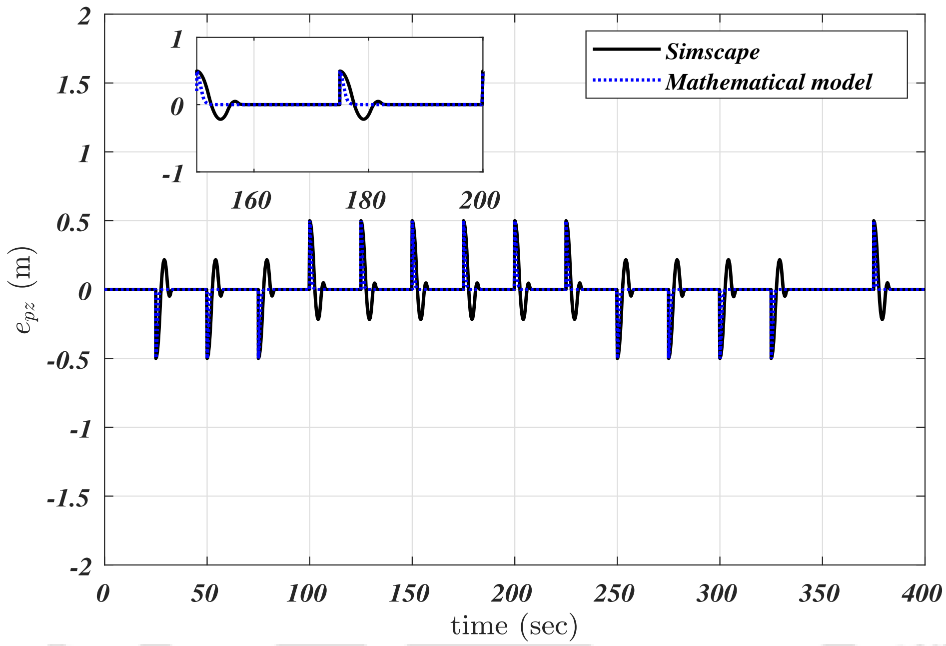

Figure 4.23: Tracking in Z-direction

the further control scheme of the snake robot which will help to perform the manipulator task in underwater conditions without external forces. The proposed approach make it possible to have 3D motion of the snake robot in an underwater environment.

4.5 Summary

In this work, a method for the 3D motion control of the snake robot is proposed using the buoyancy variation technique. The VHCs are used for the motion control in the X-Y plane and the event-based control technique is used for the motion control in the Z-direction. The STSMC is used to estimate the dynamic uncertainties of the snake robot. The robustness of the proposed technique is verified by considering two different paths following cases *viz.* straight-line path and sinusoidal path. The external disturbances are also considered to verify the robustness of the control scheme and discussed. Moreover, the PD controller is used for the motion control and error tracking in the Z-direction. It is also observed that the small change in mass results the significant velocity in vertical direction in the underwater environment. The complete implementation of the control technique (Horizontal and Vertical) on proposed motion dynamics equation results in the 3D motion of the snake robot in an underwater environment. Moreover, the proposed mathematical system is also verified using the Simscape-Multibody environment where the equation of motion is not required explicitly.

5

Manoeuvring of Underwater Snake Robot with Tail-thrust using the Actor-Critic Neural Network STSMC

Contents

5.1 Introduction

In the previous chapters, the robot comes across the drag force and added mass effect during the motion and the speed of the robot is found very low in the underwater environment. Therefore, some thrust force is required to increase its the speed. In this chapter, the speed of the snake robot is improved by implementing a thruster at the tail of a snake robot. The equation of motion for the snake robot is derived by considering the effect of the thrust force on each link. The virtual holonomic constraints (VHCs) are formulated to derive the reduced dynamical system. The control objective is defined based on the reduced order dynamics using the sliding surface approach. The STSMC scheme with reinforcement learning (i.e. actor-critic neural network) is proposed for the motion control of the snake robot in the uncertain underwater environment. The dynamic model of the snake robot (equivalent control law) is estimated using the reinforcement learning control approach. The switching control law is designed using the STSMC. The proposed control scheme not only improves the tracking errors but also reduces the chattering effect and control effort. Finally, these results are verified by comparing them with the existing control scheme. Overall, the online dynamic model estimation problem is addressed by designing the controller for the head link of the snake robot.

This chapter is organized as follows. In the section 5.2, the mathematical modeling of a snake robot with thrust force is derived. Here, the allocation of forces on each link and its effect is derived. The actor-critic neural network super twisting sliding mode control is explained in the section 5.3. The obtained results are discussed in the section 5.4 and chapter is summarized in the section 5.5.

5.2 Mathematical Modeling of the Underwater Snake Robot

The kinematics equations are similar as explained in the previous chapters. Therefore, the motion dynamics of the snake robot with tail thrust is explained in detail. The force acting on a tail is denoted as f_p. The transmission of this force on each link is different. The conversion of f_p into the local coordinates and then global coordinates are described further. The force acting along the direction of the link can be denoted as follows.

$$\mathbf{f}^{l}_{px} = [1, \cos\phi_1, \cos\phi_1 \cos\phi_2, \ldots, \cos\phi_1 \cos\phi_2 \ldots \cos\phi_{n-1}]^T f_p, \tag{5.1}$$

[4]This chapter is part of publication: B. M. Patel, & S. K. Dwivedy (2023). Manoeuvring of underwater snake robot with tail thrust using the actor-critic neural network super-twisting sliding mode control in the uncertain environment and disturbances. Neural Computing and Applications, 1-15.

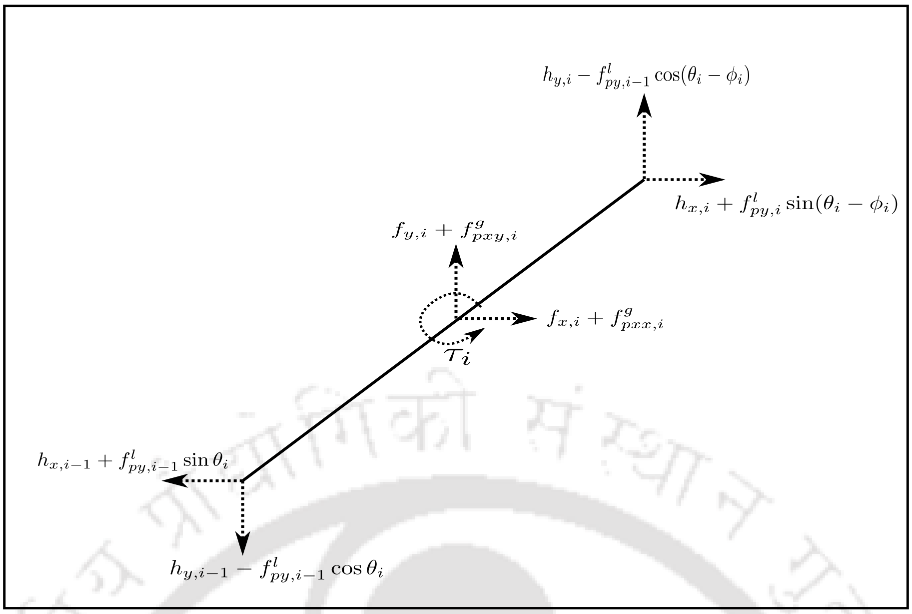

Figure 5.1: Free Body diagram of link 'i'

where, $\mathbf{f}^l_{px} = [f^l_{px,1}, f^l_{px,2}, \ldots, f^l_{px,n}]^T \in \mathbb{R}^n$. The force acting normal to the link is denoted as f^l_{py} and expressed as follows.

$$\mathbf{f}^l_{py} = \mathrm{diag}[\sin\phi_1, \sin\phi_2, \cdots, \sin\phi_{n-1}, 0]\mathbf{f}^l_{px}, \tag{5.2}$$

where, $\mathbf{f}^l_{py} = [f^l_{py,1}, f^l_{py,2}, \ldots, f^l_{py,n}]^T \in \mathbb{R}^n$. The force acting along the direction of link $\mathbf{f}^l_{px}$ can be converted into global coordinate system $(\mathbf{F}^g_{pxx}, \mathbf{F}^g_{pxy})$ as follows.

$$\mathbf{F}^g_{pxx} = \mathbf{C}_\theta \mathbf{f}^l_{px}, \tag{5.3a}$$

$$\mathbf{F}^g_{pxy} = \mathbf{S}_\theta \mathbf{f}^l_{px}, \tag{5.3b}$$

where, $\mathbf{F}^g_{pxx} = [f^g_{pxx,1}, f^g_{pxx,2}, \ldots, f^g_{pxx,n}]^T \in \mathbb{R}^n$ and $\mathbf{F}^g_{pxy} = [f^g_{pxy,1}, f^g_{pxy,2}, \ldots, f^g_{pxy,n}]^T \in \mathbb{R}^n$. The force balance equation is given for a link 'i' as follows.

$$m\ddot{x}_i = f_{x,i} + h_{x,i} + f^l_{py,i}\sin\theta_{i+1} - h_{x,i-1} - f^l_{py,i-1}\sin\theta_i, \tag{5.4a}$$

$$m\ddot{y}_i = f_{y,i} + h_{y,i} - f^l_{py,i}\cos\theta_{i+1} - h_{y,i-1} + f^l_{py,i-1}\cos\theta_i, \tag{5.4b}$$

where, $(f_{x,i}, f_{y,i})$ are the environmental forces acting on the link 'i', $(h_{x,i}, h_{y,i})$ are the joint constraint between link 'i' and '$i+1$', $(h_{x,i-1}, h_{y,i-1})$ is the joint constraint between link '$i-1$' and 'i' as shown in Fig. 5.1.

$$\begin{aligned} J\ddot{\theta}_i = u_i - u_{i-1} &- (h_{y,i-1} - f^l_{py,i-1}\cos\theta_i)l\cos\theta_i + (h_{y,i} - f^l_{py,i-1}\cos(\theta_i - \phi_i))l\cos\theta_i \\ &- (h_{x,i-1} + f^l_{py,i-1}\sin\theta_i)l\sin\theta_i - (h_{x,i} + f^l_{py,i}\sin(\theta_i - \phi_i))l\sin\theta_i + \tau_i. \end{aligned} \tag{5.5}$$

where, J is the mass moment-of-inertia of link 'i', u_i is the actuator torque between link 'i' and 'i+1', u_{i-1} is the actuator torque between link '$i-1$' and 'i', $\tau_i = -\lambda_1\dot{\theta}_i - \lambda_2\dot{\theta}_i^2 - \lambda_3\ddot{\theta}$, in which λ_1 and λ_2 are the linear and nonlinear rotational damping parameters, respectively. λ_3 is a rotational added mass effect parameter. This Eq. 5.5 can be extended further to get the entire moment balance equation of the snake robot as follows.

$$J\mathbf{I_n}\ddot{\boldsymbol{\theta}} = \mathbf{D}^T\mathbf{u} + l\mathbf{C}_\theta\mathbf{A}^T\mathbf{h}_y - l\mathbf{C}_\theta\mathbf{A}^T\mathbf{F}^l_{pyC\theta} - l\mathbf{S}_\theta\mathbf{A}^T\mathbf{h}_x - l\mathbf{S}_\theta\mathbf{A}\mathbf{F}^l_{pyS\theta} + \boldsymbol{\tau}. \tag{5.6}$$

The force balance of the entire snake robot is described below.

$$m\ddot{\mathbf{X}} = \mathbf{F}_x + \mathbf{F}^g_{pxx} + \mathbf{D}^T\mathbf{h}_x + \mathbf{D}^T\mathbf{F}^l_{pyS\theta}, \tag{5.7a}$$

$$m\ddot{\mathbf{Y}} = \mathbf{F}_y + \mathbf{F}^g_{pxy} + \mathbf{D}^T\mathbf{h}_y + \mathbf{D}^T\mathbf{F}^l_{pyC\theta}. \tag{5.7b}$$

where, $\mathbf{u} = [u_1, \cdots\cdots\cdots, u_{n-1}]^T \in \mathbb{R}^{n-1}$, $\mathbf{F}_x = [f_{x,1}, \cdots\cdots\cdots, f_{x,n}]^T \in \mathbb{R}^n$, $\mathbf{F}_y = [f_{y,1}, \cdots\cdots, f_{y,n}]^T \in \mathbb{R}^n$, $\mathbf{h}_x = [h_{x,1}, \cdots\cdots, h_{x,n-1}]^T \in \mathbb{R}^{n-1}$, $\mathbf{h}_y = [h_{y,1}, \cdots\cdots, h_{y,n-1}]^T \in \mathbb{R}^{n-1}$, $\boldsymbol{\tau} = [\tau_1, \cdots\cdots, \tau_n]^T \in \mathbb{R}^n$, $\mathbf{I}_n = \begin{bmatrix} 1 & & \\ & \ddots & \\ & & 1 \end{bmatrix}, \in \mathbb{R}^{n\times n}$. The total force in X-Y direction $\mathbf{F} = [\mathbf{F}_x, \mathbf{F}_y]^T = \mathbf{F_a} + \mathbf{F_d} + \mathbf{F}^g_{px} \in \mathbb{R}^{2n}$, where $\mathbf{F_d} \in \mathbb{R}^{2n}$ is drag force and $\mathbf{F_a} \in \mathbb{R}^{2n}$ is force due to added mass effect. The force acting on CG of the link due to the tail thrust is denoted as $\mathbf{F}^g_{px} = [\mathbf{F}^g_{pxx}, \mathbf{F}^g_{pxy}]^T$. The force $\mathbf{F}_a$ acting on the snake robot due to the added mass effect can be written as below.

$$\mathbf{F}_a = \mathbf{D}_1\mathbf{K}_1\mathbf{S}\mathbf{C}_\theta\ddot{\boldsymbol{\theta}} + \mathbf{D}_1\mathbf{K}_1\mathbf{E}\ddot{\mathbf{p}}_{xy} + \mathbf{D}_1\mathbf{K}_3, \tag{5.8}$$

where μ is added mass parameter, $\mathbf{E} = \begin{bmatrix} \mathbf{e} & \mathbf{0}_{(n,1)} \\ \mathbf{0}_{(n,1)} & \mathbf{e} \end{bmatrix} \in \mathbb{R}^{2n\times 2}$, $\mathbf{D}_1 = -\begin{bmatrix} \mathbf{C}_\theta & -\mathbf{S}_\theta \\ \mathbf{S}_\theta & \mathbf{C}_\theta \end{bmatrix}$ $\begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\mu} \end{bmatrix} \in \mathbb{R}^{2n\times 2n}$, $\boldsymbol{\mu} = \mu\mathbf{I}_n$, $\mathbf{K}_3 = \mathbf{K}_1\mathbf{N}_2 - \mathbf{K}_1\dot{\mathbf{V}}_c + \mathbf{K}_2\mathbf{N}_1 \in \mathbb{R}^{2n\times 1}$. The force due to the drag effect ($\mathbf{F}_d$) can be expressed in terms of linear drag force ($\mathbf{F}_{Ld}$) and nonlinear drag force ($\mathbf{F}_{NLd}$) such as $\mathbf{F}_d = \mathbf{F}_{Ld} + \mathbf{F}_{NLd}$. The linear drag force can be expressed as

$$\mathbf{F}_{Ld} = l\mathbf{Q}_\theta\mathbf{S}\mathbf{C}_\theta\dot{\boldsymbol{\theta}} + \mathbf{Q}_\theta\mathbf{E}\mathbf{v}_r, \tag{5.9}$$

where, $\mathbf{v}_r = \dot{\mathbf{p}} - \mathbf{v}_c$, $\mathbf{Q}_\theta = -\begin{bmatrix} c_t(\mathbf{C}_\theta)^2 + c_n(\mathbf{S}_\theta)^2 & (c_t - c_n)\mathbf{S}_\theta\mathbf{C}_\theta \\ (c_t - c_n)\mathbf{S}_\theta\mathbf{C}_\theta & c_t(\mathbf{S}_\theta)^2 + c_n(\mathbf{C}_\theta)^2 \end{bmatrix}$. The nonlinear drag force can be expressed as

$$\mathbf{F}_{NLd} = -\begin{bmatrix} c_t\mathbf{C}_\theta & -c_n\mathbf{S}_\theta \\ c_t\mathbf{S}_\theta & c_n\mathbf{C}_\theta \end{bmatrix} diag\left(abs \begin{bmatrix} \mathbf{V}_{rx} \\ \mathbf{V}_{ry} \end{bmatrix} \right) \begin{bmatrix} \mathbf{V}_{rx} \\ \mathbf{V}_{ry} \end{bmatrix}, \tag{5.10}$$

The robot changes its position during motion. So, the acceleration of CM in the X-Y plane can be obtained using Eq. 5.7a-5.7b as follows.

$$\ddot{\mathbf{p}}_{xy} = \frac{1}{n} \begin{bmatrix} \mathbf{e}^T\ddot{\mathbf{X}} \\ \mathbf{e}^T\ddot{\mathbf{Y}} \end{bmatrix} = \frac{1}{nm}\mathbf{E}^T\mathbf{F}, \tag{5.11}$$

$\because \mathbf{e}^T\mathbf{D}^T = 0$. where $\mathbf{F}$ is the total force (tail thrust and drag forces) acting on the snake robot. The serpentine motion equation can be derived by substituting $(\mathbf{h_x}, \mathbf{h_y})$ from the Eq. 5.6 to Eq. 5.5 as follows.

$$\begin{aligned} &\mathbf{M}_\theta\ddot{\boldsymbol{\theta}} + \mathbf{W}_\theta diag(\dot{\boldsymbol{\theta}})\dot{\boldsymbol{\theta}} - \mathbf{S}\mathbf{C}_\theta^T(\mathbf{f}_d + \mathbf{A}_g) + \boldsymbol{\Lambda}_1\dot{\boldsymbol{\theta}} + \boldsymbol{\Lambda}_2 diag(\mid \dot{\boldsymbol{\theta}} \mid)\dot{\boldsymbol{\theta}} \\ &- \mathbf{S}\mathbf{C}_\theta^T\mathbf{F}_{px}^g - \mathbf{G}_1 = \mathbf{D}^T\mathbf{u} + \boldsymbol{\Delta}, \end{aligned} \tag{5.12a}$$

$$\ddot{\mathbf{p}}_{xy} = \mathbf{H}_x^{-1}\Big(\mathbf{E}^T\mathbf{f}_d + \mathbf{E}^T\mathbf{D}_1\mathbf{K}_1\mathbf{S}\mathbf{C}_\theta\ddot{\boldsymbol{\theta}} + \mathbf{E}^T\mathbf{D}_1\mathbf{K}_3 + \mathbf{E}^T\mathbf{F}_{px}^g\Big). \tag{5.12b}$$

where,

$$\begin{aligned} &\mathbf{M}_\theta = J\mathbf{I_n} + ml^2\mathbf{S}_\theta\mathbf{V}\mathbf{S}_\theta + ml^2\mathbf{C}_\theta\mathbf{V}\mathbf{C}_\theta - \mathbf{S}\mathbf{C}_\theta^T\mathbf{A}_k + \boldsymbol{\Lambda}_3, \\ &\mathbf{A}_k = \mathbf{D}_1\mathbf{K}_1\mathbf{E}\mathbf{H}_x^{-1}\mathbf{E}^T\mathbf{D}_1\mathbf{K}_1\mathbf{S}\mathbf{C}_\theta + \mathbf{D}_1\mathbf{K}_1\mathbf{S}\mathbf{C}_\theta, \\ &\mathbf{A}_g = \mathbf{D}_1\mathbf{K}_1\mathbf{E}\mathbf{H}_x^{-1}\mathbf{E}^T\mathbf{D}_1\mathbf{K}_3 + \mathbf{D}_1\mathbf{K}_1\mathbf{E}\mathbf{H}_x^{-1}\mathbf{E}^T\mathbf{f}_d + \mathbf{D}_1\mathbf{K}_3, \\ &\mathbf{H}_x = nm\mathbf{I}_2 - \mathbf{E}^T\mathbf{D}_1\mathbf{K}_1\mathbf{E}, \\ &\mathbf{W}_\theta = ml^2\mathbf{S}_\theta\mathbf{V}\mathbf{C}_\theta - ml^2\mathbf{C}_\theta\mathbf{V}\mathbf{S}_\theta, \\ &\mathbf{G}_1 = l\mathbf{S}_\theta\mathbf{K}\mathbf{D}^T\mathbf{F}_{pyS\theta}^l - l\mathbf{C}_\theta\mathbf{K}\mathbf{D}^T\mathbf{F}_{pyC\theta}^l - l\mathbf{C}_\theta\mathbf{A}^T\mathbf{F}_{pyS\theta}^l - l\mathbf{S}_\theta\mathbf{A}^T\mathbf{F}_{pyS\theta}^l, \\ &\mathbf{V} = \mathbf{A}^T(\mathbf{D}\mathbf{D}^T)^{-1}\mathbf{A}, \in \mathbb{R}^{n\times n}, \\ &\boldsymbol{\Lambda_1} = \lambda_1\mathbf{I}_n, \quad \boldsymbol{\Lambda_2} = \lambda_2\mathbf{I}_n, \quad \boldsymbol{\Lambda_3} = \lambda_3\mathbf{I}_n, \end{aligned}$$

Δ is the external disturbances. The reduced dynamical system is written as follows.

$$\ddot{\theta}_n = \hat{\Upsilon}_1 + \hat{\Upsilon}_2 u_\gamma + \hat{\Upsilon}_3 u_{\phi_0} + \tilde{\Upsilon}_{\theta_n}, \tag{5.13a}$$

$$\ddot{\mathbf{p}}_{xy} = \hat{\Upsilon}_4 + \hat{\Upsilon}_5 u_\gamma + \hat{\Upsilon}_6 u_{\phi_0} + \tilde{\Upsilon}_p, \tag{5.13b}$$

where,

$$\begin{aligned}
\hat{\Upsilon}_1 &= -\frac{\mathbf{e}^T\mathbf{M}_\theta\mathbf{H}\ddot{\mathbf{\Phi}}(\gamma)\dot{\gamma}^2}{\mathbf{e}^T\mathbf{M}_\theta\mathbf{e}} - \frac{\mathbf{e}^T}{\mathbf{e}^T\mathbf{M}_\theta\mathbf{e}}\Big(\mathbf{W}_\theta\dot{\theta}^2 - \mathbf{G}_1 - \mathbf{SC}_\theta(\mathbf{f}_d + \mathbf{A}_g) - \mathbf{SC}_\theta^T\mathbf{F}_{px}^g + \mathbf{\Lambda_1}\dot{\boldsymbol{\theta}} \\
&\quad + \mathbf{\Lambda_2} diag(\mid \dot{\boldsymbol{\theta}} \mid)\dot{\boldsymbol{\theta}}\Big), \\
\hat{\Upsilon}_2 &= -\frac{\mathbf{e}^T\mathbf{M}_\theta\mathbf{H}\dot{\mathbf{\Phi}}(\gamma)}{\mathbf{e}^T\mathbf{M}_\theta\mathbf{e}}, \\
\hat{\Upsilon}_3 &= -\frac{\mathbf{e}^T\mathbf{M}_\theta\mathbf{H}\mathbf{b}}{\mathbf{e}^T\mathbf{M}_\theta\mathbf{e}}, \\
\hat{\Upsilon}_4 &= \mathbf{H}_x^{-1}\mathbf{E}^T\mathbf{f}_d + \mathbf{H}_x^{-1}\mathbf{E}^T\mathbf{f}_{px}^g + \mathbf{H}_x^{-1}\mathbf{E}^T\mathbf{D}_1\mathbf{K}_3 + \mathbf{H}_x^{-1}\mathbf{E}^T\mathbf{D}_1\mathbf{K}_1\mathbf{SC}_\theta\mathbf{H}\ddot{\boldsymbol{\phi}}(\gamma)\dot{\gamma}^2 \\
&\quad + \mathbf{H}_x^{-1}\mathbf{E}^T\mathbf{D}_1\mathbf{K}_1\mathbf{SC}_\theta\mathbf{e}\hat{\Upsilon}_1, \\
\hat{\Upsilon}_5 &= \mathbf{H}_x^{-1}\mathbf{E}^T\mathbf{D}_1\mathbf{K}_1\mathbf{SC}_\theta\mathbf{e}\hat{\Upsilon}_2 + \mathbf{H}_x^{-1}\mathbf{E}^T\mathbf{D}_1\mathbf{K}_1\mathbf{SC}_\theta\mathbf{H}\dot{\boldsymbol{\phi}}(\gamma), \\
\hat{\Upsilon}_6 &= \mathbf{H}_x^{-1}\mathbf{E}^T\mathbf{D}_1\mathbf{K}_1\mathbf{SC}_\theta\mathbf{e}\hat{\Upsilon}_3 + \mathbf{H}_x^{-1}\mathbf{E}^T\mathbf{D}_1\mathbf{K}_1\mathbf{SC}_\theta\mathbf{H}\mathbf{b}.
\end{aligned}$$

$\tilde{\Upsilon}_{\theta_n}, \tilde{\Upsilon}_p$ are the lumped disturbances in reduced dynamics of head angle and CM acceleration, respectively. Based on the reduced order system, the controller would be designed which is explained in the next section.

5.3 Controller Design

Snake robot travels in an unknown uncertain underwater environment. Therefore, there is a need for a robust control approach to deal with uncertainties. The control objective is that *to track the desired path with the desired velocity.* The error dynamics of the reduced dynamics can be written by using the Eq. 4.15 as below.

$$\ddot{\tilde{\theta}}_n = \hat{\Upsilon}_1 + \hat{\Upsilon}_2 u_\gamma + \hat{\Upsilon}_3 u_{\phi_0} - \ddot{\theta}_{ref}, \tag{5.14a}$$
$$\dot{\tilde{v}}_t = \mathbf{u}_{\theta_n}^T\hat{\Upsilon}_4 + \mathbf{u}_{\theta_n}^T\hat{\Upsilon}_5 u_\gamma + \mathbf{u}_{\theta_n}^T\hat{\Upsilon}_6 u_{\phi_0} + \dot{\theta}_n v_n - \dot{v}_{ref}. \tag{5.14b}$$

The sliding surface is designed using this error dynamics as follows section 3.2.

$$\boldsymbol{\sigma}(\bar{\xi}) = \begin{bmatrix} \sigma_1(\bar{\xi}) \\ \sigma_2(\bar{\xi}) \end{bmatrix} = \begin{bmatrix} \dot{\tilde{\theta}}_n + k_n\tilde{\theta}_n \\ \dot{\gamma} + k_v\tilde{v}_{t,r} \end{bmatrix}, \tag{5.15}$$

where, $\bar{\xi} = [\tilde{\theta}_n, \dot{\tilde{\theta}}_n, \tilde{v}_{t,r}, v_{n,r}, \gamma, \dot{\gamma}, \phi_0, \dot{\phi}_0]^T$ is state vector, $k_n, k_v > 0$ are the design parameter. The derivative of Eq. 5.15 is used to obtain the reaching law for sliding surface as the following equation.

$$\dot{\sigma}_1(\bar{\xi}) = \ddot{\bar{\theta}}_n + k_n\dot{\bar{\theta}}_n, \tag{5.16a}$$

$$\dot{\sigma}_2(\bar{\xi}) = \ddot{\gamma} + k_v\dot{\bar{v}}_{t,r}. \tag{5.16b}$$

Therefore, Eq. 5.16 can be written by substituting Eq. 5.14 as

$$\dot{\sigma}_1(\bar{\xi}) = \hat{\Upsilon}_1 + \hat{\Upsilon}_2 u_\gamma + \hat{\Upsilon}_3 u_{\phi_0} - \ddot{\theta}_{ref} + k_n\dot{\bar{\theta}}_n + \tilde{\Upsilon}_{\theta_n}, \tag{5.17a}$$

$$\dot{\sigma}_2(\bar{\xi}) = u_\gamma + k_v(\mathbf{u}_{\theta_n}^T\hat{\Upsilon}_4 + \mathbf{u}_{\theta_n}^T\hat{\Upsilon}_5 u_\gamma + \mathbf{u}_{\theta_n}^T\hat{\Upsilon}_6 u_{\phi_0} + \dot{\theta}_n v_n - \dot{v}_{ref}) + u_{\theta_n}^T\tilde{\Upsilon}_p. \tag{5.17b}$$

Now, Eq. 5.17 can represent in sliding vector form as

$$\dot{\boldsymbol{\sigma}}(\bar{\xi}) = \hat{\mathbf{f}}_\sigma + \mathbf{f}_e + \hat{\mathbf{g}}_\sigma\mathbf{u}_\sigma + \bar{\mathbf{f}}_\sigma, \tag{5.18}$$

where,

$$\hat{\mathbf{f}}_\sigma = \begin{bmatrix} \hat{\Upsilon}_1 \\ k_v(\mathbf{u}_{\theta_n}^T\hat{\Upsilon}_4 + \dot{\theta}_n v_n) \end{bmatrix}, \mathbf{f}_e = \begin{bmatrix} k_n\dot{\bar{\theta}}_n - \ddot{\theta}_{ref} \\ -k_v\dot{v}_{ref} \end{bmatrix}, \hat{\mathbf{g}}_\sigma = \begin{bmatrix} \hat{\Upsilon}_2 & \hat{\Upsilon}_3 \\ k_v(1+\mathbf{u}_{\theta_n}^T\hat{\Upsilon}_5) & k_v\mathbf{u}_{\theta_n}^T\hat{\Upsilon}_6 \end{bmatrix},$$

$$\mathbf{u}_\sigma = \begin{bmatrix} u_\gamma \\ u_{\phi_0} \end{bmatrix}, \bar{\mathbf{f}}_\sigma = \begin{bmatrix} \tilde{\Upsilon}_{\theta_n} \\ u_{\theta_n}^T\tilde{\Upsilon}_p \end{bmatrix}.$$

The uncertain parameters are taken in SMC design as nominal values based on the user choice. However, one can not predict the exact uncertainties; if more uncertain parameters exist, it becomes difficult to select. Therefore, the uncertain parameters are estimated here using the actor-critic neural network (i.e. Reinforcement Learning Control). As given in the Eq. 5.18, the uncertain value $\hat{\mathbf{f}}_\sigma$, $\hat{\mathbf{g}}_\sigma$ to be estimated using the actor-critic neural network. $\hat{\mathbf{f}}_\sigma$, $\hat{\mathbf{g}}_\sigma$ are estimated in combined form for the simplicity. Therefore, the sliding surface is modified in Eq. 5.18 according to it by introducing $\mathbf{g}_1 = \begin{bmatrix} 1 & -4.5 \\ 2 & -1 \end{bmatrix}$ positive definite matrix which is explained below. Let, take inverse of $\hat{\mathbf{g}}_\sigma$ in Eq. 5.18,

$$\hat{\mathbf{g}}_\sigma^{-1}\dot{\boldsymbol{\sigma}} = \hat{\mathbf{g}}_\sigma^{-1}\hat{\mathbf{f}}_\sigma + \mathbf{u}_\sigma + \hat{\mathbf{g}}_\sigma^{-1}\bar{\mathbf{f}}_\sigma + \hat{\mathbf{g}}_\sigma^{-1}\mathbf{f}_e. \tag{5.19}$$

Hence, all the term of Eq. 5.19 become uncertain except $\mathbf{u}_\sigma$. Therefore, $\mathbf{g}_1$ a positive definite matrix is introduced in such a way that the uncertain terms become in a single vector $\hat{\mathbf{H}}_\sigma$.

$$\mathbf{g}_1^{-1}\hat{\mathbf{H}}_\sigma = \hat{\mathbf{g}}_\sigma^{-1}\hat{\mathbf{f}}_\sigma + \hat{\mathbf{g}}_\sigma^{-1}\bar{\mathbf{f}}_\sigma + (\hat{\mathbf{g}}_\sigma^{-1} - \mathbf{g}_1^{-1})\mathbf{f}_e - (\hat{\mathbf{g}}_\sigma^{-1} - \mathbf{g}_1^{-1})\dot{\boldsymbol{\sigma}}. \tag{5.20}$$

Substituting Eq. 5.20 into Eq. 5.19, equation becomes

$$\mathbf{g}_1^{-1}\dot{\boldsymbol{\sigma}} = \mathbf{g}_1^{-1}\hat{\mathbf{H}}_\sigma + \mathbf{g}_1^{-1}\mathbf{f}_e + \mathbf{u}_\sigma. \tag{5.21}$$

Now, the Eq. 5.21 becomes

$$\dot{\boldsymbol{\sigma}} = \hat{\mathbf{H}}_\sigma + \mathbf{f}_e + \mathbf{g}_1\mathbf{u}_\sigma + \boldsymbol{\delta}. \tag{5.22}$$

$\boldsymbol{\delta}$ is lumped disturbances and uncertainties which is considered for the stability analysis. This modification helps implement the reinforcement learning super-twisting sliding mode control.

5.3.1 Super Twisting Sliding Mode Control

The equivalent control is taken as shown in Eq. 5.23. The $\mathbf{u}_{eq}$ which consists of the model dynamics is estimated using the actor-critic neural network.

$$\mathbf{u}_{eq} = -\mathbf{g}_1^{-1}\hat{\mathbf{H}}_\sigma - \mathbf{g}_1^{-1}\mathbf{f}_e = -\hat{\mathbf{K}}_\sigma - \mathbf{g}_1^{-1}\mathbf{f}_e \tag{5.23}$$

The existing switching control law is given in Eq. 5.24.

$$\mathbf{u}_{sw} = -\,\mathbf{g}_\sigma^{-1}\left(k_1 sgn(\boldsymbol{\sigma})\|\boldsymbol{\sigma}\|^{\frac{1}{2}} + k_2\boldsymbol{\sigma}\|\boldsymbol{\sigma}\|^{\frac{3}{2}} + \int_0^t k_3 sgn(\boldsymbol{\sigma})\,dt\right) \tag{5.24}$$

where, k_1, k_2, k_3 are the parameters. Therefore, the control law can be expressed as below.

$$\mathbf{u}_\sigma = \mathbf{u}_{eq} + \mathbf{u}_{sw} \tag{5.25}$$

The $\hat{\mathbf{K}}_\sigma$ represents uncertain model dynamics and it is estimated by lumped values (section 3.2) or learning algorithm techniques. The actor-critic reinforcement learning control method is used to estimate the $\hat{\mathbf{K}}_\sigma$ for the proposed work.

5.3.2 Actor-Critic Reinforcement Learning Control Algorithm

The actor-critic reinforcement learning algorithm is used to estimate $\hat{\mathbf{K}}_\sigma$. The actor neural network is designed for the action or the generate to estimate signal. The critic neural network estimates the cost (J_c) using the output of the actor neural network. The procedure for constructing the actor neural network is as follows. The implementation of control law is shown in Fig. 5.2. The actor neural network works as the action of the controller and the critic neural network works as the optimization of the previous action. Based on the action and previous critic data the centre line will be updated of the actor neural network. actor and critic algorithm works as parallel to modify the next result.

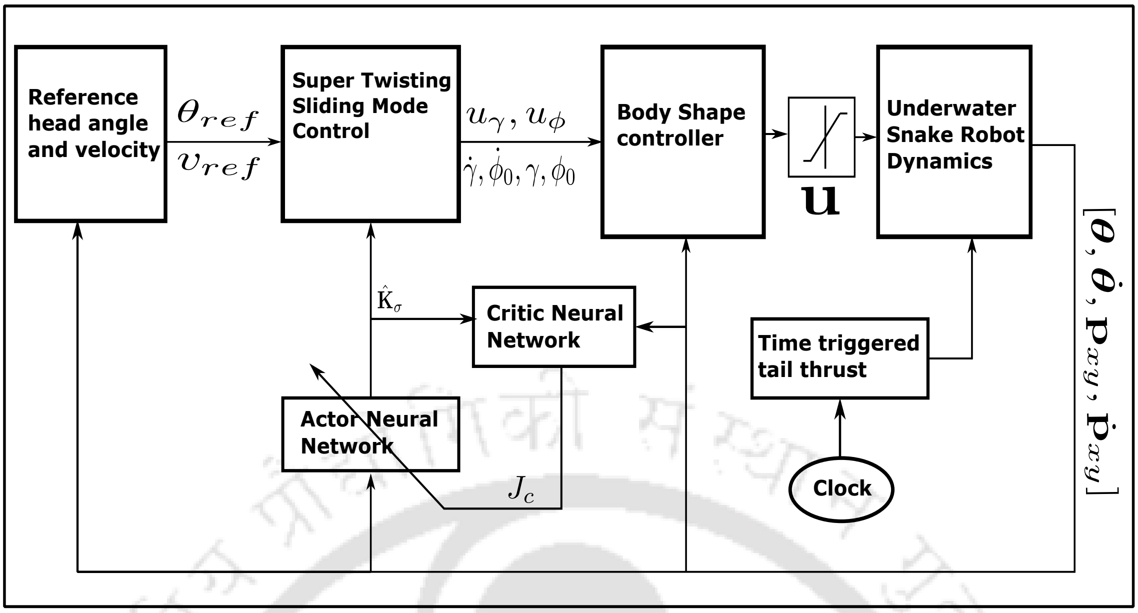

Figure 5.2: Block diagram of control law implementation

Therefore, the high peaks are observed during initial calibration of the control algorithm.

5.3.2.1 Design of Actor Neural Network

The state input $\boldsymbol{\zeta}_a = [\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}, \mathbf{p_{xy}}, \dot{\mathbf{p}}_{\mathbf{xy}}]$ is considered for the actor neural network. The input for the neural network is designed such that the information required to estimate uncertainties could be sufficient. The uncertainties are estimated using the Gaussian-based activation function. The mathematical expression of the Gaussian-based activation function $(h_j)$is given in Eq. 5.26.

$$h_j = exp\left(\frac{-\|(\boldsymbol{\zeta}_{a,j}^T - \underline{\mathbf{C}}(:,j))\|^2}{2\underline{\mathrm{b}}^2}\right) \tag{5.26}$$

The following equation is the output of the actor neural network.

$$\hat{\mathrm{K}}_\sigma = [\mathbf{W^1}, \mathbf{W^2}]^T\mathbf{h} \tag{5.27}$$

where, $\underline{\mathrm{C}}(:,j), \underline{\mathrm{b}}, [\underline{\mathrm{W}}_\mathbf{j}^\mathbf{1}, \underline{\mathrm{W}}_\mathbf{j}^\mathbf{2}]^T$ are centre vector, width and weights, respectively. These parameters can be made adaptive which helps in the improvement of the estimation for the actor output. Hence, these parameters are adaptive based on the sliding surface and critic neural network output error (η_c). The adaption law is decided based on the sliding surface of the coupled dynamical system as shown in Eq. 5.18. The actor neural network algorithm can be seen in Algorithm 5.1. Therefore, the adaption law can be decided for the centre vector and width which is used in the Gaussian-based activation function. The

Algorithm 5.1 Actor Neural Network Algorithm

1: **for** j = 1:9 **do**
2: $\quad n = j + 9$
3: $\quad \boldsymbol{\zeta}_{a,new} = \left[\boldsymbol{\zeta}_a([j, n, \mathbf{p}_{xy}, \dot{\mathbf{p}}_{xy}])\right]^T$ ▷ Input of actor neural network
4: $\quad \underline{\mathbf{C}}(:, j)_{new} = \underline{\mathbf{C}}(:, j)_{old} + \eta_c \|\boldsymbol{\sigma}\|(\zeta_{a,new} - \underline{\mathbf{C}}(:, j)_{old})$ ▷ Center-line adaption
5: $\quad \underline{\mathrm{b}} = \frac{max(abs(\zeta_{a,new} - \underline{\mathbf{C}}(:,j)_{new}))}{\sqrt{8}}$ ▷ Width of the distribution
6: $\quad h(j) = exp\left(\frac{-\|\zeta_{a,new} - \underline{\mathbf{C}}(:,j)_{new}\|^2}{2\underline{\mathrm{b}}^2}\right)$ ▷ Gaussian activation function
7: **end for**
8: $\underline{\mathbf{C}}_{old} = \underline{\mathbf{C}}_{new}$

adaption law for the weight matrix is proposed as follows.

$$\begin{bmatrix} \dot{\underline{\mathbf{W}}}^{\mathbf{1}}(\mathbf{t}) \\ \dot{\underline{\mathbf{W}}}^{\mathbf{2}}(\mathbf{t}) \end{bmatrix} = P_1 \boldsymbol{\sigma} \mathbf{h}^T \tag{5.28}$$

This equation would get integrated and utilized for the weight updating as follows.

$$\begin{bmatrix} \mathbf{W^1}(\mathbf{t}) \\ \mathbf{W^2}(\mathbf{t}) \end{bmatrix} = \begin{bmatrix} \mathbf{W^1}(\mathbf{t} - \boldsymbol{\Delta}\mathbf{t_1}) \\ \mathbf{W^2}(\mathbf{t} - \boldsymbol{\Delta}\mathbf{t_1}) \end{bmatrix} + \eta_a \begin{bmatrix} \underline{\mathbf{W}}^{\mathbf{1}}(\mathbf{t}) \\ \underline{\mathbf{W}}^{\mathbf{2}}(\mathbf{t}) \end{bmatrix} \tag{5.29}$$

The weight adaption law is designed based on the sliding surface and actor error function (η_a). The actor error function (η_a) is described as follows.

$$\eta_a = J_c - \Lambda_u \tag{5.30}$$

where, Λ_u is ultimate target to value of entire cost-to-go function, usually it is kept 0. Therefore, this weight adaption is designed such that if no dynamic uncertainties exist in the system, η_a would be zero. $\boldsymbol{\sigma}$ may not be zero due to the chattering effect in the sliding mode control scheme. Therefore, either $\boldsymbol{\sigma}$ or η_a becomes zero would make weight adaptation constant once the uncertainties become constant. The estimated ($\hat{\mathbf{K}}_\sigma$) is inputted to the STSMC.

5.3.2.2 Design of Critic Neural Network

The critic neural network is used to estimate the cost-to-go function. The estimated $\hat{\mathbf{K}}_\sigma$ output of the actor neural network is the input for the critic neural network along with state input. The input state for critic neural network is considered as $\boldsymbol{\zeta}_c = [\boldsymbol{\theta}, \mathbf{p_{xy}}, \hat{\mathbf{K}}_\sigma]$ which consists the position states and estimated uncertainties. In the critic neural network, the centre vector is kept constant. The width of a neural network is adjusted with the input. The algorithm of the critic neural network is explained in Algorithm 5.2. The

Algorithm 5.2 Critic Neural Network Algorithm

for i= 0:t **do**
  for j = 1:9 **do**
    $\boldsymbol{\zeta}_{c,new} = \left[\boldsymbol{\zeta}_c([j, \mathbf{p}_{xy}, \hat{\mathbf{K}}_\sigma])\right]^T$ ▷ Input of critic neural network
    $\underline{\mathrm{b}} = \frac{max(abs(\boldsymbol{\zeta}_{c,new} - \mathbf{C}(:,j)))}{\sqrt{8}}$ ▷ Width of the distribution
    $h(j) = exp\left(\frac{-\|\boldsymbol{\zeta}_{c,new} - \mathbf{C}(:,j)\|^2}{2\underline{\mathrm{b}}^2}\right)$ ▷ Gaussian activation function
  end for
  $\mathbf{W}_c(i) = \mathbf{W}_c(i-1) + \|\boldsymbol{\sigma}\|\eta_c(i-1)\mathbf{h}$ ▷ Width of critic neural network
  $J_c(i) = \mathbf{W}_c^T(i)\mathbf{h}$ ▷ Cost-to-go function
end for

reward function (r_d) is designed such that it can evaluate the reward according to the improvement/deterioration in the tracking error as follows.

$$r_d = [\tilde{\theta}_n, \tilde{v}_{t,r}]\mathrm{diag}(P_r)[\tilde{\theta}_n, \tilde{v}_{t,r}]^T, \tag{5.31}$$

where, $\mathbf{P}_r$ is positive definite matrix. The critic neural network error function (η_c) is formulated based on the cost-to-go function (J_c) and reward function (r_d) as follows.

$$\eta_c(t) = \beta_c J_c(t) - (J_c(t - \Delta t_2) - r_d(t)). \tag{5.32}$$

The output of the critic neural network is input for the actor neural network error function. This simultaneous process of the actor-critic neural network would improve the estimation of the $\hat{\mathbf{K}}_\sigma$. The implementation of the proposed control scheme would be tested with different external disturbances and parametric uncertainties by keeping the same control parameters. These control parameters are identified by a trial and error approach. The implementation of the proposed control scheme is shown in Fig. 5.2. The critic neural network analyses the actor's estimated values and calculates the cost. This cost improves the weights in the actor neural network. This type of dynamic programming helps to estimate the optimal model online.

5.3.3 Lyapunov Stability

Let consider the ideal neural network weight $\mathbf{W}^*$ for actual $\mathbf{K}_\sigma$ such as

$$\mathbf{K}_\sigma = \mathbf{W}^{*T}\mathbf{h}.$$

Therefore, the error function is formulated as $\tilde{\mathbf{K}}_\sigma = \mathbf{K}_\sigma - \hat{\mathbf{K}}_\sigma$ and $\tilde{\mathbf{K}}_\sigma = \tilde{\mathbf{W}}_a^T\mathbf{h}$, where, $\tilde{\mathbf{W}}_a = \mathbf{W}_a^* - \hat{\mathbf{W}}_a$ and $\mathbf{W}_a^* > \hat{\mathbf{W}}_a$. Substituting this error function in Eq. 5.18, the equation becomes

$$\dot{\boldsymbol{\sigma}}(\bar{\xi}) = \tilde{\mathbf{K}}_{\sigma} + \mathbf{g}_1\mathbf{u}_{sw} + \boldsymbol{\delta}. \tag{5.33}$$

The Lyapunov candidate V_1 can be defined as

$$\begin{aligned}
\mathbf{V_1} &= \frac{1}{2}\boldsymbol{\sigma}^T\boldsymbol{\sigma} + \frac{1}{2P_1}\tilde{\mathbf{W}}_{\mathbf{a}}^T\tilde{\mathbf{W}}_{\mathbf{a}} \\
\dot{\mathbf{V}}_1 &= \boldsymbol{\sigma}^T\dot{\boldsymbol{\sigma}} + \frac{1}{P_1}\tilde{\mathbf{W}}_{\mathbf{a}}^T\dot{\tilde{\mathbf{W}}}_{\mathbf{a}} \\
&= \boldsymbol{\sigma}^T\left(\tilde{\mathbf{K}}_{\sigma} - k_1 sgn(\boldsymbol{\sigma})\|\boldsymbol{\sigma}\|^{\frac{1}{2}} - k_2\boldsymbol{\sigma}\|\boldsymbol{\sigma}\|^{\frac{3}{2}} - \int_0^t k_3 sgn(\boldsymbol{\sigma})\,dt\right) - \tilde{\mathbf{W}}_a^T\boldsymbol{\sigma}^T\mathbf{h} + \|\boldsymbol{\sigma}\|\|\boldsymbol{\delta}\| \\
&\text{After substituting } \tilde{\mathbf{K}}_{\sigma} = \tilde{\mathbf{W}}_a^T\mathbf{h} \\
&\leq -k_1\|\boldsymbol{\sigma}\|^{\frac{3}{2}} - k_2\|\boldsymbol{\sigma}\|^{\frac{7}{2}} - \|\boldsymbol{\sigma}\|\int_0^t k_3 dt + \|\boldsymbol{\sigma}\|\|\boldsymbol{\delta}\| \\
&\leq 0,
\end{aligned}$$

where, k_1, k_2, k_3 are the positive gain parameters. The stability of the actor-critic algorithm can be analysed as follows. $\tilde{\eta}_a = \eta_a^* - \eta_a$ where $\eta_a^* > \eta_a$, $\tilde{\eta}_c = \eta_c^* - \eta_c$ where $\eta_c^* > \eta_c$.

$$\begin{aligned}
V_2 &= \frac{1}{2}\tilde{\eta}_a^2 + \frac{1}{2}\tilde{\eta}_c^2 \\
\dot{V}_2 &= -\tilde{\eta}_a\dot{\eta}_a - \tilde{\eta}_c\dot{\eta}_c \\
&= -\tilde{\eta}_a\dot{J}_c - \tilde{\eta}_c\beta\dot{J}_c - \tilde{\eta}_c\|r_d\| \\
&J_c \text{ depends on } \tilde{W}_c \\
&\therefore \\
&\leq -\tilde{\eta}_a\frac{\partial J_c}{\partial\tilde{W}_c}\frac{d\tilde{W}_c}{dt} - \tilde{\eta}_c\beta\frac{\partial J_c}{\partial\tilde{W}_c}\frac{d\tilde{W}_c}{dt} - \tilde{\eta}_c\|r_d\| \\
&\leq -\tilde{\eta}_c\|r_d\| \quad \left(\because \frac{d\tilde{W}_c}{dt} = 0\text{as per the iteration}\right) \\
&\leq 0.
\end{aligned}$$

Here, two Lyapunov candidates are considered. The first candidate shows the stability of system after implementing STSMC and the second part shows the stability of the proposed actor-critic neural network stability. If neural network controller is unstable, there might chance of the instability in dynamic system. Therefore, two separate Lyapunov candidates are shown to verify the system and proposed actor-critic network stability.

5.4 Results and Discussion

The mass $m = 0.9047$ kg, length $2l = 0.18$ m is considered for each link of a snake robot. It is assumed that the tail thrust($f_p = 0.4$ N) is constant which means the spiral-type tail constantly moves in one direction. The water current velocity $v_{c,x} = -0.01$ m/s and $v_{c,y} = -0.01$ m/s are considered for X-Y direction respectively. The control parameters for the actor-critic neural network are shown in Table 5.1.

Table 5.1: Control Parameters

Actor-critic	$P_1 = 5$, $\beta_c = 0.5$
Sliding surface	$k_n = 10$, $k_v = 16$
Super-Twisting	$k_1 = 10$, $k_2 = 10$, $k_3 = 2$
Body-Shape parameters	$\alpha = \frac{\pi}{6}$ rad, $\beta = \frac{5\pi}{18}$ rad, $k_p = 20$, $k_d = 5$, $k_i = 10$

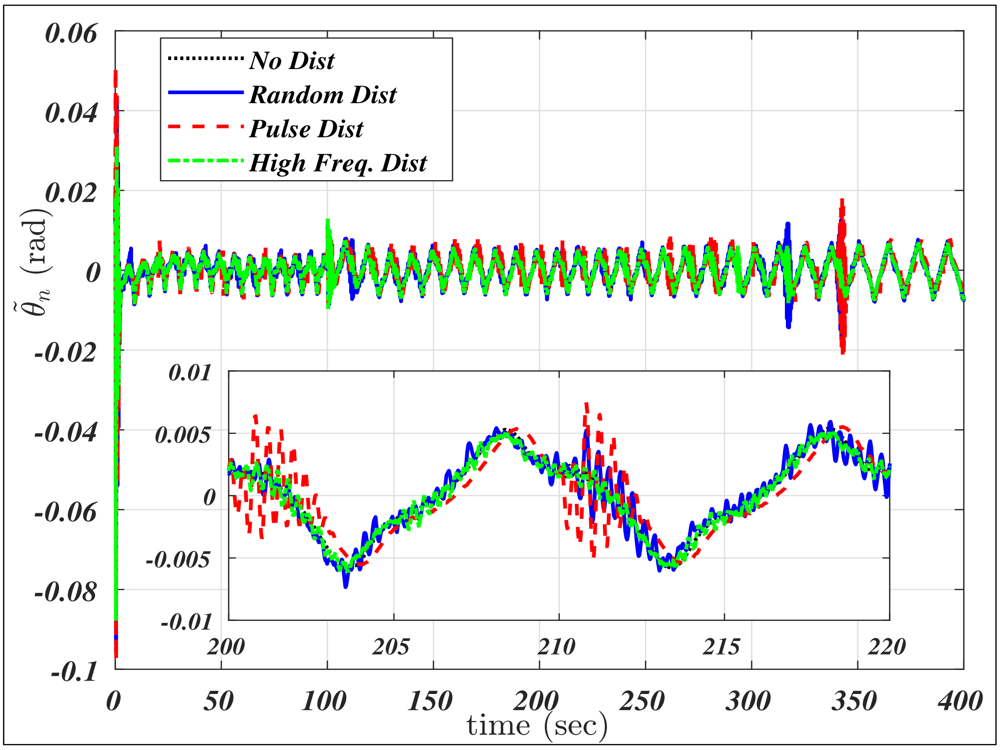

Figure 5.3: Head angle tracking error

The proposed control scheme is compared with the existing STSMC scheme. The control parameters for comparing both control schemes are kept the same except for the actor-critic parameter because it is not in the existing control (section 2.3) scheme. The lumped value of the uncertain term $\hat{\mathbf{K}}_\sigma$ is considered $[0.5, -0.5]^T$ for the existing control scheme which is nearer to estimated uncertain values of the proposed control scheme. It is noted that these lumped values are not used for the proposed ACSTSMC scheme.

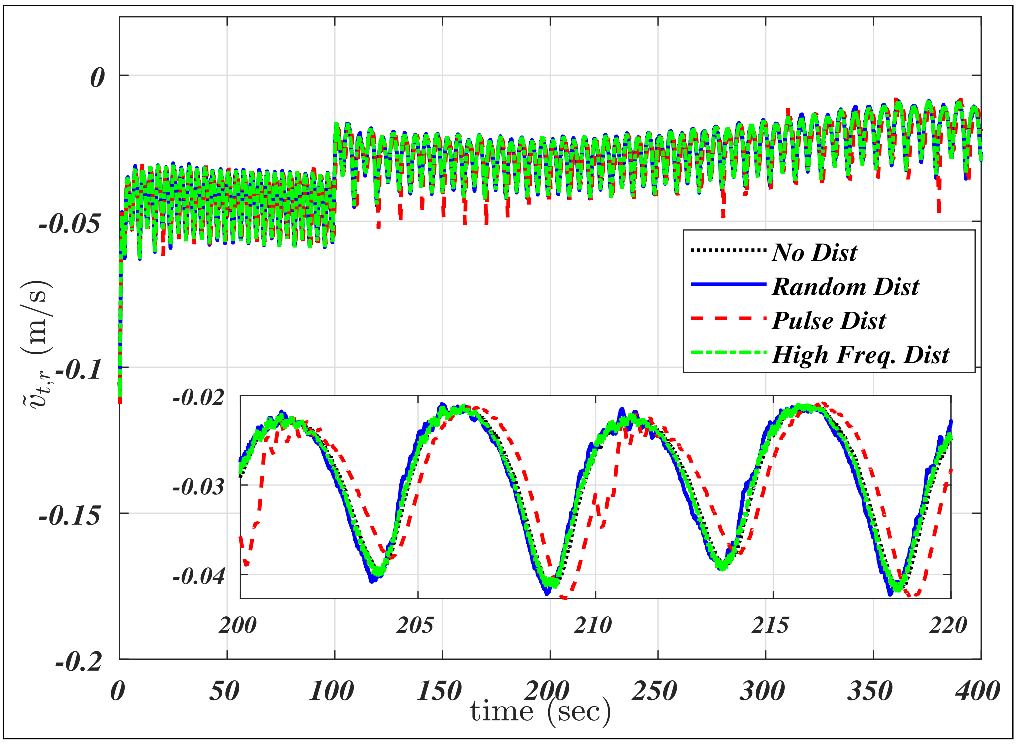

Figure 5.4: Velocity tracking error

Figure 5.3 shows the tracking error of the head angle ($\tilde{\theta}_n$). It is observed that the tracking error is near zero. Some peaks are observed during the pulse disturbance condition however those are within the limit. The tangential velocity is the next target for the tracking to fulfil the control objective. Figure 5.4 shows the tracking error of the tangential velocity. Error improvement is observed when the tail thrust force is in action. It is noted that the tail-thruster is started after 100 sec to show the effect of the tail thrust in a proposed dynamic system. This proves the proposed dynamic model of the underwater snake robot and its effect on each link are correct. This is also verified by observing the joint velocity tracking ($\dot{\gamma}$) in Fig. 5.5.

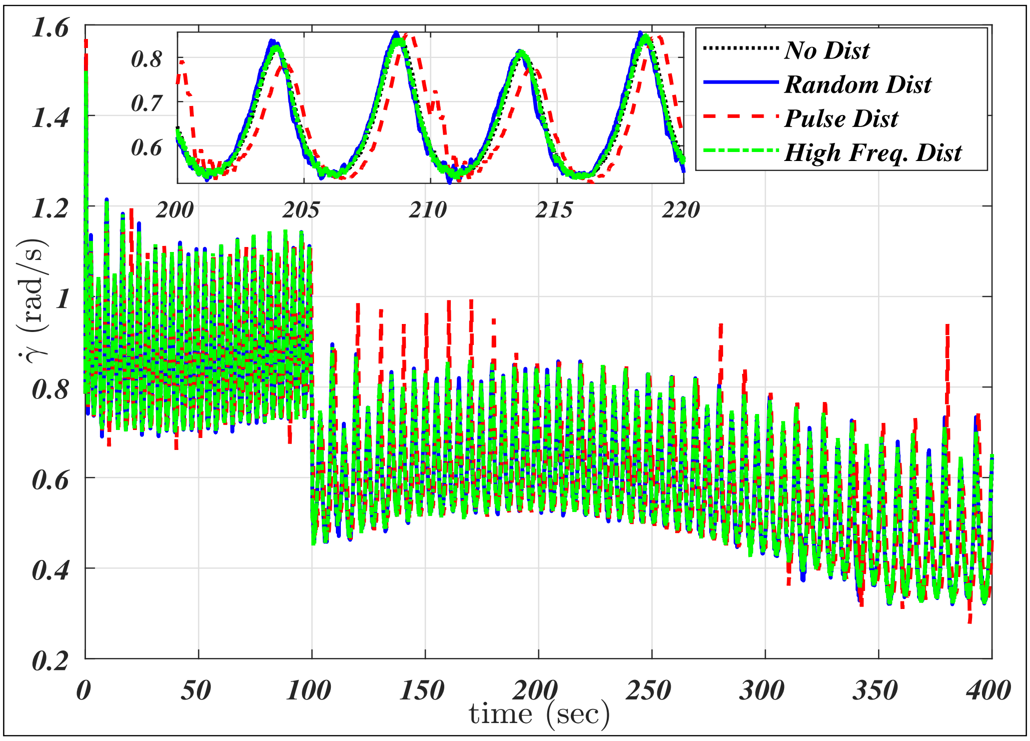

Figure 5.5: Joint angular velocity

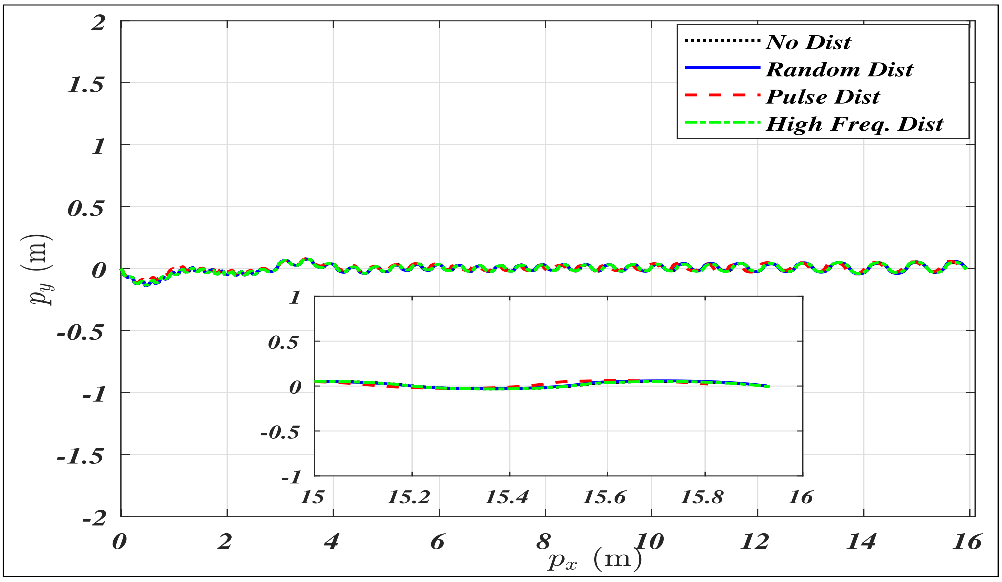

Figure 5.6: CM position of snake robot

Initially, the joint velocity is higher before the thruster starts as shown in Fig. 5.5. Once, the thrust force comes across the link, the controller decides autonomously to reduce the joint speed. The reduction in the joint speed is observed after 100 sec. This has happened to maintain the control objective to track the desired velocity. The straight path is considered to follow by the snake robot. It is shown in Fig. 5.6. Almost the same speed of the snake robot is observed using the proposed ACSTSMC scheme for all disturbance conditions as per Fig. 5.6. The phase angle (ϕ_0) keeps the robot on track. The phase angle is shown in Fig. 5.7.

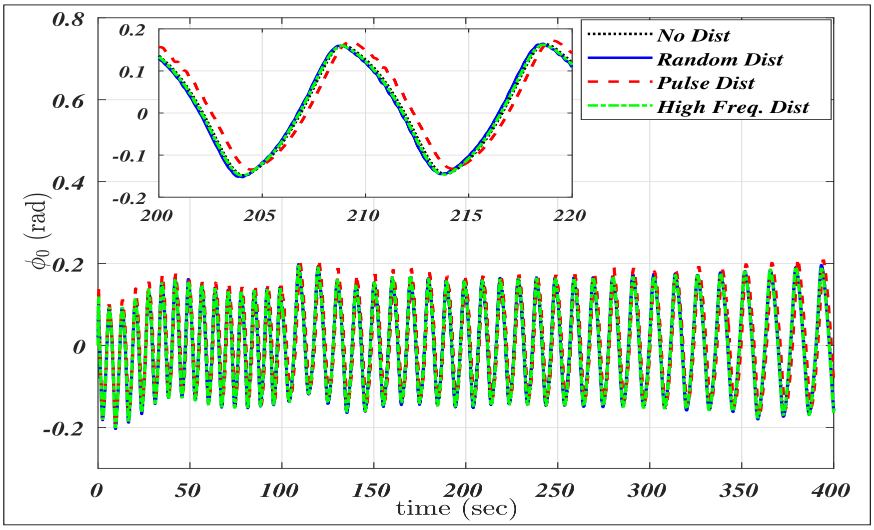

Figure 5.7: Phase angle

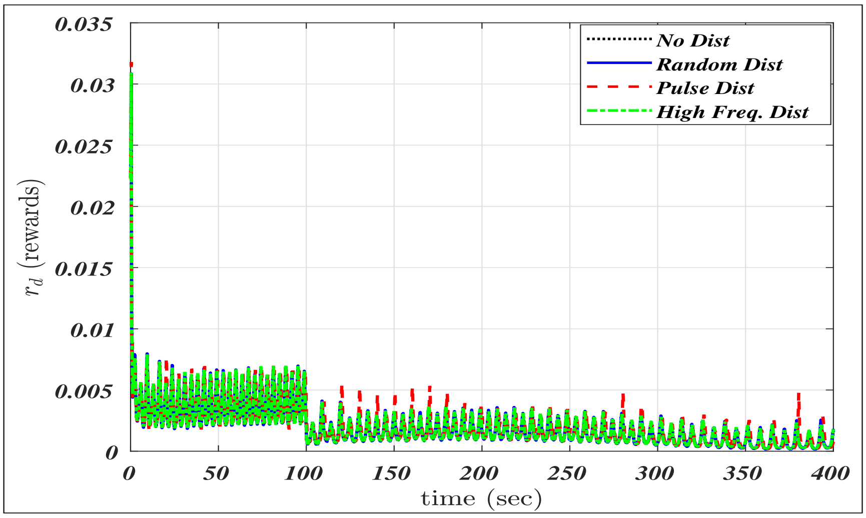

Figure 5.8: Rewards of the actor-critic neural network

The reward is an important part of the critic network. The reward value of each iteration is shown in Fig. 5.8. This reward value is used to decide the critic output which is used for adapting the centre vector and weights of the actor neural network. It helps to estimate the uncertainties in the dynamic system. The estimated uncertainties are shown in Fig. 5.9 for the different cases of the disturbances. Variations in estimated values are observed for the different cases of disturbances.

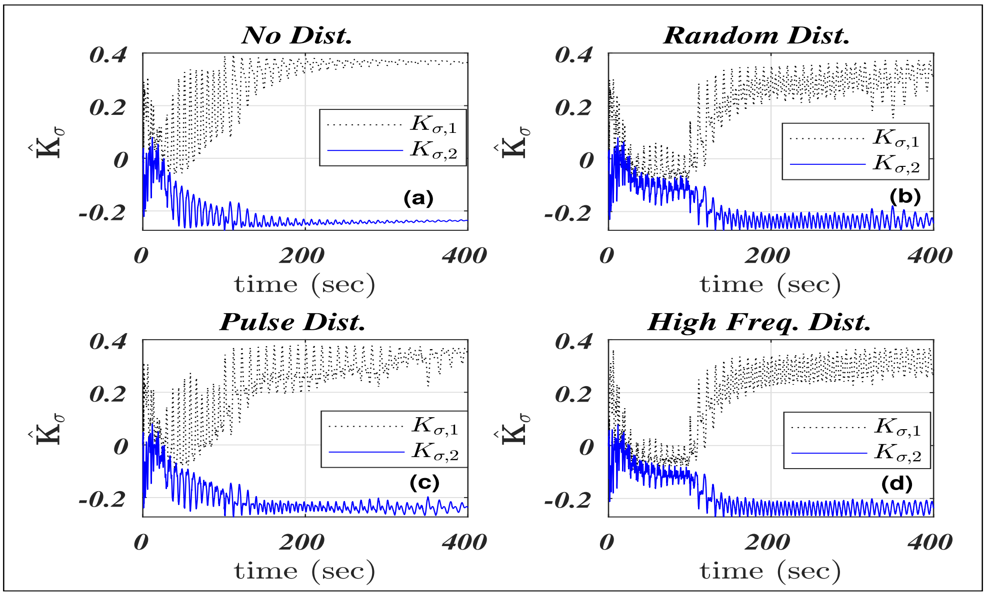

Figure 5.9: Estimated uncertainties

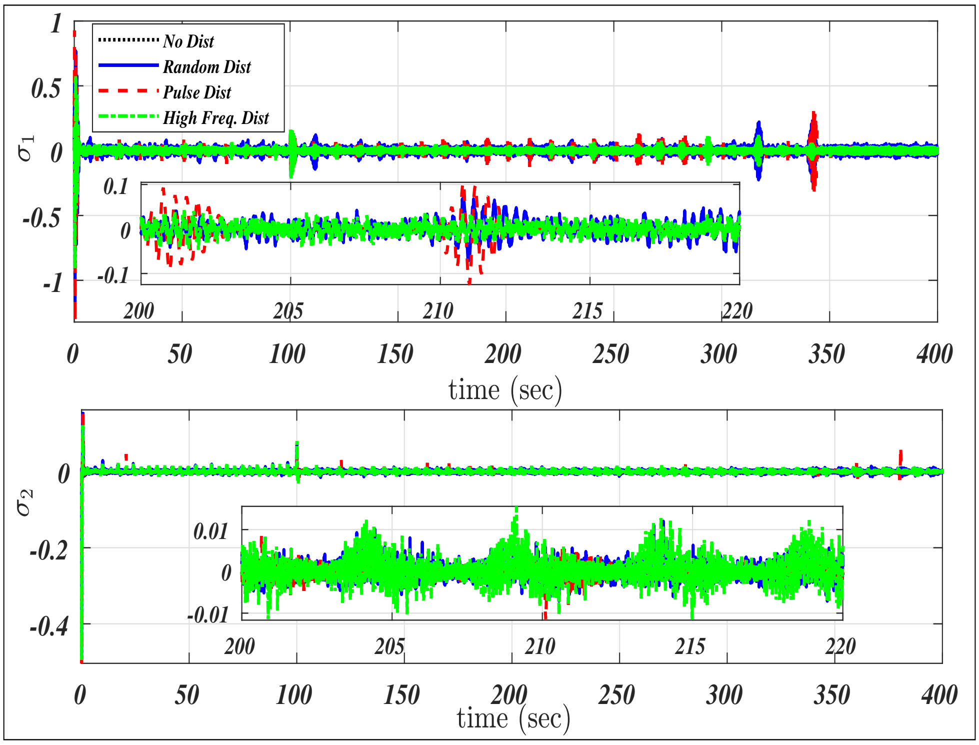

Figure 5.10: Sliding surface

Figure 5.10 shows the sliding surface. It is needed to be zero to achieve the control objective. There is also less chattering required in sliding surfaces for smooth tracking operation. The minimum chattering effect in sliding surfaces results in a reduction in chattering in torque. This can be observed in Fig. 5.11. It is also observed that using the proposed control scheme not only required less control effort but also reduced the chattering effect. It is noted that the control saturation is used in this work to meet actuator constraints.

The exact quantified results of the chattering effect are shown in Table 5.2. Case-1 is "No Disturbance", Case-2 is "Random Disturbance", Case-3 is "Pulse Disturbance" and Case-4 is "High-Frequency Disturbance" are considered. The results are compared for the proposed control technique and the existing control technique for each case. It is observed that a sufficient reduction in chattering is possible by using the proposed control scheme. Moreover, the minimum chattering happens for the different cases of disturbances by using the proposed control scheme. The required torque for the motion of the snake robot is shown in Table 5.3, namely control effort. Generally, the minimum control effort is required for the long life of the battery and also increases the manoeuvrability of the robot. Therefore, it is necessary to reduce the control effort. The control scheme is designed not only for the reduction in chattering and control effort but also for the improvement of the tracking error. The tracking error comparison is shown in Table 5.4. The improvement in tracking errors is found to be better using the proposed control scheme compare to existing STSMC.

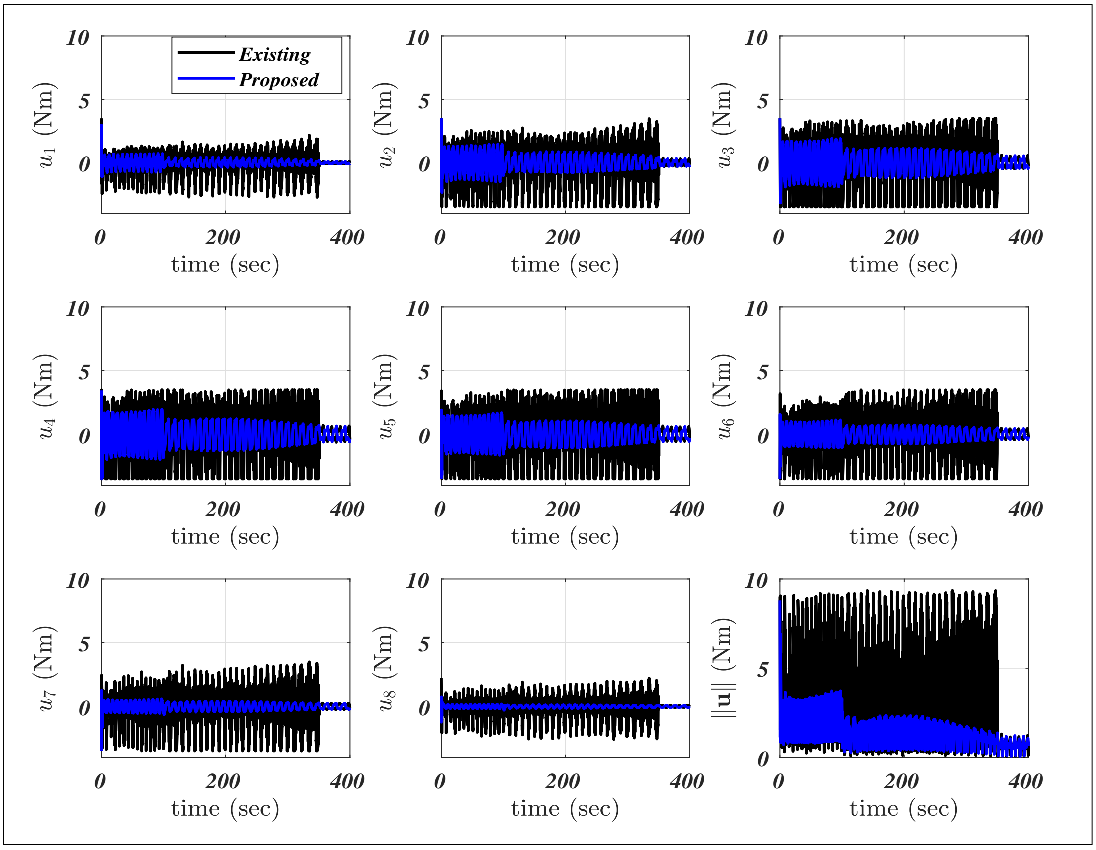

Figure 5.11: Torque comparison of existing and proposed control scheme

Table 5.2: Chattering Effect

	Case-1		Case-2		Case-3		Case-4	
	Proposed	Existing	Proposed	Existing	Proposed	Existing	Proposed	Existing
u_1	4.1670	23.0880	12.9250	20.7500	4.6150	20.6960	6.3400	20.8060
u_2	2.0860	17.1450	3.5460	15.3410	2.4810	15.4360	3.2200	15.4360
u_3	1.4760	20.3230	3.0500	18.3240	2.0780	18.3220	2.4370	18.3850
u_4	1.3750	21.9880	2.9300	19.9140	2.0590	19.9460	2.4240	19.9950
u_5	1.3060	21.2320	3.0710	19.1440	1.9830	19.1090	2.7380	19.1740
u_6	1.1760	19.4930	3.4040	17.5650	2.0440	17.6170	3.0810	17.6310
u_7	1.1020	17.2280	3.9100	15.4580	2.1450	15.4570	4.0890	15.5410
u_8	1.0420	22.2430	12.0510	19.9710	2.1310	19.8240	6.1110	19.9630
$\|\mathbf{u}\|$	2.5710	38.7670	5.1550	35.0400	3.9950	34.9990	4.7430	35.1760
$\tilde{v}_{t,r}$	0.0197	0.1022	0.0375	0.0940	0.0214	0.0939	0.0368	0.0941
$\tilde{\theta}_n$	0.0061	0.5183	0.0211	0.4629	0.0211	0.4634	0.0186	0.4641

Table 5.3: Control effort

	Case-1		Case-2		Case-3		Case-4	
	Proposed	Existing	Proposed	Existing	Proposed	Existing	Proposed	Existing
u_1	0.2524	0.6927	0.2559	0.6300	0.2572	0.6319	0.2559	0.6329
u_2	0.6360	1.3922	0.6396	1.2786	0.6446	1.2800	0.6379	1.2815
u_3	0.9365	1.8481	0.9406	1.7076	0.9464	1.7078	0.9384	1.7094
u_4	1.0105	2.0027	1.0150	1.8518	1.0190	1.8521	1.0127	1.8536
u_5	0.8595	1.8974	0.8648	1.7519	0.8669	1.7513	0.8621	1.7536
u_6	0.5831	1.6394	0.5897	1.5007	0.5905	1.5014	0.5863	1.5036
u_7	0.2989	1.2315	0.3072	1.1177	0.3070	1.1187	0.3041	1.1214
u_8	0.0885	0.6305	0.1005	0.5696	0.0982	0.5708	0.1021	0.5727
$\|\mathbf{u}\|$	1.8821	4.2481	1.8946	3.9079	1.9026	3.9093	1.8891	3.9143

Table 5.4: Tracking Error

	Case-1		Case-2		Case-3		Case-4	
	Proposed	Existing	Proposed	Existing	Proposed	Existing	Proposed	Existing
$\tilde{v}_{t,r}$	0.0308	0.0327	0.0308	0.0323	0.0312	0.0323	0.0308	0.0323
$\tilde{\theta}_n$	0.0032	0.0500	0.0034	0.0445	0.0035	0.0446	0.0032	0.0446

5.5 Summary

In this work, the equation of motion for the snake robot equipped with a thruster at the tail is derived on each link. The VHCs approach is used for deriving the reduced dynamics. The STSMC with reinforcement learning algorithm is used to estimate the uncertainties and disturbances in the motion dynamics. The centre vector, width and weight adaption algorithms are described for the actor-critic neural network. An improvement in the tracking errors of the head angle and tangential velocity is found. The reduction in the chattering effect and control effort is found using the proposed control scheme. The results are qualitatively and quantitatively compared with the existing control scheme and the improved performance is found using the proposed control scheme.

6

Motion Control of an Aquatic Snake Robot with Actuators Failure using Neural Network STSMC

Contents

6.1 Introduction

In this work, the snake robot is modelled and simulated in the Simscape multibody environment, eliminating the need for explicit mathematical modelling. Due to the precence of large number of actuators, there may be situation in which single or more actuators may be failed. Understanding the robot's motion in the event of such failures is crucial. Hence, this work examines different scenarios of actuator malfunction. Motion control is achieved through virtual holonomic constraints (VHCs) based approach, considering various uncertainties in underwater environments. To deal with these uncertainties, the model-based robust control approach known sliding mode control (SMC) is applied to a simplified dynamic system, which is formulated through VHCs. A radial basis function (RBF) neural network estimates the model, which is then integrated into the SMC control strategy. To mitigate the chattering issue associated with SMC, the STSMC strategy is employed. Consequently, the RBF-STSMC scheme is proposed for controlling the motion of a snake robot in underwater conditions during single or multiple actuator failures and uncertain environments. This proposed scheme is evaluated against existing control strategies.

The chapter is organized as follows. The snake robot modelling in the Simscape multibody environment and various cases of the actuators failure are given the section 6.2. The RBF-STSMC scheme is explained in the section 6.3. The obtained results are discussed in the section 6.4 and the chapter is summarized in the section 6.5.

6.2 Simscape Modelling of the Snake Robot

The mathematical model of a snake robot is widely discussed in the chapter 2. In this chapter, the snake robot is modelled within the Simscape multibody framework. Here, it is not necessary to explicitly use the equation of motion for the snake robot. Instead, the construction of the robot utilizes the link and joint blocks available in the Simscape library, as illustrated in Fig. 6.1. The assembly process involves connecting the 'P_link' (frame) of 'Link 8' to the 'F node' of the joint, and linking the 'B node' of the joint to the 'N_link' (frame) of 'Link 9', as depicted in Fig. 6.1. The terms 'P_link' and 'N_link' represent the frames of the preceding and subsequent links, respectively, which are connected via the joint. Environmental uncertainties are introduced by applying external forces to the 'F3 node' of the links. The 't node' on the joint indicates the input

[5]This chapter is part of publication: B. M. Patel, R.G. Valenti & S. K. Dwivedy (2024). Motion control of an aquatic snake robot with actuators failure and environmental uncertainties using neural network super twisting sliding mode control. Journal of Mechanisms and Robotics, (Submitted).

torque applied to the joint, whereas 'q' and 'w' represent the angular position and angular velocity, respectively. Sensors are attached to the 'R node' on the link to measure the link states, which are crucial for controller design. Following this methodology, the entire robot is constructed in the Simscape environment, resulting in the assembled snake robot as shown in Fig. 6.2.

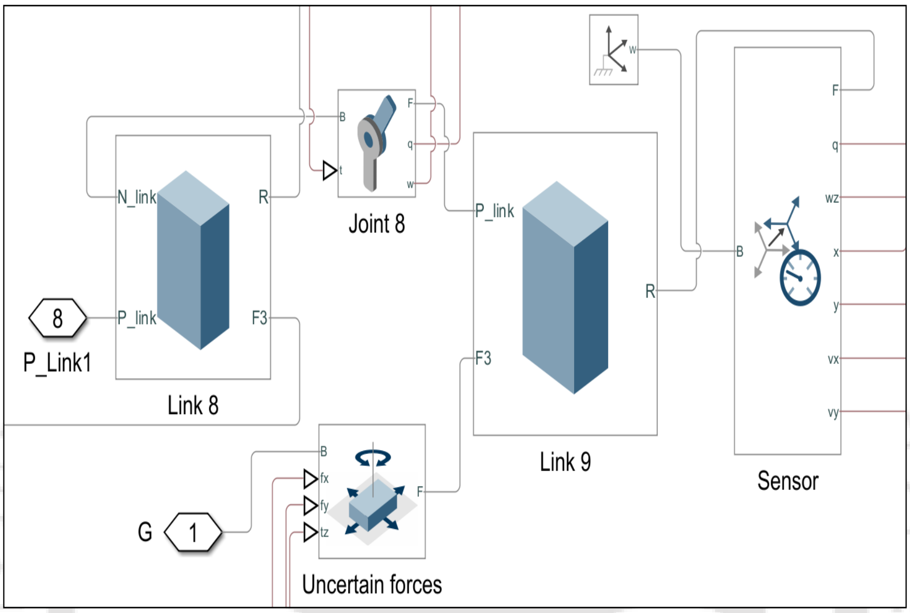

Figure 6.1: Simscape model of Link and Joint

To design the model-based controller, it is necessary to know the mathematical model of a snake robot. Here, a summary of the essential mathematical model required for the controller design is provided. The detailed equation of motion is given in the chapter 2 which are again written below for the completeness.

$$\mathbf{M}_{\boldsymbol{\theta}}\ddot{\boldsymbol{\theta}} + \mathbf{W}_{\boldsymbol{\theta}}\dot{\boldsymbol{\theta}}^2 + \mathbf{\Lambda_1}\dot{\boldsymbol{\theta}} + \mathbf{\Lambda_2}diag(|\dot{\boldsymbol{\theta}}|)\dot{\boldsymbol{\theta}} - l\mathbf{S}\mathbf{C}_\theta^T\mathbf{f}_d = \mathbf{D}^T\mathbf{u}, \tag{6.1a}$$

$$\ddot{\mathbf{p}}_{xy} = \frac{1}{nm}\mathbf{E}^T\mathbf{f}_d. \tag{6.1b}$$

where,

$$\mathbf{M}_\theta = J\mathbf{I_n} + ml^2\mathbf{S}_\theta\mathbf{V}\mathbf{S}_\theta + ml^2\mathbf{C}_\theta\mathbf{V}\mathbf{C}_\theta,$$
$$\mathbf{W}_\theta = ml^2\mathbf{S}_\theta\mathbf{V}\mathbf{C}_\theta - ml^2\mathbf{C}_\theta\mathbf{V}\mathbf{S}_\theta,$$
$$\mathbf{V} = \mathbf{A}^T(\mathbf{D}\mathbf{D}^T)^{-1}\mathbf{A}, \in \mathbb{R}^{n\times n},$$
$$\mathbf{\Lambda_1} = \lambda_1\mathbf{I}_n, \quad \mathbf{\Lambda_2} = \lambda_2\mathbf{I}_n,$$
$$\mathbf{f}_{Ld} = l\mathbf{Q}_\theta\mathbf{S}\mathbf{C}_\theta\dot{\boldsymbol{\theta}} + \mathbf{Q}_\theta\mathbf{E}\mathbf{v}_r, \quad where \quad \mathbf{v}_r = \dot{\mathbf{p}} - \mathbf{v}_c,$$

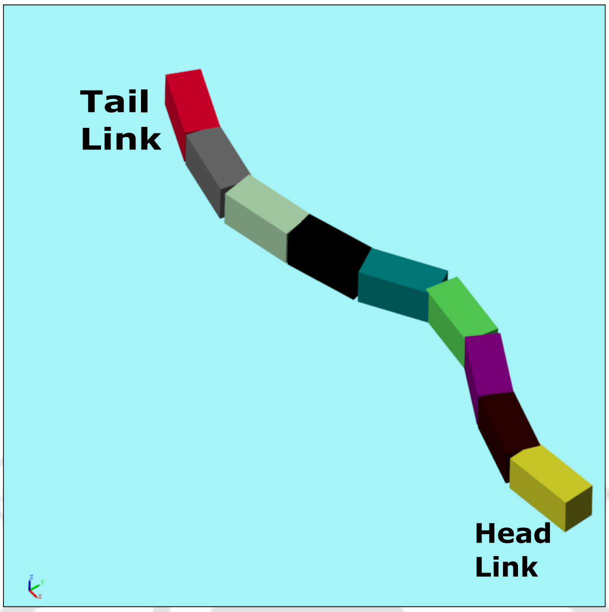

Figure 6.2: Visualization model of snake robot

$$
\begin{aligned}
&\mathbf{f}_{NLd} = -\begin{bmatrix} c_t\mathbf{C}_\theta & -c_n\mathbf{S}_\theta \\ c_t\mathbf{S}_\theta & c_n\mathbf{C}_\theta \end{bmatrix} diag\left(abs \begin{bmatrix} \mathbf{V}_{rx} \\ \mathbf{V}_{ry} \end{bmatrix} \right) \begin{bmatrix} \mathbf{V}_{rx} \\ \mathbf{V}_{ry} \end{bmatrix}, \\
&\mathbf{f}_d = \mathbf{f}_{Ld} + \mathbf{f}_{NLd}, \\
&\begin{bmatrix} \mathbf{V}_{rx} \\ \mathbf{V}_{ry} \end{bmatrix} = \begin{bmatrix} \mathbf{C}_\theta & \mathbf{S}_\theta \\ -\mathbf{S}_\theta & \mathbf{C}_\theta \end{bmatrix} \begin{bmatrix} \dot{\mathbf{X}} - \mathbf{V_x} \\ \dot{\mathbf{Y}} - \mathbf{V_y} \end{bmatrix}, \\
&\mathbf{SC}_\theta = \begin{bmatrix} \mathbf{K}^T \boldsymbol{S_\theta} \\ -\mathbf{K}^T \boldsymbol{C_\theta} \end{bmatrix}, \in \mathbb{R}^{2n\times n} \\
&\mathbf{Q}_\theta = -\begin{bmatrix} c_t(\mathbf{C}_\theta)^2 + c_n(\mathbf{S}_\theta)^2 & (c_t - c_n)\mathbf{S}_\theta\mathbf{C}_\theta \\ (c_t - c_n)\mathbf{S}_\theta\mathbf{C}_\theta & c_t(\mathbf{S}_\theta)^2 + c_n(\mathbf{C}_\theta)^2 \end{bmatrix}, \\
&\mathbf{V_x} = \mathbf{e}v_{cx}, \quad \mathbf{V_y} = \mathbf{e}v_{cy}, \\
&\mathbf{I}_n = \begin{bmatrix} 1 & & & \\ & \ddots & & \\ & & \ddots & \\ & & & 1 \end{bmatrix}, \in \mathbb{R}^{n\times n}.
\end{aligned}
$$

6.2.1 Cases of the Actuator Failure

As discussed in the previous section, the snake robot is comprised of several links and joints, each equipped with an actuator. There exists a chance for actuator failure, necessitating the optimization of the remaining motors to achieve maximum efficiency. This study examines four different scenarios of actuator failure, detailed as follows:

- Case-I: No actuator failure.
- Case-II: The second actuator has failed.
- Case-III: The fourth actuator has failed.
- Case-IV: Both the second and fourth actuators have failed.

The numbering of the links is from the tail to the head, with the 1^{st}-link being the tail and the last-link being the head. A control scheme suitable for these actuator failure scenarios in an uncertain underwater environment is proposed and will be elaborated upon in the subsequent section.

6.3 Controller Design

The SMC strategy is similar to the chapter 2. As in this work, the dynamics of the robot is not considered explicitly, the expanded terms of the control law is considered to avoid the singularities of the solver. Utilizing this approach, the STSMC is constructed as follows.

$$\boldsymbol{\sigma}(\bar{\xi}) = \begin{bmatrix} \sigma_1(\bar{\xi}) \\ \sigma_2(\bar{\xi}) \end{bmatrix} = \begin{bmatrix} \dot{\tilde{\theta}}_n + k_n\tilde{\theta}_n \\ \dot{\gamma} + k_v\tilde{v}_{t,r} \end{bmatrix} \tag{6.3}$$

where, $k_n, k_v > 0$ are the design parameter. The control law can be defined by equating $\dot{\boldsymbol{\sigma}}(\bar{\xi}) = \mathbf{0}$ as follows.

$$u_{\phi_0} = -\hat{f}_1 + \ddot{\theta}_{ref} - k_n\dot{\tilde{\theta}}_n + u_{sw_1} \tag{6.4a}$$
$$u_{sw_1} = -k_1|\sigma_1|^{1/2}\text{sgn}(\sigma_1) - k_2\sigma_1|\sigma_2|^{3/2} - k_1 z_1 \tag{6.4b}$$
$$u_\gamma = -\hat{f}_2 + k_v\dot{v}_{t,ref} + u_{sw_2} \tag{6.4c}$$
$$u_{sw_2} = -k_2|\sigma_2|^{1/2}\text{sgn}(\sigma_2) - k_1\sigma_2|\sigma_1|^{3/2} - k_2 z_2 \tag{6.4d}$$
$$\dot{z}_1 = k_z\text{sgn}(\sigma_1) \tag{6.4e}$$
$$\dot{z}_2 = k_z\text{sgn}(\sigma_2) \tag{6.4f}$$

u_{sw_1} and u_{sw_2} represent the reaching laws of the STSMC. $\hat{f}_1$ and $\hat{f}_2$ denote the uncertain components of the robot dynamics, estimated through the RBFNN. The inputs used for training the neural network include the angular position and velocity of the joint, the sliding surface, and the orientation of the snake robot's head link which is different from the previously considered input in chapter 3. These states are measured using the sensors as described in the section 6.2. The block diagram of estimating an uncertain model using a neural network is depicted in Fig. 6.3. In this, the centre line and weight adaption are

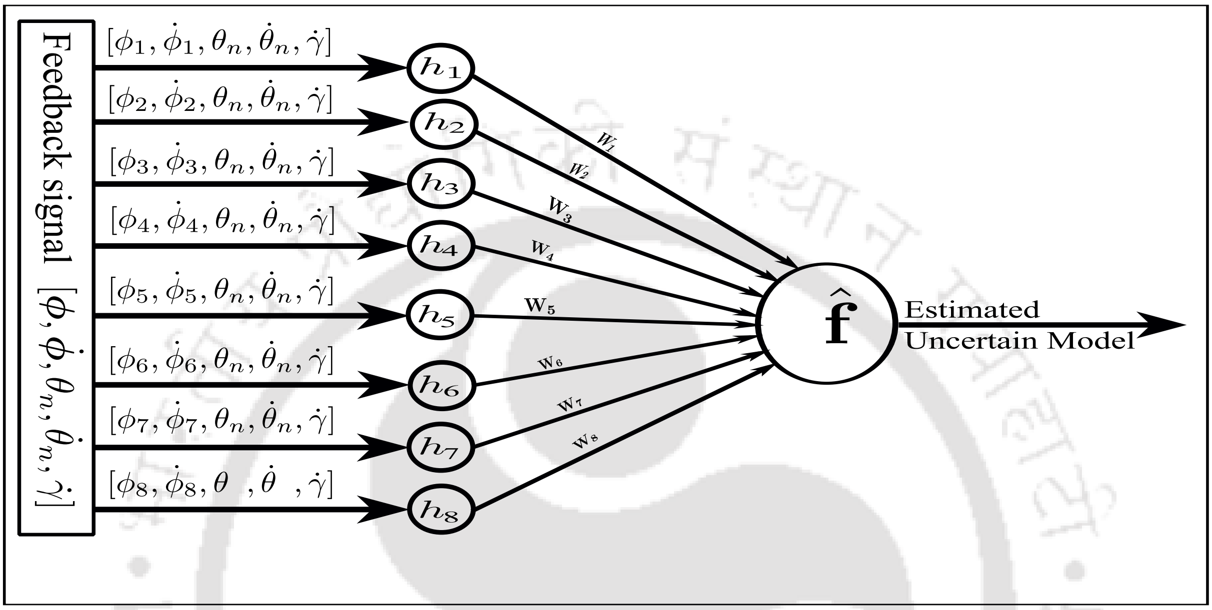

Figure 6.3: Block diagram of neural network

considered. The algorithm for training the neural network is outlined in Algorithm 6.1.

Algorithm 6.1 Neural Network Algorithm

1: **for** j = 1:number of joints **do**

2: $\quad n = j$+number of joints

3: $\quad \zeta_{new} = \left[\zeta([j, n, \theta_n, \dot{\theta}]_n, \dot{\gamma})\right]^T$ ▷ Input of neural network

4: $\quad \underline{\mathbf{C}}(:, j)_{new} = \underline{\mathbf{C}}(:, j)_{old} + \|\boldsymbol{\sigma}\|(\zeta_{new} - \underline{\mathrm{C}}(:, j)_{old})$ ▷ Center-line adaption

5: $\quad \underline{\mathrm{b}} = \frac{max(abs(\zeta_{new} - \underline{\mathrm{C}}(:,j)_{new}))}{\sqrt{5}}$ ▷ Width of the distribution

6: $\quad h(j) = exp\left(\frac{-\|\zeta_{new} - \underline{\mathrm{C}}(:,j)_{new}\|^2}{2\underline{\mathrm{b}}^2}\right)$ ▷ Gaussian activation function

7: **end for**

8: $\dot{\hat{\mathbf{W}}} = P_1\boldsymbol{\sigma}\mathbf{h}^T + P_2\|\boldsymbol{\sigma}\|^{4/3}\boldsymbol{\sigma}\mathbf{h}^T$ ▷ Weight adaption law

9: $\hat{\mathbf{f}} = \hat{\mathbf{W}}^T\mathbf{h}$ ▷ Estimated uncertain dynamics

10: $\underline{\mathbf{C}}_{old} = \underline{\mathbf{C}}_{new}$ ▷ Updated center line

The control algorithm implementation is shown in Fig. 6.4. The benefit of using

STSMC over SMC is the reduction of the chattering effect of the joint torque. Therefore, STSMC is proposed over SMC using a neural network approach. However, the results are also compared with RBF-SMC and the baseline control scheme, as mentioned in the following subsection.

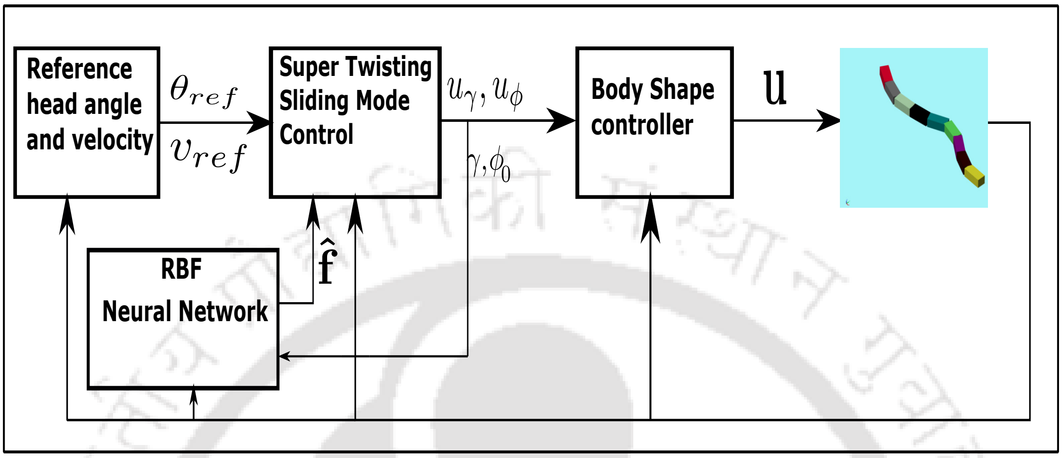

Figure 6.4: Block diagram of control law implementation

While using SMC, the control approach remains the same with the only changes occurring in the reaching law (u_{sw_1}, u_{sw_2}) which can be written as follows.

$$u_{sw_1} = k_3 \mathrm{sgn}(\sigma_1) \tag{6.5a}$$
$$u_{sw_2} = k_3 \mathrm{sgn}(\sigma_2) \tag{6.5b}$$

6.3.1 Lyapunov Stability

Let the Lyapunov candidate be chosen as described below.

$$V_1 = \frac{1}{2}\sigma_1^2 + \frac{1}{2}\sigma_2^2 + \frac{1}{2}z_1^2 + \frac{1}{2}z_2^2$$

$$\dot{V}_1 = \sigma_1(\hat{f}_1 - \ddot{\theta}_{ref} + k_n\dot{\bar{\theta}}_n + u_{\phi_0}) + \sigma_2(u_\lambda + k_v\hat{f}_2 - \dot{v}_{ref}) + z_1\dot{z}_1 + z_2\dot{z}_2$$

substituting Eq. 6.4

$$= \sigma_1(u_{sw_1}) + \sigma_2(u_{sw_2}) + z_1 k_z \mathrm{sgn}(\sigma_1) + z_2 k_z \mathrm{sgn}(\sigma_2)$$

$$= \sigma_1(-k_1|\sigma_1|^{1/2}\mathrm{sgn}(\sigma_1) - k_2\sigma_1|\sigma_2|^{3/2} - k_1 z_1) + \sigma_2(-k_2|\sigma_2|^{1/2}\mathrm{sgn}(\sigma_2) - k_1\sigma_2|\sigma_1|^{3/2} - k_2 z_2) + z_1 k_z \mathrm{sgn}(\sigma_1) + z_2 k_z \mathrm{sgn}(\sigma_2)$$

$$\leq -k_1|\sigma_1|^{3/2} - k_2\sigma_1^2|\sigma_2|^{3/2} - k_1\sigma_1 z_1 - k_2|\sigma_2|^{3/2} - k_1\sigma_2^2|\sigma_1|^{3/2} - k_2\sigma_2 z_2 + z_1 k_z \mathrm{sgn}(\sigma_1) + z_2 k_z \mathrm{sgn}(\sigma_2)$$

$$\leq -k_1|\sigma_1|^{3/2} - k_2\sigma_1^2|\sigma_2|^{3/2} - k_2|\sigma_2|^{3/2} - k_1\sigma_2^2|\sigma_1|^{3/2} - (k_1 - k_z)|\sigma_1 z_1| - (k_2 - k_z)|\sigma_2 z_2|$$
$$\leq 0.$$

From the analysis presented above, it is necessary for k_1 and k_2 to be positive, and they must satisfy the conditions $k_1 \geq k_z$ and $k_2 \geq k_z$ to ensure the system's stability.

Further, the weight adaption is also considered in the RBF-neural network. Therefore, the stability of the weight function is also required.
Let consider the ideal neural network weights $\mathbf{W}^*$ for actual $\mathbf{f}$ (unknown dynamics) such as

$$\mathbf{f} = \mathbf{W}^{*T}\mathbf{h}.$$

Therefore, the approximation weight functions are formulated as $\tilde{\mathbf{W}} = \mathbf{W}^* - \hat{\mathbf{W}}$ where, $\mathbf{W}^* > \hat{\mathbf{W}}$.
Let's consider the Lyapunov candidate of the weight function as follows.

$$\begin{aligned}
V_2 &= \frac{1}{2P_1}\tilde{\mathbf{W}}^T\tilde{\mathbf{W}} \\
\dot{V}_2 &= -\frac{1}{P_1}\tilde{\mathbf{W}}^T\dot{\hat{\mathbf{W}}} \\
&\text{substituting } \hat{\mathbf{W}} \text{ from the algorithm} \\
&= -\frac{1}{P_1}\tilde{\mathbf{W}}^T(P_1\boldsymbol{\sigma}\mathbf{h}^T + P_2\|\boldsymbol{\sigma}\|^{4/3}\boldsymbol{\sigma}\mathbf{h}^T) \\
&\leq -\|\tilde{\mathbf{W}}^T\boldsymbol{\sigma}\|\mathbf{h}^T - \frac{P_2}{P_1}\|\tilde{\mathbf{W}}\|\|\sigma\|^{7/3}\mathbf{h}^T \\
&\leq 0.
\end{aligned}$$

This derivation shows that $P_2 \geq P_1$ can fulfil the stability condition $\dot{V}_1 + \dot{V}_2 \leq 0$. Here, two Lyapunov candidates are considered only for ease of the derivation.

6.3.2 Baseline Controller

The baseline control scheme is given below by following [79]. The results of the proposed control scheme are compared with this control scheme. In this control scheme, u_{ϕ_0} and u_γ had the same error function as the proposed control scheme. Therefore, a comparison between this control scheme and the proposed control scheme is considered.

$$u_{\phi_0} = -\frac{1}{\epsilon}\sigma_1 - k_{\phi_1}\phi_0 - k_{\phi_2}\dot{\phi}_0 \tag{6.6a}$$

$$u_\gamma = -k_\gamma\sigma_2 \tag{6.6b}$$

6.4 Results and Discussion

This section presents a comparison of the control schemes in under actuator failure in uncertain underwater conditions. The control parameters are selected for all cases based on achieving the same tracking error for Case I that is no actuator failure. The control parameters are given in Table-6.1. The parameters are kept the same for all the cases of the actuator failures. To check the robustness of the proposed control scheme, two

Table 6.1: Control parameters

Controller	Parameters
Joint controller (PID)	$k_p = 20$, $k_d = 5$, $k_i = 10$
Sliding surface	$k_n = 10$, $k_v = 12.5$
Existing control scheme	$k_{\phi_1} = 0.1$, $k_{\phi_2} = 0.1$, $k_\gamma = 1$
STSMC	$k_1 = 8$, $k_2 = 7$, $k_z = 2$
SMC	$k_3 = 10$
RBF-NN	$P_1 = 5$, $P_2 = 2$
Reference Trajectory	$k_t = 0.025$, $k_h = 0.02$

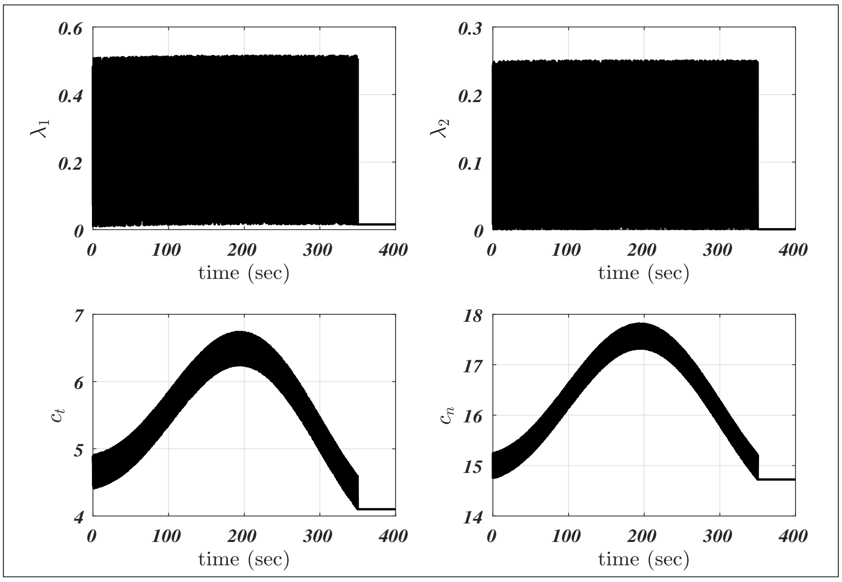

Figure 6.5: Uncertain parameters of drag force and fluidic torque

different path following cases are considered *viz.* straight-line path and sinusoidal path following. The uncertainties in the drag forces and fluidic torque parameters are also considered to compare the proposed control scheme with the existing one. The graphical representation of these uncertainties is shown in Fig. 6.5. Moreover, the quantitative analyses of the results are performed using the chattering indicator and error tracking. All cases of actuator failures for the straight-line path following are discussed in the next subsection.

6.4.1 Straight Line Path Following

This subsection explains the straight-line path following of the snake robot. Here, four cases are considered as mentioned in section 6.2.1. The CG tracking of a snake robot

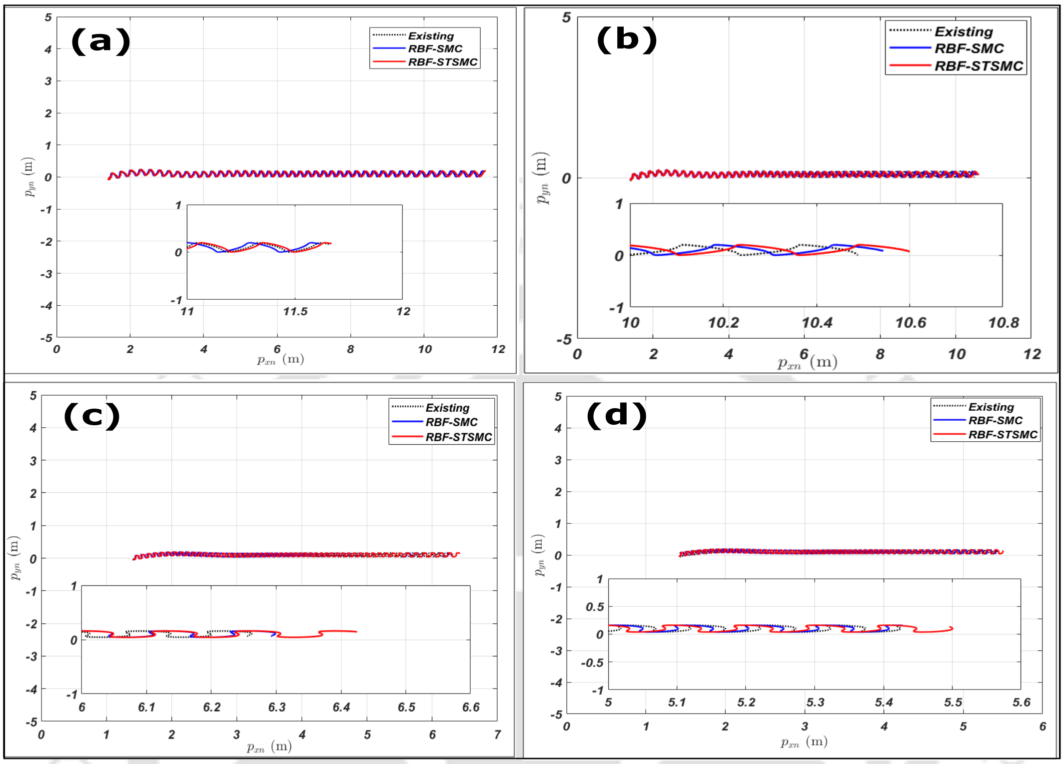

Figure 6.6: Straight path trace of head link-CG (a) Case-I (b) Case-II (c) Case-III (d) Case-IV

head is shown in Fig. 6.6. Unlike, the motion of CG of the snake robot is considered in the previous chapters, in the present chapter, the motion of a CG of the head link is considered. As shown in Fig. 6.1, a sensor is placed on the head link to measure the position and orientation that can be used to track the motion of a robot. Figure 6.6a shows the tracking of the CG of a head link for the no actuator failure case. It is observed that the distance covered by the snake robot is almost the same for all the control schemes, and based on this result, the control parameters are selected and kept the same for the entire work. The failure of the second actuator is illustrated in Fig. 6.6b, where it is evident that RBF-STSMC outperforms other control schemes in terms of the distance covered by the robot. Fig. 6.6c shows the scenario of the fourth actuator failing during the snake robot's motion. This failure impacts the robot's motion more significantly than in previous cases, with the distance covered by the snake robot reducing to approximately 40% of that in Case-I. The failures of the second and fourth actuators, as depicted in Fig. 6.6d, highlight the superior performance of RBF-STSMC in these scenarios. The distance covered by the snake robot within the same timeframe can be assessed through the tangential velocity

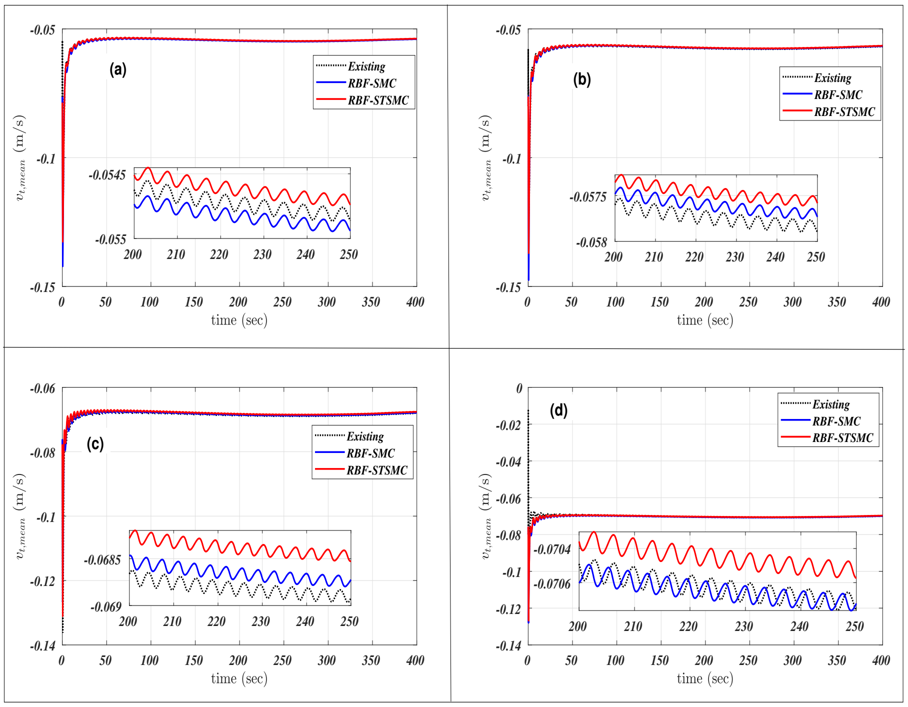

Figure 6.7: Tangential velocity tracking: straight-line path following (a) Case-I (b) Case-II (c) Case-III (d) Case-IV

tracking, as illustrated in Fig. 6.7. The findings reveal that the performance of RBF-STSMC is notably superior, especially when the fourth actuator fails, compared to other control schemes. Additionally, the issue of chattering, prevalent in sliding mode control approaches, is addressed by employing a chattering indicator, the results of which are presented in Table-6.2. RBF-STSMC demonstrates a significant reduction in chattering compared to RBF-SMC, contributing to extended actuator lifespan.

The tracking error, quantified in Table 6.3, shows that RBF-STSMC enhances the tangential velocity of the snake robot more effectively than other controllers. Although the head angle tracking error is better with the existing control scheme, the head tracking error remains minimal, ensuring the robot stays on the desired path. In this study, speed is prioritized since the head tracking error is within acceptable limits, and RBF-STSMC has proven to be effective in covering more distance in a given time period. The robustness of the control scheme is further evaluated through a sinusoidal path following scheme, which is discussed in the subsequent subsection.

Table 6.2: Chattering Indicator- straight-line path following

	Case-I		Case-II		Case-III		Case-IV	
	STSMC	SMC	STSMC	SMC	STSMC	SMC	STSMC	SMC
u_1	8.7727	120.7290	6.4354	73.9691	11.0941	113.7198	7.3992	75.1344
u_2	15.8783	207.0876	0.0000	0.0000	15.0841	169.0757	0.0000	0.0000
u_3	18.7466	243.3820	16.3267	152.0243	15.1166	148.1866	13.0732	86.9234
u_4	18.7810	245.3965	19.7533	210.9780	0.0000	0.0000	0.0000	0.0000
u_5	19.0056	243.7453	18.8018	242.7367	20.3836	157.4009	19.9174	165.6240
u_6	18.3012	238.2107	17.9864	246.6568	17.9346	207.5218	18.4757	217.2762
u_7	14.0922	202.9531	13.7561	208.5257	11.3354	193.1877	12.1495	198.4951
u_8	7.6653	119.9965	7.4247	120.3391	6.8748	117.7637	7.2722	119.0088
$\|\mathbf{u}\|$	30.3160	364.4885	30.1971	316.4186	30.2802	157.9404	26.5915	167.5715

Table 6.3: Error in RMS- straight-line path following

	Case-I			Case-II		
	STSMC	SMC	Existing	STSMC	SMC	Existing
$\tilde{v}_t$	0.0549	0.0550	0.0554	0.0582	0.0584	0.0593
$\tilde{v}_{t,mean}$	0.0542	0.0544	0.0543	0.0570	0.0572	0.0573
$\tilde{\theta}_n$	0.0022	0.0027	0.0001	0.0022	0.0028	0.0001
	Case-III			Case-IV		
	STSMC	SMC	Existing	STSMC	SMC	Existing
$\tilde{v}_t$	0.0698	0.0702	0.0708	0.0714	0.0716	0.0720
$\tilde{v}_{t,mean}$	0.0680	0.0683	0.0685	0.0701	0.0703	0.0703
$\tilde{\theta}_n$	0.0015	0.0022	0.0001	0.0016	0.0022	0.0001

6.4.2 Sinusoidal Path Following

The robustness of the control scheme is tested using a sinusoidal path. Figure 6.8 illustrates the case of sinusoidal path following. Similar to the straight-line path scenario, it is observed that the robot's ability to cover distance is adversely affected when the fourth actuator fails. However, the performance of RBF-STSMC remains consistently superior across all cases of actuator failure when compared to other control schemes except for case-III, as can be verified in Fig. 6.9. In case-III, a variation between 4.9 m to 5 m is observed, attributed to the computational complexity in the NN-based control scheme, as illustrated in Fig. 6.8c. However, it is noted that the distance covered by the snake robot using RBF-STSMC is increased compared to other control schemes when actuators fail.

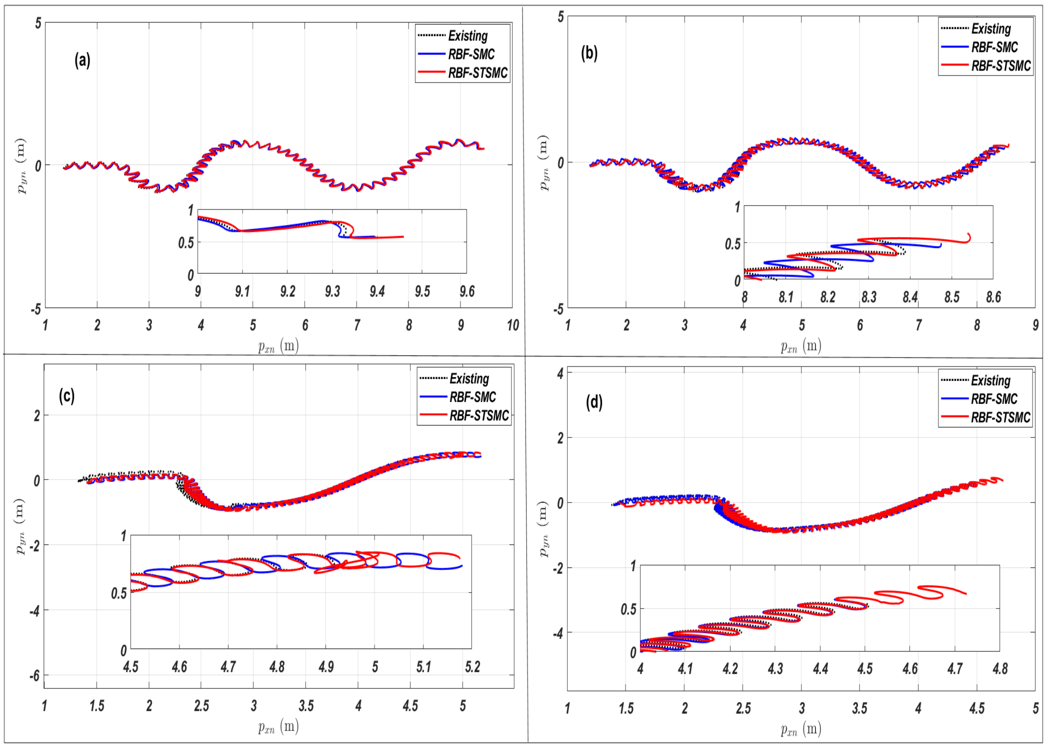

Figure 6.8: Sinusoidal path trace of head link-CG (a) Case-I (b) Case-II (c) Case-III (d) Case-IV

Table 6.4: Chattering Indicator - sinusoidal path following

	Case-I		Case-II		Case-III		Case-IV	
	STSMC	SMC	STSMC	SMC	STSMC	SMC	STSMC	SMC
u_1	32.7142	123.9954	18.4889	74.4725	48.8340	113.6056	16.0623	75.0791
u_2	52.5479	210.7967	0.0000	0.0000	44.2422	165.7429	0.0000	0.0000
u_3	57.0535	247.4493	44.8046	162.7581	42.3460	146.0010	21.8916	86.0903
u_4	64.3517	249.9318	54.2486	221.9298	0.0000	0.0000	0.0000	0.0000
u_5	62.4572	248.6238	57.0336	247.5827	47.3803	168.7824	35.1315	172.6513
u_6	61.4397	243.4432	52.8768	246.6829	50.6910	217.3897	37.6391	225.7405
u_7	50.7913	207.5119	46.2776	209.0709	48.3765	196.9695	34.3811	203.2581
u_8	38.8118	124.3860	32.4657	123.7126	39.7267	117.5394	27.5539	118.8190
$\|\|\mathbf{u}\|\|$	105.8348	387.6268	85.2277	345.0936	83.7776	187.7827	54.0669	194.5175

Table 6.4 displays the chattering effect in torque during sinusoidal path following. Here, chattering is observed to be higher compared to that in a straight-line path for the same control scheme, which is attributed to the complexity of the path. Nonetheless, the

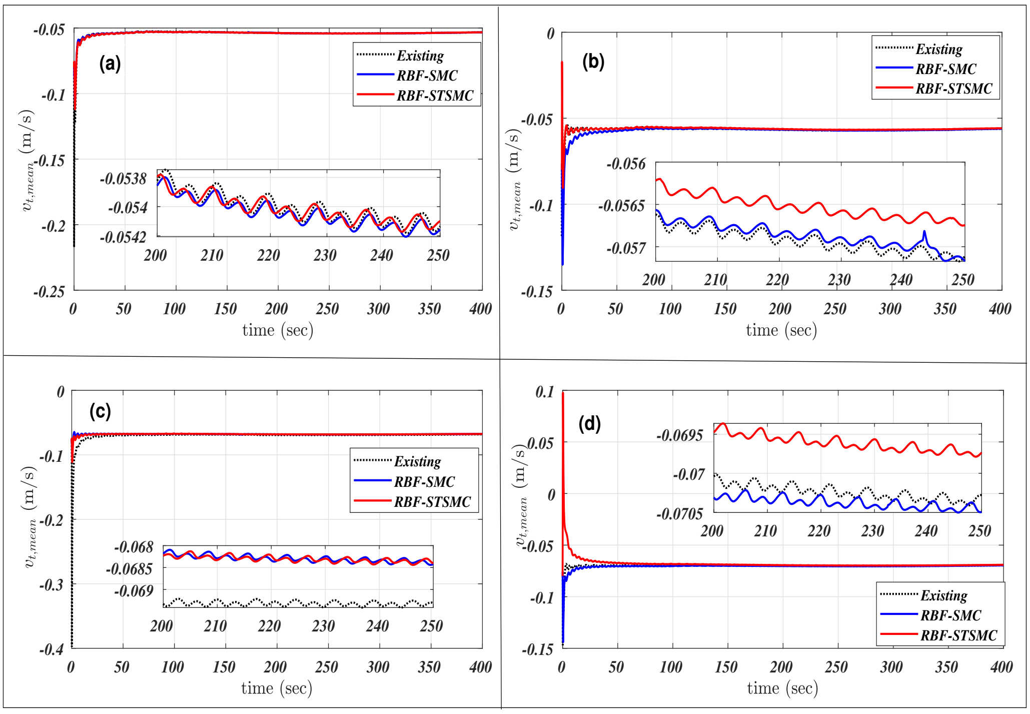

Figure 6.9: Tangential velocity tracking: sinusoidal path following (a) Case-I (b) Case-II (c) Case-III (d) Case-IV

performance of RBF-STSMC surpasses that of RBF-SMC in both scenarios. Table 6.5 presents the tracking error in RMS, where the performance of RBF-STSMC is found to be in good agreement.

Table 6.5: Error in RMS -sinusoidal path following

	Case-I			Case-II		
	STSMC	SMC	Existing	STSMC	SMC	Existing
v_t	0.0544	0.0550	0.0549	0.0576	0.0578	0.0604
$v_{t,mean}$	0.0536	0.0537	0.0536	0.0561	0.0566	0.0566
$\tilde{\theta}_n$	0.0093	0.0754	0.0007	0.0065	0.0071	0.0051
	Case-III			Case-IV		
	STSMC	SMC	Existing	STSMC	SMC	Existing
v_t	0.0707	0.0715	0.0716	0.0710	0.0712	0.0715
$v_{t,mean}$	0.0680	0.0680	0.0690	0.0692	0.0702	0.0700
$\tilde{\theta}_n$	0.0388	0.0861	0.0039	0.0049	0.0038	0.0004

The snake robot is constructed using multibody dynamics software. This developed model allows for the direct deployment of new control techniques and comparison of results. The most suitable control scheme can then be deployed to the physical snake robot, significantly reducing the chance of experiment failure through this simulation approach.

6.5 Summary

In this work, the snake robot is modelled within the Simscape multibody environment. This accounts for uncertainties in drag forces and fluidic torque during underwater motion. The VHCs-based RBF-STSMC approach is employed to control the snake robot's motion, considering various scenarios of actuator failure. Particularly, the failure of the fourth actuator significantly affects the robot's motion, leading to a reduction in speed, which is an expected consequence. This study aims to identify which control scheme offers the best performance under such conditions. The performance of RBF-STSMC is found to surpass that of other control schemes. A major advantage of the neural network approach is its ability to eliminate the need for estimating lumped uncertainties in the model, thereby avoiding the overestimation of parameters when designing model-based controls.

7

Conclusion and Scope for Future Work

Contents

7.1 General Conclusion

The dynamics and control of a snake robot with nine links and eight joints actuated by servo motors in uncertain underwater environmental conditions is attempted in this thesis. Five different problems are addressed in this work. The equations of motion of the snake robot have been derived using Newton's 2^{nd} law for the planar motion. Then, it is reduced by using virtual holonomic constraints. This approach has a unique benefit where the dynamics of the multi-link snake robot is reduced to that of a single-link (head-link) system on which control law is deployed. The head-link angle and tangential velocity of a snake robot are controlled using the control scheme. In all the problems, the control parameters are selected based on the Lyapunov stability analysis.

In the first problem, the sliding mode control (SMC) approach is used with the existing dynamics of a snake robot. The chattering effect which is a major concern using the SMC scheme, is reduced by using the super twisting sliding mode control (STSMC) scheme. The aim of this work is to achieve the direction control of a snake robot in an uncertain underwater environment with the desired velocity. The performance of the STSMC scheme is compared with the adaptive sliding mode control (ASMC) scheme and SMC scheme. A significant reduction in chattering is observed using the STSMC. This robust control scheme is used to deal with uncertainties in drag forces and fluidic torque during the motion of a snake robot. These uncertainties are considered as lumped parameters in controller design.

In the second problem, the same mathematical model is considered for the planar motion of a snake robot where the radial basis function neural network (RBF-NN) approach is used for the estimation of the uncertain environment. Here, the RBF-NN with ASMC scheme is used for the motion control of a snake robot. The RBF-NN is designed in such a way that the weights, width and centre-line of the neural networks are adaptive. To estimate the uncertain dynamics, the designed sliding surface, link angle velocity and orientation, CG position and velocity of the snake robot are considered as the input for neural networks. The results of the proposed RBF-NN with the ASMC scheme are found to be better than the ASMC scheme in terms of chattering reduction and improvement in error tracking while following a desired path.

In the third problem, the planar motion (X-Y plane) of the snake robot is extended to 3D motion in the underwater environment by including the vertical motion (Z-direction). The vertical motion is generated using the buoyancy variation technique. Here, the equation of motion is derived by including the added mass effect and vertical motion of the snake robot along with the external disturbances. The horizontal motion is controlled using the existing STSMC scheme and the vertical motion is achieved using the pump operations, for which the event-based control scheme and PD control scheme are used. Moreover, the proposed dynamics are verified using the simscape multibody environment where the equation of motion is not required explicitly.

It is observed that the speed of a snake robot is very less in underwater conditions and that can be improved by external thrust force which is addressed in the fourth problem. Here, the equation of motion of the snake robot is derived considering the tail-link equipped with a thruster. The actor-critic neural network with super twisting sliding mode control scheme is used to deal with the uncertainties and disturbance during the motion of the snake robot. The proposed control scheme is compared with the existing STSMC scheme and the reduction in chattering effect and control effort is found. The speed of the snake robot is increased by using the thruster.

As the snake robot has many actuators, there may be a chance of failure. This is addressed in the fifth problem. The control scheme is designed such that the robot motion can be sustained in the presence of the actuator failure. The snake robot is constructed in the Simscape multibody environment. The radial basis function neural network super twisting sliding mode control scheme is used to deal with various uncertainties and actuator failure. The proposed control scheme is compared with the existing control scheme, and an improvement in performance is observed.

Following specific conclusions are made from the above mentioned studies.

7.2 Specific Conclusion

- The STSMC scheme is proposed to track the head link tangential velocity and orientation of a snake robot. The given reference head angle direction and tangential velocity are followed by the snake robot in an uncertain underwater environment. The significant reduction in the chattering of the input angular (joint) velocity of 94.15% and 88.23% from the SMC and ASMC schemes, respectively, is found for the STSMC scheme. A significant reduction in the chattering of the input torque is also found. The overall reduction in terms of the norm in torque of 88.50% and 77% from SMC and ASMC schemes, respectively, is found for the proposed control scheme.

- The adaptive RBF neural network sliding mode control scheme is proposed to follow the desired path for the snake robot with the desired tangential velocity of the head-link of a snake robot. The reduction in chattering of the input angular (joint) velocity of 42.62% from the ASMC scheme is found for the proposed scheme. A significant reduction in the chattering of the input torque is also found. The overall reduction in terms of the norm in torque of 37.55% from the ASMC scheme is found for the proposed control scheme. Improvements in the tracking error of tangential velocity and head link orientation are 1.93% and 12.9%, respectively.

- For the 3D motion of a snake robot, two types of path following cases *i.e.* straight line path following and sinusoidal path following are considered for the horizontal motion control. The effectiveness of STSMC is tested by introducing external disturbances along with environmental uncertainties such as random disturbances, pulse disturbances and high-frequency disturbances. The chattering effect has been compared for the various disturbances. When there is no disturbance, very less chattering is observed which is obvious. During the straight line path following case, the chattering of torque in terms of norms due to pulse disturbances is 29.75% and 28.04% less than random and high-frequency disturbances, and similarly for the sinusoidal path following case due to the pulse disturbances, is 35.02% and 28.07% less than random and high-frequency disturbances. The tracking error is almost the same for every case of disturbances and path following cases.

- When a thruster is considered on the tail-link of the snake robot using an actor-critic neural network, the reduction in the chattering of torque (in norms) compared to the existing control scheme is found to be 93.36%, 85.28%, 88.58% and 86.51% for the cases of no disturbances, random disturbance, pulse disturbance and high-frequency disturbances, respectively. The improvement in error tracking of tangential velocity using the proposed control scheme compared to the existing control scheme are 5.81%, 4.64%, 3.40% and 4.64% for no disturbances, random disturbance, pulse disturbance and high-frequency disturbances, respectively. Similarly, the head angle tracking is improved by 93.6%, 92.35%, 92.15% and 92.82% for the above mentioned disturbances cases. Moreover, the reduction in control effort in terms of norms using the proposed control scheme compared to the existing control scheme is 55.69%, 51.50%, 50.73% and 51.73% for the above mentioned disturbances.

- The RBF-STSMC scheme is used for the motion control of a snake robot during the actuator failure. The four cases (No actuator failed, second, fourth and combined second and fourth actuators failed) of actuators failures are considered. Two-path following cases *i.e. straight line path following and sinusoidal path following* are considered in this work to check the robustness of the proposed control scheme during the actuator's failure along with environmental uncertainties. The improvement in error tracking of tangential velocity using RBF-STSMC scheme compared to the baseline control approach for the straight line path following are 1.85%, 1.51%, 0.83% and for sinusoidal path following are 4.63%, 1.26%, 0.70%. Moreover, the reduction in the chattering effect of the torques for each case using the proposed control scheme compared to the RBF-SMC scheme is about 85%.

As observed from the general and specific conclusions, the present work is a novel attempt for dynamics and control studies of the snake robot in both planar and 3D motion in uncertain underwater environments. Both classical and modern control theories have been used in this work for various path-following cases. The various cases studied in this work can be used by researchers in this field to develop a wide range of snake robots for many different applications. The following section briefly describes the scope for future work.

7.3 Scope for Future Work

- **Gait Pattern:** In the present work, only the lateral undulation motion gait pattern is considered. One may consider other different gait patterns such as concertina motion, rectilinear motion, crotaline motion for the various underwater environmental conditions i.e., narrow areas, dense water, etc. This will change the motion dynamics of the snake robot.

- **Limited known states:** In this work, all the states (joint angles and CG position of the snake robot and their derivatives) are known to estimate the dynamical systems that can be improved with limited known states for designing the control scheme. One may extend this work with limited known states of the system and design the control scheme.

- **Snake robot with manipulator:** In the present work, only a simple motion study of the snake robot has been carried out. This work can be extended for many applications such as ship haul cleaning, repairing and maintenance of underwater pipes, cable routing, etc. where a manipulator can be included on the head link. One may derive the detailed dynamical equation of the snake robot with manipulator operation and its control approach. This will help to develop cost effective mobile manipulators in the underwater environment for performing various tasks.

- **Hardware implementation:** In this work, the theoretical analysis and numerical modelling of a 9-link snake robot is carried out. One may extend this work for n-numbers of links. A physical model (as shown in Fig. 7.1) is developed using a servo motor on each joint, and a PID controller is used for its motion on land and underwater environment using Arduino Mega (ATmega - 2560) micro-controller board. Due to the problem of procuring costly sensors and microcontrollers, the developed control schemes have not been implemented. One may use the developed control schemes on the physical snake robot in future.

Figure 7.1: Experimental set-up of snake robot, Link1, Link2

List of Publications (From Thesis)

International Journals Publication

1. **B. M. Patel** and S. K. Dwivedy (2023). Virtual holonomic constraints based super twisting sliding mode control for motion control of planar snake robot in the uncertain underwater environment. Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering, 237(8), 1480-1491.

2. **B. M. Patel** and S. K. Dwivedy (2023). Manoeuvring of underwater snake robot with tail thrust using the actor-critic neural network super-twisting sliding mode control in the uncertain environment and disturbances. Neural Computing and Applications, 1-15.

3. **B. M. Patel** and S. K. Dwivedy (2024). 3D dynamics and control of a snake robot in uncertain underwater environment. Robotica, 1-28.

4. **B. M. Patel**, J. Narayan & S. K. Dwivedy (2024). Manoeuvring planar snake robot in uncertain underwater condition using adaptive neural network sliding mode control. International Journal of Dynamics and Control, 1-19.

5. **B. M. Patel**, R.G. Valenti & S. K. Dwivedy (2024). Motion control of an aquatic snake robot with actuators failure and environmental uncertainties using neural network super twisting sliding mode control. Multibody System Dynamics, (Under Review).

Conference Proceedings

1. **B.M. Patel**, J. Narayan and S.K. Dwivedy (2022, March). Symbiotic organism search-based locomotion of underwater snake robot in various environments. In 2022 2nd International Conference on Image Processing and Robotics (ICIPRob) (pp. 1-6). IEEE.

2. **B. M. Patel** and S. K. Dwivedy (2023, July). Manoeuvring control of the snake robot using multi-variable super twisting sliding mode control in the uncertain underwater environment. In Proceedings of the 2023 6th International Conference on Advances in Robotics (pp. 1-6).

Blog Published by MathWorks

1. Snake-like Robot Modeling and Navigation.

List of Publications (Collaboration during PhD)

International Journals Publication

1. J. Narayan, M. Abbas, **B.M. Patel** and S. K. Dwivedy (2023). Adaptive RBF neural network-computed torque control for a pediatric gait exoskeleton system: an experimental study. Intelligent Service Robotics, 16(5), 549-564.

Conference Proceedings

1. J. Narayan, M. Abbas, **B.M. Patel** and S. K. Dwivedy. (2022, March). A singularity-free terminal sliding mode control of an uncertain paediatric exoskeleton system. In 2022 5th International Conference on Advanced Systems and Emergent Technologies (IC_ASET) (pp. 198-203). IEEE.

2. J. Narayan, **B. M. Patel**, M. Abbas, G. Shivhare and S. K. Dwivedy. (2022, July). Cooperative control of a pediatric exoskeleton system for active-assist gait rehabilitation. In 2022 IEEE International conference on electronics, computing and communication technologies (CONECCT) (pp. 1-6). IEEE (**Best paper award**).

3. J. Narayan, M. Abbas, **B. M. Patel**, Jhunjhunwala, S. and S. K. Dwivedy. (2023, July). Improved fast terminal sliding mode control for a pediatric gait exoskeleton system: theory and experimental results. In Proceedings of the 2023 6th International Conference on Advances in Robotics (pp. 1-6).

Brief Bio of the Author

Bhavik M. Patel is from Vadodara, Gujarat in India. He graduated with Bachelor of Engineering (B.E.) in Mechanical Engineering from Dr. S. & S. S. Ghandhy Government Engineering College, Surat affiliated with Gujarat Technological University in 2017. In 2020, He has completed his M.Tech degree in Machine Design from the Department of Mechanical Engineering, Indian Institute of Engineering Science and Technology, Shibpur in the research area of vibration and control under the supervision of Prof. Shyamal Chatterjee and during this study, he was motivated for the PhD. In 2020, He joined Department of Mechanical Engineering, IIT Guwahati as PhD student under the supervision of Prof. S. K. Dwivedy. The research work is carried out during the period July 2020 to July 2024 in the field of dynamics and control of robotic system. The author's research focuses on study of dynamics and control approaches for the motion of a snake robot in underwater environment. His work was recognized by the MathWorks and published a blog on it. During PhD, he was directly involved in the various activities such as Research and Industrial Conclave, Research Scholar Forum - ME and DiSHA club as active member.

References

[1] R. K. Katzschmann, J. DelPreto, R. MacCurdy, and D. Rus, "Exploration of underwater life with an acoustically controlled soft robotic fish," *Science Robotics*, vol. 3, no. 16, 2018.

[2] M. Calisti, F. Corucci, A. Arienti, and C. Laschi, "Dynamics of underwater legged locomotion: Modeling and experiments on an octopus-inspired robot," *Bioinspiration & biomimetics*, vol. 10, no. 4, p. 046 012, 2015.

[3] B. Allotta *et al.*, "The arrows project: Adapting and developing robotics technologies for underwater archaeology," *IFAC-PapersOnLine*, vol. 48, no. 2, pp. 194–199, 2015.

[4] P. Liljebäck, Ø. Stavdahl, K. Y. Pettersen, and J. T. Gravdahl, "Mamba-a waterproof snake robot with tactile sensing," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 2014, pp. 294–301.

[5] R. L. Baines, J. W. Booth, F. E. Fish, and R. Kramer-Bottiglio, "Toward a bio-inspired variable-stiffness morphing limb for amphibious robot locomotion," in *2019 2nd IEEE International Conference on Soft Robotics (RoboSoft)*, IEEE, 2019, pp. 704–710.

[6] H. Godaba, J. Li, Y. Wang, and J. Zhu, "A soft jellyfish robot driven by a dielectric elastomer actuator," *IEEE Robotics and Automation Letters*, vol. 1, no. 2, pp. 624–631, 2016.

[7] J. Liu, Y. Tong, and J. Liu, "Review of snake robots in constrained environments," *Robotics and Autonomous Systems*, vol. 141, p. 103 785, 2021, ISSN: 0921-8890.

[8] A. Sahoo, S. K. Dwivedy, and P. Robi, "Advancements in the field of autonomous underwater vehicle," *Ocean Engineering*, vol. 181, pp. 145–160, 2019, ISSN: 0029-8018.

[9] J. Evans, P. Redmond, C. Plakas, K. Hamilton, and D. Lane, "Autonomous docking for intervention-auvs using sonar and video-based real-time 3d pose estimation," in *Oceans 2003. Celebrating the Past... Teaming Toward the Future (IEEE Cat. No. 03CH37492)*, IEEE, vol. 4, 2003, pp. 2201–2210.

[10] G. Marani, S. K. Choi, and J. Yuh, "Underwater autonomous manipulation for intervention missions auvs," *Ocean Engineering*, vol. 36, no. 1, pp. 15–23, 2009.

[11] P. Sanz, R. Marın, J. Sales, G. Oliver, and P. Ridao, "Recent advances in underwater robotics for intervention missions. soller harbor experiments," *Low Cost Books, Spain*, 2012.

[12] D. Ribas, N. Palomeras, P. Ridao, M. Carreras, and A. Mallios, "Girona 500 auv: From survey to intervention," *IEEE/ASME Transactions on mechatronics*, vol. 17, no. 1, pp. 46–53, 2011.

[13] P. J. Sanz *et al.*, "Trident: A framework for autonomous underwater intervention missions with dexterous manipulation capabilities," *IFAC Proceedings Volumes*, vol. 43, no. 16, pp. 187–192, 2010.

[14] L. Brignone, E. Raugel, J. Opderbecke, V. Rigaud, R. Piasco, and S. Ragot, "First sea trials of hrov the new hybrid vehicle developed by ifremer," in *Oceans 2015-genova*, IEEE, 2015, pp. 1–7.

[15] J. Sverdrup-Thygeson, E. Kelasidi, K. Y. Pettersen, and J. T. Gravdahl, "The underwater swimming manipulator—a bioinspired solution for subsea operations," *IEEE Journal of Oceanic Engineering*, vol. 43, no. 2, pp. 402–417, 2017.

[16] P. Liljebäck, K. Y. Pettersen, Ø. Stavdahl, and J. T. Gravdahl, *Snake robots: modelling, mechatronics, and control*. Springer Science & Business Media, 2012.

[17] R. Bauchot, *Snakes: a natural history*. Sterling Publishing Company, Inc., 2006.

[18] J. Gray, "The mechanism of locomotion in snakes," *Journal of experimental biology*, vol. 23, no. 2, pp. 101–120, 1946.

[19] B. C. Jayne, "Muscular mechanisms of snake locomotion: An electromyographic study of lateral undulation of the florida banded water snake (nerodia fasciata) and the yellow rat snake (elaphe obsoleta)," *Journal of Morphology*, vol. 197, no. 2, pp. 159–181, 1988.

[20] B. R. Moon and C. Gans, "Kinematics, muscular activity and propulsion in gopher snakes," *Journal of Experimental Biology*, vol. 201, no. 19, pp. 2669–2684, 1998.

[21] A. A. Transeth, K. Y. Pettersen, and P. Liljebäck, "A survey on snake robot modeling and locomotion," *Robotica*, vol. 27, no. 7, pp. 999–1015, 2009.

[22] R. Ariizumi and F. Matsuno, "Dynamic analysis of three snake robot gaits," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1075–1087, 2017.

[23] M. Javaheri Koopaee, C. Pretty, K. Classens, and X. Chen, "Dynamical modelling and control of snake-like motion in vertical plane for locomotion in unstructured environments," in *ASME 2019 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, American Society of Mechanical Engineers Digital Collection, 2019.

[24] P. Liljebäck, K. Y. Pettersen, Ø. Stavdahl, and J. T. Gravdahl, "A review on modelling, implementation, and control of snake robots," *Robotics and Autonomous systems*, vol. 60, no. 1, pp. 29–40, 2012.

[25] J. Liu, Y. Tong, and J. Liu, "Review of snake robots in constrained environments," *Robotics and Autonomous Systems*, vol. 141, p. 103 785, 2021.

[26] S. Hirose, *Biologically inspired robots: Serpentile locomotors and manipulators*, 1993.

[27] M. Mori and S. Hirose, "Development of active cord mechanism acm-r3 with agile 3d mobility," in *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No. 01CH37180)*, vol. 3, 2001, pp. 1552–1557.

[28] S. Yu, S. Ma, B. Li, and Y. Wang, "An amphibious snake-like robot with terrestrial and aquatic gaits," in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 2960–2961.

[29] A. Crespi, A. Badertscher, A. Guignard, and A. J. Ijspeert, "Amphibot i: An amphibious snake-like robot," *Robotics and Autonomous Systems*, vol. 50, no. 4, pp. 163–175, 2005.

[30] A. Crespi and A. J. Ijspeert, "Amphibot ii: An amphibious snake robot that crawls and swims using a central pattern generator," in *Proceedings of the 9th international conference on climbing and walking robots (CLAWAR 2006)*, 2006, pp. 19–27.

[31] B. Klaassen and K. Paap, "Gmd-snake2: A snake-like robot driven by wheels and a method for motion control," in *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C)*, vol. 4, 1999, 3014–3019 vol.4.

[32] S. Hirose, A. Morishima, S. Tukagosi, T. Tsumaki, and H. Monobe, "Design of practical snake vehicle: Articulated body mobile robot kr-ii," in *Fifth International Conference on Advanced Robotics 'Robots in Unstructured Environments*, vol. 1, 1991, pp. 833–838.

[33] S. Takaoka, H. Yamada, and S. Hirose, "Snake-like active wheel robot acm-r4.1 with joint torque sensor and limiter," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011, pp. 1081–1086.

[34] J. Borenstein, M. Hansen, and A. Borrell, "The omnitread ot-4 serpentine robot—design and performance," *Journal of Field Robotics*, vol. 24, no. 7, pp. 601–621, 2007.

[35] W. Wang, G. Zong, H. Zhang, and Z. Deng, *A reconfigurable mobile robots system based on parallel mechanism*. INTECH Open Access Publisher, 2008.

[36] C. Wright *et al.*, "Design of a modular snake robot," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 2007, pp. 2609–2614.

[37] M. Yim, Y. Zhang, K. Roufas, D. Duff, and C. Eldershaw, "Connecting and disconnecting for chain self-reconfiguration with polybot," *IEEE/ASME Transactions on Mechatronics*, vol. 7, no. 4, pp. 442–451, 2002.

[38] S. Murata, E. Yoshida, A. Kamimura, H. Kurokawa, K. Tomita, and S. Kokaji, "M-tran: Self-reconfigurable modular robotic system," *IEEE/ASME Transactions on Mechatronics*, vol. 7, no. 4, pp. 431–441, 2002.

[39] A. Castano, W.-M. Shen, and P. Will, "Conro: Towards deployable robots with inter-robots metamorphic capabilities," *Autonomous Robots*, vol. 8, no. 3, pp. 309–324, 2000.

[40] D. S. Rollinson, "Control and design of snake robots," 2014.

[41] K. Y. Pettersen, "Snake robots," *Annual Reviews in Control*, vol. 44, pp. 19–44, 2017.

[42] C. D. Onal and D. Rus, "Autonomous undulatory serpentine locomotion utilizing body dynamics of a fluidic soft robot," *Bioinspiration & biomimetics*, vol. 8, no. 2, p. 026 003, 2013.

[43] A. M. Kohl, E. Kelasidi, K. Y. Pettersen, and J. T. Gravdahl, "A control-oriented model of underwater snake robots exposed to currents," in *2015 IEEE Conference on Control Applications (CCA)*, 2015, pp. 1585–1592.

[44] A. A. Transeth, R. I. Leine, C. Glocker, and K. Y. Pettersen, "3-d snake robot motion: Nonsmooth modeling, simulations, and experiments," *IEEE Transactions on Robotics*, vol. 24, no. 2, pp. 361–376, 2008.

[45] Z. Cao, D. Zhang, and M. Zhou, "Modeling and control of hybrid 3-d gaits of snake-like robots," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 10, pp. 4603–4612, 2021.

[46] Z. Cao, D. Zhang, and M. Zhou, "Direction control and adaptive path following of 3-d snake-like robot motion," *IEEE Transactions on Cybernetics*, vol. 52, no. 10, pp. 10 980–10 987, 2022.

[47] D. Li *et al.*, "Anti-disturbance path-following control for snake robots with spiral motion," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 12, pp. 11 929–11 940, 2023.

[48] E. Kelasidi, K. Y. Pettersen, J. T. Gravdahl, S. Stromsoyen, and A. Sorensen, "Modeling and propulsion methods of underwater snake robots," in *IEEE Conference on Control Technology and Applications (CCTA)*, 2017, pp. 819–826.

[49] E. Kelasidi, K. Y. Pettersen, P. Liljebäck, and J. T. Gravdahl, "Locomotion efficiency of underwater snake robots with thrusters," in *IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, 2016, pp. 174–181.

[50] T. Ma, L. Wu, Z. Lin, C. Ren, and S. Ma, "Energy optimization of an underwater swimming manipulator with rotary thrusters and rolling joints," in *41st Chinese Control Conference (CCC)*, 2022, pp. 3717–3722.

[51] X. Jiang, F. Yang, and S. Shi, "Design and full-link trajectory tracking control of underwater snake robot with vector thrusters under strong time-varying disturbances," *Ocean Engineering*, vol. 266, p. 113 012, 2022.

[52] A. S. Tijjani, A. Chemori, and V. Creuze, "A survey on tracking control of unmanned underwater vehicles: Experiments-based approach," *Annual Reviews in Control*, vol. 54, pp. 125–147, 2022.

[53] H. Date and Y. Takita, "Adaptive locomotion of a snake like robot based on curvature derivatives," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 2007, pp. 3554–3559.

[54] D. Li, Z. Pan, H. Deng, and L. Hu, "Adaptive path following controller of a multijoint snake robot based on the improved serpenoid curve," *IEEE Transactions on Industrial Electronics*, vol. 69, no. 4, pp. 3831–3842, 2021.

[55] J. Mukherjee, S. Mukherjee, and I. N. Kar, “Sliding mode control of planar snake robot with uncertainty using virtual holonomic constraints,” *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 1077–1084, 2017.

[56] Q. Yongqiang, Y. Hailan, R. Dan, K. Yi, L. Dongchen, L. Xiaoting, *et al.*, “Path-integral-based reinforcement learning algorithm for goal-directed locomotion of snake-shaped robot,” *Discrete Dynamics in Nature and Society*, vol. 2021, 2021.

[57] M. Sato, M. Fukaya, and T. Iwasaki, “Serpentine locomotion with robotic snakes,” *IEEE Control Systems Magazine*, vol. 22, no. 1, pp. 64–81, 2002.

[58] X. Xiao and R. Murphy, “A review on snake robot testbeds in granular and restricted maneuverability spaces,” *Robotics and Autonomous Systems*, vol. 110, pp. 160–172, 2018, ISSN: 0921-8890.

[59] J. Lim, W. Yang, Y. Shen, and J. Yi, “Analysis and validation of serpentine locomotion dynamics of a wheeled snake robot moving on varied sloped environments,” in *2020 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, 2020, pp. 1069–1074.

[60] P. Liljeback, I. U. Haugstuen, and K. Y. Pettersen, “Path following control of planar snake robots using a cascaded approach,” *IEEE Transactions on Control Systems Technology*, vol. 20, no. 1, pp. 111–126, 2012.

[61] C. D. Onal and D. Rus, “Autonomous undulatory serpentine locomotion utilizing body dynamics of a fluidic soft robot,” *Bioinspiration & biomimetics*, vol. 8, no. 2, p. 026 003, 2013.

[62] K. Dowling, “Limbless locomotion: Learning to crawl,” in *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)*, IEEE, vol. 4, 1999, pp. 3001–3006.

[63] T. Ludeke and T. Iwasaki, “Exploiting natural dynamics for gait generation in undulatory locomotion,” *International Journal of Control*, vol. 93, no. 2, pp. 307–318, 2020.

[64] M. Mori and S. Hirose, “Three-dimensional serpentine motion and lateral rolling by active cord mechanism acm-r3,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1, 2002, 829–834 vol.1.

[65] K. Lipkin *et al.*, “Differentiable and piecewise differentiable gaits for snake robots,” in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2007, pp. 1864–1869.

[66] S. Park, H. Lee, and D. Lee, “Robust motion control of robotic systems with environmental interaction via data-driven inversion of cpg,” in *2020 20th International Conference on Control, Automation and Systems (ICCAS)*, 2020, pp. 692–698.

[67] Z. Bing, C. Lemke, L. Cheng, K. Huang, and A. Knoll, “Energy-efficient and damage-recovery slithering gait design for a snake-like robot based on reinforcement learning and inverse reinforcement learning,” *Neural Networks*, vol. 129, pp. 323–333, 2020.

[68] A. J. Ijspeert and A. Crespi, "Online trajectory generation in an amphibious snake robot using a lamprey-like central pattern generator model," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, 2007, pp. 262–268.

[69] P. Prautsch, T. Mita, and T. Iwasaki, "Analysis and control of a gait of snake robot," *IEEJ Transactions on Industry Applications*, vol. 120, no. 3, pp. 372–381, 2000.

[70] P. Prautsch and T. Mita, "Control and analysis of the gait of snake robots," in *Proceedings of the 1999 IEEE International Conference on Control Applications (Cat. No.99CH36328)*, vol. 1, 1999, 502–507 vol. 1.

[71] P. Prautsch, T. Mita, and T. Iwasaki, "Analysis and control of a gait of snake robot," *IEEJ Transactions on Industry Applications*, vol. 120, no. 3, pp. 372–381, 2000.

[72] M. Sato, M. Fukaya, and T. Iwasaki, "Serpentine locomotion with robotic snakes," *IEEE Control Systems Magazine*, vol. 22, no. 1, pp. 64–81, 2002.

[73] J. GRAY, "The Mechanism of Locomotion in Snakes," *Journal of Experimental Biology*, vol. 23, no. 2, pp. 101–120, 1946, ISSN: 0022-0949.

[74] E. Kelasidi, P. Liljeback, K. Y. Pettersen, and J. T. Gravdahl, "Innovation in underwater robots: Biologically inspired swimming snake robots," *IEEE Robotics Automation Magazine*, vol. 23, no. 1, pp. 44–62, 2016.

[75] 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*, vol. 1, no. 1, pp. 1–15, 2014.

[76] 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*, vol. 24, no. 3, pp. 884–899, 2016.

[77] H. Wang, H. Zhang, Z. Wang, and Q. Chen, "Impulsive control and stability analysis of biped robot based on virtual constraint and adaptive optimization," *Advanced Control for Applications*, vol. 2, no. 2, e32, 2020.

[78] S. Celikovsky and M. Anderle, "Stable walking gaits for a three-link planar biped robot with two actuators based on the collocated virtual holonomic constraints and the cyclic unactuated variable," *IFAC-PapersOnLine*, vol. 51, no. 22, pp. 378–385, 2018, 12th IFAC Symposium on Robot Control SYROCO 2018, ISSN: 2405-8963.

[79] A. M. Kohl, E. Kelasidi, A. Mohammadi, M. Maggiore, and K. Y. Pettersen, "Planar maneuvering control of underwater snake robots using virtual holonomic constraints," *Bioinspiration & Biomimetics*, vol. 11, no. 6, p. 065 005, 2016.

[80] D. Li, Z. Pan, H. Deng, and L. Hu, "Adaptive path following controller of a multijoint snake robot based on the improved serpenoid curve," *IEEE Transactions on Industrial Electronics*, 2021.

[81] E. Kelasidi, K. Pettersen, A. Kohl, and J. Gravdahl, "An experimental investigation of path following for an underwater snake robot with a caudal fin," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 11 182–11 190, 2017, 20th IFAC World Congress, ISSN: 2405-8963.

[82] A. Zhang, S. Ma, B. Li, M. Wang, X. Guo, and Y. Wang, "Adaptive controller design for underwater snake robot with unmatched uncertainties," *Science China Information Sciences*, vol. 59, no. 5, pp. 1–15, 2016.

[83] A. Zhang, S. Ma, B. Li, and M. Wang, "Curved path following control for planar eel robots," *Robotics and Autonomous Systems*, vol. 108, pp. 129–139, 2018, ISSN: 0921-8890.

[84] B. Bandyopadhyay, S. Janardhanan, S. K. Spurgeon, *et al.*, "Advances in sliding mode control," *Lecture Notes in Control and Information Sciences*, vol. 440, 2013.

[85] V. Utkin, J. Guldner, and J. Shi, *Sliding mode control in electro-mechanical systems*. CRC press, 2017.

[86] J. Mukherjee, I. N. Kar, and S. Mukherjee, "Adaptive sliding mode control for head-angle and velocity tracking of planar snake robot," in *2017 11th Asian Control Conference (ASCC)*, 2017, pp. 537–542.

[87] A. Orucevic, M. Wrzos-Kaminska, M. E. B. Lysø, K. Y. Pettersen, and J. T. Gravdahl, "Automatic alignment of underwater snake robots operating in wakes of bluff bodies," *Control Engineering Practice*, vol. 147, p. 105 904, 2024.

[88] W. Remmas, A. Chemori, and M. Kruusmaa, "Fault-tolerant control allocation for a bio-inspired underactuated auv in the presence of actuator failures: Design and experiments," *Ocean Engineering*, vol. 285, p. 115 327, 2023.

[89] D. Papageorgiou, G. Þ. Sigurðardóttir, E. Falotico, and S. Tolu, "Sliding-mode control of a soft robot based on data-driven sparse identification," *Control Engineering Practice*, vol. 144, p. 105 836, 2024.

[90] G. Ramamurthi, B. Bandyopadhyay, H. Arya, and G. K. Singh, "Tracking control for nonminimum phase mimo micro aerial vehicle – a dynamic sliding manifold approach," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 230, no. 9, pp. 1001–1029, 2016.

[91] J. Liu, "Neural network sliding mode control," in *Radial Basis Function (RBF) Neural Network Control for Mechanical Systems: Design, Analysis and Matlab Simulation*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 113–132, ISBN: 978-3-642-34816-7.

[92] X. Liu, R. Gasoto, Z. Jiang, C. Onal, and J. Fu, "Learning to locomote with artificial neural-network and cpg-based control in a soft snake robot," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 7758–7765.

[93] D. Neumerkel, J. Franz, and L. Krueger, "Real-time drive control with neural networks," *IEE CONTROL ENGINEERING SERIES*, pp. 179–202, 1995.

[94] M. Abu-Khalaf and F. L. Lewis, "Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network hjb approach," *Automatica*, vol. 41, no. 5, pp. 779–791, 2005.

[95] A. Elkenawy, A. M. El-Nagar, M. El-Bardini, and N. M. El-Rabaie, “Full-state neural network observer-based hybrid quantum diagonal recurrent neural network adaptive tracking control,” *Neural Computing and Applications*, vol. 33, no. 15, pp. 9221–9240, 2021.

[96] L. Guo and H. Zhao, “Online adaptive optimal control algorithm based on synchronous integral reinforcement learning with explorations,” *Neurocomputing*, vol. 520, pp. 250–261, 2023.

[97] W. He, H. Gao, C. Zhou, C. Yang, and Z. Li, “Reinforcement learning control of a flexible two-link manipulator: An experimental investigation,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 12, pp. 7326–7336, 2021.

[98] F. F. M. El-Sousy and F. A. F. Alenizi, “Optimal adaptive super-twisting sliding-mode control using online actor-critic neural networks for permanent-magnet synchronous motor drives,” *IEEE Access*, vol. 9, pp. 82 508–82 534, 2021.

[99] M. Li, Z. Cai, J. Zhao, J. Wang, and Y. Wang, “Disturbance rejection and high dynamic quadrotor control based on reinforcement learning and supervised learning,” *Neural Computing and Applications*, vol. 34, no. 13, pp. 11 141–11 161, 2022.

[100] Z. Song, J. Yang, X. Mei, T. Tao, and M. Xu, “Deep reinforcement learning for permanent magnet synchronous motor speed control systems,” *Neural Computing and Applications*, vol. 33, pp. 5409–5418, 2021.

[101] C. Liu, G. Wen, Z. Zhao, and R. Sedaghati, “Neural-network-based sliding-mode control of an uncertain robot using dynamic model approximated switching gain,” *IEEE Transactions on Cybernetics*, vol. 51, no. 5, pp. 2339–2346, 2021.

[102] M. Korayem, M. Safarbali, and N. Y. Lademakhi, “Adaptive robust control with slipping parameters estimation based on intelligent learning for wheeled mobile robot,” *ISA Transactions*, 2024.

[103] H. Li, X. Hu, X. Zhang, H. Chen, and Y. Li, “Adaptive radial basis function neural network sliding mode control of robot manipulator based on improved genetic algorithm,” *International Journal of Computer Integrated Manufacturing*, pp. 1–15, 2023.

[104] X. Fang, L. Wang, and K. Zhang, “Adaptive recurrent neural network intelligent sliding mode control of permanent magnet linear synchronous motor,” *Neural Computing and Applications*, vol. 36, no. 1, pp. 349–363, 2024.

[105] X. Yunlang, S. Feng, S. Xinyi, G. Liang, H. Shuo, and Y. Xiaofeng, “A composite neural network-based adaptive sliding mode control method for reluctance actuator maglev system,” *Neural Computing and Applications*, vol. 35, no. 21, pp. 15 877–15 890, 2023.

[106] X. Yu, Y. Fu, P. Li, and Y. Zhang, “Fault-tolerant aircraft control based on self-constructing fuzzy neural networks and multivariable smc under actuator faults,” *IEEE Transactions on Fuzzy Systems*, vol. 26, no. 4, pp. 2324–2335, 2018.

[107] A. M. Nejad, S. M. Hosseini, B. Sobhani, and A. Harifi, “Fault-tolerant controller based on artificial intelligence combined with terminal sliding mode and pre-described performance function applied on ship dynamic position stabilization systems,” *Engineering Applications of Artificial Intelligence*, vol. 132, p. 107 890, 2024.

[108] S. A. Woods, R. J. Bauer, and M. L. Seto, “Automated ballast tank control system for autonomous underwater vehicles,” *IEEE Journal of Oceanic Engineering*, vol. 37, no. 4, pp. 727–739, 2012.

[109] B. Tiwari and R. Sharma, “Study on system design and integration of variable buoyancy systems for underwater operation,” *Defence Science Journal*, vol. 71, no. 1, pp. 124–133, 2021.

[110] J. Lunze and D. Lehmann, “A state-feedback approach to event-based control,” *Automatica*, vol. 46, no. 1, pp. 211–215, 2010.

[111] M. Abbas, J. Narayan, and S. K. Dwivedy, “Event-triggered adaptive control for upper-limb robot-assisted passive rehabilitation exercises with input delay,” *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 236, no. 4, pp. 832–845, 2022.

[112] K. Nath and M. K. Bera, “Integral sliding mode control of networked robotic manipulator: A dynamic event-triggered design,” *Advanced Robotics*, pp. 1–13, 2022.

[113] J. Kadiyam, S. Mohan, D. Deshmukh, and T. Seo, “Simulation-based semi-empirical comparative study of fixed and vectored thruster configurations for an underwater vehicle,” *Ocean Engineering*, vol. 234, p. 109 231, 2021.

[114] A. K. KS, J. Kadiyam, and S. Mohan, “Comparative performance investigations of the intervention-class underwater vehicle with different possible thruster configurations using eight identical thrusters,” *Ocean Engineering*, vol. 288, p. 116 147, 2023.

[115] D. Xu, Z. Yan, H. Xie, L. Lu, Z. Xiao, and B. Han, “Demand-driven optimization method for thruster configuration of underwater vehicles considering capability boundary and system efficiency,” *Ocean Engineering*, vol. 310, p. 118 749, 2024.

[116] D. Zhang, R. Ju, and Z. Cao, “Reinforcement learning-based motion control for snake robots in complex environments,” *Robotica*, pp. 1–15, 2024.

[117] H. N. Le and N. T. Vo, “Modeling and analysis of an ruu delta robot using solidworks and simmechanics,” *International Journal of Dynamics and Control*, pp. 1–13, 2024.

[118] S. Modak and R. K. K, “Kinematics and singularity analysis of a novel hybrid industrial manipulator,” *Robotica*, vol. 42, no. 2, pp. 579–610, 2024.

[119] C. Cosenza, V. Niola, S. Pagano, and S. Savino, “Theoretical study on a modified rocker-bogie suspension for robotic rovers,” *Robotica*, vol. 41, no. 10, pp. 2915–2940, 2023.

[120] T. I. Fossen, *Handbook of marine craft hydrodynamics and motion control.* John Wiley & Sons, 2011.

[121] J. Colgate and K. Lynch, "Mechanics and control of swimming: A review," English (US), *IEEE Journal of Oceanic Engineering*, vol. 29, no. 3, pp. 660–673, 2004, ISSN: 0364-9059.

[122] M. Maggiore and L. Consolini, "Virtual holonomic constraints for euler–lagrange systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 4, pp. 1001–1008, 2013.

[123] H. K. Khalil, *Nonlinear systems third edition*. 2002, vol. 115.

[124] S. P. Bhat and D. S. Bernstein, "Finite-time stability of continuous autonomous systems," *SIAM Journal on Control and Optimization*, vol. 38, no. 3, pp. 751–766, 2000.

[125] F. Plestan, Y. Shtessel, V. Brégeault, and A. Poznyak, "New methodologies for adaptive sliding mode control," *International Journal of Control*, vol. 83, no. 9, pp. 1907–1919, 2010.

[126] R. Hendel, F. Khaber, and N. Essounbouli, "Adaptive high-order sliding mode controller-observer for mimo uncertain nonlinear systems," *Asian Journal of Control*, vol. 22, no. 6, pp. 2309–2329, 2020.