

**Modular Continuum Mobile Robot: Design, Modeling,
and Motion Planning**

by
Yinan Sun

A Dissertation
Submitted to the Faculty
of the
WORCESTER POLYTECHNIC INSTITUTE
in partial fulfillment of the requirements for the
Degree of Doctor of Philosophy
in
Robotics Engineering

August 2022

APPROVED:

Prof. Cagdas D. Onal, Advisor, Robotics Engineering Department, Worcester Polytechnic Institute

Prof. Berk Calli, Dissertation Committee, Robotics Engineering Department, Worcester Polytechnic Institute

Prof. Carlo Pinciroli, Dissertation Committee, Robotics Engineering Department, Worcester Polytechnic Institute

Prof. Ming Luo, Dissertation Committee, School of Mechanical and Materials Engineering, Washington State University

Acknowledgements

- My advisor: Prof. Cagdas D. Onal, for mentoring and guiding me through my master and PhD life;
- Committee Members: Prof. Berk Calli, Prof. Carlo Pinciroli, Prof. Ming Luo, for all your valuable contribute to this dissertation;
- Colleagues: Erik H. Skorina, Zhenyu Wan, Tianyu Cheng, Yun Qin, Shoushan Chiang, Aihaitijiang Abudula, Raagini Rameshwar, Hao Yang, Yuqi Jiang, Junius Santoso, Louis-Claude Walter, Gabrielle Conard, Dan Moyer, Arnold Muralt, Robin Hall, Mason Mitchell, Steven Hyland, Michael da Silva, and everyone else in WPI Soft Robotics Lab, you guys are awesome!
- Friends: Yao Shen, Xuan Liu, Gloria Lance, Anany Dwivedi, Pengbo Wang, Jiazhang Chen, Yibo Zhang, Haoran Wang, Ruibo Yan, Zhentian Qian, Renato Gasoto, for all your kindness and support;
- My parents, for all your love, support, and always proud of my career.
- This dissertation is based upon work supported by the National Science Foundation (NSF) under Grant Nos. IIS-1551219, CMMI-1728412, and CMMI-1752195. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

Abstract

Snakes and other creatures with flexible continuous bodies are a remarkable source of inspiration for mobile search-and-rescue robots. Their unique slender body structure and multiple locomotion modes are well-suited to negotiate with highly unstructured and unpredictable environments. The design, manufacturing, modeling, and control techniques of snake robots make it possible to imitate the structure, mechanical properties, and locomotion gaits of snakes, opening up new possibilities in robotics research. Unlike the creatures, most of existing snake robots are modular designed with rigid structures for locomotion conducting. The lack of flexibility reduces the snake robots' potential of adapt to complex environment.

Continuum robots are also well-suited to rescue and exploration applications because of their flexible and continuous structure. Continuum solutions are widely utilized on novel manipulators, presenting promising potential as robotic tools because of their ability to safely conform to the shape of the objects and environments they interact with. The unique property to perform shape changing on their structure, such as length, bending angle, and bending direction continuously along their arc length, makes them possible to negotiate much more complicated environments. However, considering their versatility, continuum robots also offer great potential to be configured as mobile robots for search-and-rescue applications in unstructured environments.

In this dissertation, novel mobile robots are presented, combining the modular design of snake robots and the flexible continuous structure from continuous robots, which enables their potential of multiple locomotion modes and the ability to negotiate complex environments.

This dissertation introduce our research work on a continuum mobile snake robot purely inspired by natural snakes with slender body, and a continuum salamander robot inspired partially by salamanders, combined with suitable engineering techniques for better performance.

The continuum soft snake robot is made of modules that can actively deform in 3D. We conducted rigorous studies to access its performance under a range of conditions, including gait parameters, number of modules, and different environmental conditions. We developed a flexible 3D-printed wave-spring sheath to support the robot modules, increasing the snake's performance in climbing steps three-fold. Finally, we introduce a simulator and a numerical model to provide a real-time simulation of the soft robotic snake. With the help of the real-time simulator, it is possible to develop and test new locomotion gaits for the soft robotic snake within a short period of time, compared to experimental trial and error. As a result, the soft robotic snake presented in this dissertation is able to traverse diverse surfaces, perform several bio-inspired and custom gaits, and climb over steps.

The continuum salamander robot consists of a cable-driven bellows-like origami module mounted between sets of powered wheels. The origami structure allows the body to deform as necessary to adapt to complex environments and terrains, while the wheels allow the robot to reach speeds of up to 303.1 mm/s (2.05 body-length/s). Considering the unique locomotion of this mobile robot, we analyze the locomotion under experimental results and reasonable assumptions, then build a Euler-spiral-based kinematics model to describe the continuum robot motion. Based on the model, we also explore feasible motion planning methods and deploy on the robot in real world experiments.

Dissertation Structure

The dissertation contains a background discussion, details of technical research methods and results, and some ideas for future work.

Chapter 1 discusses the background, literature review, and contributions of this dissertation.

Chapter 2 discusses research work on a continuum modular snake robot.

Chapter 3 discusses research work on a continuum modular salamander robot.

Chapter 4 concludes this dissertation, compares the two robots developed in this dissertation, and discusses related future research.

Chapter 5 is an Appendix, including extra experiments, exploring the locomotion behavior of the continuum modular snake robot, along with a table of recent related research on search and rescue robots with flexible structures as a comparison to this dissertation.

Table of Contents

Acknowledgements	i
Abstract	ii
Dissertation Structure	iv
List of Figures	xv
List of Tables	xvii
1 Introduction	1
1.1 Background and Motivation	1
1.1.1 Inspiration from Snake Robots	2
1.1.2 Inspiration from Continuum Robots	2
1.2 Literature Review	3
1.2.1 Snake Robot	3
1.2.2 Continuum Robot	4
1.3 Dissertation Contributions	6
1.3.1 Continuum Modular Snake Robot	7
1.3.2 Continuum Modular Salamander Robot	8
2 Continuum Soft Snake Robot	10
2.1 Design and Fabrication	11
2.1.1 Modular Design	11
2.1.2 Compound Material	12
2.1.3 Fabrication	12

2.1.4	Wave Spring Sheath Constraint	14
2.1.5	System Architecture	15
2.2	Robot Locomotion	16
2.2.1	Motion Capabilities of Soft Robotic Modules	16
2.2.2	Locomotion Capabilities of the Modular 3-D Soft Robotic Snake	17
2.3	Real Time Simulation	25
2.3.1	Nvidia Flex based simulation	26
2.3.2	Pressure Dynamics	28
2.3.3	Buckling effect and wave spring replication	29
2.4	Discussion and Conclusion	30
3	Continuum Salamander Robot	31
3.1	Design Details	33
3.1.1	Origami Module	33
3.1.2	Propulsion Wheel System	40
3.1.3	Signal Flow of Salamander Robot	43
3.2	Robot Demonstration	44
3.2.1	Linear Locomotion Speed on Different Surfaces	45
3.2.2	Climbing on a Ramp	47
3.2.3	Negotiating Narrow Passageways	48
3.2.4	Robustness through Radial Symmetry	49
3.2.5	Vertical Climb	50
3.2.6	Multi-module Cooperation	51
3.3	Kinematics Modeling	52
3.3.1	Inverse Kinematics for Origami Actuator	52
3.3.2	Kinematics Model for Single Module Salamander Robot	54
3.4	Motion Planning	60
3.4.1	Motion Planner with Lattice A-star	60
3.4.2	CHOMP Based Motion Planner	62

3.4.3	Comparison between Lattice based A-star and CHOMP based motion planner	68
3.4.4	Path Following	71
3.4.5	Motion Planner Deployed on Robot	72
3.5	Discussion and Conclusion	72
4	Conclusion and Future Research	75
4.1	Summary of Work and Contributions	75
4.1.1	Research Conclusion	75
4.1.2	Comparison Between Snake and Salamander Robot	76
4.2	Future Research	76
4.2.1	Soft Snake Robot	76
4.2.2	Continuum Salamander Robot	77
4.2.3	Automatic Gait Generator	79
5	Appendix	80
5.1	Modular Soft Snake Robot Locomotion Data on Different Surfaces	80
5.1.1	Undulatory Locomotion	80
5.1.2	Sidewinding Locomotion	83
5.2	Recent Research about Flexible Mobile Robots	87

List of Figures

1.1	(a) ACM-R5 snake robot developed by researchers in TITECH (Yamada (2005)). (b) Fire fighter snake robot developed by researchers from NTNU (Liljeback et al. (2006)). (c) Obstacle aided snake robot Aiko developed by researchers in NTNU (Transeth et al. (2008a)). (d) CMU modular snake robot capable of multiple locomotion modes (Wright et al. (2012)). (e) JHU snake robot capable of climbing over large scale object using a snake-like partitioned gait (Gart et al. (2019)). (f) TUKE robotic snake capable of snake like pipe climbing locomotion (Virgala et al. (2021)).	5
1.2	(a) MIT pneumatic continuum robot extremely compliant to the unstructured environment (Marchese et al. (2014)). (b) UCSB growing robot, capable of reaching target while dodging the obstacles (Greer et al. (2019)).	7
2.1	Modular designed soft snake robot.	10
2.2	Left: Inflation and deflation curve with shaded error region. (the green line represent the time when signal input is applied, and the red line represent the end of the signal input). The inflation pressure is 17 psi. Right: A bending module in testing.	11
2.3	Fabrication process of the 3-D soft robotic bending module. (a) The mold of a single linear-stretched pneumatic. (b) The single actuator without thread. (c) The threaded single actuator. (d) The mold of the outer body, which is used to combine the three threaded single actuators. (e) The demolded soft 3-D bending segment. (f) The soft segment sealed with acrylic plates and the vent screw. (g) The assembled soft robotic bending module.	13

2.4	Top-left: An example of a soft robotic module buckling in a cantilever configuration. The lower actuation chamber is pressurized while a downwards force is applied to the tip; Bottom-left: A fully integrated module and the wave spring sheath 3-D printed out of NinjaFlex; Top-right: Snake robot made of five modules with NinjaFlex sheaths mounted; Bottom-right: Wave spring sheath with and without horizontal support 3-D printed out of Objet FLX9085.	14
2.5	Left: Green data points represent the trajectory of the central point at the top end-plate (tip) of the module when performing the hexagonal gait. The trajectory, which also can be considered as the workspace, is a deformed hexagon on a (red) spherical. Right: A soft robotic module in testing. The blue and white dots show the status of the three actuation chambers. Blue dots mean inflated states and white dots mean deflated states.	15
2.6	The max tip height achieved by the bending module in cantilever configuration under tip loading. With reduced buckling effect, the actuators bends higher.	17
2.7	Top-Left: Trajectory of the soft robotic snake CoM (central of mass) when locomotion frequency is 1.50 Hz. Top-Middle: Trajectory of the soft robotic snake CoM when locomotion frequency is 1.75 Hz. Bottom-Left: Trajectory of the soft robotic snake CoM when locomotion frequency is 2.00 Hz. Bottom-Middle: Soft robotic snake performing lateral undulation locomotion from right side to left side in real world. Right: Error between simulation result and real world experiment result with relate to distance traveled.	19
2.8	(a) Lateral undulatory locomotion on paper; (b) Lateral undulatory locomotion on grass; (c) Mathematical Model calculated result of overall trajectory of central point under lateral undulatory locomotion with same parameter on different surfaces. Yellow circle shows the start point. Red and blue circle shows the end point of locomotion on paper and grass respectively	21

2.9	Top-Left: Trajectory of the soft robotic snake CoM (central of mass) when locomotion frequency is 0.75 Hz. Top-Middle: Trajectory of the soft robotic snake CoM when locomotion frequency is 0.875 Hz. Bottom-Left: Trajectory of the soft robotic snake CoM when locomotion frequency is 1.00 Hz. Bottom-Middle: Soft robotic snake performing sidewinding locomotion on a paper surface. Right: Error between simulation result and real world experiment result with relate to distance traveled.	22
2.10	Step climbing in real world experiment (top) and in real-time simulation (center). Bottom: Actuation States of the proposed soft robotic modules. Each subplot displays the cross-section of the soft module with three pneumatic chambers around a central slot for wiring. Red chambers are pressurized, and white chambers are depressurized.	23
2.11	A snapshot of the graphical user interface (GUI) of Nvidia Flex during a sidewinding locomotion simulation. Parameters can be either attached in the slider tabs for quick adjustment, or simply listed on the screen for data observation.	27
3.1	(top) Modular designed salamander robot consists of a cable-driven origami actuator, a propulsion mechanism, a power package, and a remote communication module to receive and send messages individually. Extra sensors can also be attached to help mapping optical and thermal environments around if necessary. (bottom) Single module salamander robot conducting Simultaneous Localization and Mapping (SLAM).	32
3.2	(left) Yoshimura pattern plot in CorelDraw software for laser cutting. Folding lines marked red are cut in dashed lines and the Black lines are cut through. (middle) Laser cutted PET sheet folded into Yoshimura patter. (top-right) Top folds. (bottom-right) Bottom folds.	34
3.3	(left)Two Yoshimura pattern folded PET sheets combined together through fork-slot mechanism. (middle) Top view of an assembled triangular origami bellow. (bottom-right) Front view of an assembled triangular origami bellow.	35

3.4	(left)Front view of the PCB Chip Diagram (dot locations highlighted in yellow). (right) Bottom view of the PCB, with cable slots for the three DC motors on the module.	36
3.5	(a) Fully stretched origami module, with the length of 148mm, (b) Fully compressed origami module, the length is 57mm, 38% of the maximum length, (c) Maximum bending angle of our origami module is 106 deg.	38
3.6	(top-left) Top view of the 3D printed case with shaft holder and mounting holes for the motors. (top-middle) Bottom view of the 3D printed case for PCB holding. (top-right) Top view of an assembled origami module with motors. (bottom-left) Bottom view of an assembled origami module with the PCB board. (bottom-middle) Front view of an assembled origami module of the motor cable slot on PCB board through the preset hollow out structure on 3D printed case. (bottom-right) Side view of an assembled origami module of the motor shafts fixed on bearings in shaft holders.	39
3.7	(left) CAD rendering of belt transmission design, (middle) Fabricated belt transmission design. (right) Fully assembled salamander robot with belt transmission. .	41
3.8	(left) CAD rendering of gear transmission design, (middle) Fabricated gear transmission design. (right) Fully assembled salamander robot with gear transmission. .	42
3.9	(top-left) CAD rendering of active wheels with suspension design. (bottom-left) 3D printed parts with motors mounted. (right) Assembled wheel set with notation of each part.	42
3.10	Wiring chart of a single module salamander robot.	44
3.11	Signal flow chart of a single module salamander robot.	45
3.12	Salamander robot going through a maze.	46
3.13	(a) Overlapping between the traditional 4-wheel robot with the maze, (b) Salamander robot with the same length can deform and go through the maze.	47
3.14	Salamander robot climbing on a 60 deg carpet surface slope.	48

3.15	During the experiment, salamander robot was turned over several times, while the velocity stayed the same. The yellow arrows show the turning over directions. . . .	49
3.16	A single module salamander robot moving vertically along a narrow gap for a body length in 16 s with time stamp.	50
3.17	(left) A 3 module salamander robot turning right. (top-right) A 3 module salamander robot moving straight forward. (bottom-right) A 3 module salamander robot raising two ends.	51
3.18	A 3 module salamander robot climbing over a 15 cm high obstacle.	52
3.19	(left) Sketch of the origami continuum module. The three parameters to present module states are marked red. The green arc represent the central axis of the module and C represent it central of circle. (right) Inverse kinematics parameters and coordinate overlaid on a bending continuum origami module	53
3.20	(left) Research about a simplified snake robot modeling (Rezapour et al. (2014)) (right) A typical kinematics model sketch of a two-wheel mobile robot.	55
3.21	Simplified sketch of a single module salamander robot. The pink dumbbell-like shape on two ends represent the two wheel sets of the robot. The bending block in between represent the origami module. Notations on the sketch are explained in Section 3.3.2	55
3.22	Two Euler spirals with different ratio of length to curvature.	58
3.23	A piecewise curve consists of three pieces of trimmed Euler spirals	60
3.24	A sketch of lattice based A-star searching algorithm. The curves represent the lattice generated from each node. The green curves represent the searching result. .	61
3.25	Two example of lattice based A-star search with different obstacle setting. The left trial searched 49735 nodes in 6.36 s to reach a solution. The right trial searched 883165 nodes in 48.89 s	62
3.26	Trajectory updated by gradient descent from iteration 1 to iteration 10.	66

3.27	Trajectory updated by gradient descent with orientation and curvature constraint applied. Yellow circles represent the virtual obstacles generated in the previous iteration. The beginning orientation angle is set to be -90 deg, and the ending orientation angle is also set to be -90 deg.	69
3.28	(left) Trajectory trapped in local optimization. (right) Restart when detect possible local optimization. Light red curve represent the most promising candidate, and the dark red curve represent the trajectory in current iteration.	70
3.29	A sketch of LOS algorithm. Dashed circle with radius of R represent the area to switch way point.	72
3.30	(left) Motion planning result. The red curve represent the best trajectory generated in planner after 60 iterations. (right) Path tracking during traversing. The blue, red, and yellow points represent the robot position, projection point, and reference point respectively. The orange line represent orientation of the robot.	73
3.31	Motion planning and path tracking deployed on a single module salamander robot with time stamp.	73
4.1	CAD model presenting the simplified mechanical structure of the origami module.	78
4.2	Real time simulation of a simplified multi-module salamander robot with Pybullet.	79
5.1	Trajectories highlighting the maximum undulation speed (in body-lengths per second) for each number of modules in the Snake. Different frequencies, offsets, and surface materials were tested, and the fastest shown here. For three segment robot, we choose the parameter: rubber surface, 1.75 Hz frequency, $\frac{2}{3}\pi$ phase difference; for the four segment robot, we choose the parameter: rubber surface, 1.50 Hz frequency, $\frac{2}{3}\pi$ phase difference; for the five segment robot, we choose the parameter: paper surface, 2.00 Hz frequency, $\frac{3}{4}\pi$ phase difference.	82

5.2	Speed and deviation measured with curvature of the trajectory for five-segment soft snakes performing undulatory locomotion on a paper surface with different undulation frequencies and segment offsets tested in real world experiments and simulation. (top-left): Speed in experiments; (top-right): Curvature of the trajectory in experiment; (bottom-left): Speed in simulation; (bottom-right): Curvature of the trajectory in simulation.	83
5.3	Trajectories of the snake performing undulatory locomotion on different surfaces. For each surface chosen, we choose the parameters which maximized the speed. On grass, we choose parameters: 2.00 Hz frequency, $\frac{3}{4}\pi$ phase difference; on carpet, we choose parameter: 1.50 Hz frequency, $\frac{5}{6}\pi$ phase difference; on paper, we choose parameter: 2.00 Hz frequency, $\frac{3}{4}\pi$ phase difference; on rubber, we choose parameter: 2.00 Hz frequency, $\frac{2}{3}\pi$ phase difference.	84
5.4	Trajectories highlighting the maximum undulation speed (in body-lengths per second) for each number of modules in the Snake. Different frequencies, offsets, and surface materials were tested, and the fastest shown here. For three segment robot, we choose the parameter: rubber surface, 0.875 Hz frequency, $\frac{1}{3}\pi$ phase difference; for the four segment robot, we choose the parameter: rubber surface, 1.00 Hz frequency, $\frac{1}{2}\pi$ phase difference; for the five segment robot, we choose the parameter: rubber surface, 0.875 Hz frequency, $\frac{1}{2}\pi$ phase difference.	87
5.5	Speed and deviation measured with curvature of the trajectory for five-segment soft snakes performing sidewinding locomotion on a paper surface with different undulation frequencies and segment offsets tested in real world experiments and simulation. (top-left): Speed in experiments; (top-right): Curvature of the trajectory in experiment; (bottom-left): Speed in simulation; (bottom-right): Curvature of the trajectory in simulation.	88

5.6 Trajectories of the snake performing sidewinding locomotion on different surfaces. For each surface chosen, we choose the parameters which maximized the speed. On grass, we choose parameters: 1.00 Hz frequency, $\frac{5}{6}\pi$ phase difference; on carpet, we choose parameter: 1.00 Hz frequency, $\frac{5}{6}\pi$ phase difference; on paper, we choose parameter: 0.875 Hz frequency, $\frac{1}{2}\pi$ phase difference; on rubber, we choose parameter: 0.875 Hz frequency, $\frac{1}{2}\pi$ phase difference. 89

List of Tables

2.1	Dimension of a 4 segments modular soft robotic snake.	12
2.2	Parameters of gait functions.	21
2.3	State of each segment to lift each wheel in step climbing locomotion. “/” means switching, for example, “2/1” means switching between state 2 and state 1 under the frequency of ω . The states are explained in Figure 2.10.	24
3.1	Linear speed with belt or gear transmission system on different surfaces	46
3.2	Maximum angle of climb for modules with belt or gear transmission system on different surfaces	48
3.3	Parameters appeared in inverse kinematics for origami module.	53
3.4	Parameters appeared in single module salamander robot kinematics modeling.	57
3.5	Parameters appeared in Euler spiral related equations	59
3.6	Comparison between salamander motion planner based on lattice A-star and CHOMP	71
4.1	Comparison between continuum snake robot and salamander robot.	76
5.1	Sliding and rolling friction factor of different surfaces	81
5.2	Undulation locomotion velocity by body length per second (BL/s), 3 segments (total length 404 mm). The fastest result is marked red.	84
5.3	Undulation locomotion velocity by body length per second (BL/s), 4 segments (total length 510 mm). The fastest result is marked red.	85
5.4	Undulation locomotion velocity by body length per second (BL/s), 5 segments (total length 616 mm). The fastest result is marked red.	86

5.5	Sidewinding locomotion velocity by body length per second (BL/s), 3 segments (total length 404 mm). The fastest result is marked red.	89
5.6	Sidewinding locomotion velocity by body length per second (BL/s), 4 segments (total length 510 mm). The fastest result is marked red.	90
5.7	Sidewinding locomotion velocity by body length per second (BL/s), 5 segments (total length 616 mm). The fastest result is marked red.	91
5.8	Taxonomy of recent flexible mobile robot research.	92

Chapter 1

Introduction

1.1 Background and Motivation

Over 100,000 people die each year from natural disasters like earthquakes, tornado, hurricanes, and wildfires. World's most recent deadly events include the Indian Ocean tsunami and the earthquake in Haiti. Search and rescue operation are deployed to save lives affected, while this is actually a difficult and dangerous task for the human rescuers.

Robotics has been developed since mid-20th century. Nowadays, the state of art robots help human society in many aspect with extraordinary ability, even capable of doing Parkour (a military training course attempting to pass obstacles efficiently without assisting equipment). It is expected that robots can also help human beings in difficult and dangerous tasks like search and rescue mission. However, rescuers need to search for survivors trapped in rubble like structure, which is highly unstructured and unpredictable environment. Most robots are not designed for this situation. Existing robots might either be too big to fit through, or too small, that would be stopped by large size obstacles. Also, rescuers are not only expected to get over rubble like structure, but also get inside of it for human rescue. Our goal is to build a robot capable of autonomously traversing over or inside of unstructured and unpredictable environments with agility and robustness.

1.1.1 Inspiration from Snake Robots

Snakes are a remarkable source of inspiration for mobile search-and-rescue robots. Their unique slender body structure and multiple modes of locomotion are well-suited to movement in narrow passages and other difficult terrain.

Snake robots may be a solution for traversing these complicated environments. As very flexible and agile creatures, biological snakes evolved to move efficiently in different kinds of terrains because of their unique structure and locomotion gaits. In addition, the softness of snake bodies increase their adaptability and flexibility which makes them compliant to contact with the environment. These features make snake-inspired robots suitable for search-and-rescue missions in uncertain and constrained environments.

1.1.2 Inspiration from Continuum Robots

Another subset of bio-inspired robotics research is continuum robots, which is inspired by flexible and continuous biological structures like snakes, elephant trunks, and octopuses. In most of the related research, continuum robots are utilized on novel manipulators, presenting promising potential as robotic tools because of their ability to safely conform to the shape of the objects and environments they interact with. Traditional manipulators are usually rigid and heavy, with limited degrees of freedom, which reduce their dexterity in complicated, cluttered, or maze-like environments. Continuum manipulators are able to change their shape, such as length, bending angle, and bend to change direction continuously along their arc length, which makes them possible to navigate much more complicated environments. Consequently, continuum manipulators are widely used in flexible arm research. However, considering their versatility, continuum robots also offer great potential to be configured as mobile robots for search-and-rescue applications in unstructured environments.

1.2 Literature Review

1.2.1 Snake Robot

Snake robots have been created with traditional rigid structures as shown in Figure 1.1. Hirose et al. (Hirose and Yamada (2009)) studied the kinematics of snake motion and developed multiple snake like robots with active or passive joints and wheels. Choset's research group (Wright et al. (2007, 2012); Marvi et al. (2014); McKenna et al. (2008)) developed a rigid modular snake actuated by servo motors mounted in series, which is capable of multiple gaits including climbing up inside a wall opening and raising its head to observe the environment behind an obstacle. Gravdahl's research group developed an underwater snake robot, propelling with eel like locomotion. Crespi and Ijspeert et al. (Ijspeert and Crespi (2007); Crespi and Ijspeert (2006)) developed a rigid snake robot actuated with DC motors. This robot can conduct planar locomotion with CPG (central pattern generator) on land and swim in water. Transeth et al. (Transeth et al. (2007, 2008b,a)) developed a snake robot "Aiko", which can do obstacle-aided locomotion in addition to frequently used locomotion methods like lateral undulation and sidewinding gaits. In order to increase adaptive functionality to complex terrain, Kano et al. (Kano et al. (2017)) developed a decentralized control scheme, which enables the robotic snake to generate reasonable locomotion depending on surrounding environment. Considering the large size obstacle in the environment, researchers in JHU designed a snake robot capable of snake-like partitioned gait for object climbing (Gart et al. (2019)). The TUKE robotic snake can mimic pipe climbing locomotion of snakes which enables the snake to traverse inside of pipe structure (Virgala et al. (2021)).

Most of the existing snake robots consist of rigid links serially connected by revolution joints, forming a high degree of freedom system to approximate snake locomotion. However, this discretization with rigid joints is different from the continuous deformation of a biological snake. It requires many more segments of rigid links to create a close approximation of the body motion, which results in more weight, complexity in coordination of the redundant degrees of freedom, and increased energy consumption. In addition, rigid links have to be as short as possible to make sure

they don't jam in tight corners, which reduces movement options for a rigid snake, particularly in narrow places.

Our research work presented a continuum pneumatic soft robotic snake. In order to create anisotropic friction, which is necessary in serpentine locomotion, we utilize passive wheels. Godage's research group (Arachchige et al. (2021)) designed a soft robotic snake without passive wheels, and fulfill the propulsion by inward and outward rolling locomotion. However the velocity is limited, and the system require a considerable number of tethering tubes for pressure input. Another research work (Branyan et al. (2020)) utilize kirigami pattern to create a novel "snake skin", able to create anisotropic friction for snake like robot. However for our snake structure, it is difficult to come up with a skin cover without reducing performance. We also developed a composite silicone rubber material to improve the robustness of 3-D snake robot modules to increase their bending amplitude and performed preliminary tests with lateral undulation and sidewinding gaits of an early research work of a 3-D soft robotic snake comprising four of these modules (Qin et al. (2018)). In a separate study, we studied soft hybrid wave-spring actuators (Skorina and Onal (2020)), which is used for soft manipulation systems to be able to carry large payloads by reducing the buckling effect.

1.2.2 Continuum Robot

Researchers utilize continuum manipulators in minimally invasive surgery (Qi et al. (2018)) and inspection tasks in nuclear facilities (Dong et al. (2019)), in which the work spaces are usually cluttered while being unreachable or unsafe for human beings. Another pressing application is search-and-rescue. During search-and-rescue missions, rescuers need to search for survivors trapped in piles of rubble, which are highly unstructured and unpredictable environment. Rescue tasks are usually dangerous and difficult for human rescuers. However, continuum robots present considerable potential as low-cost and efficient rescue robots because of their ability to navigate negotiate complicated environments by performing shape changing on their structure, such as length, bending angle, and bending direction continuously along their body. While continuum manipulators have been shown to work well in similar environments, such as narrow rubble-filled tunnels of an

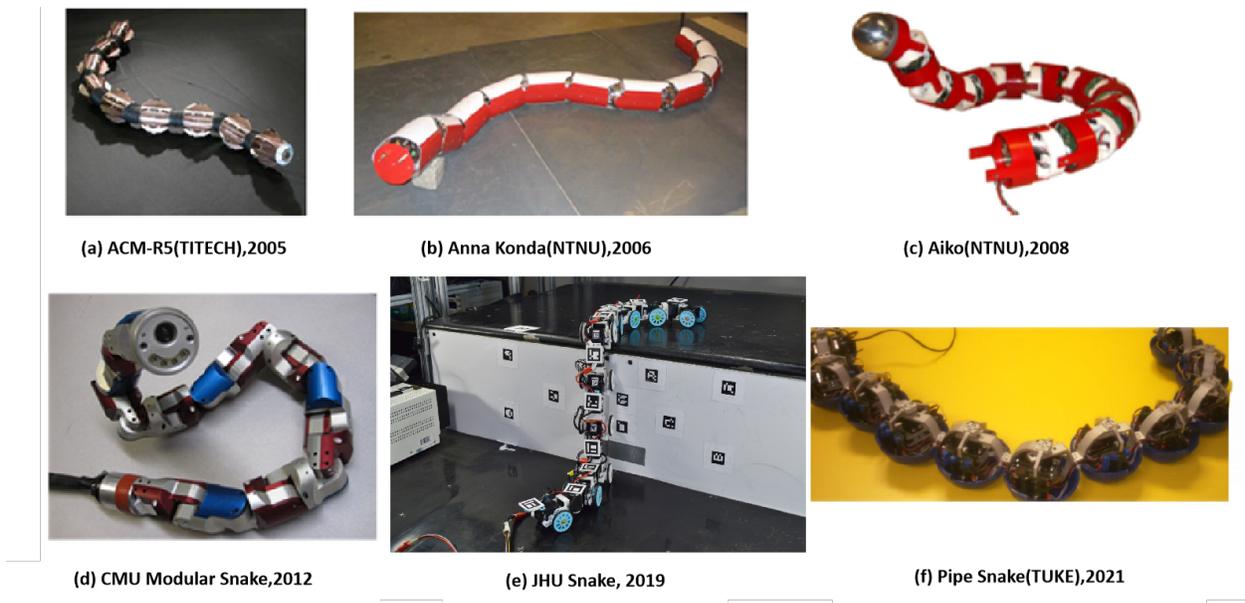


Figure 1.1: (a) ACM-R5 snake robot developed by researchers in TITECH (Yamada (2005)). (b) Fire fighter snake robot developed by researchers from NTNU (Liljeback et al. (2006)). (c) Obstacle aided snake robot Aiko developed by researchers in NTNU (Transeth et al. (2008a)). (d) CMU modular snake robot capable of multiple locomotion modes (Wright et al. (2012)). (e) JHU snake robot capable of climbing over large scale object using a snake-like partitioned gait (Gart et al. (2019)). (f) TUKE robotic snake capable of snake like pipe climbing locomotion (Virgala et al. (2021)).

archaeological site (Coad et al. (2020)), the distance the end of the robot can reach is limited by the length of the manipulator. As a result, mobile robots utilizing continuum structures serve as a promising alternative.

Researchers have utilized different types of actuation methods for continuum robots, including pneumatic, hydraulic, and motor-cable driven. Hydraulic and pneumatic actuation are commonly used on continuum robotic manipulators as shown in Figure 1.2. For example, researchers designed a pneumatic or origami-inspired continuum manipulator, actuated by an external air compressor through the balloons inside (Fathi et al. (2019); Marchese et al. (2014)). Another example is vine robot (Coad et al. (2020); Greer et al. (2019)), which is a kind of pneumatic continuum robotic manipulator able to continuously grow from the robot tip. Vine robots have been proved to be incredibly robust against sharp objects and adept at navigating unstructured environments

or narrow openings. In another work (Xavier et al. (2019)), the authors combine hydraulic soft actuators serially to create a worm-like robot that can climb up inside of a smooth pipe. Hydraulic and pneumatic continuum robots are relatively flexible with continuously deformable property. However, a pressurized pneumatic or hydraulic source with tubes, valves, and compressors or pumps are needed for these two actuation methods, which would hinder the movement of tetherless mobile robots, especially in the case of search-and-rescue applications since mobile robots without bulky or heavy payloads will be significantly more efficient in unstructured environments or narrow openings.

Compared to hydraulic or pneumatic actuation, motor-cable driven actuation could be a better choice for autonomous mobile robot since it can be operated with an on-board electronic system powered by small rechargeable batteries. In addition, motor-cable driven continuum robots tend to be easier to model due to the accuracy and predictability of their motion (Webster III and Jones (2010)). There has been much research work on the design, fabrication, and modeling of cable-driven continuum manipulators (Godage et al. (2019); Wooten et al. (2018); Yeshmukhametov et al. (2019)), but seldom work has been done on cable-driven continuum mobile robots.

In order to design and fabricate a lightweight robot for more agile performance, we decide to utilize the origami technique to design and fabricate the continuously deformable robot structure. Origami is known as a traditional art of folding two dimensional sheets, for example paper and plastic sheets, in certain crease patterns to create complex three dimensional structures. It is widely used as a promising research direction towards light and deformable structures with active or passive degrees of freedom, while maintaining considerable resistance to unexpected deformations (Onal et al. (2014)). With the unique features, origami technique is widely used to develop light weighted, deformable, and low cost robots (Zhang et al. (2016); Banerjee et al. (2018)).

1.3 Dissertation Contributions

In this dissertation, we utilize the unique properties of continuum robots, combining with the snake robot design, to create mobile robots with the potential of multiple locomotion modes and capable

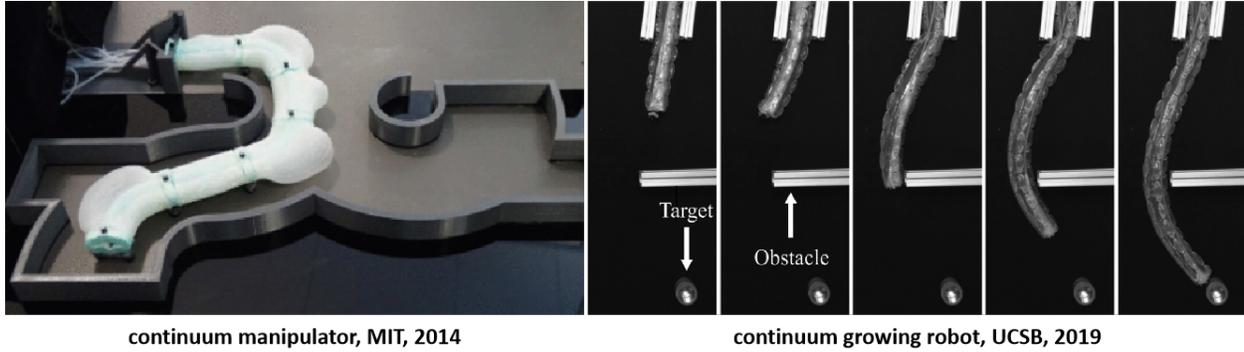


Figure 1.2: (a) MIT pneumatic continuum robot extremely compliant to the unstructured environment (Marchese et al. (2014)). (b) UCSB growing robot, capable of reaching target while dodging the obstacles (Greer et al. (2019)).

of autonomously negotiating with unexpected and unstructured environments. The contribution consists of the following two sections:

1.3.1 Continuum Modular Snake Robot

A continuum soft snake robot is designed and fabricated and the main contributions of this work include:

- We present a 3D soft robotic snake with
 - a modular system architecture comprising identical modules with integrated 3D soft actuation, valving, and electronics;
 - wave-spring sheaths that enhance the resistance against buckling and twisting when the snake modules are under large external forces.
- We demonstrate and verify different locomotion gaits and methods with the 3D soft robotic snake prototype, including
 - Lateral undulation,
 - Sidewinding locomotion, and
 - A novel step climbing locomotion which is developed in the real-time simulation conveniently and rapidly as compared with experimental iterations.

- We present a real-time simulation for this complex pneumatic soft robotic system to provide an intuitive environment for more flexible design and assessment of various gaits with significant advantages of developing and testing in a rapid pace.

1.3.2 Continuum Modular Salamander Robot

Considering our primal motivation is to build a robotics solution against complex environments and unpredictable outdoor environment, a pure biomimetic approach could be limited by existing soft robotics technique. On the contrary, it is reasonable to be inspired by biological creatures while using engineering judgment to build a novel robot with the potential to overcome existing issues for example the buckling effect discussed in Section 2.1.4.

Similar to snakes, salamanders are also very flexible animals with long deformable slender bodies, while the limbs conduct locomotion instead of body undulation, so they do not have to generate certain pattern of deformation to move. Instead, the flexible body only helps to change direction to potentially traverse narrow paths when they move forward, which decouples propulsion and body deformation.

The inspiration from salamander and engineering judgement result in a novel mobile robot with continuum origami module and active wheels. In this case, instead of generating locomotion, the continuum module is only used to deform the body for traversal, and the active wheels provide the locomotion. We name this novel mobile robot as mobile continuum salamander robot.

The main contributions of this work including developing a novel flexible origami mobile robot with continuous locomotion, and a feasible motion planning method for the robot. The mobile robot comes with the properties below:

- Light weight, low cost, and deformable;
- Relatively high maximum speed and climbing ability;
- Extremely fit for narrow and complicated maze-like terrain;
- Modular design, with demonstrated potential of completing more complicated tasks when serially assembled.

The motion planning method is feasible for robots with continuous locomotion, Trajectories generated by this motion planning method has the following properties:

- Smooth and collision free;
- Determined beginning and ending orientation;
- Constrained maximum curvature.

Chapter 2

Continuum Soft Snake Robot

This section presents the design, fabrication, verification, and real time simulation of a modular soft snake robot based on our previous research about a pneumatic snake robot (Luo et al. (2015)). With 3-degree-of-freedom (DoF) modules, the proposed modular soft snake robot can perform both planar locomotion, for example lateral undulation, and three-dimensional (3D) locomotion like climbing over obstacle. Three different locomotion methods, lateral undulation, sidewinding locomotion, and step climbing, are presented and tested on different surfaces and with different numbers of modules. A simulation environment is developed in parallel using NVidia Flex, which is a particle-based simulation technique developed by NVidia Gameworks.

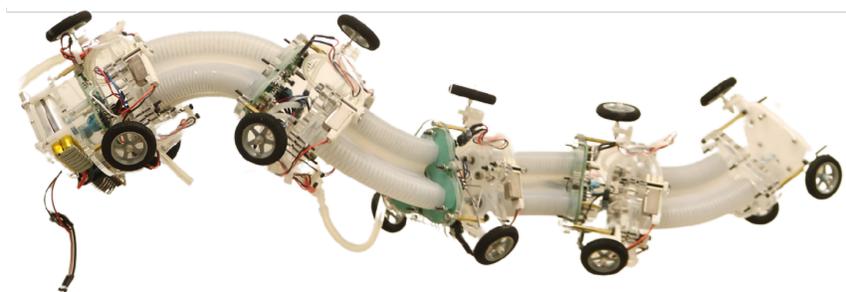


Figure 2.1: Modular designed soft snake robot.

This chapter is organized as follows: In Design and Fabrication section, we describe the design details and fabrication methods utilized for our soft robotic snake and wave-spring sheath; In Robot Locomotion section, we demonstrate three different locomotion gaits with our proposed

soft robotic snake platform, and we compare the locomotion path with our simulation results; In Real Time Simulation section, we introduce the GPU-based NVidia Flex simulator based on Physx engine, details about real-time soft matter simulation, seamless multi-physics mechanics and more details about simulation environment construction for our pneumatic soft snake robot.

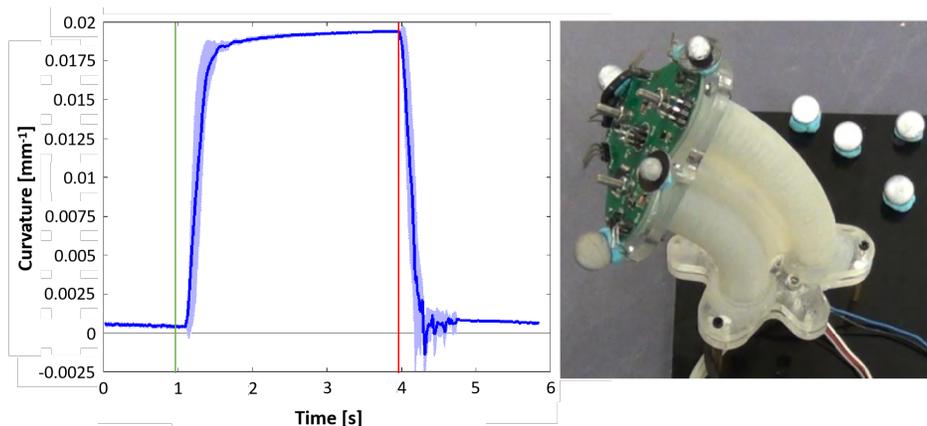


Figure 2.2: Left: Inflation and deflation curve with shaded error region. (the green line represent the time when signal input is applied, and the red line represent the end of the signal input). The inflation pressure is 17 psi. Right: A bending module in testing.

2.1 Design and Fabrication

2.1.1 Modular Design

Our 3D soft robotic snake incorporates a modular architecture based on our previous prototypes (Qin et al. (2018)), as shown in Figure 2.1. Dimension of a four-segment modular soft robotic snake is listed in Table 2.1 Modular design allows broken modules to be easily replaced and repaired separately, increasing the maintenance efficiency and system reliability.

In addition, the modular design makes it possible to scale the number of the modules, which would provide redundancy. A greater number of modules is helpful especially for step climbing experiments.

Each module consists of a pneumatic silicone rubber soft bending body with three embedded chambers, three solenoid valves and a peripheral controller printed circuit board (PCB). The three

Table 2.1: Dimension of a 4 segments modular soft robotic snake.

Parameter	Value (mm)
Module length	70
Height	100
Inner chamber diameter	8
Passive wheel diameter	20
Total length	510

valves control the air flow of the three chambers independently and the states of the valves are controlled by the peripheral controller. A main controller mounted in the head module connects to the peripheral controllers through an I²C communication bus to send desired values of valve duty cycles mapped from the target pressure. Peripheral controllers operate the solenoid valves with a pulse-width modulation (PWM) signal according to the received duty cycle, so that the chambers in each module can be actuated with a corresponding pressure.

2.1.2 Compound Material

In our previous research (Qin et al. (2018)) we used Ecoflex 0030 as the material of the 3D soft robotic snake robot. We observed buckling when the soft snake robot was performing a sidewinding locomotion since the modules were under relatively high loads while operating. To find an ideal material, a range of ratios of Ecoflex 0030, Ecoflex 0050 and DragonSkin 10 are used to build snake robot modules. The stiffness and modular bending performance are tested. The results show that the combination of 40% Ecoflex 0030 and 60% DragonSkin 10 by weight has the best performance against buckling. With this mixture, the buckling is eliminated when the 3D soft robotic snake performs sidewinding locomotion.

2.1.3 Fabrication

Each module includes a soft bending segment constructed from 3 linear-stretched pneumatic actuators, the three valves used to control the air flow in this segment, and a module controller circuit

to drive these valves. The states of the valves on each module are controlled by a main controller, which is connected to each slave controller by I^2C communication. The linear-stretched pneumatic actuators consist of tubes of silicone rubber wrapped in helix of thread. When pressurized, the shallow angle of the thread (approaching perpendicular to the tube) prevents the tube from deforming radially into a sphere, and instead causes it to extend.

Three of these linear-stretched pneumatic actuators are fused together with silicone to form a soft bending segment. When one of the actuators in a segment is pressurized, the un-actuated material causes it to bend in the opposite direction. Each chamber is sealed at both ends using a pair of acrylic plates bolted together around a flange of silicone, so the flange of silicone can be seen as a gasket (Luo et al. (2017)). A vent screw (A screw with a hole drilled through it) is used on one side of each chamber to provide access for pneumatic pressure. Tubes connect each chamber to a corresponding valve. Each of these valves is given commands by the custom slave controller board mounted on one end of the module (replacing one of the acrylic plates). The fabrication steps are shown in Figure 2.3.

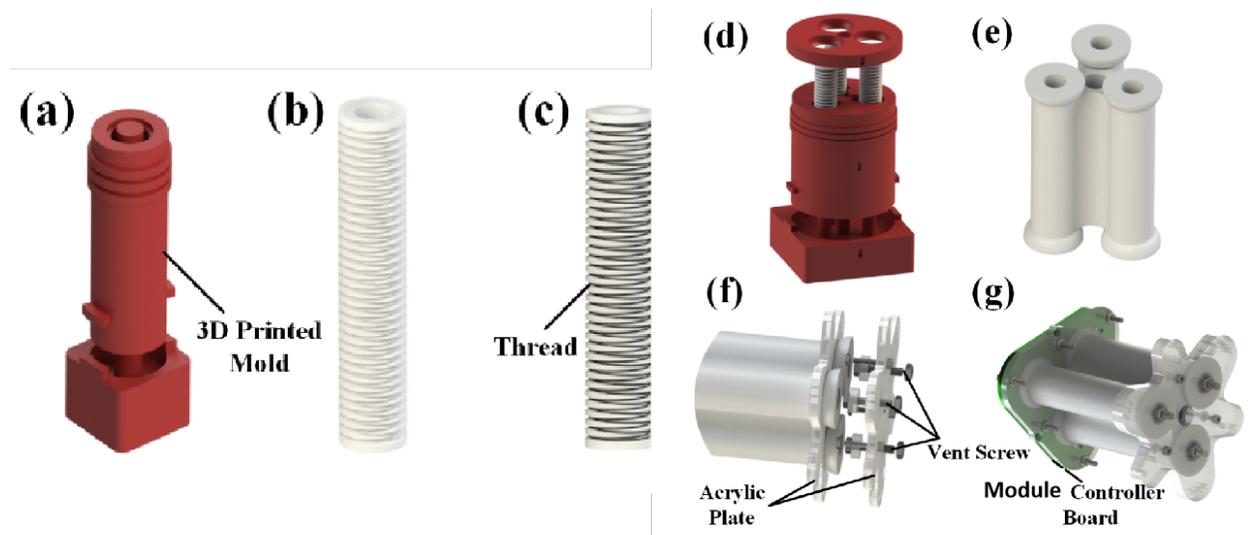


Figure 2.3: Fabrication process of the 3-D soft robotic bending module. (a) The mold of a single linear-stretched pneumatic. (b) The single actuator without thread. (c) The threaded single actuator. (d) The mold of the outer body, which is used to combine the three threaded single actuators. (e) The demolded soft 3-D bending segment. (f) The soft segment sealed with acrylic plates and the vent screw. (g) The assembled soft robotic bending module.

2.1.4 Wave Spring Sheath Constraint

Under unexpected external forces, silicone rubber modules will buckle even if silicone with relatively high elastic modulus is used, which limits the performance of soft robots. As shown in Figure 2.4 (left), the pressurized chamber buckles, elongating without bending the entire module. This limits the ability of the soft robotic snake to traverse obstacles and perform complex gaits or motions in 3-D. For example, when performing step climbing locomotion, soft robotic snake modules buckle under the influence of gravity. As a result, locomotion can not be performed as desired, reducing the maximum height the soft robotic snake can traverse.

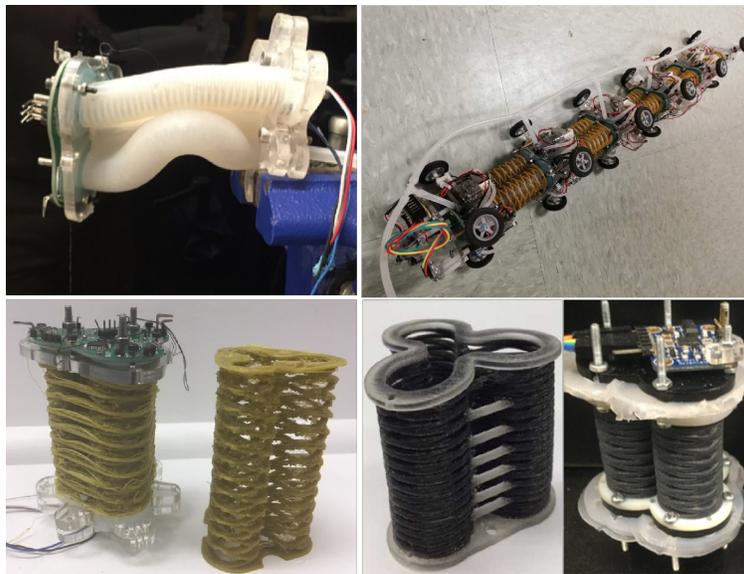


Figure 2.4: Top-left: An example of a soft robotic module buckling in a cantilever configuration. The lower actuation chamber is pressurized while a downwards force is applied to the tip; Bottom-left: A fully integrated module and the wave spring sheath 3-D printed out of NinjaFlex; Top-right: Snake robot made of five modules with NinjaFlex sheaths mounted; Bottom-right: Wave spring sheath with and without horizontal support 3-D printed out of Objet FLX9085.

We address this problem by encasing the snake modules in 3D-printed wave springs based on our previous research (Skorina and Onal (2020)). Wave springs consist of a stacked series of flexible waves and are traditionally used in applications where a traditional coil spring may be too large. We utilize this type of flexible wave in order to have a sheath for the snake module that can selectively extend, allowing the module to bend, while also resisting buckling. Flexible 3D

printing allows us to modify this structure to conform to the contours of the bending modules, and we add horizontal supports to further resist buckling.

We made two different designs of sheaths shown in Figure 2.4, with two different materials, Objet FLX9085 and NinjaFlex. We observed that the Objet FLX9085 wave springs were expensive and prone to breakage while NinjaFlex wave springs were more reliable and more elastic.

2.1.5 System Architecture

The 3D soft robotic snake is a modular robot composed of at least three soft robotic modules mounted in series. The modules are connected with screws and spacers, as well as wires, tubes and passive wheels located in short gaps between adjacent modules. We mount the main controller, an Arduino Pro Mini with Atmega328 chip, at the head of the soft robotic snake. Two 2-cell lithium polymer batteries (the voltage is 7.4 V) are mounted at the head and the tail respectively, to power the control circuit and the valves and maintain balance. In addition, we add a Bluetooth module on the main controller to enable remote commands to switch between locomotion modes.

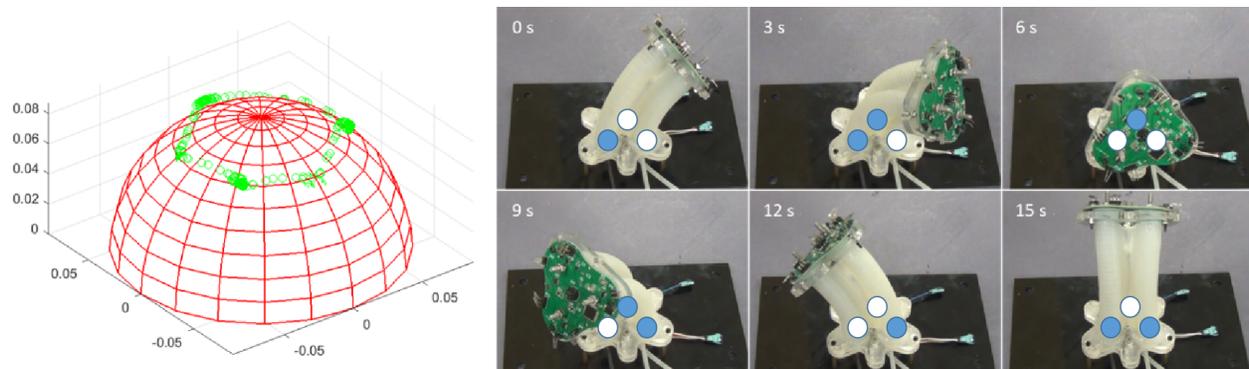


Figure 2.5: Left: Green data points represent the trajectory of the central point at the top end-plate (tip) of the module when performing the hexagonal gait. The trajectory, which also can be considered as the workspace, is a deformed hexagon on a (red) spherical. Right: A soft robotic module in testing. The blue and white dots show the status of the three actuation chambers. Blue dots mean inflated states and white dots mean deflated states.

2.2 Robot Locomotion

2.2.1 Motion Capabilities of Soft Robotic Modules

We first performed experiments to evaluate the 3-D soft robotic modules for their ability to generate the desired motions and to withstand reasonable external loads for realistic locomotion gaits of a 3-D soft robotic snake.

Inflation and deflation

Thirty sets of experiments are conducted to test the inflation and deflation performance of the modules in bending, which provides us with dynamic response information relevant for the bending waves used in the locomotion gaits. During the experiments, the soft robotic module is inflated to bend in a step response, then deflated to recover after a certain amount of time, following a square wave pattern. Figure 2.2 shows the mean inflation and deflation curve with the shaded region representing standard deviation between experiments. Based on these experiments, we quantify the inflation and deflation time constants as between 0.23-0.44 s. In order to fully inflate and deflate the chambers as possible, for each chamber it would take 0.46-0.88 s. Thus, we confine the gait frequency in the locomotion experiments to be no higher than 2 Hz, so that there is a reasonable chance for the actuators to be fully actuated.

Hexagonal Gait

In order to achieve sidewinding locomotion, the soft robotics modules should bend such that their end-plates (tips) move in circular paths with a desired phase delay between adjacent modules. For ease of implementation and to increase computational efficiency, we approximate this ideal circular gait and develop a hexagonal gait which can be simply implemented by binary inflation-deflation for each actuation chamber without controlling the pressure which would be required to achieve precise circular tip trajectories. Figure 2.5 shows that the tip trajectory of the modules performing the hexagonal gait is tracing a deformed hexagon projected on the spherical workspace of the module.

Wave Spring Sheath

The goal of adding the custom flexible wave spring sheaths to the soft robotic modules is to increase the resistance to buckling under external force. We set up experiments to test the wave spring sheaths. A bending module with different kinds of custom wave spring sheaths mounted is set in a cantilever configuration with a downward tip load as shown in Figure 2.4 (Left). We gradually increase the pressure in the lower actuation chamber, and measure the maximum module height and bending angle it can reach before buckling. As shown in Figure 2.6, the sheaths have significant effect on reducing buckling when pressurized actuators are under external force. In addition, the Objet sheath with horizontal supports is generally the best performer. However, under higher loads, it performs similar to the NinjaFlex sheaths. Given the advantage in cost and reliability of the NinjaFlex sheaths, we select the NinjaFlex Sheath as the best choice.

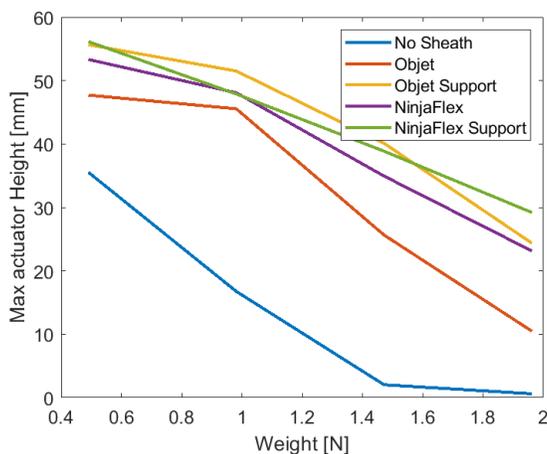


Figure 2.6: The max tip height achieved by the bending module in cantilever configuration under tip loading. With reduced buckling effect, the actuators bends higher.

2.2.2 Locomotion Capabilities of the Modular 3-D Soft Robotic Snake

In this section, we connect four bending modules in series to create a robotic snake and test undulatory and sidewinding locomotion on it. Also, we test custom step climbing locomotion gaits on an enhanced version of robotic snake with an additional module and wave-spring sheath constraint structure for improved ability to resist gravitational forces during climbing. The robotic snake use

low-level valve duty cycle commands that vary along the length of the robot according to desired functions or algorithmic sequences that generate the specific gaits. For undulatory and sidewinding locomotion, we use four modules in the soft robotic snake prototype to reduce overall length. For step climbing locomotion, we use five modules to increase the propulsion and balance of the body during lift off. The experiments are conducted with varying settings to study their effect in locomotion performance. We use a motion capture system to record the trajectory of the soft robotic snake. In parallel, we conduct the simulation studies under the same settings and compare the simulation results with the experimental results in the real world.

Undulatory Locomotion

Undulatory locomotion is a well known locomotion gait of biological snakes, which is also easily conducted by many robotic snakes. With a bidirectional bending motion in the horizontal plane, the snake thrusts its body from side to side, creating a travelling wave, and propels itself with the effect of anisotropic friction provided by snake scales. On our soft robotic snake, the anisotropic friction is provided by the passive wheels mounted on the robot and the motion pattern is identical to the one observed in biological snakes. To perform undulatory locomotion, all the modules undergo bidirectional bending with a certain phase delay between adjacent modules creating a traveling wave of curvature. To generate this motion, we use Equation 2.1 to control the robotic snake (K_{ij} describes the status of each channel in segments. ω and β refer to the frequency and phase delay in the gaits. Detailed description of the parameters are shown in Table 2.2).

$$K_{ij} = \text{sgn}(\sin(2\pi\omega t + i\beta + j\pi) + \phi), \quad (2.1)$$

$$i \in (0, 1, 2 \dots N - 1), j \in (0, 1).$$

During this locomotion, we do not actuate the top chamber in each segment. Geometrically, if we would like to make sure the segment bending on the same surface, it is needed to “half actuate” the top chamber. While in the real world experiment, the bending on upward direction will be largely reduced by gravity. As the result, the lateral undulation will stay on the same plane.

We conduct experiments to demonstrate undulatory locomotion on a paper surface with frequency (ω) of 1.5 Hz, 1.75 Hz, and 2 Hz. The phase delay is set to be $2\pi/3$, which results in 1.25 traveling curvature waves along the body. For comparison, the same tests are conducted in the real-time simulation for our soft robotic snake. In Figure 2.7, the trajectory of the soft robotic snake CoM (central of mass) are presented. The blue lines represent the result from simulation, while the red lines represent the result from real world experiments. In real world experiment, the robotic snake robot can reach velocity of 140.25 mm/s (0.275 body length/s) under 2 Hz.

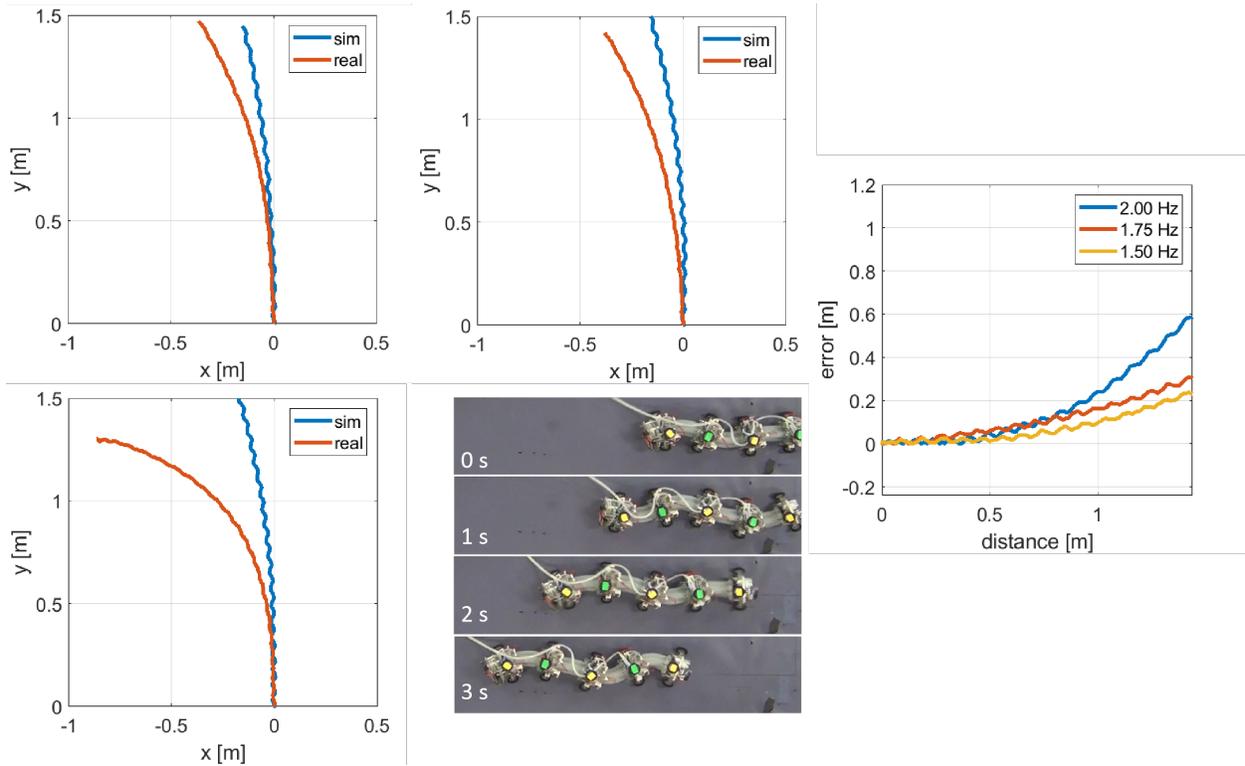


Figure 2.7: Top-Left: Trajectory of the soft robotic snake CoM (central of mass) when locomotion frequency is 1.50 Hz. Top-Middle: Trajectory of the soft robotic snake CoM when locomotion frequency is 1.75 Hz. Bottom-Left: Trajectory of the soft robotic snake CoM when locomotion frequency is 2.00 Hz. Bottom-Middle: Soft robotic snake performing lateral undulation locomotion from right side to left side in real world. Right: Error between simulation result and real world experiment result with relate to distance traveled.

The robotic snake is theoretically supposed to move in a straight line with constant input and parameter setting. Because of the fabrication inaccuracy, uneven paper surface, and unexpected

sliding friction, the robotic snake can not move in a straight line. The simulation results present a similar tendency and velocity with the real world experiments, while the trajectories in simulation results are closer to a straight line since there are no fabrication inaccuracies and the surface can be set to be perfectly even. Deviation in simulation is caused by noise added in contact calculation, and the model for hyper-elastic material is simplified for calculation efficiency, which result in error under large deformation, thus, the simulation shows some deviation. Higher frequency have little effect on simulation results, while in real world experiments, higher frequency causes more unexpected sliding during undulatory locomotion, thus, the deviation increases. Our previous work presents a method to fulfill trajectory following for soft robotic snakes with an iterative learning controller (Luo et al. (2020)). While in this work, in order to test the performance of the soft robotic snake and compare with the simulation, we do not add any controller during the experiments.

Experiments of modular snake robot with different number of module (from 3 modules to 5 modules) on different surfaces, including paper, rubber, carpet, and grass, are also conducted as a supplementary research on the property of our modular soft snake robot and the real-time simulation developed, and the results are described in Appendix section 5.1. On the grass surface we tested, sliding friction and rolling friction factors were similar to each other, as shown in Table 5.1. This disrupts the ability of the undulatory locomotion to function, and results in strange snake behavior. We observed that the snake on grass would move backwards, as shown in Figure 2.8 (a), (b). To verify that this was the expected result, we used the friction factors from Table 5.1 in a mathematical model of undulatory locomotion we had previously developed and simulate the movement, discussed in more detail in (Luo et al. (2014)). The results can be found in Figure 2.8 (c), where on grass the snake moved much slower and in the opposite direction.

Sidewinding Locomotion

Sidewinding is a variation of the serpentine motion that makes use of all 3 degrees-of-freedom in each segment. It physically lifts parts of its body off the ground, giving the body an S-shape with two points of contact with the ground to push the snake in a diagonal direction. Sidewinding locomotion utilizes static friction force instead of sliding on the surface, thus, it may have the

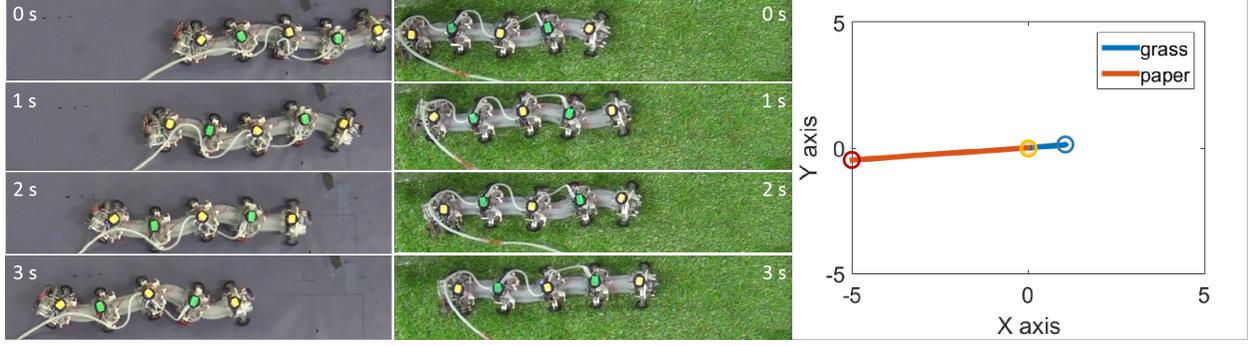


Figure 2.8: (a) Lateral undulatory locomotion on paper; (b) Lateral undulatory locomotion on grass; (c) Mathematical Model calculated result of overall trajectory of central point under lateral undulatory locomotion with same parameter on different surfaces. Yellow circle shows the start point. Red and blue circle shows the end point of locomotion on paper and grass respectively

Table 2.2: Parameters of gait functions.

Symbol	Description
i	sequence of the segment(0 represents the 1 st segment)
j	sequence of the channel(0: bottom left; 1: bottom right; 2: top)
K_{ij}	status of j^{th} channel in i^{th} segment(-1: deflation; 0: stay still; 1: inflation)
ω	frequency of the undulation gait
t	total running time
β	phase difference between adjacent segments
ϕ	offset of the travelling wave
N	number of segments

advantage of higher energy efficiency, but spends energy against gravity to lift the body up. As discussed in Section 2.2.1, To realize this locomotion gait in the soft robotic snake, we developed a hexagonal gait to approximate the circular bending motion observed in biological snakes. In order to conduct the locomotion, we use Equation 2.2 to control the soft robotic snake (K_{ij} describes the status of each channel in segments. ω and β refer to the frequency and phase delay in the gaits. Detailed description of the parameters are shown in Table 2.2).

$$K_{ij} = \text{sgn}\left(\sin\left(2\pi\omega t + i\beta + j\frac{2\pi}{3}\right) + \phi\right), \quad (2.2)$$

$$i \in (0, 1, 2 \dots N - 1), j \in (0, 1, 2).$$

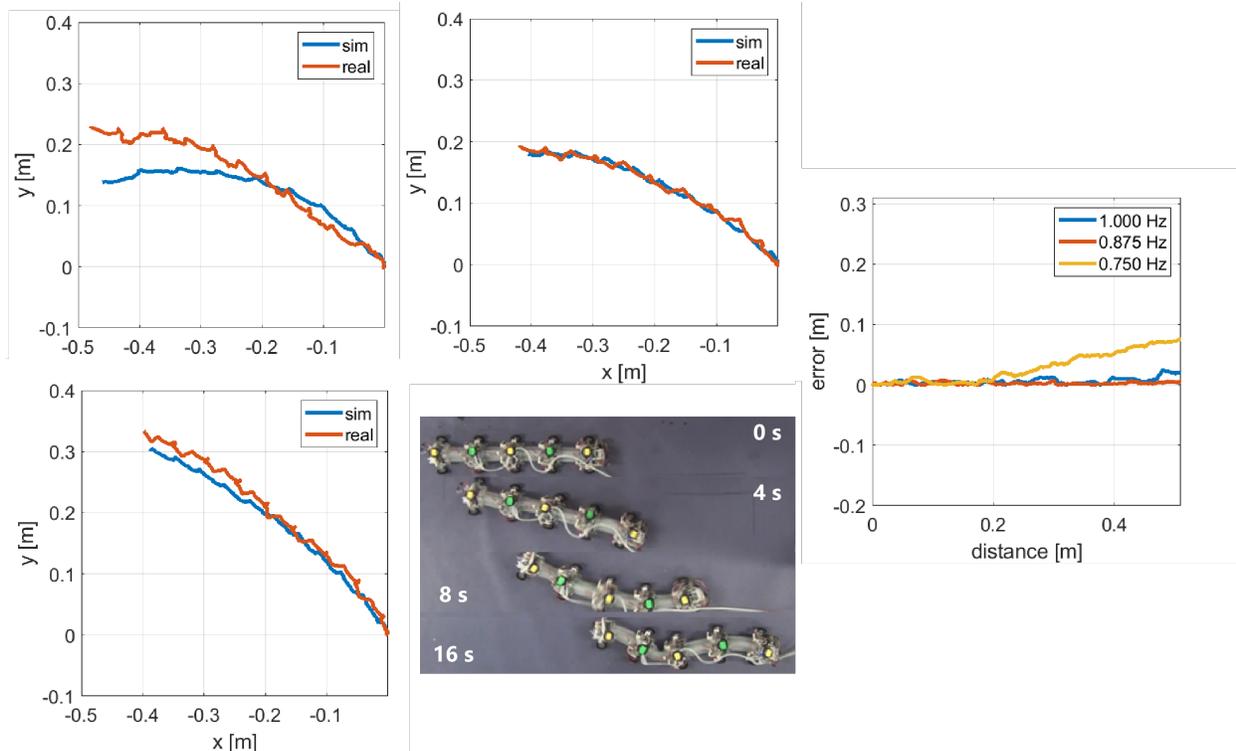


Figure 2.9: Top-Left: Trajectory of the soft robotic snake CoM (central of mass) when locomotion frequency is 0.75 Hz. Top-Middle: Trajectory of the soft robotic snake CoM when locomotion frequency is 0.875 Hz. Bottom-Left: Trajectory of the soft robotic snake CoM when locomotion frequency is 1.00 Hz. Bottom-Middle: Soft robotic snake performing sidewinding locomotion on a paper surface. Right: Error between simulation result and real world experiment result with relate to distance traveled.

We conduct experiments to demonstrate sidewinding locomotion on a paper surface with frequency (ω) of 0.75 Hz, 0.875 Hz, and 1 Hz. The phase delay (β) is set to be $\pi/2$, which results in one full traveling curvature wave along the body. For comparison, the same tests are conducted in the real-time simulation for our soft robotic snake. In Figure 2.9, the trajectory of the soft robotic snake CoM (central of mass) are presented. The blue curves represent the simulation results, while the red lines represent the results from real world experiments. In real world experiment, the robotic snake robot can reach velocity of 76.50 mm/s (0.150 body length/s) under 0.875 Hz.

Compared to lateral undulation locomotion, sidewinding locomotion has better fit between simulation and experimental result. When performing sidewinding, the segments on robotic snake

was applied with extra force from gravity and ground compared to lateral undulatory locomotion, result in lower deformation. Meanwhile, the model for hyperelastic material is set to be low order Mooney-Rivlin inside the simulator as a trade off between efficiency and accuracy. As a result, the simulator has a better fitting of lower hyperelastic material deformation which happened in sidewinding locomotion and the step climbing locomotion.

Since sidewinding locomotion is a 3 dimensional, contact based locomotion, there is seldom unexpected sliding compared to undulatory locomotion, also, the effects of surface variations are reduced. As a result, our simulation results are more similar to our real-world experimental results.

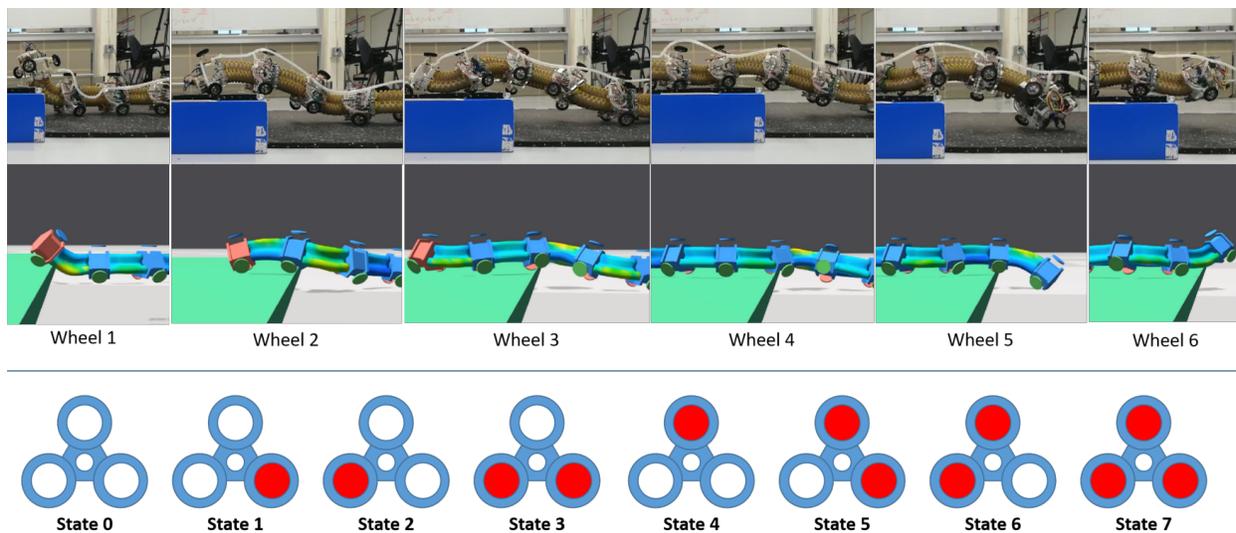


Figure 2.10: Step climbing in real world experiment (top) and in real-time simulation (center). Bottom: Actuation States of the proposed soft robotic modules. Each subplot displays the cross-section of the soft module with three pneumatic chambers around a central slot for wiring. Red chambers are pressurized, and white chambers are depressurized.

Step Climbing Locomotion

To explore the proposed 3-D soft robotic snake’s ability to operate on non-planar environments, we developed a custom locomotion sequence based on the climbing motion of real snakes, which allows our robot to climb up a step. At least three modules are needed on the ground for the robot to perform snake-like lateral undulation locomotion and power the robot to move forward. The step climbing motion will result in intermediate states with several modules that can not touch the

Table 2.3: State of each segment to lift each wheel in step climbing locomotion. “/” means switching, for example, “2/1” means switching between state 2 and state 1 under the frequency of ω . The states are explained in Figure 2.10.

	segment1	segment2	segment3	segment4	segment5
wheel1	3	2/1	2/1	2/1	2/1
wheel2	4	4	2/1	2/1	4
wheel3	2/1	4	3	2/1	2/1
wheel4	2/1	2/1	4	3	2/1
wheel5	2/1	2/1	2/1	4	4
wheel6	5/6	2/1	2/1	2/1	3

ground when climbing up high steps due to the restriction of the module length. Thus, in order to gain higher thrust and better balance we added one more module to the robot, and created this gait for a 5-module version of the robotic snake, without loss of generality. As a result, this version offers greater balance for some of the modules to be lifted off the ground.

The snake modules are strong enough for lateral undulation and sidewinding locomotion. However, when lifting themselves off the ground, the external force applied on each segment would easily result in buckling, thus, the original soft robotic snake in our previous work fails to climb up. This task exemplifies the need for engineering directional stiffness for soft robots, enabling a level of rigidity in some directions and compliance in others. We achieve this anisotropic stiffness response with a flexible 3-D printed sheath as discussed in Section 3.1.3, which significantly improves the performance in step climbing, as determined previously through individual module experiments.

In order to climb on a step, we need a specific locomotion gait developed as we have done for undulatory and sidewinding locomotion. Instead of trial and error in real-world experiments, we tested and adjusted possible gaits in our real-time simulation. By creating extra constraint with corresponding parameters in the simulator, the snake robot in simulation has similar anti-buckling performance with the soft robotic snake mounted with 3D printed sheath (as shown in Figure 2.10) in real world. With the help of our simulator, a group of gaits are developed to make sure our snake

robot can climb on a 7 cm step, which we set as a challenge since it is as high as the snake itself. The step climbing locomotion gait is described in Table 2.3 and Figure 2.10.

In the process of climbing up a step, the robot first moves forward with its head (first module) raised up (bending upwards). After the passive wheels of the first module land steadily on the step, the first module bends downward to get a foothold force and raises the second module while the rest of the body follows lateral undulation to continue pushing forward. After the passive wheels of the second segment land steadily on the step, the head module resumes its regular lateral undulatory serpentine locomotion to move forward, while the second module bends downward in order to avoid slipping backward. Meanwhile, the third module is lifted up to repeat the same procedure as the second module. After landing on the ground of the upper step steadily, the third module can return to its normal serpentine locomotion and lift the tail part of the robot. With the rest of the modules pulling it forward with lateral undulation, the tail can land on the upper step easily. This gait essentially generates a simple vertical wave that travels once from head to tail on top of the generic lateral undulation gait which continues pushing the body forward, revealing an interesting way to fuse gaits for 3D motion enabled by our 3D soft robotic modules.

2.3 Real Time Simulation

For rigid mobile robots, real-time simulations are easily performed using existing simulator tools, while for soft mobile robots with deformable structures, real-time simulation is an active research area because of the limitations in current simulation methods. Huang et al. (Huang et al. (2020)) developed a discrete differential geometry based numerical simulation method for mobile soft robots with limbs. The computational efficiency of this method enables it to fulfill real-time simulation for a star-shaped rolling robot. Duriez (Duriez (2013)) presented a real-time finite element method (FEM) based simulation and used it as a part of the controller to control a 3-D silicone rubber soft robot inspired by a parallel motion platform. In recent work, we explored the potential of NVidia Flex as a real-time simulator for soft snake robots in 2-D only (Gasoto et al. (2019)). This system provides a powerful tool to seamlessly simulate hyper-elastic structures and rigid bod-

ies. Building on our experience, we present a real-time simulator prototype in this article, based on NVidia Flex to simulate our 3-D soft robotic snake, which is a much more complicated soft robotic system. We use this method to build a real-time simulation and help us quickly develop and test new locomotion gait patterns for our soft robotic snake. As a demonstration, we present experimental results that lets the robotic snake to go over a 7 cm step using a a custom locomotion gait first generated in simulation and directly utilized in the real world without modification.

2.3.1 Nvidia Flex based simulation

Wavespring sheath reduced the uncertainty of the system while the unexpected deformation and sliding still exist which makes simplified model less accurate for simulation or controller design.

FEM method is often used on complex deformation situation, while most of the FEM softwares are too slow for such a complex system like our pneumatic soft snake robot. We turn to utilize a particle based engine called Nvidia Flex to build a real time simulation for our snake robot. NVidia Flex is a particle-based simulation technique developed by NVidia Gameworks. It split the soft matters into particles connected with spring-like constraints, while the particles and rigid bodies are calculated via Lagrangian multi-physics mechanics. Figure 2.11

The 3D robotic snake is simulated using a particles and rigid bodies system derived from Lagrangian multi-physics mechanics, given in the general form as Equation 2.3 that includes external and gyroscopic forces.

$$\begin{aligned}
M\ddot{q} - f(q, \dot{q}) - J_b^T \lambda_b - J_n^T \lambda_n - J_f^T \lambda_f &= 0 \\
c_b(q, p) + E\lambda_b &= 0 \\
0 \leq c_n(q) \perp \lambda_n &\geq 0 \\
\forall i \in A, D_i^T \dot{q} + \frac{|D_i^T \dot{q}|}{\lambda_{f,i}} \lambda_{f,i} &\geq 0 \\
\forall i \in \mathcal{J}, \lambda_{f,i} &= 0
\end{aligned} \tag{2.3}$$

Interactions between particles are defined as a set of bi-lateral and unilateral constraints (c_b and c_n , respectively), and are associated with their respective Lagrange multipliers λ_b and λ_n . With the use of a compliance (or inverse stiffness) term defined in the block-diagonal matrix $E \in R^{n_b \times n_b}$,

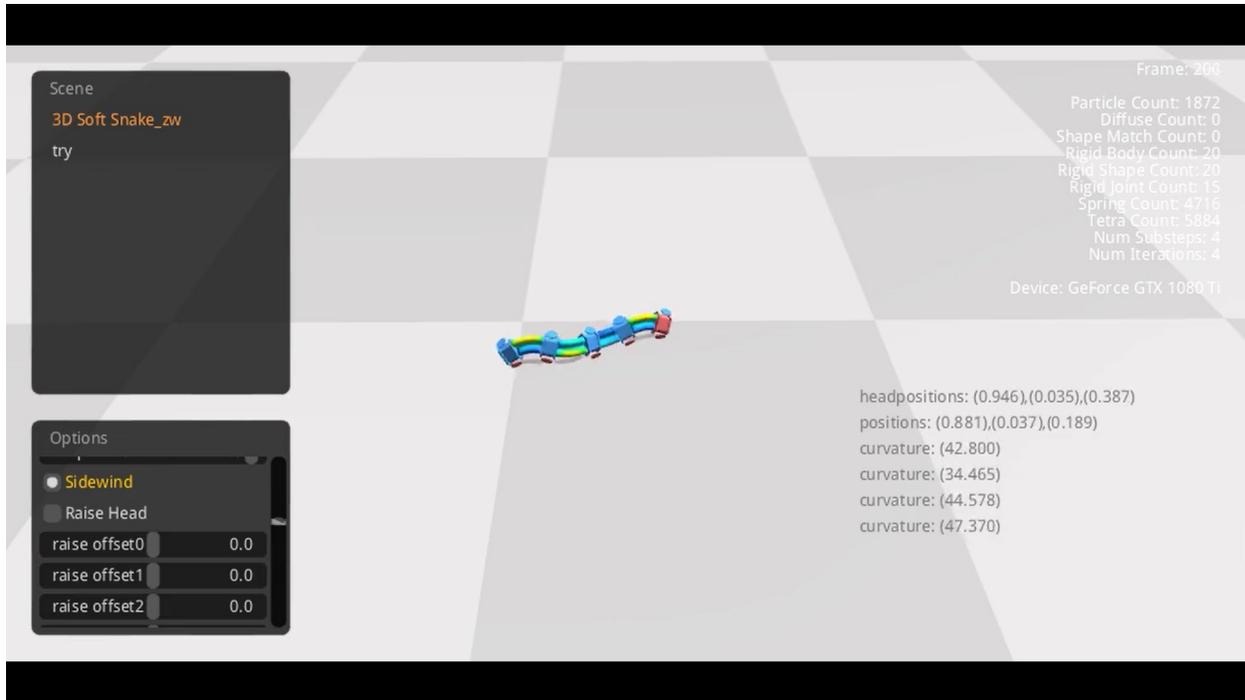


Figure 2.11: A snapshot of the graphical user interface (GUI) of Nvidia Flex during a sidewinding locomotion simulation. Parameters can be either attached in the slider tabs for quick adjustment, or simply listed on the screen for data observation.

bilateral constraints can be used to store elastic energy potentials, as described in Backman et al. (Backman et al. (2012)). The target pressures for the link modules are included in the vector p , which are parameters for a select set of bilateral constraints to simulate the resulting expansion of the links. The contact and frictional forces are based on Coulomb's model, which defines an admissible cone of contact forces (Stewart and Trinkle (1996)), and defined by the unilateral contact constraints, and μ_i the friction coefficient for the i^{th} contact. The frictional forces for a contact are parameterized by $\lambda_{f,i}$, with a corresponding basis D_i defining the surface tangent plane at the contact point. Active contacts are defined in the set $A = \{i \in (1, \dots, n_c) | \mu_i \lambda_{n,i} > 0\}$ with inactive contacts \mathcal{J} being its complement. The constraint Jacobians J_b, J_n contain the gradient of bilateral and normal constraint functions with respect to q , and we define the set of frictional basis vectors as the matrix $J_f = [D_1, \dots, D_{n_c}]^T$.

Tetrahedral finite-elements are used to model the solid chamber material. Assuming a constant strain element and a linear isotropic constitutive model, each tetrahedron defines a 6-dimensional

bidirectional constraint vector,

$$c_{tetra}q + E_{tetra}\lambda = 0 \quad (2.4)$$

where $c_{tetra}(q) = [\epsilon_{xx}, \epsilon_{yy}, \epsilon_{zz}, \epsilon_{yz}, \epsilon_{xz}, \epsilon_{xy}]$ is the vector of corotational strains in Voigt notation, and E_{tetra} is the constant element compliance matrix, given by:

$$E_{tetra} = \frac{1}{V_e Y} \begin{bmatrix} 1 & -v & -v & 0 & 0 & 0 \\ -v & 1 & -v & 0 & 0 & 0 \\ -v & -v & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1+v & 0 & 0 \\ 0 & 0 & 0 & 0 & 1+v & 0 \\ 0 & 0 & 0 & 0 & 0 & 1+v \end{bmatrix} \quad (2.5)$$

Where V_e is the element volume, Y and v are the material Young's modulus and Poisson's ratio, respectively.

2.3.2 Pressure Dynamics

In order to simulate the pressure on the chambers, since each chamber is constrained by a helical thread wound around its axis (Skorina et al. (2018); Qin et al. (2018)), the only direction available for expansion is along the axis. Therefore, the bilateral constraints on the axis direction for particles that make up the cavity receive an actuation parameter based on the pressure applied. Since the elastic model used is linear with stress, the expansion ratio with the pressure is given with respect to the rest pose by the factor $\epsilon(p)$ as: $\epsilon(p) = 1 + \frac{p}{Y}$. Given the limited airflow provided by the valves, the pressure delivery must obey the following relationship given by:

$$v^2 = \frac{2}{\rho}(p - p_s) \quad (2.6)$$

Where ρ is the air density p is the target pressure in the chamber, p_s is the source pressure, and v is the air flow to the chamber. This results in the following discrete time update equation:

$$\Delta p_i = \frac{p_i(t+h) - p_i(t)}{p_s} \quad (2.7)$$

Where t is the current time, p_i is the pressure in chamber at current simulation step, and h is a fixed time step. Each time-step performs 4 iterations of Newton's method, with each linear system solved approximately using 20 Preconditioned Conjugate Residual (PCR) iterations to ensure a fixed computational cost. The chamber deflates in an isobaric curve, until its volume ceases to reduce, then slowly releasing the remaining air until stabilizing with atmospheric pressure. Therefore the deflation ratio is limited to a threshold T_p , resulting in the following differential equations for pressure update:

$$p_i(t+h) = \begin{cases} p_i(t) + p_s \Delta p_i^2 k_i, & \text{inflating;} \\ p_i(t) - \min(p_i(t) k_d, T_p), & \text{deflating.} \end{cases} \quad (2.8)$$

Where k_i, k_d are the inflation and deflation damping parameters respectively, tuned according to experimental data for the link.

2.3.3 Buckling effect and wave spring replication

Just like in the real soft modules, the effect of adding pressure on the simulated modules causes buckling of the expanded chamber, due to forces pushing against it. In order to avoid buckling to reproduce the effect of the proposed wave-spring sheath, a set of constraints were added to replicate the behavior of the sleeve. Stiff bidirectional constraints were added between particles that make up the exterior wall of each chamber, such that it won't let one chamber to buckle against another.

2.4 Discussion and Conclusion

In this chapter, a 3D soft robotic snake and accompanying 3D-printed wave spring sheath are designed, fabricated, and validated. Experiments were performed to test the performance of a single soft robotic module, as well as the locomotion of the snake robot under undulation and sidewinding gaits. A simulation platform based on NVidia Flex was developed and verified using experiments on the physical snake. Developing a step climbing gait, we were able to command the 3D soft robotic snake to traverse a 7 cm high step, demonstrating its ability to function in non-planar environments for the first time, while showing the ability of the NVidia Flex system to seamlessly simulate robotic systems with hyperelastic structures and rigid bodies in real time. The 3-D printed wave spring can help with the buckling behavior, but cannot fully eliminate it, which still restricts the ideal climbing capability, we will extend our current design advantages and keep researching better structures to address this issue.

Considering the fact that hyper-elasticity result in unpredictable system because the structure is easy to conduct nonneglectable deformation under external force, a better module for snake robot can help to improve the performance and stability of the soft robotic snake. Besides, an artificial snake skin instead of the passive wheel and mounted on the snake robot can be more reliable to generate anisotropic friction for a much more compact structure. With a much more stable snake robot, more gaits would need be developed and applied on the robot to make it a universal mobile platform which is especially powerful in searching and rescuing.

Our real-time simulation presents a way to generate reliable information quickly, especially for gait development. The soft robotic snake is a complex pneumatic hyperelastic system to simulate. In order to fulfill real-time simulation requirements, there is a trade-off between computational complexity and accuracy of the results. The ability of the NVidia Flex platform to use GPU based computation is very beneficial for our simulations.

Chapter 3

Continuum Salamander Robot

In this section we introduce the design, fabrication and experimental evaluation of our modular continuum mobile salamander robot with origami structure. Figure 3.1 shows a single module salamander robot with active wheels and suspension propulsion wheel system.

Experiments are conducted to test the performance of the robot. Our results show that, the salamander robot can achieve a maximum speed of 226.4 mm/s, climb up to 60.3 deg slope, execute sharp turns with a minimum radius of 79.9 mm (0.54 body length), go straight up along a narrow gap vertical to ground, go through a maze which a 4-wheeled rigid mobile robot with the same length can not achieve, and keep operating even though it turns over or randomly tossed.

Inverse kinematics of the origami actuator are conducted for controller. Kinematics modeling of the single module salamander robot is calculated for motion prediction and path planning.

Motion planning methods for continuous locomotion of salamander robot is discussed, and two feasible motion planning solutions based on lattice A-star and Covariant Hamiltonian Optimization for Motion Planning (CHOMP) are generated and compared. Also an experiment with ground truth feedback from motion capture system is conducted to deploy the motion planning method on our robot in real world.

Furthermore, with more modules connected in series, the assembled robot offers the potential to achieve better stability in climbing ramps or going over obstacles. An experiment of step climbing is conducted with manually tuned climbing locomotion gait.

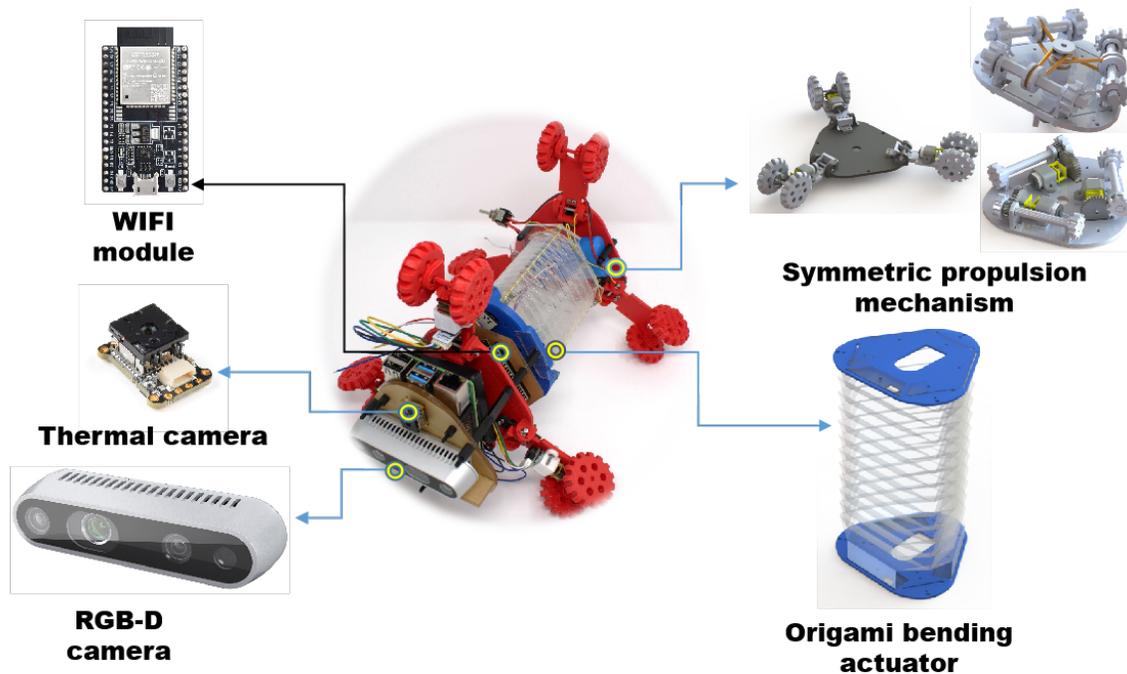


Figure 3.1: (top) Modular designed salamander robot consists of a cable-driven origami actuator, a propulsion mechanism, a power package, and a remote communication module to receive and send messages individually. Extra sensors can also be attached to help mapping optical and thermal environments around if necessary. (bottom) Single module salamander robot conducting Simultaneous Localization and Mapping (SLAM).

This chapter is organized as follows: In design details section, we describe the design of origami module, propulsion system, and the overall system architecture. In robot demonstration section, we demonstrate the velocity, climbing ability, ability of negotiating narrow and unstructured environments, and robustness against unpredictable environments. Also, the traversing ability of multi-module salamander robot is demonstrated by climbing over a 15 cm obstacle. In kinematics modeling, we introduce the inverse kinematics for origami modules and kinematics modeling of a single module salamander robot. In motion planning section, we introduce two feasible motion planner for robots with continuous locomotion pattern. In the end, we conclude this chapter in discussion and conclusion section.

3.1 Design Details

In general, each module of the salamander robot consists of a cable-driven origami actuator, a propulsion mechanism, a power package, a remote communication module to receive and send messages individually, and other sensors for certain requirements. In this section, we mainly introduce the design of origami actuator and the propulsion mechanism.

3.1.1 Origami Module

The continuum salamander robot consists of one or more three-DoF origami modules, based on the origami continuum bending modules similar to the work presented in our previous research (Luo et al. (2018a)). Each module behaves as an independent joint, and when one end is fixed, the other side of the module has an approximately hemispherical workspace. An overall look of the earlier version of our origami module can be find in Figure 3.5, in which the driving DC motors are directly fixed on the PCB board for a more compact design. In later design, we separate the motors and PCB board while fix both of them on a 3D printed case, as shown in Figure 3.6.

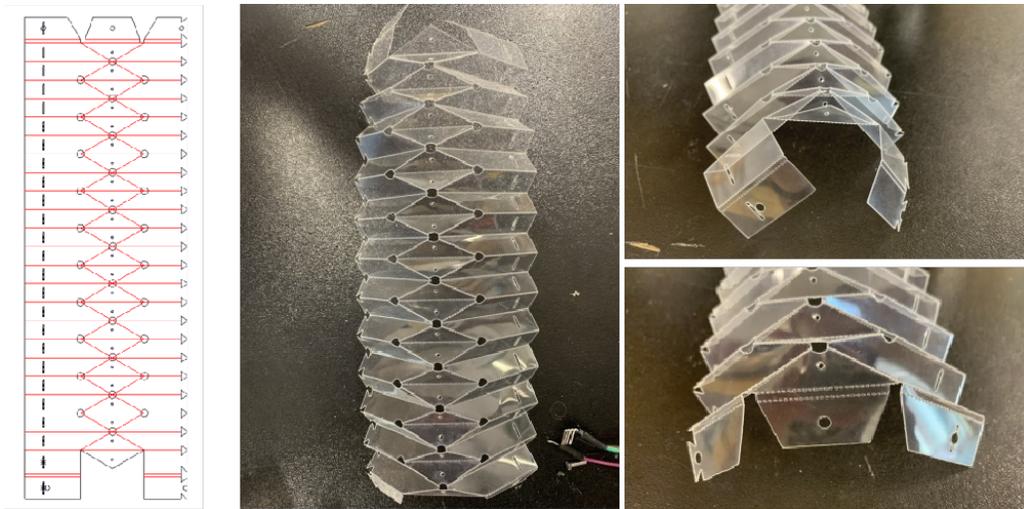


Figure 3.2: (left) Yoshimura pattern plot in CorelDraw software for laser cutting. Folding lines marked red are cut in dashed lines and the Black lines are cut through. (middle) Laser cutted PET sheet folded into Yoshimura patten. (top-right) Top folds. (bottom-right) Bottom folds.

Origami Triangular Bellow Structure

The flexible mechanism of these modules consists of the Yoshimura pattern: a bellows-like origami crease pattern, as shown in Figure 3.2. To establish such a mechanism, three pieces of 7-mil (178- μm) thick folded polyethylene terephthalate (PET) (following our origami fabrication process (Onal et al. (2012); Santoso et al. (2017))) are assembled through tab and slot patterns as shown in Figure 3.3. A sheet of 1/8" thick laser-cut acrylic board and a module control circuit board are connected through the origami body. Three N20 gear DC motors (gear ratio 1:150) are bolted on the module to actuate each module. These motors have 3D printed spools or metallic shaft connector used as spools on their output shafts, which are connected to nylon fishing lines as cables. These cables are attached to the acrylic plate or a 3D printed case on the other end of the module. In addition, the cables are threaded through a series of holes in the origami body, ensuring that the cable always remains perpendicular to the control board and motors and maximizing the bending moment for the module.

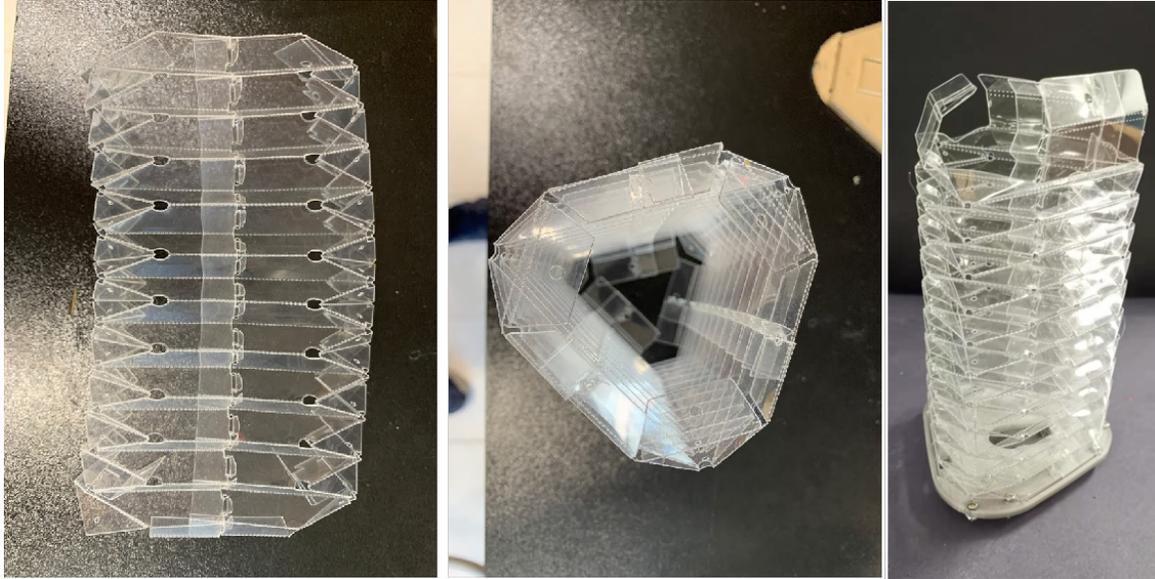


Figure 3.3: (left) Two Yoshimura pattern folded PET sheets combined together through fork-slot mechanism. (middle) Top view of an assembled triangular origami bellow. (bottom-right) Front view of an assembled triangular origami bellow.

Circuit Design

Each origami continuum module consists of a compliant plastic body, an acrylic end plate, and a custom-made PCB, using an 8-bit ATmega32U4 as microprocessor, for embedded control. Further, a total of four electrical wires run through the cavity of the origami structure in a helical shape, providing power and communication between continuum module and other components on the module. The module control boards communicate to a signal receiving WIFI module (ESP32) by using Inter-Integrated Circuit (I^2C) protocol, then communicate to the host WIFI module on desktop computer via ESP-NOW remote communication protocol.

When a motor retracts its cable, while the other two keep still, the differences in cable length will clearly cause the origami module to bend towards the shorter cable. Identically, when all three motors retract their cables, the entire module will contract axially. The motors and cables are mounted in a triangular pattern, giving each module a full three degrees of freedom (DoF) in the form of bending curvature, bending direction, and segment length. Detailed description about the kinematics model of the origami module can be found in Section 3.3. The controller board

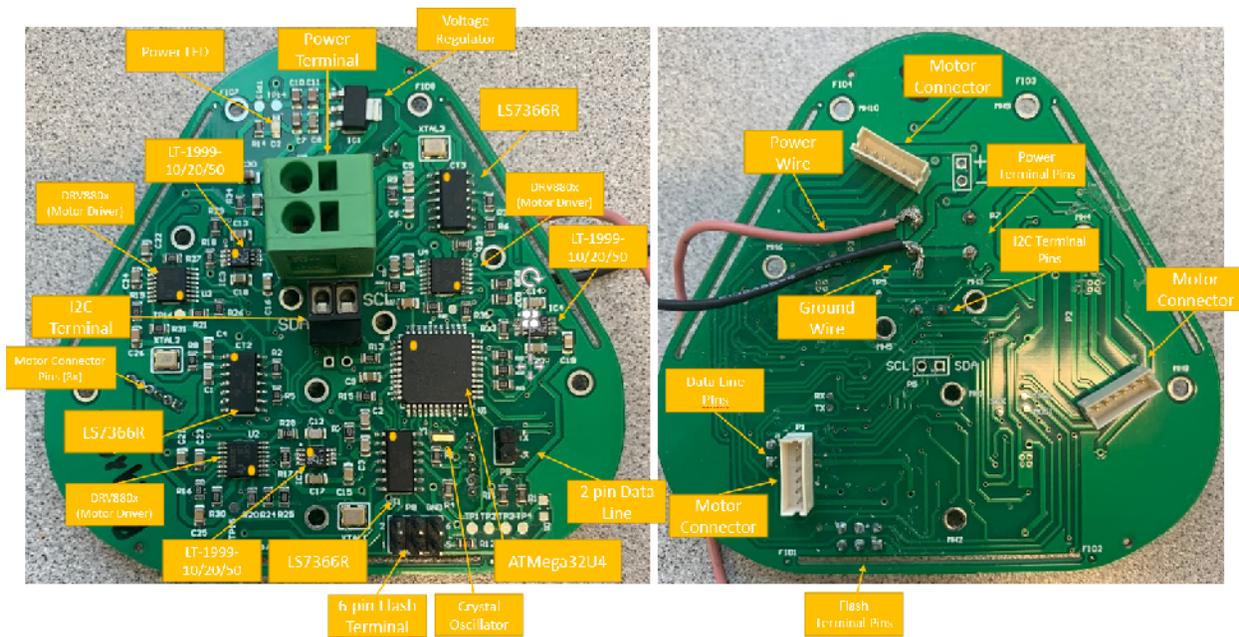


Figure 3.4: (left)Front view of the PCB Chip Diagram (dot locations highlighted in yellow). (right) Bottom view of the PCB, with cable slots for the three DC motors on the module.

contains three channels of motor drivers and decoders, each of which being able to receive encoder data from the motor and implement low-level feedback control on the cable lengths. In addition, three current sensors are deployed in each motor control channel to detect possible overload for the micro-controller to abort the system if necessary. Figure 3.4 shows the front and back view of the custom PCB board.

Salamander robot’s circuit boards have shrunk in size by nearly half compared with their predecessors (shown in Figure 3.4), which makes the entire body smaller. Motor sets are placed triangulated on bottom side of the module. By adjusting the motor to pull or release the cable, the module can bend in different directions and control the movement direction of the module. Specifically, the main controller reads the value of motor encoders, then calculates and outputs the corresponding PWM for the motors according to the codes. This low-level control mechanism has been illustrated in our prior work (Santoso et al. (2017)), either. .

Module Assembly

In the elder version of our salamander robot, the bottom side of the origami triangular bellow is directly fixed on the PCB board which also housing three driving motors for a more compact design as shown in Figure 3.5. While for stability and reliability sake, in our new design the bottom side of the origami triangular bellow is fixed on a 3D printed case on the bottom side of the origami module (shown in Figure 3.6), which is attached directly to the rear wheel array discussed in Section 3.1.2. The 3D-printed case housing three N20 gear DC motors (gear ratio 1:150) and the module control circuit board (as shown in Figure 3.6). This case, which separates the motors and circuit board, was added as an improvement to the design presented in our previous work (Sun et al. (2020)) to simplify the assembly process and prevent the motors from stressing and damaging the circuit board as they did when they were directly attached to the board.

The fully assembled 3-D origami module is shown in Figure 3.5 and Figure 3.6. The maximum length of the module is 148 mm. In our previous work (Santoso et al. (2017)), the axial and torsional stiffness are measured, also, the forward kinematic and inverse kinematic model of our cable driven continuum origami actuator is generated assuming the shape to be a circular arc in 3-D space. Some mechanical properties of the original module are listed here: the total weight is 110 g, integrated with actuation, sensing, and control systems. The origami structure has a 7.311 Nm/rad (0.128 Nm/degree) torsional stiffness while being capable of bending in two directions and changing arc-length down to a fully collapsed state. The module is capable of passively supporting a 1-kg mass at its tip, or four additional serially connected modules, bending approximately 6 degrees.

To actuate the module, each motor controls the length of a nylon fishing line that runs from a 3mm-to-3mm brass connector on the motor's output shaft to the rear acrylic plate. The cables are threaded through a series of holes in the origami body, ensuring that the cable always remains perpendicular to the control board and motors and maximizing the bending moment for the module. When all of the cables are the same length, the origami body is straight and can have a maximum length of 148 mm. If all three motors retract their cables, the entire module will contract axially. However, when a motor retracts its cable while the other two motors remain stationary, the differences in cable length will cause the origami module to bend towards the shorter cable. The

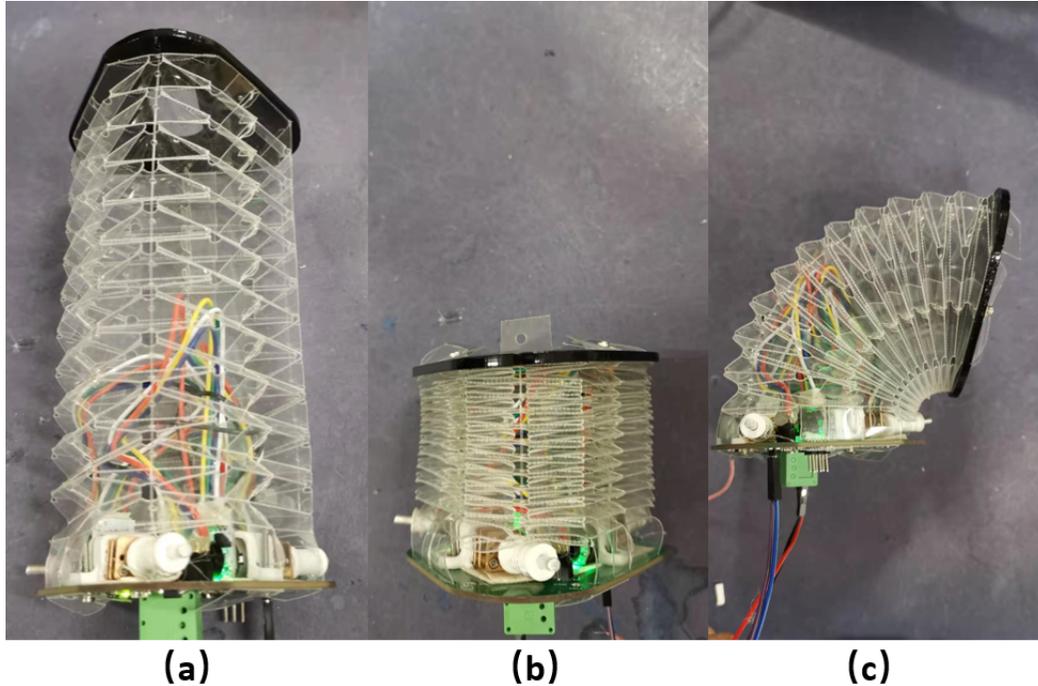


Figure 3.5: (a) Fully stretched origami module, with the length of 148mm, (b) Fully compressed origami module, the length is 57mm, 38% of the maximum length, (c) Maximum bending angle of our origami module is 106 deg.

motors and cables are mounted in a triangular pattern, giving each module a full three degrees of freedom (DoF) in the form of bending curvature, bending direction and length of the central axis, as illustrated in Figure 3.6.

More details of the kinematics of the module can be found in our prior work (Santoso et al. (2017)), which lays out the derivation of the forward and inverse kinematic models of our cable driven continuum origami actuator assuming the shape of the bent actuator to be a circular arc in 3-D space as well as presents the physical properties of the module, such as the axial and torsional stiffness. For reference, the relevant mechanical properties of the original module found in (Santoso et al. (2017)) are listed here: the total weight is 110 g, integrated with actuation, sensing, and control systems. The origami structure has a 7.311 Nm/rad (0.128 Nm/degree) torsional stiffness while being capable of bending in two directions and changing arc-length down to a fully collapsed state. The module is capable of passively supporting a 1-kg mass at its tip, equivalent to four additional serially connected modules, while only bending approximately 6 degrees.

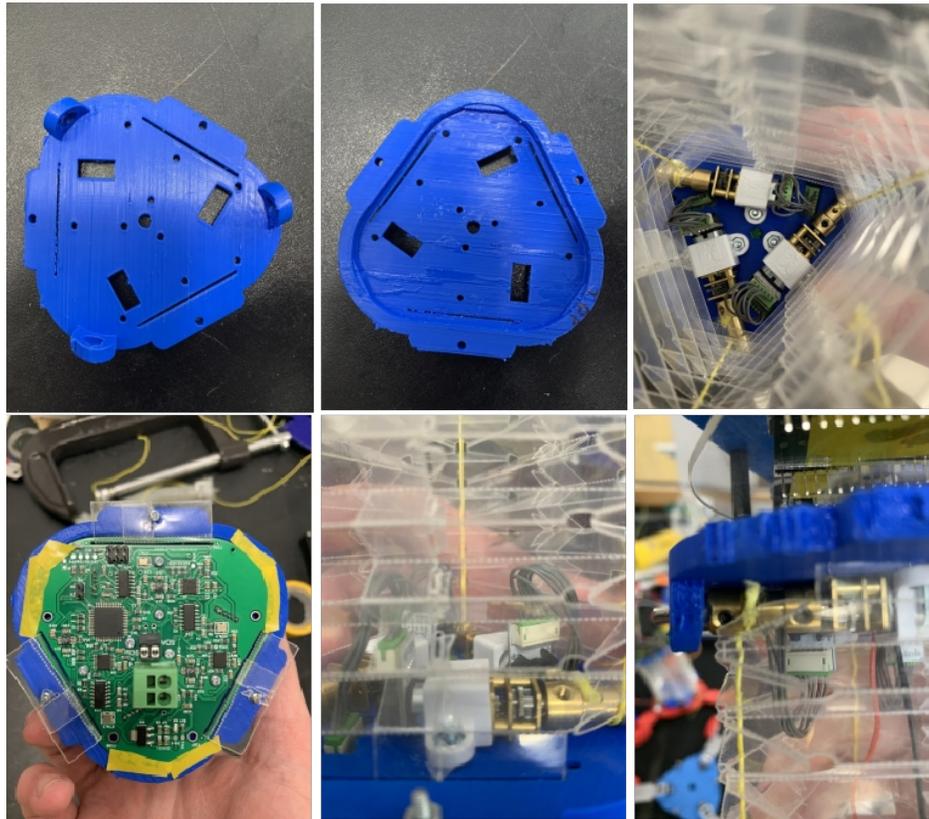


Figure 3.6: (top-left) Top view of the 3D printed case with shaft holder and mounting holes for the motors. (top-middle) Bottom view of the 3D printed case for PCB holding. (top-right) Top view of an assembled origami module with motors. (bottom-left) Bottom view of an assembled origami module with the PCB board. (bottom-middle) Front view of an assembled origami module of the motor cable slot on PCB board through the preset hollow out structure on 3D printed case. (bottom-right) Side view of an assembled origami module of the motor shafts fixed on bearings in shaft holders.

3.1.2 Propulsion Wheel System

Locomotion gaits of traditional robots fail when the robots fall over in unstructured environment. Thus, it is necessary to increase the robustness against the unexpected environments, while still being compact enough to go through small openings. This is the reason that our propulsion mechanisms are designed radially symmetric. In the following sections, we explored several different mechanisms with different properties for different possible situations. .

Belt Transmission

To minimize weight and required electrical power we developed another method to use a single propulsion motor that drives all wheels. To transmit the power of this motor placed vertically in the center of the section, we designed a system of belts transmitting to the wheel axes as shown in Figure 3.7.

This system is composed of three shafts placed on the sides of the triangular section using shaft holders. A cylindrical part with three slots is placed on the top of the motor to convey each belt to its shaft. A pulley is placed on each shaft to hold the belt in place. Slots on the cylindrical part and pulley are designed with tread to increase the friction between belt and slot. Every part of this system is 3D printed with Acrylonitrile Butadiene Styrene (ABS) and meant to be easily replaced. The belts are made of Urethane. The solderable feature allowing us to modify the belt length as we required. All the parts fit together without glue or screws, except for the motor holder. The contact surface of the wheels are covered by silicon, producing more friction with the ground and improving the locomotion.

This method allowed us to drive all the wheels with a single motors, but we found that the urethane belts were unreliable, easily wearing out. In addition, they were prone to slippage, reducing the maximum power that could be applied to the wheels. As a result, this design makes our robot more compact, and light weighted, suitable for extremely narrow environment with less requirement on velocity and thrust force.

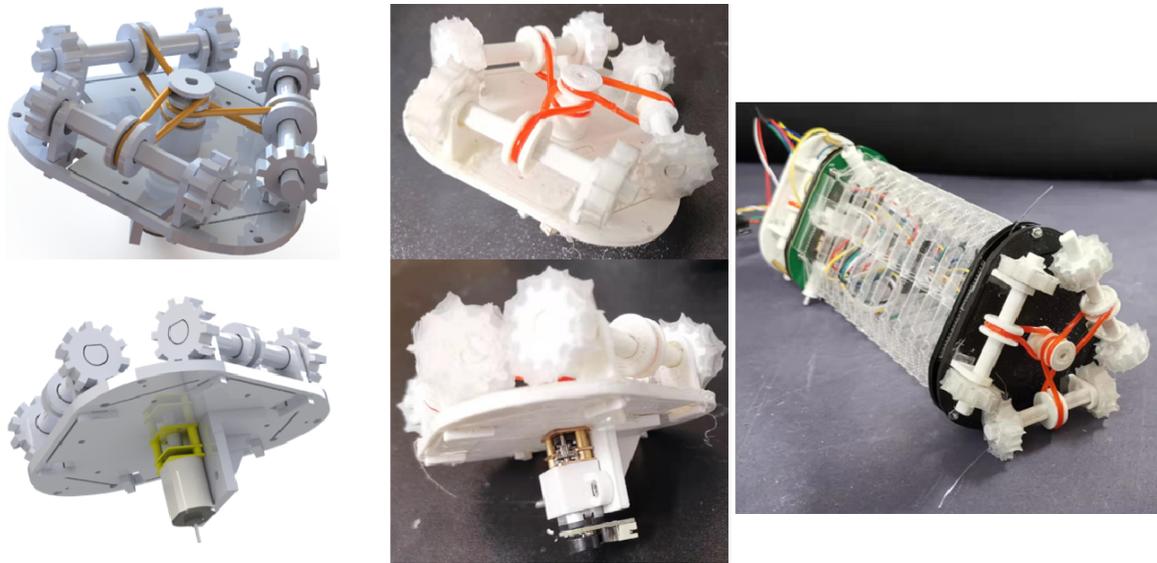


Figure 3.7: (left) CAD rendering of belt transmission design, (middle) Fabricated belt transmission design. (right) Fully assembled salamander robot with belt transmission.

Gear Transmission

Another method we designed used three separate motors, each driving a pair of wheels. As shown in Figure 3.8 in this system, there are three shafts placed on each side of the triangular section, with two wheels attached on each of them. Every shaft is driven by a DC motor parallel to the shaft through a gear transmission with gear ratio of 31:14. This gear ratio was chosen to increase the speed of the wheels but was limited by the need to limit the size of the driving gear. One problem with this method was the extra weight and power consumed by the three motors.

This design is made to be more stable, with much high velocity and thrust force although it is heavier than the belt transmission design.

Active Wheels with Suspension

In the previous two sections, we introduced wheel set designs utilizing either a single motor with belt-transmission or three motors with gear-transmission system to actuate the wheels. In this section, several improvements have been made to the propulsion system.

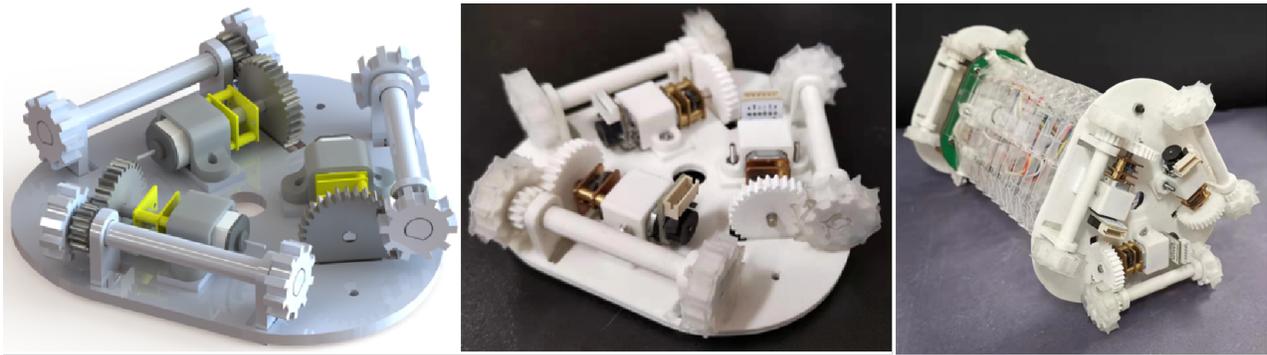


Figure 3.8: (left) CAD rendering of gear transmission design, (middle) Fabricated gear transmission design. (right) Fully assembled salamander robot with gear transmission.

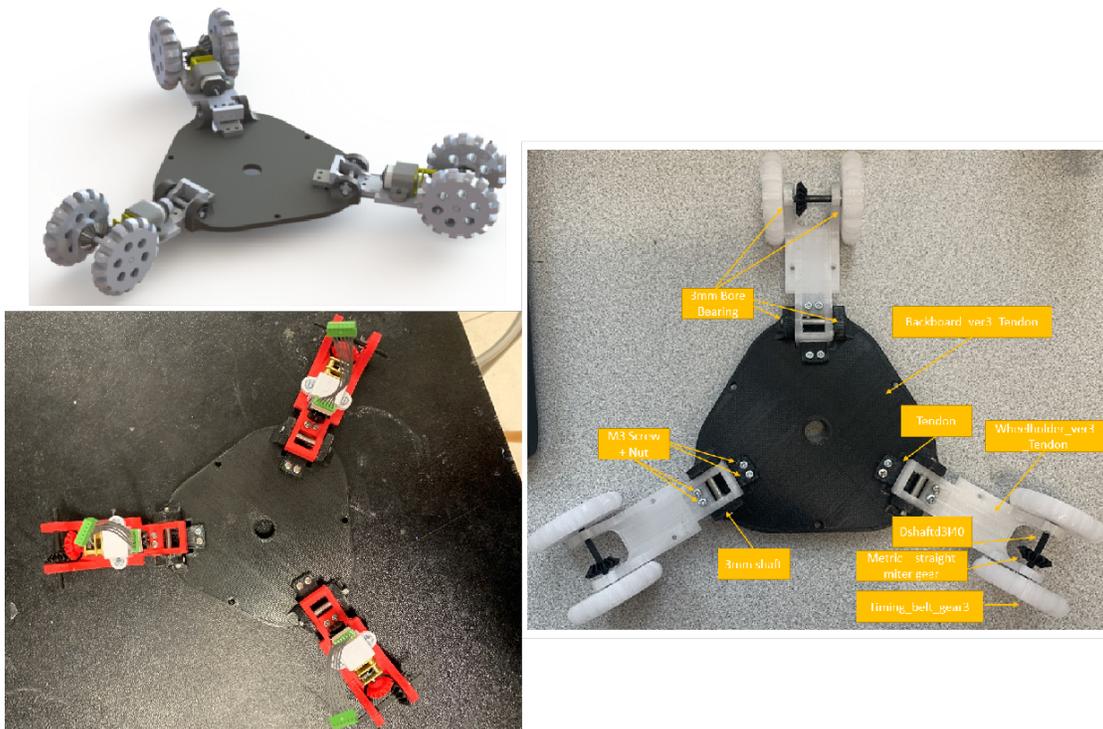


Figure 3.9: (top-left) CAD rendering of active wheels with suspension design. (bottom-left) 3D printed parts with motors mounted. (right) Assembled wheel set with notation of each part.

In order to obtain greater stability and adaptability in response to different surroundings, such as uneven surfaces or narrow openings, a wheel array with a wider, axially symmetric base equipped with three motors and suspension capabilities was developed (see Figure 3.9). Each array consists of three 3D-printed wheel holders each connected to a corner of a 3D-printed triangular plate via a 3mm shaft. The wheel holder's rotation about the shaft is limited by a rectangular Ninjabflex "tendon" that is fixed underneath the shaft between the wheel holder and the main plate. The softness of the tendon provides a spring-like force against the wheel holder, providing suspension for the wheels. The wheels themselves are attached to the holders using 3mm D shafts, each with a miter gear between the wheels.

To provide wheel drive, a motor is attached to each of the wheel holders on the wheel array using plastic motor brackets. Each motor's output shaft features a miter gear that drives the corresponding wheel-shaft gear. Wheel array on the other side of the salamander robot is not equipped with motors. Thus, the wheel array follows the front array passively.

Considering the fact that searching and rescue missions are mainly deployed in outdoor environments, we select the active wheels with suspension propulsion mechanism as our default wheel system option in real world application situation because of its potential to be adaptable to unpredictable environment.

3.1.3 Signal Flow of Salamander Robot

Considering the developing efficiency, we use ESP32 as the WIFI module for remote communication and use Adafruit motor driver feather board as the velocity control board for the three motors on the wheel set. The electronic modules (custom PCB board for origami actuator, ESP32 WIFI module and Adafruit motor driver feather board) are connected with jumping wires following the wiring chart in Figure 3.10.

Figure 3.11 shows an overall control principle scheme, showing the signal flow of a single module salamander robot. In general, after calculating the deformation and robot velocity requirement during the manipulation, the control signal is sent from a remote station to the robot via WIFI

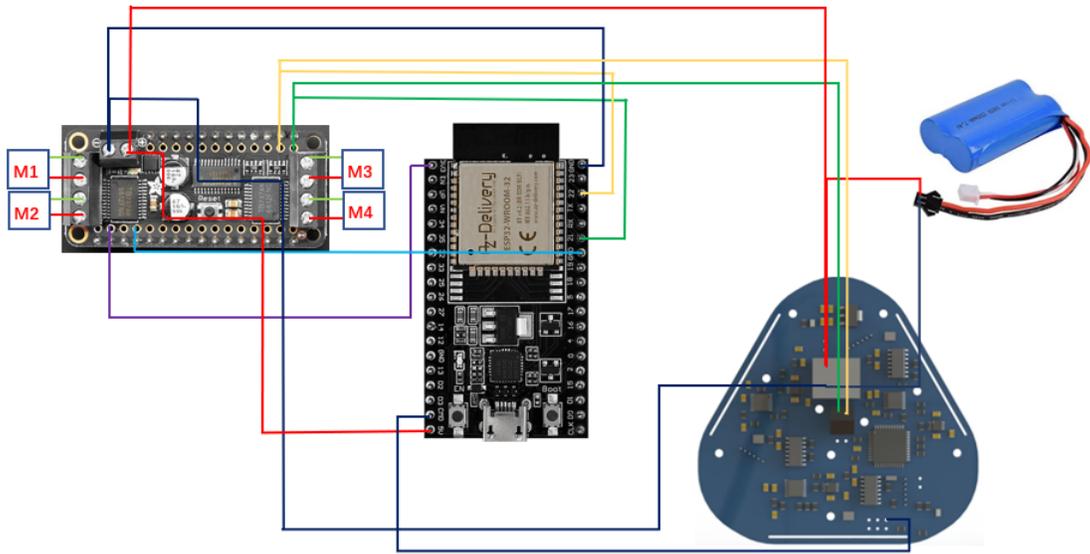


Figure 3.10: Wiring chart of a single module salamander robot.

modules, according to received signal, the local circuit control the actuator and propulsion system separately.

For the origami actuator, the custom PCB board received the deformation state via I^2C protocol, then map it to the motor rotation angle in microcontroller. With motor drivers and feedback through encoders, the actuator is able to deform to expected shape through a implemented PID controller. Meanwhile, the current sensor detects possible overload in the circuit. For propulsion mechanism, there a commercially available onboard driver module is deployed to control the motors rotational speed according to the received signal via I^2C protocol.

3.2 Robot Demonstration

We performed a range of experiments to test the capabilities of the Salamander robot, quantify the performance of different transmission methods against each other. We tested their speeds on a range of surfaces, as well as their ability to traverse a constrained environment, turn over, climb

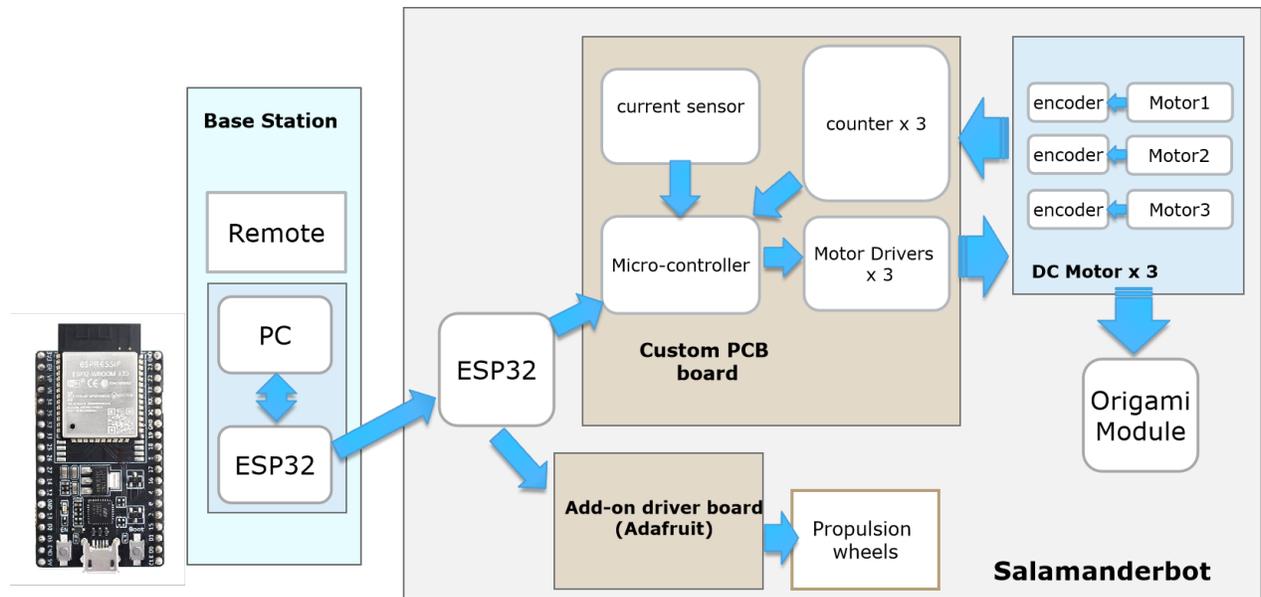


Figure 3.11: Signal flow chart of a single module salamander robot.

up ramps of varying inclines, and climbing straight along a gap vertical to ground surface. In the end, we assembled three single module salamander robot into a 3-segment salamander robot, and climbing over a 15 cm obstacle with manually tuned gait.

3.2.1 Linear Locomotion Speed on Different Surfaces

Linear speed of the salamander robot is tested on five different surfaces, including tile, paper, rubber, carpet, and grass. The voltage of the motors is set at 11.9 V. Modules are tested with belt transmission system and gear transmission system separately, and are expected to go straight-line distance of 2 ft (609.6 mm). For each setting, we conducted the experiment for five times, and calculated the average speed. The result is shown in Table 3.1 (Considering the fact that active wheels with suspension design are more of a trade-off design for real world environment instead of seeking the best performance on certain index, this design does not join the comparison). Modules assembled with gear transmission travels faster than modules assembled with belt transmission since the maximum load of the belt transmission is limited by the friction between the belts and pulley slots. Sliding between belts and slot will reduce the transmission efficiency. Under extreme conditions, for example on a grass surface, the transmission will not be able to propel the module

Table 3.1: Linear speed with belt or gear transmission system on different surfaces

surface	transmission	belt [mm/s]	gear [mm/s]
	tile		94.3
paper		91.0	303.1
rubber		85.4	242.3
carpet		80.7	278.8
grass		0.2	202.3

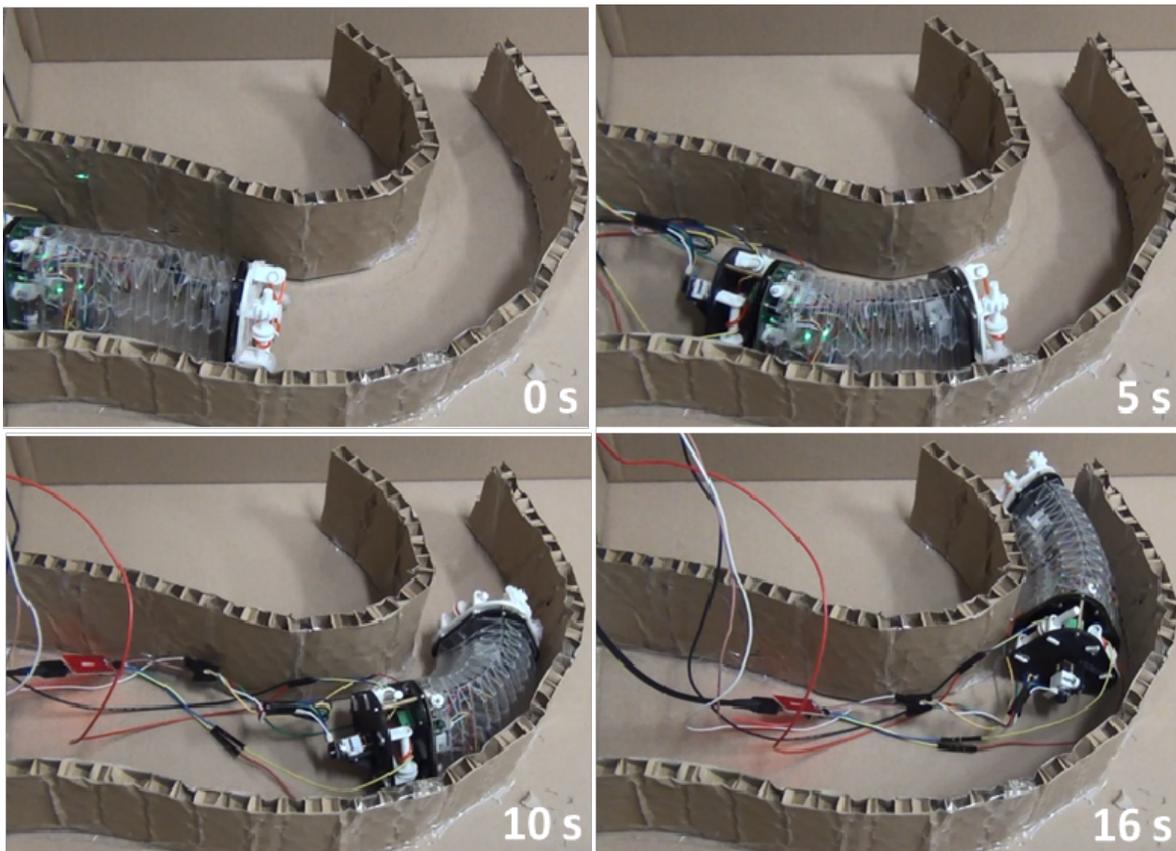


Figure 3.12: Salamander robot going through a maze.

since friction is too high, which is the reason that the the speed of the salamander robot on grass with belt transmission is close to zero. For gear transmission system, the transmission ratio is 14:31, and will stay still unless the gear teeth wear out.

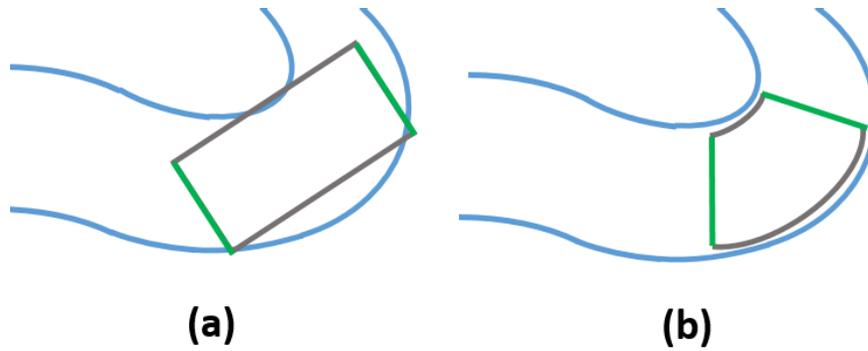


Figure 3.13: (a) Overlapping between the traditional 4-wheel robot with the maze, (b) Salamander robot with the same length can deform and go through the maze.

3.2.2 Climbing on a Ramp

Light weight feature of origami structure could increase its performance in climbing. Thus, we conducted the experiment to test the maximum incline angle of climb for the modules assembled with different transmission systems (Considering the fact that active wheels with suspension design are more of a trade-off design for real world environment instead of seeking the best performance on certain index, this design does not join the comparison).

In this experiment, a slope with changeable angle of climb is created. The surface of slope is covered with carpet or rubber to create different surfaces with different friction. We define the standard of "climbability" as "able to climb 1 ft forward in 30 s". The result is shown in Table 3.2. Modules assembled with belt transmission system are limited with the maximum load so the climbing ability is also lower than modules assembled with the gear transmission system. For gear transmission system, although it can provide enough force for propulsion, the climbing ability is still limited with the maximum friction between the wheel and surface. As for the gear transmission on rubber, unlike the other three conditions, the module fell off the slope instead of staying still. So technically, for gear transmission, if the friction coefficient of the wheel is higher, the climbing ability of the module will keep increasing. In addition, the stability of the robot could potentially be improved if more modules are connected in series, which is a subject of our future work.

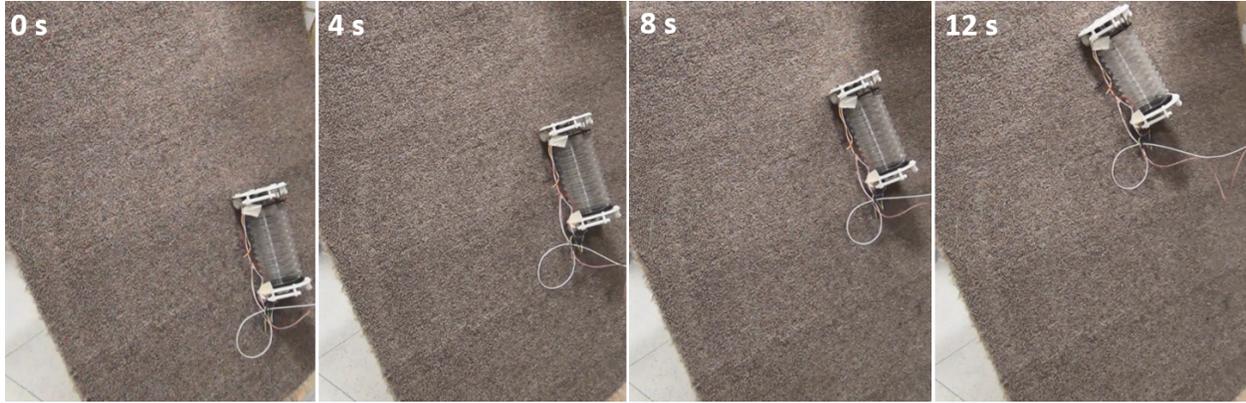


Figure 3.14: Salamander robot climbing on a 60 deg carpet surface slope.

Table 3.2: Maximum angle of climb for modules with belt or gear transmission system on different surfaces

surface	transmission	belt [deg]	gear [deg]
	rubber		18
carpet		36	60

3.2.3 Negotiating Narrow Passageways

In this test, a simplified maze is constructed, which is extremely difficult to traverse with a traditional rigid four-wheel mobile robot. A data cable is connected from the control PCB to a computer through a tiny TTL-USB device, so that we can control the movement of the module from computer with UART to guide the module through the maze. As shown in Fig. 3.12, the module can successfully navigate the maze in 16 s. The maze is designed to be extremely narrow, that there would be no space for a similar snake robot to conduct serpentine locomotion to go forward. Meanwhile, the sharp turn near the end of the maze would cause a traditional four-wheel robot to get jammed as shown in Fig. 3.13. For such complex environments, which could be found in search-and-rescue or exploration tasks, salamander robot is a perfect agent.

Before the experiment, the module is programmed with basic motion primitives so that salamander robot can adjust its posture to achieve steering. Although primitives are already encoded, the robot is remote controlled. Thus, the reaction time of human being likely slows down the speed. In future work, tactile sensors and/or a camera can help the robot automatically analyse the

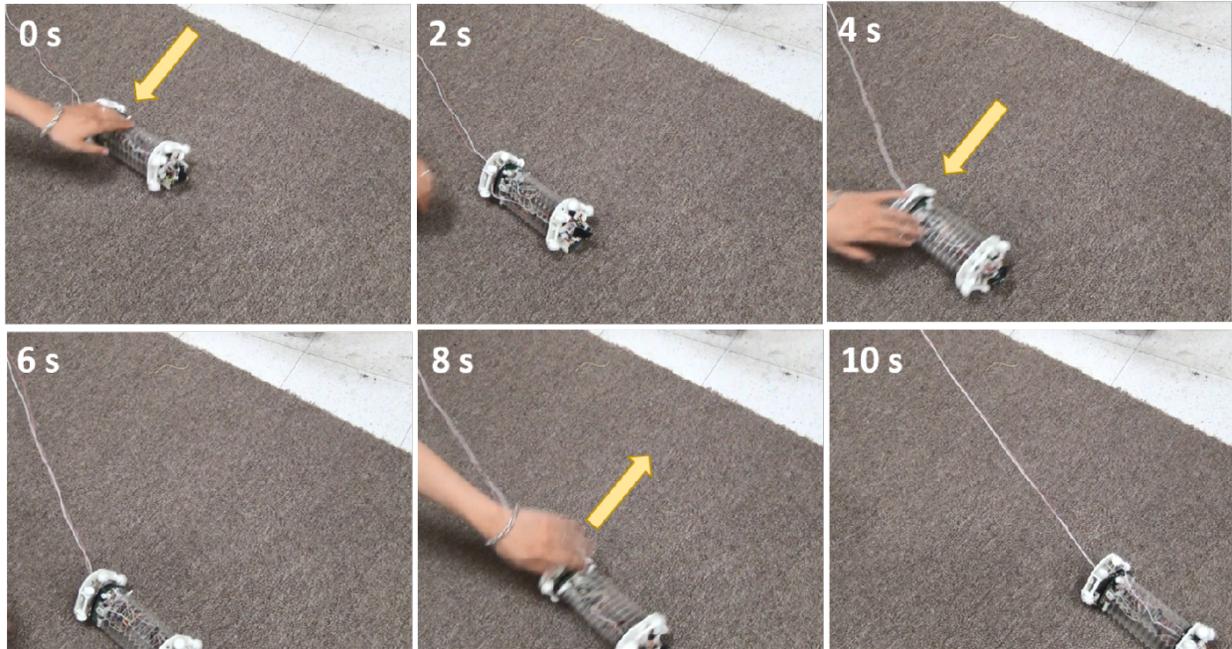


Figure 3.15: During the experiment, salamander robot was turned over several times, while the velocity stayed the same. The yellow arrows show the turning over directions.

environment and execute the proper motion. In this case, we believe that the traversal speed would be comparable to the characterized linear speed values.

3.2.4 Robustness through Radial Symmetry

Propulsion wheels distributed on the three surfaces on the module may be considered as a waste of energy when the salamander robot is moving on flat surfaces. However, this design makes it possible to overcome some extreme cases. To validate the robustness of the robot, we manually turned over the robot body, or randomly tossed it on the ground. It turns out that the salamander robot is still able to travel as usual as shown in Fig. 3.15. Also, if the three pairs of wheels touch the surface at the same time, which could be a tube or a narrow crack, the propulsion could be able to carry the module up a large incline.

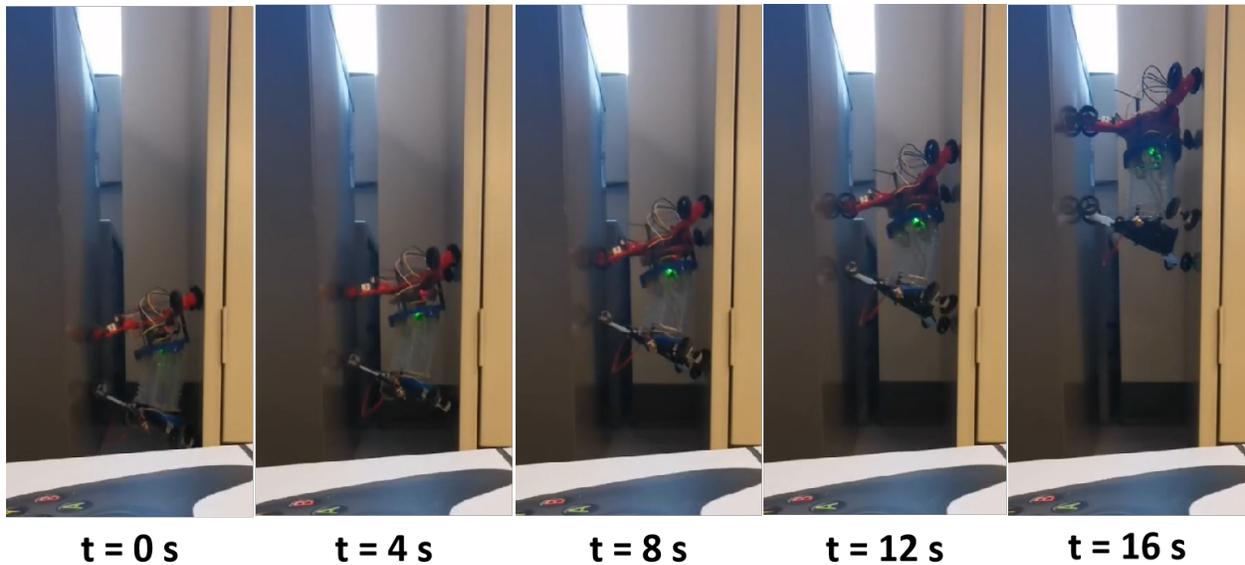


Figure 3.16: A single module salamander robot moving vertically along a narrow gap for a body length in 16 s with time stamp.

3.2.5 Vertical Climb

Suspension design in the propulsion wheel system is supposed to ensure the robot be adaptive to unexpected environment, and even utilize the reactive force from the environment to help with propulsion. In order to demonstrate the hypothesis, in this test a single module salamander robot equipped with suspension propulsion wheel system are set in a narrow gap environment vertical to the ground surface for a "free walking". Before the test, there is no pre-programmed gait or manual manipulation during the test influencing the origami actuator or the wheel system, to ensure that the robot is passive reacting to the environment.

As shown in Figure 3.16, the suspension and under-actuated origami module passively adapt to the narrow gap while applying normal stress to the gap surface, which result in friction force for the robot to move vertically upward a body length distance in 16 s. This test demonstrate that the salamander robot is able to to passively adapt to unexpected environment with the flexibility of the suspension and origami actuator.

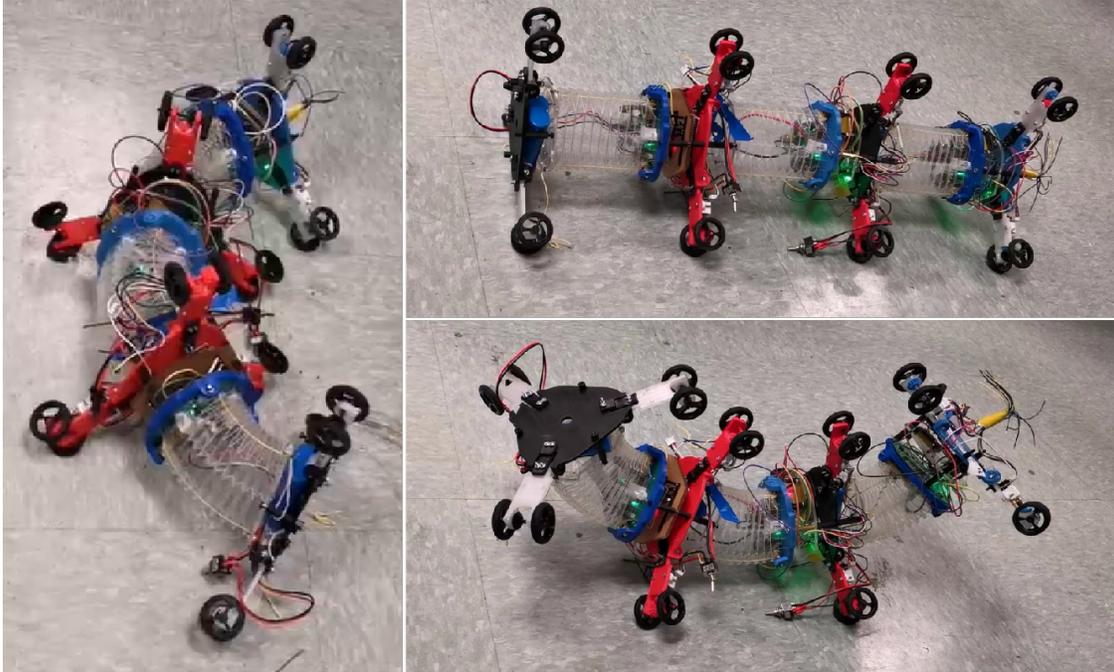


Figure 3.17: (left) A 3 module salamander robot turning right. (top-right) A 3 module salamander robot moving straight forward. (bottom-right) A 3 module salamander robot raising two ends.

3.2.6 Multi-module Cooperation

More complex task, for example going over large size obstacle is unlikely to be fulfilled by a single module salamander, which is the reason to build multi-module salamander robot for a better performance against more difficult situation. In this test, three continuum salamander robot are fabricated and serially connected to form a three module salamander robot. Our salamander robot is highly modularized design with individual power package and wireless communication module equipped, thus, the hardware setting is almost the same, and the modules are only mechanically connected (shown in Figure 3.17) while each module can be considered as an independent unit in the wireless network cooperating with each other to accomplish the locomotion gait.

Figure 3.18 shows a three module salamander robot climbing over a 15 cm obstacle. During this test, the modules are pre-programmed with several gaits. The system is manually manipulated by sending gait command to each module. As a result, the robot is manually manipulated to perform climbing locomotion successfully. In the future research, a dynamic model will be generated, combining with the motion planner generated for our robot (introduced in Section 3.4.2)



Figure 3.18: A 3 module salamander robot climbing over a 15 cm high obstacle.

to autonomously perform climbing locomotion more efficiently. As we can notice, in the end of the locomotion, the multi module salamander robot is manipulated to fall off the obstacle, while the locomotion is still able to travel. The result further demonstrate the robustness of the axially symmetric propulsion system design against unpredictable environment.

3.3 Kinematics Modeling

In this section, inverse kinematics of the origami module and kinematics modeling of salamander robot is discussed and calculated for robot steering control, motion primitive generating, and path prediction. This section consists of two subsections, the inverse kinematic for the origami actuator for predicting the shape of the main body during steering control, and the kinematics model for the single module salamander robot motion primitive generating, in order to predict the path of the robot under certain input.

3.3.1 Inverse Kinematics for Origami Actuator

As a typical continuum actuator, our inverse kinematics model is based on existing research work (Jones and Walker (2006)) with detailed derivation. We define the state of the actuator with three

Table 3.3: Parameters appeared in inverse kinematics for origami module.

Parameters	Explanation
s	length of the module central axis
θ	angle between top and bottom surfaces of the module
ϕ	first we make a projection of the top surface's normal on the bottom surface, and ϕ is the rotate angle of the projection

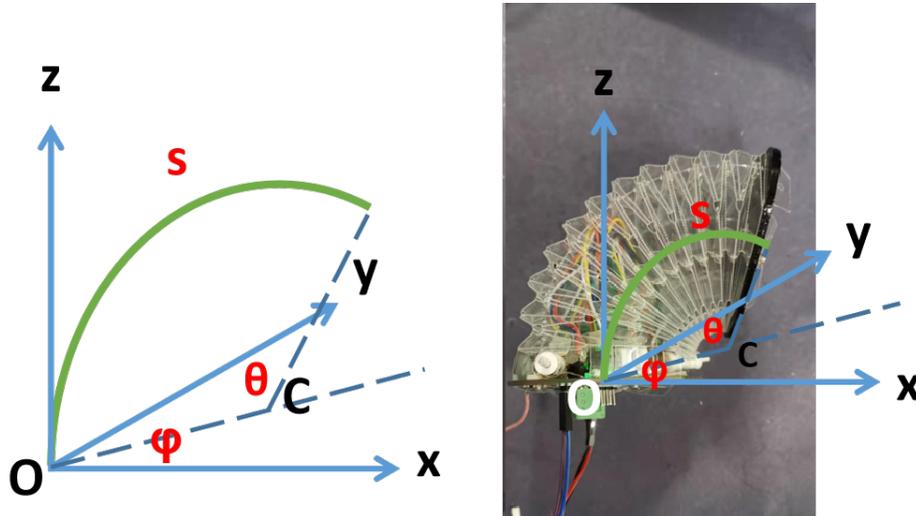


Figure 3.19: (left) Sketch of the origami continuum module. The three parameters to present module states are marked red. The green arc represent the central axis of the module and C represent it central of circle. (right) Inverse kinematics parameters and coordinate overlaid on a bending continuum origami module

parameters, including the length of the neutral axis, the angle between two ends of the module, and the bending direction of the module. The parameters are present in Figure 3.19.

The three parameters and its description are listed in Table 3.3

In this continuum actuator, the input is the length of three cables embedded in the actuator, which is directly related to motor rotation. Figure 3.19 (right) shows the inverse kinematics parameters and coordinate overlaid on a bending continuum origami module. The origin point is set to the geometric center of the bottom plate of the actuator, with x-axis pointing the right side of the bottom plate (cable 1 locate on vertical direction of x-axis; Cable 2 locate at negative x-axis area and cable 3 locate at positive x-axis area), and z-axis pointing vertically upward of the bottom

plate, forming a Cartesian coordinate. The relationship between the cable lengths and the three state parameters can be written as:

$$\begin{aligned}
 l_1 &= 2\sin\left(\frac{\kappa s}{2}\right)\left(\frac{1}{\kappa} - d\sin(\phi)\right) \\
 l_2 &= 2\sin\left(\frac{\kappa s}{2}\right)\left(\frac{1}{\kappa} + d\sin\left(\frac{\pi}{3} + \phi\right)\right) \\
 l_3 &= 2\sin\left(\frac{\kappa s}{2}\right)\left(\frac{1}{\kappa} - d\cos\left(\frac{\pi}{6} + \phi\right)\right)
 \end{aligned} \tag{3.1}$$

In this equation, l_1 , l_2 , and l_3 represent the length of cable 1, cable 2, and cable 3. d is the distance from the geometric center of the bottom plate to the cable attachment point.

A specific condition is that, when the salamander robot is moving on a planar surface (assuming cable 1 locates on the top). Steering locomotion only requires the origami module to rotate horizontally. In this case, ϕ is set to be 0 deg or 180 deg, thus we can easily get a relationship between l_1 , l_2 , and l_3 that $2l_1 = l_2 + l_3$, for a faster calculation. This relationship is also used in the kinematics modeling section for salamander robot.

3.3.2 Kinematics Model for Single Module Salamander Robot

Modeling related research works have been conducted on snake robot (Rezapour et al. (2014); Luo et al. (2014)) and wheel robot. Figure 3.20 shows typical modeling sketch for snake robots and wheel robots. Unlike snake robot or traditional wheel robot, the salamander robot use active wheel for propulsion while deforming its body at the same time for steering as shown in Figure 3.21. In this section, a kinematics model of the single module salamander robot on 2D planar surface is discussed and derived under certain assumptions. Also a path generator based on the kinematics model is derived for path planning and prediction.

The front side of the salamander robot consists of active wheel set with all the electronic modules and the motors for continuum origami module, taking up more than half of the total weight. Consider the fact that the origami structure and backward passive wheel set is much more light weighted than the front side, it is reasonable to assume that the front active wheel set can perform

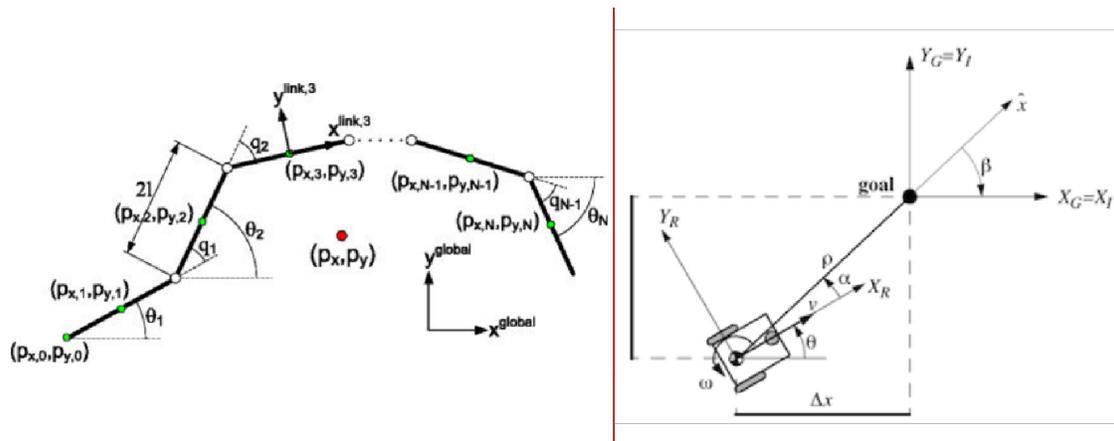


Figure 3.20: (left) Research about a simplified snake robot modeling (Rezapour et al. (2014)) (right) A typical kinematics model sketch of a two-wheel mobile robot.

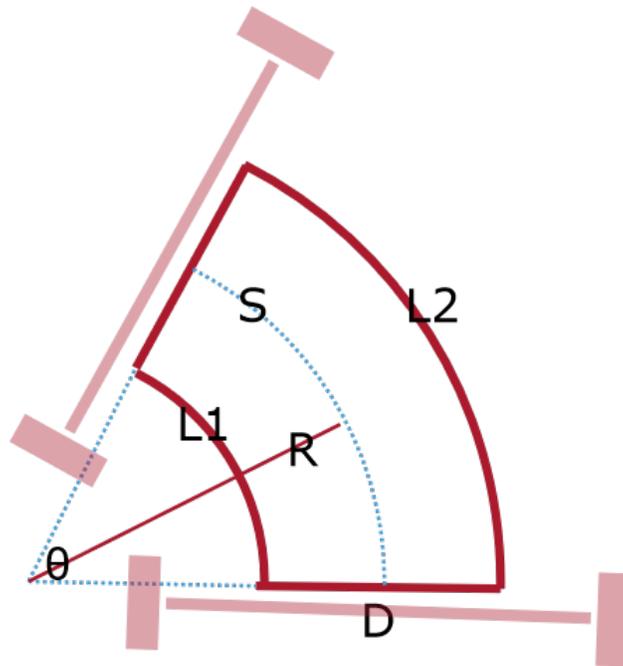


Figure 3.21: Simplified sketch of a single module salamander robot. The pink dumbbell-like shape on two ends represent the two wheel sets of the robot. The bending block in between represent the origami module. Notations on the sketch are explained in Section 3.3.2

pure rolling while the back passive wheel set can slide laterally when the module deforms. Other assumptions related to the kinematics modeling are listed below:

- Movement in horizontal plane;
- Wheels are point or line contact with ground (wheel not deformed), which means the radius of wheels keep constant;
- Steering axis normal to surface;
- Pure rolling for the front wheel set (velocity differential can be created for the two wheels on surface), while the passive back wheel set can slide
- Considering the fact that when the module deformation is constant, the salamander robot moves in an arc with same curvature of the neutral axis, here we assume that evolving curvature of robot trajectory is also same with curvature of the neutral axis in short periods.
- Angular velocity of motors can be constant

The following calculations meant to derive the relationship between the curvature and robot speed (notations mentioned in the calculation are explained in Figure 3.21 and Table 3.4). Considering the fact that $L_1 + L_2 = 2L$, the rotational velocity value of the two bottom motors on the origami module are opposite to each other.

$$\frac{L_2}{\theta} - \frac{L_1}{\theta} = D$$

$$L_1 + L_2 = 2L$$

$$\frac{2(L - L_1)}{D} = \theta = \frac{L}{R} = L \times \kappa$$

$$\kappa = \frac{2}{DL}(L - L_1)$$

$$L - L_1 = 2\pi r \alpha$$

$$\kappa = \frac{4r}{DL} \alpha$$

Table 3.4: Parameters appeared in single module salamander robot kinematics modeling.

Parameters	Explanation
L_1, L_2	length of the cable on the left and right
S	length of the neutral axis
L	original length of L_1, L_2, S
θ	angle between top and bottom layer
D	distance between two wheels
κ	curvature of neutral axis
$\Delta\kappa$	variation of curvature of neutral axis
r	radius of the spool on motor
α	absolute value of rotation angle of the two motors on origami module
$\Delta\alpha$	absolute value of rotational velocity of the two motors on origami module
ω_1	rotation velocity of the bottom left motors on wheel set
ω_2	rotation velocity of the bottom right motors on wheel set
v	linear velocity of the single module salamander robot

$$\Delta\kappa = \frac{4r}{DL} \Delta\alpha$$

$$v = \frac{(\omega_1 + \omega_2)r_1}{2}$$

According to the calculation result, the curvature evolving linearly with motor rotation on origami module. Since motor rotation velocity is constant, curvature evolves linearly with time, which means that curvature evolves in constant speed.

The last equation in the calculation procedure shows linear velocity of the single module has a linear relationship with the rotational velocity of the motors on wheel mechanism. Considering the fact that motor rotational velocity can be constant in a certain period, the variation of curvature $\Delta\kappa$, which is same with the variation robot trajectory curvature according to the assumption, is linear with robot velocity.

As we know, a curve whose curvature changes linearly with its curve length is named Euler spiral as shown in Figure 3.22. Thus, the motion primitive of the single module salamander robot is a partial Euler spiral when the origami module is deforming. The motion primitive will be an arc

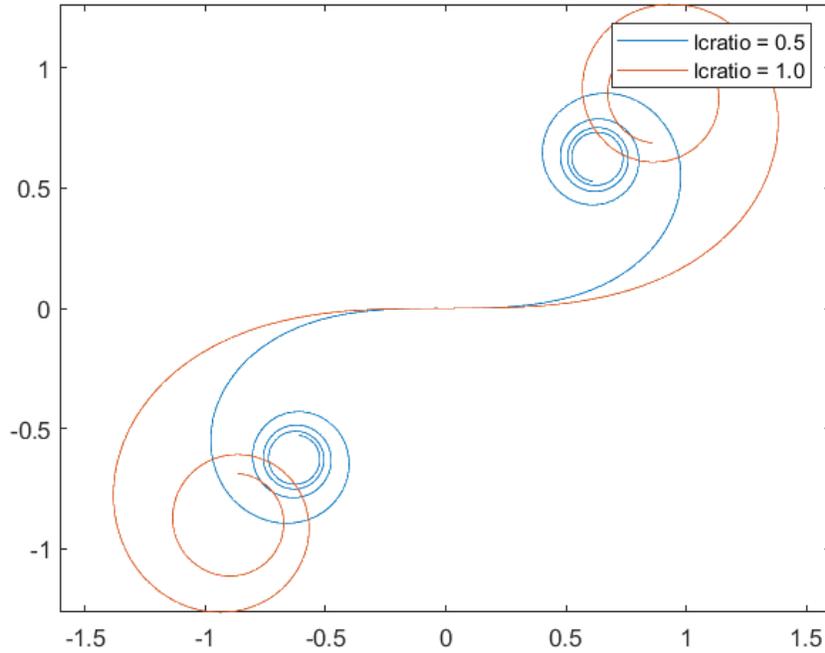


Figure 3.22: Two Euler spirals with different ratio of length to curvature.

with same curvature of the neutral axis. Equation of Euler spiral and related parameters are listed in Equation 3.2 and Table 3.5 respectively. To be specific, with the ratio of length to curvature, which is the same with ratio of the curvature variation and robot velocity, motion primitive of single module salamander robot can be described with a piece on a certain Euler spiral. With the current and end curvature, it is able to "cut" a piece from this spiral to find the exact piece on the Euler spiral as predicted path.

For example, if the current curvature variation and robot velocity ratio is 1.0, we know the predicted path can be find on the Euler spiral with ratio of length to curvature equal to 1.0. Then, with the information that during this period, the curvature changes from 1 to 2, we are able to trim the Euler spiral from position $x = \int_0^1 \cos(t^2)dt$, $y = \int_0^1 \cos(t^2)dt$ to $x = \int_0^2 \cos(t^2)dt$, $y = \int_0^2 \cos(t^2)dt$ as the predicted path.

Table 3.5: Parameters appeared in Euler spiral related equations

Parameters	Explanation
R	radius of curvature
t	length of the spiral from origin point to current position
RL	ratio of length to curvature
θ_{Euler}	slope angle at current position

$$\begin{aligned}
x &= \frac{1}{a} \int_0^{at} \cos(t^2) dt \\
y &= \frac{1}{a} \int_0^{at} \sin(t^2) dt \\
a &= \frac{1}{\sqrt{2RL}} \\
\theta_{Euler} &= t^2
\end{aligned} \tag{3.2}$$

Whenever the curvature variation changes, we repeat the steps above to get a new piece of Euler spiral, and attach to the previous path while keeping the entire path continuously derivable. With the slop angle difference between previous curve end and current curve beginning $\Delta\theta_{Euler}$ and the position difference between previous curve end and current curve beginning Δx_t and Δy_t , the transformation matrix $T(\Delta\theta_{Euler})$ can be generated:

$$T(\Delta\theta_{Euler}) = \begin{bmatrix} \cos(\Delta\theta_{Euler}) & \sin(\Delta\theta_{Euler}) & \Delta x_t \\ -\sin(\Delta\theta_{Euler}) & \cos(\Delta\theta_{Euler}) & \Delta y_t \\ 0 & 0 & 1 \end{bmatrix}$$

With the transformation matrix $T(\Delta\theta_{Euler})$, the current curve can be "rotate" to attach smoothly on the previous curve forming the completed path prediction. Thus, the overall path prediction of the single module salamander robot is generated as shown in Figure 3.23.

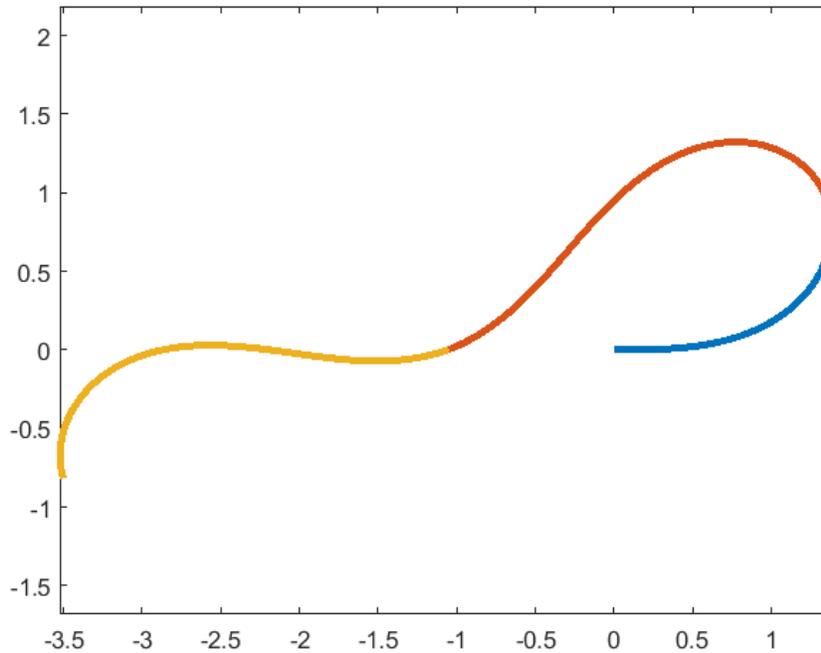


Figure 3.23: A piecewise curve consists of three pieces of trimmed Euler spirals

3.4 Motion Planning

In this section, we introduce a motion planning algorithm for Salamander robot. The algorithm is supposed to create a workable motion planner for our robot considering the obstacle, smoothness, beginning orientation, ending orientation, and maximum curvature.

Considering the fact that our proposed working environment is narrow. And the path optimizer afterward can change the path a lot. We would like to have a workable motion plan for our robot. Two possible motion planner, Lattice Based A-star Algorithm and Covariant Hamiltonian Optimization for Motion Planning (CHOMP) are discussed in this section. Both of the planner are programmed and run in Matlab. The two motion planner are compared in Section 3.4.3.

3.4.1 Motion Planner with Lattice A-star

Unlike traditional A-star searching algorithm, in lattice based A-star, every time an unexplored child node is visit, several possible path are created with our kinematics model as the recent fu-

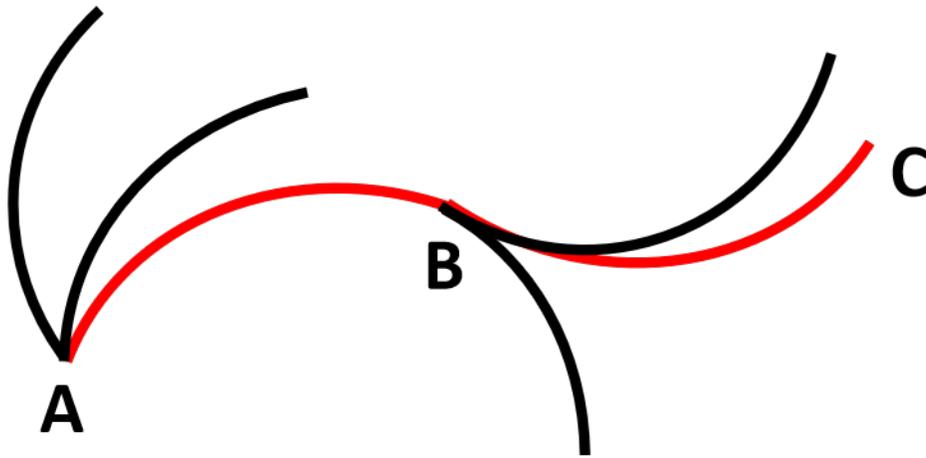


Figure 3.24: A sketch of lattice based A-star searching algorithm. The curves represent the lattice generated from each node. The green curves represent the searching result.

ture possible motion, which is able to be performed by the robot. The possible paths are named "lattice", and the end nodes of these "lattice" are record as an unexplored child node for further searching. Figure 3.24 present the theory of lattice based A-star algorithm

Figure 3.25 shows a lattice based A-star for our robot in simulation. The searching process is set to stop once we are close enough to the goal. While because of our unique motion, we can hardly make sure the robot can exactly reach the goal and with right orientation in the searching, because the robot can only follow the lattice to move. during the searching, cost for each movement are the same, while it goes infinite while hitting the obstacle. And the heuristic is the Euclidean distance between the current search node and the goal.

Lattice based A-star motion planner ensures the feasibility of the trajectory generated for single module salamander robot. However orientation constraint incorporation does not working well with this planner, which means the beginning or ending orientation of the trajectory can not be fixed, which limited the performance under the situations that orientation matters.

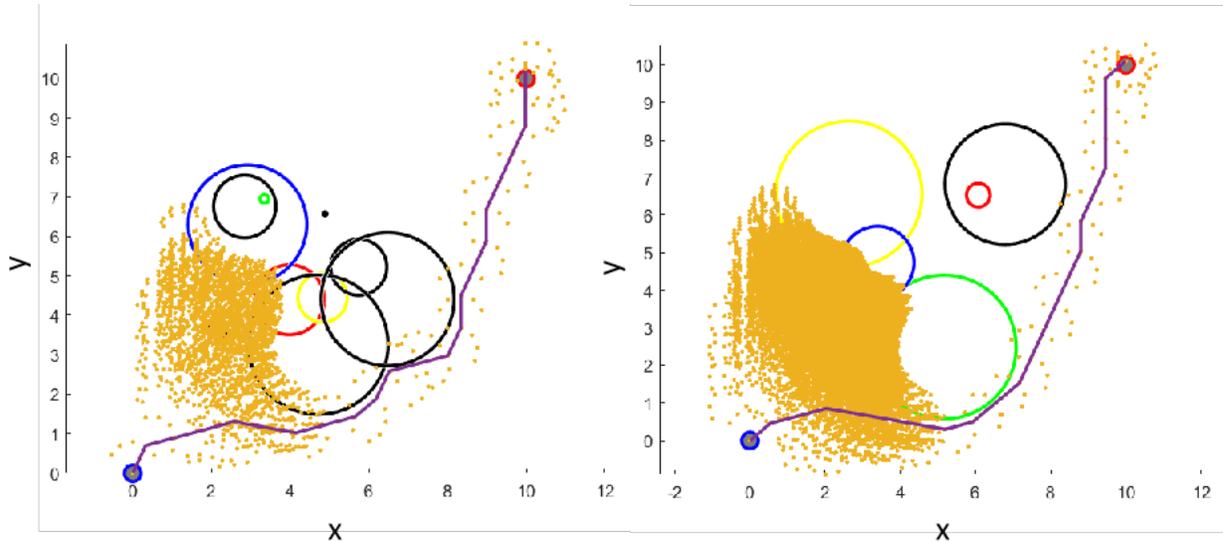


Figure 3.25: Two example of lattice based A-star search with different obstacle setting. The left trial searched 49735 nodes in 6.36 s to reach a solution. The right trial searched 883165 nodes in 48.89 s

3.4.2 CHOMP Based Motion Planner

In this section we introduce a motion planner based on CHOMP, which is short for Covariant Hamiltonian Optimization for Motion Planning (Ratliff et al. (2009); Zucker et al. (2013)).

It is a motion planning method to generate and optimizing trajectories simultaneously for robotic systems, A covariant gradient update ensures it quickly converges to a smooth and collision free locally optimal trajectory. Due to the unique property of CHOMP, related research has been developed with modified motion planning method based on CHOMP for different situations with novel properties (Mukadam et al. (2016); Byravan et al. (2014); Kalakrishnan et al. (2011)).

This motion planner consist of following modules: Gradient descent with regard to obstacle in the environment and smoothness of the trajectory it self, constraint deployment on orientation and maximum curvature, update rules and re-sampling procedure. The pseudo code of this motion planner is shown in Algorithm 1.

Input: obstacle, end points/orientation, maximum curvature, maximum iteration;

for $k = I$ **do**

```

     $v_{ori} = f_{constraint}(\xi_k, v_{start}, v_{end});$           /* Constraint for orientation */
     $obs_{virt} = f_{virtualobs}(\xi_k, curv_{max});$            /* Virtual obstacle */
     $[F_{obs}, \nabla F_{obs}] = f_{obstacle}(\xi_k, obs, obs_{virt}, \epsilon);$ 
     $[F_{smooth}, \nabla F_{smooth}] = f_{smooth}(\xi_k);$ 
     $U(\xi_k) = F_{obs}(\xi_k) + \lambda F_{smooth}(\xi_k);$ 
     $\nabla U(\xi_k) = \nabla F_{obs}(\xi_k) + \lambda \nabla F_{smooth}(\xi_k);$ 
    if  $U(\xi_k < U_{best})$  then
        |  $\xi_{best} = \xi_k;$ 
        |  $U_{best} = U(\xi_k);$                                /* Record best result */
    end
     $\xi_{k+1} = \xi_k - \frac{1}{\eta} A^{-1} \nabla U(\xi_k) + \alpha_{const} A^{-1} v_{ori};$           /* Update */
     $U;$                                                   /* Restart */

```

end

Algorithm 1: Motion Planning algorithm based on CHOMP

Gradient Descent and Cost Function

In this algorithm, we discrete the trajectory into a set of n points $(q_0, \dots, q_n + 1)$ which consists of the trajectory ξ_k , then compute dynamical quantities via finite differencing. Thus, we focus on finite dimensional optimization.

During optimization, the cost function of a trajectory contains two parts: an obstacle term F_{obs} , which measures the cost of being near obstacles; and a prior term F_{smooth} , which measures dynamical quantities of the robot such as smoothness and acceleration, same with the original CHOMP algorithm (Zucker et al. (2013)). We generally assume that F_{smooth} is independent of the environment while F_{obs} mainly depends on the distance between current trajectory to the obstacles in the environment. Thus, the cost function can be written as follows:

$$U(\xi_k) = F_{obs}(\xi_k) + \lambda F_{smooth}(\xi_k)$$

In the function, $U(\xi_k)$ represent the cost function value for the k^{th} trajectory ξ_k . λ is a scale parameter for F_{smooth} which will increase along with the iteration considering the fact that in the beginning of the iteration, the main target is to generate a collision free path, while it is more reasonable to focus on trajectory smoothness afterward.

To be specific, the smoothness term F_{smooth} is equal to the sum of squared derivatives along the trajectory. In the continuous situation, it can be written as:

$$F_{smooth} = \frac{1}{2} \int_0^1 \left\| \frac{d}{dt} \xi(t) \right\|^2 dt$$

In discrete condition, the trajectory is split into $n + 2$ points (the two fixed ends are included), and the equation is written as follows:

$$F_{smooth} = \frac{1}{2} \sum_{t=1}^n \left\| \frac{q_{t+1} - q_t}{\Delta t} \right\|^2 = \frac{1}{2} \|K\xi + e\|^2 = \frac{1}{2} \xi^T A \xi + \xi^T b + c$$

K is a finite differencing matrix while e conclude the two fixed points on the trajectory q_0 and q_{n+1} . In this equation $A = K^T K$, $b = K^T e$, and $c = \frac{1}{2} e^T e$. Hence F_{smooth} is a quadratic equation with Hessian matrix A . The matrix A can also be regarded as the matrix to calculate acceleration of all the unfixed points on the trajectory. Matrix A will also be used in constraint deployment for a smooth update on the trajectory.

The obstacle related term F_{obs} considering the distance between the points on trajectory to the obstacles. Lower distance result in higher obstacle term value. Besides, a "quick" pass through the obstacle is not expected for our trajectory considering the fact that the trajectory is expected to be away from obstacle instead of simply pass the obstacle quick enough to lower F_{obs} value. Thus, F_{obs} is written as follows in continuous condition:

$$F_{obs} = \int_0^1 c(\xi(t)) \left\| \frac{d}{dt} \xi(t) \right\| dt$$

In this equation, $c(\xi(t))$ represent the obstacle potential function, which calculate the cost depend on the distance between the trajectory and the obstacles. For a smooth and derivable potential,

$c(\xi(t))$ is written as follows, same with the original CHOMP algorithm (Zucker et al. (2013)):

$$c(\xi(t)) = \begin{cases} -d(\xi(t)) + \frac{1}{2}\epsilon, & \text{if } d(\xi(t)) < 0 \\ \frac{1}{2\epsilon}(d(\xi(t)) - \epsilon)^2, & \text{if } 0 \leq d(\xi(t)) \leq \epsilon \\ 0, & \text{otherwise} \end{cases}$$

In this equation, $d(\xi(t))$ represents the distance between current trajectory to nearest obstacle, and ϵ defines the effecting range of the obstacles.

Thus, for the discrete condition, F_{obs} is written as:

$$F_{obs} = \sum_{t=1}^n c(q_t) \left\| \frac{q_{t+1} - q_t}{\Delta t} \right\| = c^T \|K\xi + e\|$$

To perform gradient descent, first we approximate the cost function with a first-order Taylor expansion:

$$U(\xi) \approx U(\xi_k) + \nabla U(\xi_k)^T (\xi - \xi_k)$$

In order to make small changes in the average acceleration of the resulting trajectory as possible, an extra term $\frac{\lambda}{2}(\xi - \xi_k)^T A(\xi - \xi_k)$ is added, thus, the optimization problem becomes:

$$\xi_{k+1} = \arg \min_{\xi} \left\{ U(\xi_k) + \nabla U(\xi_k)^T (\xi - \xi_k) + \frac{\lambda}{2} (\xi - \xi_k)^T A(\xi - \xi_k) \right\}$$

λ represent the regularization coefficient as a trade off between minimizing the cost function value and minimize the change in average acceleration of the existing trajectory. As a result the gradient descent update rule can be written as follows:

$$\xi_{k+1} = \xi_k - \frac{1}{\lambda} A^{-1} \nabla U(\xi_k)$$

Figure 3.26 shows the iteration of motion planning by gradient descent.

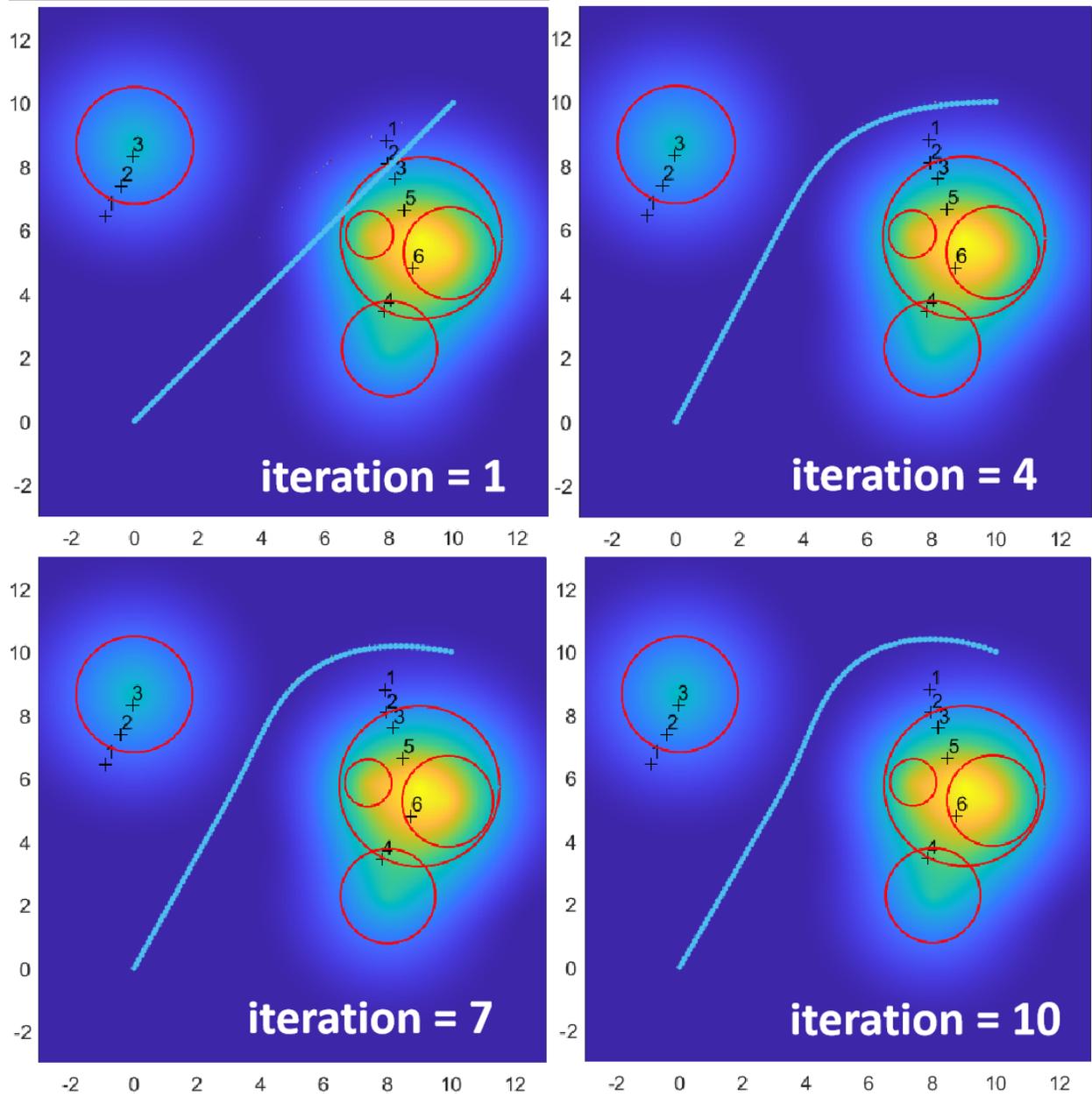


Figure 3.26: Trajectory updated by gradient descent from iteration 1 to iteration 10.

Constraint

In this section, we introduce method of constraint deployment for both the ends orientation and the maximum curvature considering the curvature limitation of the salamander robot.

Orientation constraint is applied utilizing the smooth projection method from original CHOMP algorithm (Ratliff et al. (2009)). The correction vector set is calculated from current trajectory to expected position, then map it via metric tensor A^{-1} and vertical mapping matrix for a smooth update. The orientation constraint is deployed on the points near beginning and ending points.

Considering the situation that correction vector set could have a reduced influence on the trajectory when the angle between current orientation and expected orientation is around $(2n+1)\pi, n \in \mathbb{Z}$, the angle between correction vector and current orientation is clamped between $-\frac{\pi}{6}$ and $\frac{\pi}{6}$.

The procedure of orientation constraint deployment method is listed below:

- Calculate a set of vector from affected current points to estimated point;
- If the differential angle is too large, "clamp" it to a certain range, increasing the update effect when the angle between current orientation and expected orientation is around $(2n+1)\pi, n \in \mathbb{Z}$;
- Map the vectors vertical to the path;
- Map the set of vector by multiply A^{-1} , then scale it in the path update.

Curvature constraint is applied by adding virtual obstacle with the central of circle and curvature radius same with the parts exceeding maximum curvature along the trajectory. Gradient potential generated by the circular obstacle helps to "push" the trajectory in the next iteration until the maximum curvature decrease to an acceptable value. The procedure of curvature constraint deployment method is listed below:

- Calculate the curvature of each point on the path;
- Find the points with curvature larger than maximum value;

- Create virtual circular obstacles with the same central of the circle and radius of curvature for these points;
- Add the virtual circular along with existing obstacles for the obstacle field and gradient calculation in the next iteration.

Figure 3.27 shows the iteration of motion planning by gradient descent with orientation and curvature constraint deployed.

Update rule and re-sampling

The update rule including gradient descent process and smooth projection. Both of the terms are scaled down to a proper value for better converge. And the update equation is as follows:

$$\xi_{k+1} = \xi_k - \frac{1}{\eta} A^{-1} \nabla U(\xi_k) + \alpha_{const} A^{-1} v_{ori}$$

During the iterations, it is possible for the trajectory to reach a local optimization. Thus, in the algorithm, once the possible local optimization is detected, the algorithm will restart from a new initial guess as a random minor arc connecting two ends. Figure 3.28 shows the case when the trajectory is trapped in local optimization.

At last, the trajectory with lowest cost function value will be considered as the most promising candidate.

3.4.3 Comparison between Lattice based A-star and CHOMP based motion planner

Unlike traditional A-star which split the working space into grids or nodes in advance for searching, lattice based A-star algorithm create child nodes during searching procedure to ensure the feasibility of the generated path. Thus, the nodes to be explore is theoretically possible to be unlimited, especially when the density of obstacle increase. Such property result in longer period and larger storage space requirement for searching. Also A-star algorithm can not directly have constraint on

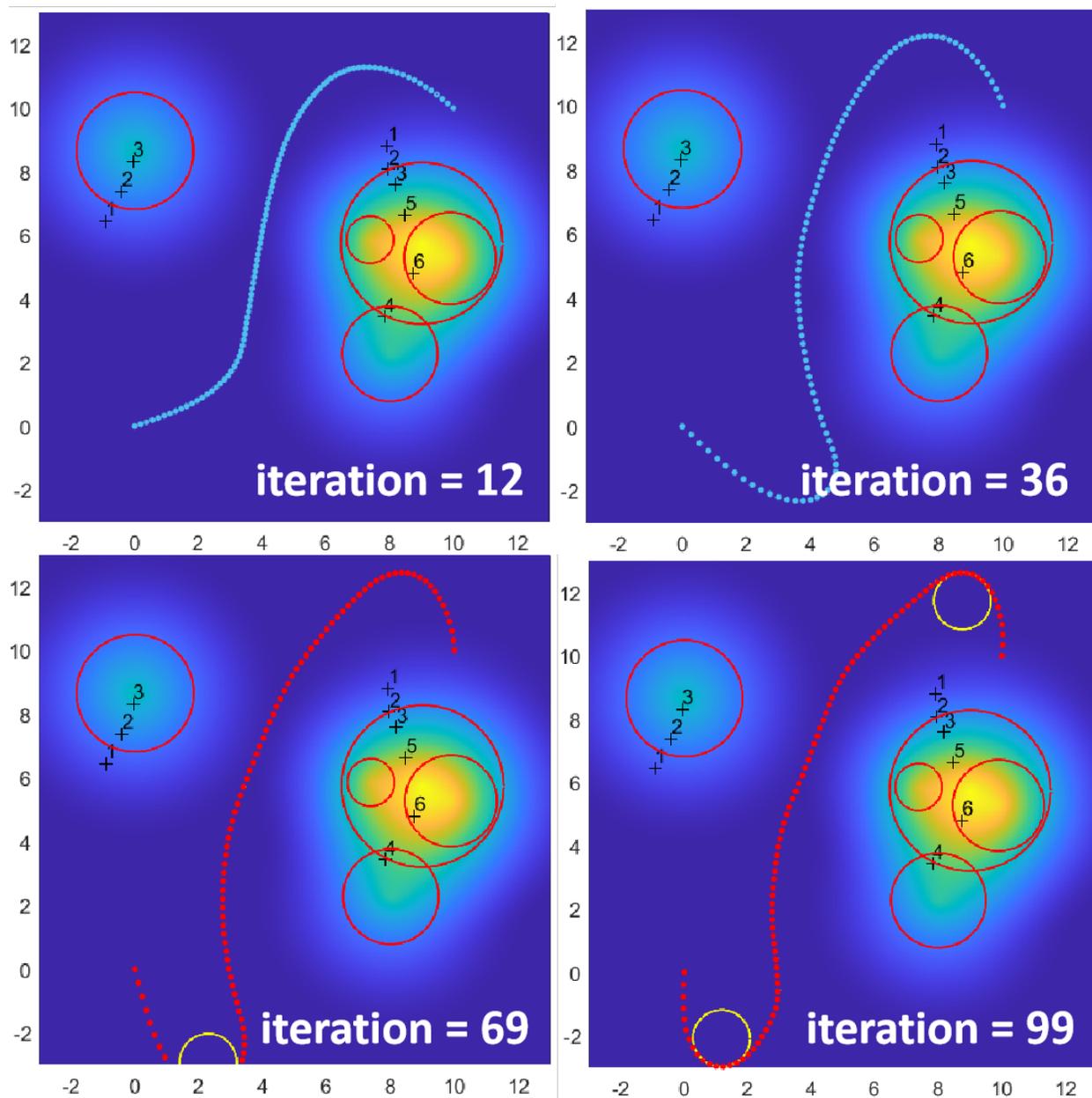


Figure 3.27: Trajectory updated by gradient descent with orientation and curvature constraint applied. Yellow circles represent the virtual obstacles generated in the previous iteration. The beginning orientation angle is set to be -90 deg, and the ending orientation angle is also set to be -90 deg.

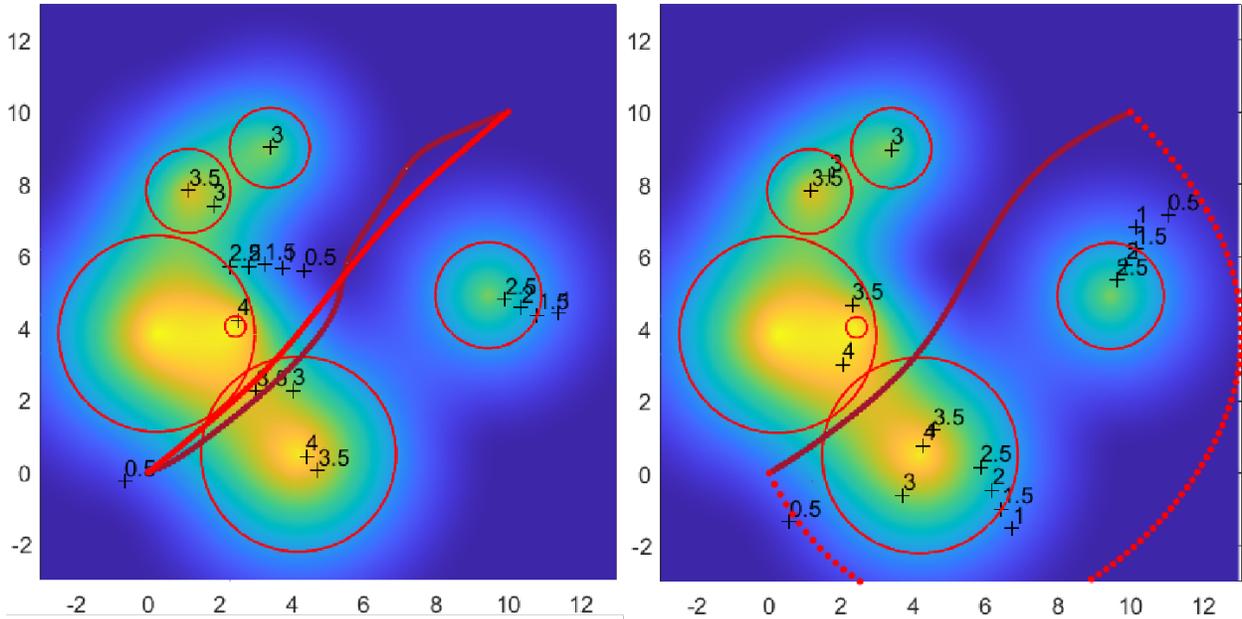


Figure 3.28: (left) Trajectory trapped in local optimization. (right) Restart when detect possible local optimization. Light red curve represent the most promising candidate, and the dark red curve represent the trajectory in current iteration.

starting and ending orientation. Modifying cost function by adding orientation related term with variable scale parameter is a possible method. However it turns out not efficiently incorporated by the search process, which means it is not possibly feasible to constraint the beginning or ending orientation of the trajectory during searching process.

CHOMP utilized covariant gradient descent update to generate and optimizing trajectories for robotic systems to ensure a quick converge to a smooth and collision free locally optimal trajectory. Meanwhile, the constraint deployment can be directly attached to the optimization procedure or obstacle related gradient potential while keep the smoothness of the entire trajectory. On the other hand, gradient of all the points along the trajectory needs to be calculated in each iteration. Also, the optimization procedure can not always ensure a feasible trajectory. A detailed comparison between the two planner is listed in Table 3.6. Considering CHOMP based motion planner embodies more advantages on our salamander robot, it is selected as the default option for motion plan.

Table 3.6: Comparison between salamander motion planner based on lattice A-star and CHOMP

Lattice based A-star	CHOMP
longer solving period	shorter solving period
possibly larger storage space requirement	smaller storage space requirement
difficult to implement constraints	constraints can be applied with an optimization procedure
feasible trajectory by calculated motion primitive	possibly feasible trajectory by deploying constraint
less calculation in each searching iteration	more calculation in each searching iteration

3.4.4 Path Following

Given the feasible path for salamander robot generated by our motion planner. A tracking algorithm named line of sight (LOS) is applied for path tracking (Fossen et al. (2003)).

Figure 3.29 presents a sketch of the LOS algorithm. In each iteration, projection point on the current tracking path is calculated. The projection point is moved towards the way point to get the reference point. Then create a vector from current robot position to the reference point as the reference orientation named LOS vector. LOS vector is used as the steering reference for the robot controller to follow as the robot moving forward. When the robot is close enough to the current way point, the algorithm switched to the next way point on the path. Details of LOS algorithm procedure in each iteration are listed as follows:

- Find the projection point of the robot on the currently tracked path (if the robot is close enough to the current way point, switch to the next way point and update the tracked path);
- Move the projection forward along the reference path for a certain distance to get the reference point (if the projection point is close enough to the way point, then select the way point as the reference point);
- Create a vector from current position to reference point as the LOS vector;
- Control the robot to follow the LOS vector.

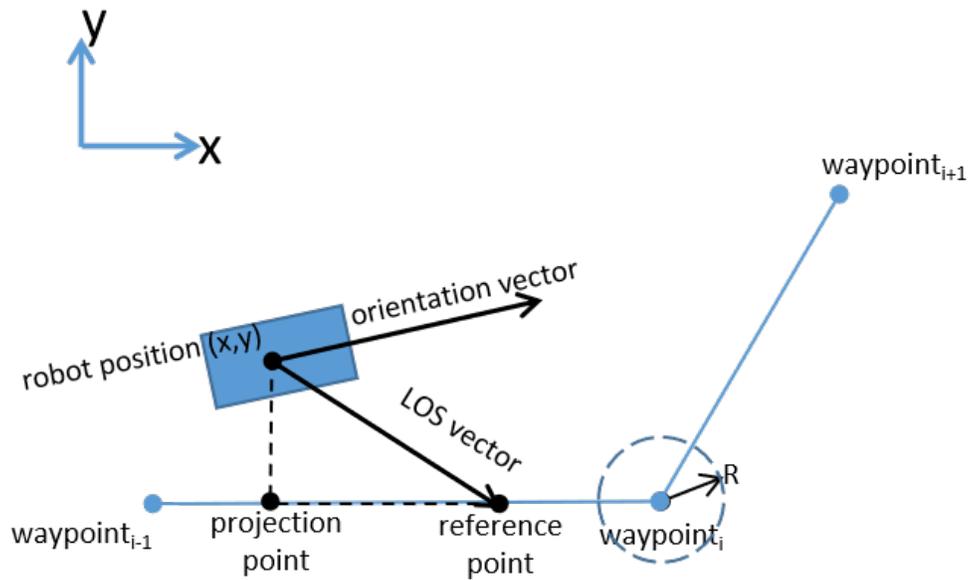


Figure 3.29: A sketch of LOS algorithm. Dashed circle with radius of R represent the area to switch way point.

3.4.5 Motion Planner Deployed on Robot

The entire system is based on Matlab. Motion capture system is used to capture the position and orientation of the robot as the feedback for the PD robot controller. The position of obstacles and target are captured and recorded for motion planner. After calculating the reference vector with LOS tracing algorithm, control signal is sent to the robot through ESP32 WIFI module. Figure 3.30 shows a single module salamander robot traverse to the target avoiding the objects, with the beginning orientation angle set to be 90 deg, and the ending orientation angle set to be 0 deg. Figure 3.31 shows the trajectory generated before traversing, and the path tracking during the experiment.

3.5 Discussion and Conclusion

This chapter introduced the salamander robot, a robot combining the morphological flexibility of a soft robot with the capabilities of traditional wheeled robots. The salamander robot combines a cable-driven origami continuum manipulator with six pairs of active wheels, which can drive it up inclines and through constrained environments. The salamander robot can reach a maximum linear

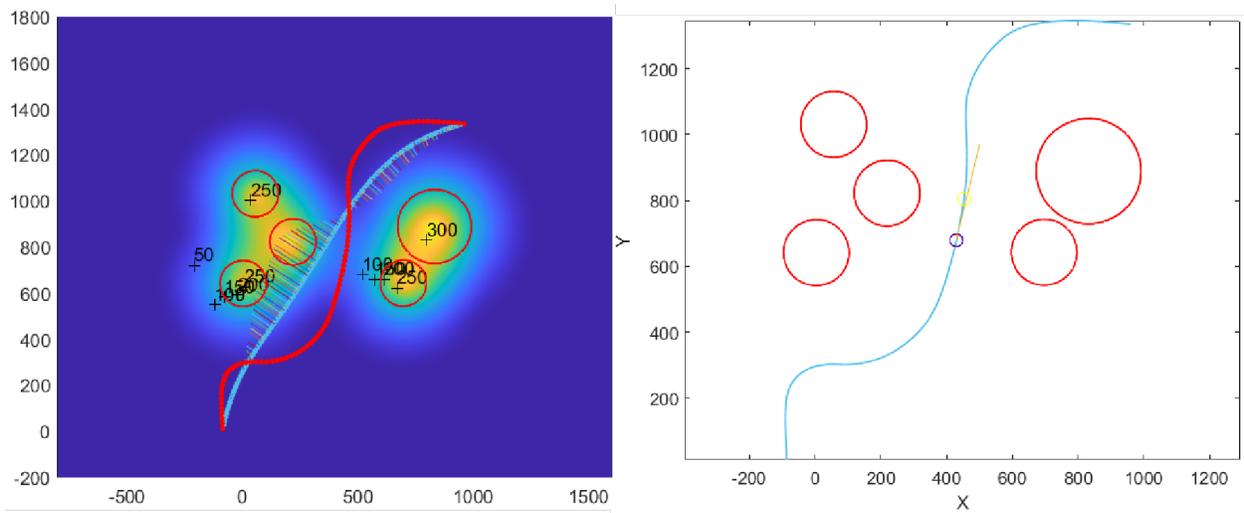


Figure 3.30: (left) Motion planning result. The red curve represent the best trajectory generated in planner after 60 iterations. (right) Path tracking during traversing. The blue, red, and yellow points represent the robot position, projection point, and reference point respectively. The orange line represent orientation of the robot.

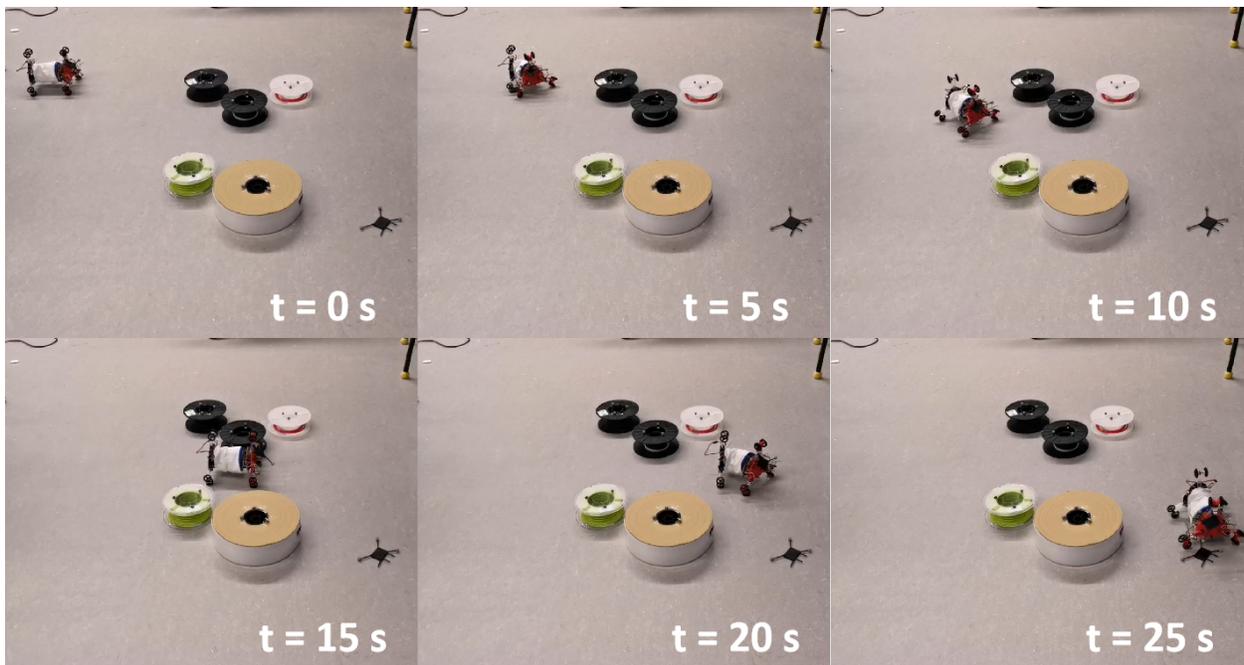


Figure 3.31: Motion planning and path tracking deployed on a single module salamander robot with time stamp.

speed of 303.1 mm/s, travel up a 60 deg slope, traverse through narrow environment with sharp turns or vertically climbing along narrow gap, and stay robust against possible external interrupt on the locomotion.

During our research process, one weakness of the wheels of the salamander robot was their small size. While this was a result of the desire to minimize the profile of the robot, it also reduces their ability to effectively come in contact with the environment. Thus we investigated ways of having flexible structure attached to the wheels and developed active wheels with suspension mechanism as our default propulsion system for outdoor environment. Such mechanism enables the wheels to extend farther away from the body of the salamander robot while still being able to compress as the wheels turned around, expanding the environments the salamander robot is capable of traversing. Besides, the highly modularized design ensures the simplicity of assembling a multi module salamander robot. For demonstration, three salamander robot modules are simply mechanically connected to form a three-module salamander robot able to traverse over large size obstacle.

In addition, inverse kinematics is discussed with an inverse kinematics (IK) solver generated for robot control. Kinematics model for single module salamander robot is generated for path prediction and motion planning. Two motion planners based on lattice A-star and CHOMP algorithm are generated and compared to each other. In the end, the motion planning method is deployed on the robot for autonomous traversing in real world.

Chapter 4

Conclusion and Future Research

4.1 Summary of Work and Contributions

4.1.1 Research Conclusion

In this dissertation, we introduced two kinds of continuum mobile robots with potential of autonomously traversing unpredictable and unstructured environments. The dissertation includes design details, real time simulation, modeling, and motion planning method.

For the modular continuum snake robot, we focus on the design, locomotion, the buckling effect, and the real time simulation. While for the modular continuum salamander robot, we introduce the design, demonstrating experiments, modeling, and motion planning.

Some of the recent research about mobile robots with adaptable flexible characteristics for search and rescue purpose are listed in Appendix section 5.2 as a comparison. Compare to these existing research, our continuum mobile robots are able to perform multiple locomotion mode, and faster than most of the mobile robots with flexible structure, especially for the salamander robot. Also, the salamander robot is untethered, and able to perform sharp turning under control. The suspension provide better potential to adapt to the unstructured environment.

Table 4.1: Comparison between continuum snake robot and salamander robot.

Snake Robot	Salamander Robot
Maximum Velocity: 165.5 mm/s	Maximum Velocity:303.1 mm/s
Maximum Curvature: $0.35 m^{-1}$	Maximum Curvature: $10.52 m^{-1}$
Higher Flexibility	Lower Flexibility
Simpler System	More Complex System
Tethered Power Supply	Untethered Power Supply
Easy to Scale Down	Depends on Circuit Size
Susceptible to Interference	Origami structure provides higher torsional resistance.
Less Electronic Components	More Electronic Components (vulnerable to external environment)
Extra space needed for propulsion (performing snake-like locomotion gait).	Propulsion and deformation are decoupled, thus no extra space needed.

4.1.2 Comparison Between Snake and Salamander Robot

Comparing the two robots, since the salamander robot is more of engineering judgement rather than pure biomimetic, it has better performance on velocity, and steering. Also the electronic system enables the untethered power supply, and the high torsional resistance structure make the system more suitable for modeling. Which makes it suitable for outdoor unstructured environment. The snake robot, however, is a more biomimetic robot with higher flexibility with less electronic components and simpler system the design make it easier to scale down for special situations like surgery or underwater exploration. A detailed comparison is presented in Table 4.1.

4.2 Future Research

4.2.1 Soft Snake Robot

Real Time Simulation with Higher Accuracy

We measured that the simulation speed has increased about 30% recently because of the steadily increasing performance of new GPU hardware. In future work, we will keep exploring the potential

of simulations based on Nvidia Flex, and plan to implement different models for hyperelastic materials to increase the application and accuracy of the real-time simulation for any soft robotic system.

Multiple Function

In addition, future work could include extending our snake-like mobility platforms to operate in the field, addressing challenges in perception, data analysis, power autonomy, and range of motions toward a universal mobility platform useful in real-world applications such as search and rescue in complex environments.

4.2.2 Continuum Salamander Robot

Dynamics Modeling and Controller Design

In order to further understand the continuum mobile robot system we have developed, kinematics and dynamics modeling is a necessary and challenging research work, especially for multi module version salamander robot. Considering the fact that origami structure is light weighted, the robot structure can be approximately simplified as a revolution joint connected mobile units as shown in Figure 4.1. Thus, the dynamics of the multi module salamander robot could be similar to articulated vehicles with longer and light weighted revolution joint connecting much shorter mobile units.

Kinematics and Dynamics modeling can also help to build better controller for autonomous navigation. Related research works explored the dynamics modeling of articulated vehicle (Azadi et al. (2014); Bouteldja and Cerezo (2017)) and the model predictive controller based on it (Nayl et al. (2015); Schoels et al. (2020); Pankert and Hutter (2020)). This can be the research direction of modeling and control research for the salamander robot system in future work.

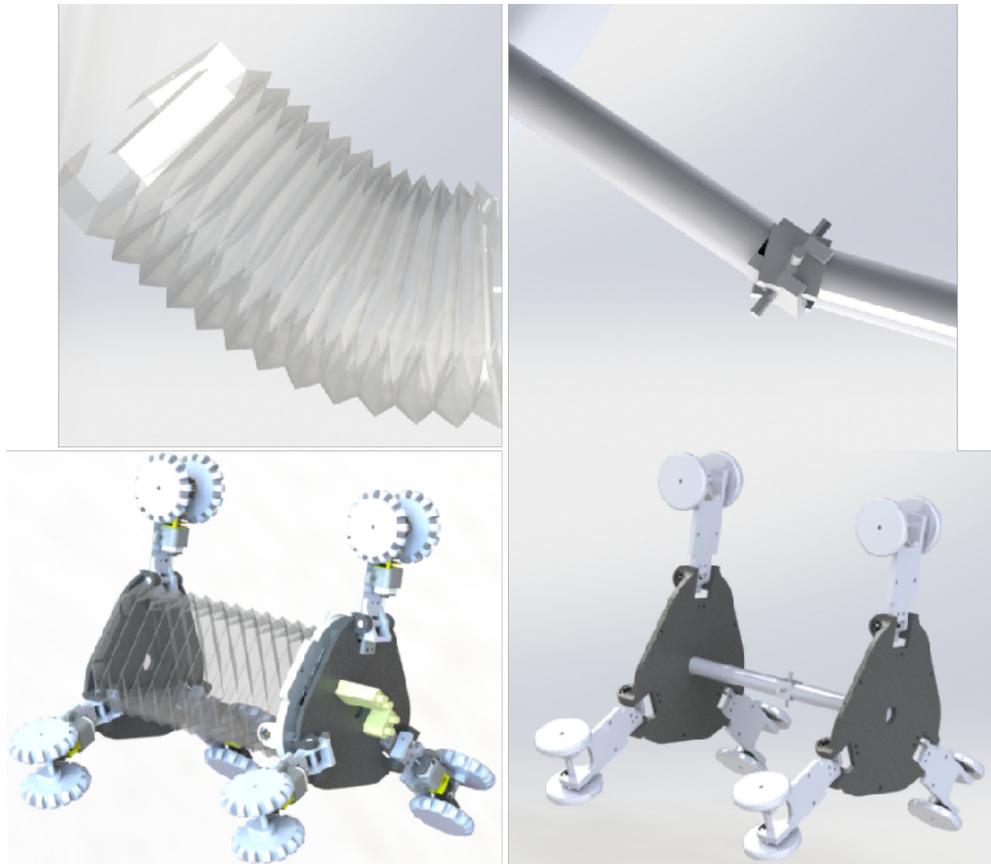


Figure 4.1: CAD model presenting the simplified mechanical structure of the origami module.

Hardware Optimization

Hardware optimization is included for more stable electronic system and better performance. For now the circuit components are highly coupled with each other which makes the circuit repairing much more difficult. Jumping wires are used to connect commercially available modules and our custom PCB boards, which further increase the uncertainty.

In future work, the circuit will be designed in a more distributed architecture. Motor drivers, decoder, and current sensors will be deployed on a smaller scale PCB board for motor driving. Also standard connector will be used in the circuit to get rid of the jumping wires.

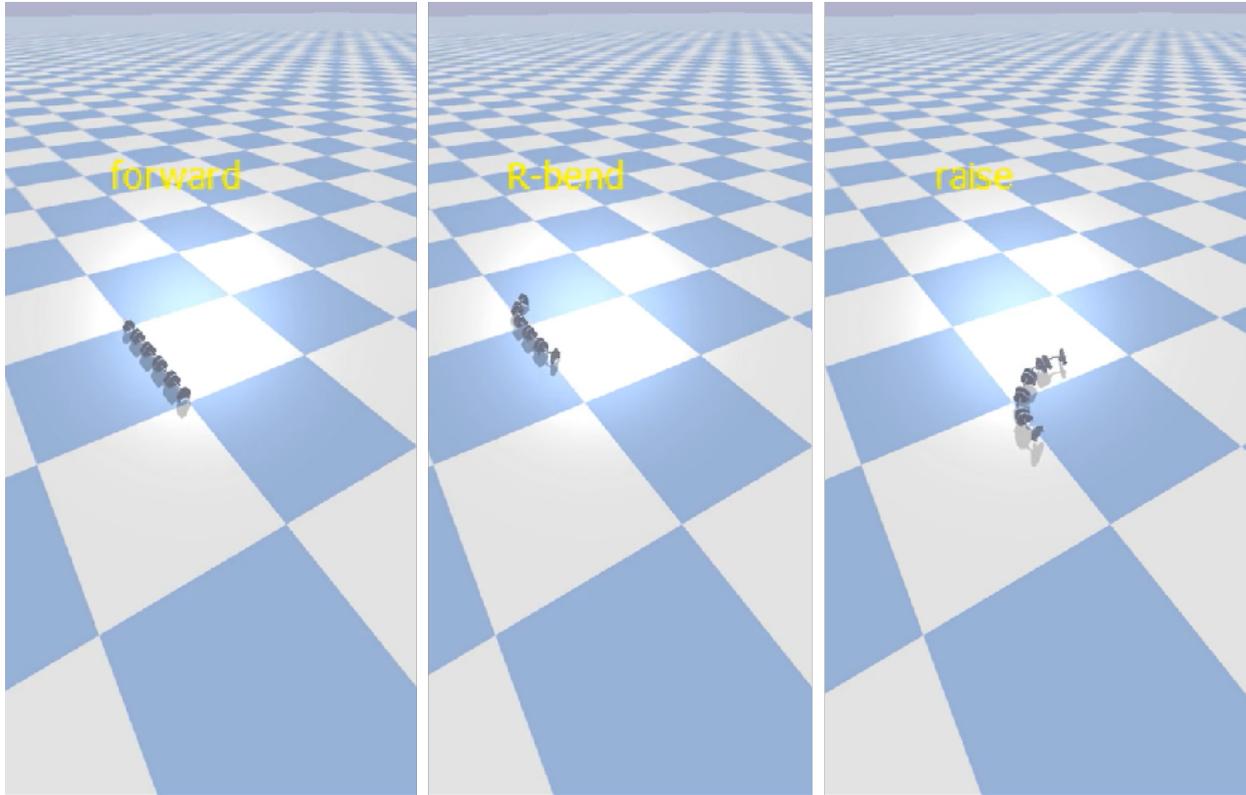


Figure 4.2: Real time simulation of a simplified multi-module salamander robot with Pybullet.

4.2.3 Automatic Gait Generator

During the research work of manipulating multi-module salamander robot climbing over large obstacle, we find it highly inefficient to manually generate traversing locomotion for different terrains. Thus, an automatic gait generator is necessary for the robot traversing unpredictable environments.

A simplified simulation with learning algorithm could be more suitable for autonomous gait generation for different environments than manually tuning the gait, like one of our research work on snake robot locomotion generating utilizing custom Proximal Policy Optimization (PPO) algorithm (Liu et al. (2020)).

For our salamander robot, we selected Pybullet as the simulation environment as shown in Figure 4.2 considering its performance and friendly socket with learning packages like Stable baseline and Pytorch.

Chapter 5

Appendix

5.1 Modular Soft Snake Robot Locomotion Data on Different Surfaces

In this section, experiments are conducted to test the performance of the single module. Sidewinding and undulatory locomotion are tested on four different surfaces (the friction factors of the surfaces are recorded in Table 5.1). Meanwhile, simulation for single module experiments and snake robot locomotion is conducted with our real-time simulator as a comparison.

During the experiments of undulatory locomotion and sidewinding locomotion, the snake is set to move straight forward under different groups of parameters, and the trajectories of soft robotic snake are recorded through a motion capture system. For each group of experiment, speed of soft robotic snake is calculated. Meanwhile, curvature of the trajectory is calculated as a measure of deviation.

5.1.1 Undulatory Locomotion

As mentioned in Section 2.2.2, undulatory locomotion is a classical s-shape movement of snake. With a bidirectional bending in the horizontal plane, the snake thrust its body from side to side, creating a travelling wave, and propel itself forward by using anisotropic friction on its passive

Table 5.1: Sliding and rolling friction factor of different surfaces

	Sliding Friction	Rolling Friction
paper	5.83	0.58
rubber	4.17	0.42
carpet	9.17	1.42
grass	9.17	8.33

wheels. The travelling wave it forms that each segment follows is shown in Equation 2.1 with the notions explained in Table 2.2.

We performed a variety of experiments, ranging the gait frequency ω from 3π to 4π , and the phase difference from $\frac{2}{3}\pi$ to $\frac{5}{6}\pi$, and number of modules used to assemble soft snake robot from 3 to 5 on 4 different surfaces including rubber, paper, carpet, and grass. Different parameter combinations have a large impact on the lateral undulatory locomotion of the soft robotic snake. There are four main factors to be discussed here: number of segments, frequency, phase difference and friction factor.

From the Table 5.2, 5.3, and 5.4, we observed that the number of segments play an important role in the speed of undulation. This is because that, with more segments, the snake can be pushed forward with more power from the extra segments. With more segments, the snake can be pushed forward with more power from the extra segments. As the result, more segments tends to increase the speed of the undulatory locomotion. As shown in Figure 5.1, increasing number of snake module can results in a larger deviation while the speed can be much higher. But considering the speed by length, its effect on speed will be reduced when the number of segments is higher, since the speed per body length decreases as the number of segments increases. Table 2 4 shows snake behavior with different numbers of segments on different surfaces, where we can see that the optimal frequency varies. Higher frequency can reduce the time for each duty cycle, while different structure and surfaces end up with different resistance on the segments, as a result, the segment cannot reach its maximum bending angle within a duty cycle. The varying optimal frequency is the result of the trading off between higher frequency and higher bending angle. Also, higher frequency re-

sults in unpredictable trajectory because errors in each duty cycle accumulate, which result in larger errors after certain period of time. In Figure 5.2, the lateral undulatory locomotion is performed

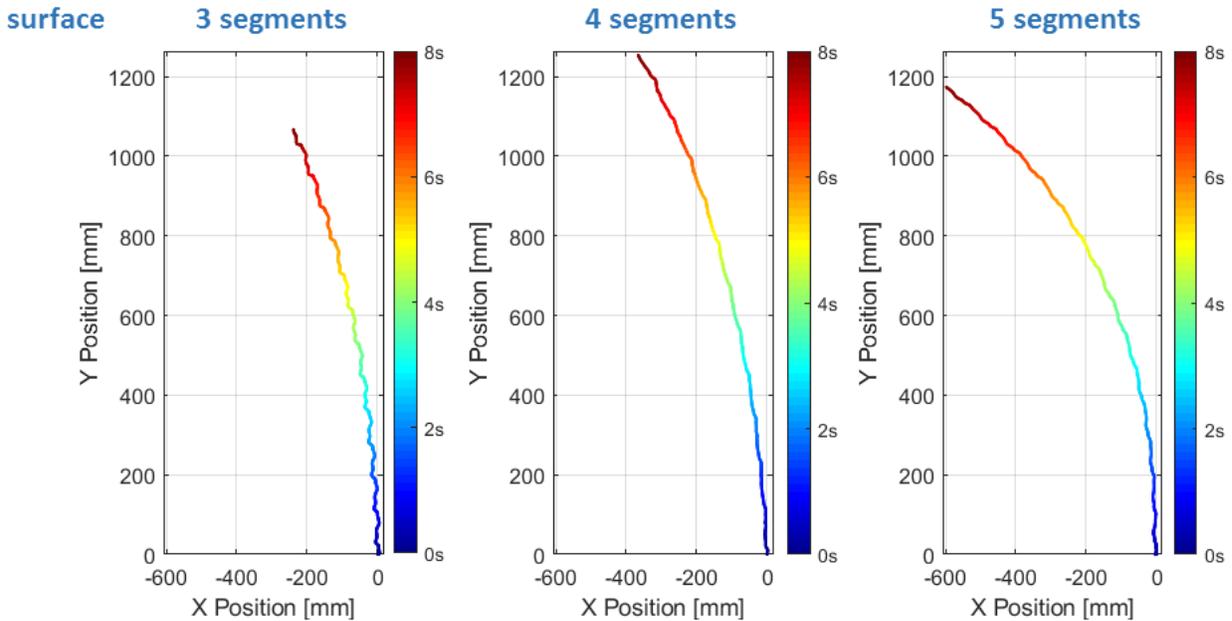


Figure 5.1: Trajectories highlighting the maximum undulation speed (in body-lengths per second) for each number of modules in the Snake. Different frequencies, offsets, and surface materials were tested, and the fastest shown here. For three segment robot, we choose the parameter: rubber surface, 1.75 Hz frequency, $\frac{2}{3}\pi$ phase difference; for the four segment robot, we choose the parameter: rubber surface, 1.50 Hz frequency, $\frac{2}{3}\pi$ phase difference; for the five segment robot, we choose the parameter: paper surface, 2.00 Hz frequency, $\frac{3}{4}\pi$ phase difference.

on a surface with low frequency, which makes the actuator stretch much faster. As a result, higher frequency usually results in higher speed since the segment can always reach its maximum bending angle. gait is supposed to move along positive Y axis, while the deviation increase compared to the lower frequency locomotion under same phase difference on same surface. In real time simulation result shown in Figure 5.2, although the exact result is not the same, it shows the similar trend. With higher frequency, the robot tends to have higher speed and deviation.

The next factor to talk about is the phase difference β in Equation 2.1, which represents the difference in phase between a segment and the adjacent segments. Compared with the number of segments and the surface, phase difference has a smaller influence on the speed of the undulation within acceptable range for this gait. However, when the phase difference is not in appropriate

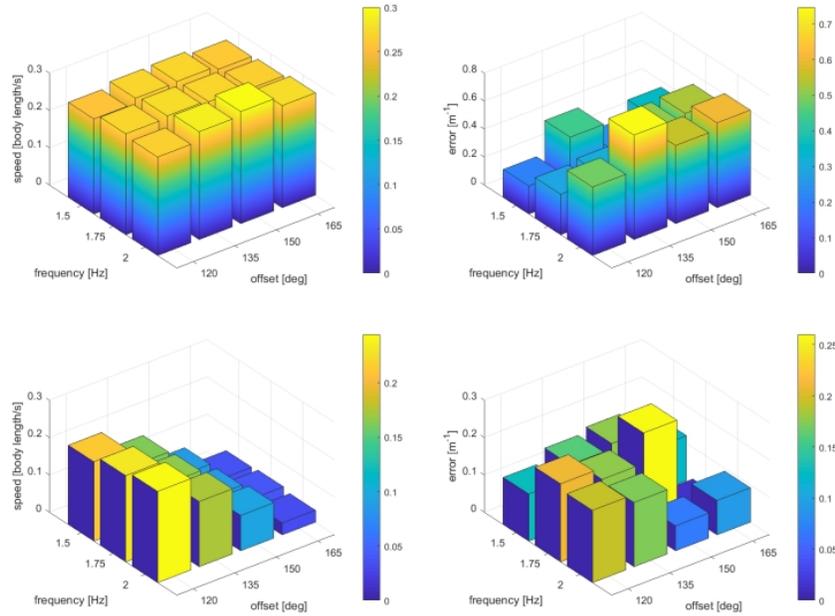


Figure 5.2: Speed and deviation measured with curvature of the trajectory for five-segment soft snakes performing undulatory locomotion on a paper surface with different undulation frequencies and segment offsets tested in real world experiments and simulation. (top-left): Speed in experiments; (top-right): Curvature of the trajectory in experiment; (bottom-left): Speed in simulation; (bottom-right): Curvature of the trajectory in simulation.

range, the snake will behave much worse. With phase difference below $\frac{1}{2}\pi$, the speed becomes quite slow, and the trajectory is not like a lateral undulation. When moving to higher phase difference, the velocity and trajectory goes normal. So we experiment from $\frac{2}{3}\pi$ to $\frac{11}{12}\pi$ with an interval of $\frac{1}{12}\pi$ to find the optimal parameters which is around $\frac{2}{3}\pi$ to $\frac{3}{4}\pi$. Friction factor also have an influence on the trajectory. As shown in Table 5.1, paper and rubber surfaces have the lowest friction factor, while carpet and grass surface has the same sliding friction factor but different rolling friction factors, with carpet's being larger than grass, which results in the difference as shown in Figure 5.3 that lower friction result in higher speed.

5.1.2 Sidewinding Locomotion

As mentioned in Section 2.2.2, undulatory locomotion is a variation of a snake's serpentine motion that makes use of three degrees-of-freedom of each segment. It physically lifts part of its body off

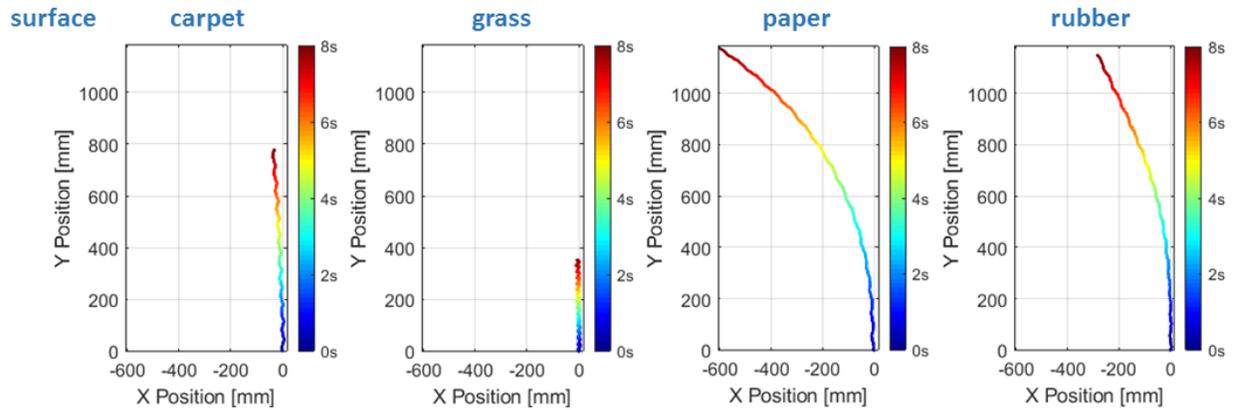


Figure 5.3: Trajectories of the snake performing undulatory locomotion on different surfaces. For each surface chosen, we choose the parameters which maximized the speed. On grass, we choose parameters: 2.00 Hz frequency, $\frac{3}{4}\pi$ phase difference; on carpet, we choose parameter: 1.50 Hz frequency, $\frac{5}{6}\pi$ phase difference; on paper, we choose parameter: 2.00 Hz frequency, $\frac{3}{4}\pi$ phase difference; on rubber, we choose parameter: 2.00 Hz frequency, $\frac{2}{3}\pi$ phase difference.

Table 5.2: Undulation locomotion velocity by body length per second (BL/s), 3 segments (total length 404 mm). The fastest result is marked red.

Gait Frequency ($\frac{2\pi}{\omega}$)	Phase Difference (β)	Velocity on Rubber	Velocity on Paper	Velocity on Carpet	Velocity on Grass
1.50 Hz	$\frac{2}{3}\pi$	0.294 BL/s	0.317 BL/s	0.176 BL/s	0.040 BL/s
1.50 Hz	$\frac{3}{4}\pi$	0.300 BL/s	0.308 BL/s	0.172 BL/s	0.041 BL/s
1.50 Hz	$\frac{5}{6}\pi$	0.301 BL/s	0.317 BL/s	0.173 BL/s	0.043 BL/s
1.50 Hz	$\frac{11}{12}\pi$	0.300 BL/s	0.303 BL/s	0.186 BL/s	0.039 BL/s
1.75 Hz	$\frac{2}{3}\pi$	0.333 BL/s	0.319 BL/s	0.165 BL/s	0.057 BL/s
1.75 Hz	$\frac{3}{4}\pi$	0.326 BL/s	0.330 BL/s	0.174 BL/s	0.056 BL/s
1.75 Hz	$\frac{5}{6}\pi$	0.314 BL/s	0.322 BL/s	0.189 BL/s	0.049 BL/s
1.75 Hz	$\frac{11}{12}\pi$	0.313 BL/s	0.296 BL/s	0.182 BL/s	0.068 BL/s
2.00 Hz	$\frac{2}{3}\pi$	0.317 BL/s	0.268 BL/s	0.036 BL/s	0.075 BL/s
2.00 Hz	$\frac{3}{4}\pi$	0.303 BL/s	0.265 BL/s	0.014 BL/s	0.091 BL/s
2.00 Hz	$\frac{5}{6}\pi$	0.308 BL/s	0.249 BL/s	0.020 BL/s	0.085 BL/s
2.00 Hz	$\frac{11}{12}\pi$	0.302 BL/s	0.251 BL/s	0.036 BL/s	0.087 BL/s

Table 5.3: Undulation locomotion velocity by body length per second (BL/s), 4 segments (total length 510 mm). The fastest result is marked red.

Gait Frequency ($\frac{2\pi}{\omega}$)	Phase Difference (β)	Velocity on Rubber	Velocity on Paper	Velocity on Carpet	Velocity on Grass
1.50 Hz	$\frac{2}{3}\pi$	0.308 BL/s	0.264 BL/s	0.115 BL/s	0.057 BL/s
1.50 Hz	$\frac{3}{4}\pi$	0.293 BL/s	0.274 BL/s	0.113 BL/s	0.070 BL/s
1.50 Hz	$\frac{5}{6}\pi$	0.306 BL/s	0.257 BL/s	0.114 BL/s	0.075 BL/s
1.50 Hz	$\frac{11}{12}\pi$	0.305 BL/s	0.277 BL/s	0.116 BL/s	0.075 BL/s
1.75 Hz	$\frac{2}{3}\pi$	0.289 BL/s	0.256 BL/s	0.106 BL/s	0.078 BL/s
1.75 Hz	$\frac{3}{4}\pi$	0.299 BL/s	0.259 BL/s	0.100 BL/s	0.071 BL/s
1.75 Hz	$\frac{5}{6}\pi$	0.282 BL/s	0.273 BL/s	0.119 BL/s	0.075 BL/s
1.75 Hz	$\frac{11}{12}\pi$	0.272 BL/s	0.280 BL/s	0.137 BL/s	0.074 BL/s
2.00 Hz	$\frac{2}{3}\pi$	0.280 BL/s	0.275 BL/s	0.124 BL/s	0.098 BL/s
2.00 Hz	$\frac{3}{4}\pi$	0.282 BL/s	0.271 BL/s	0.123 BL/s	0.096 BL/s
2.00 Hz	$\frac{5}{6}\pi$	0.279 BL/s	0.254 BL/s	0.124 BL/s	0.100 BL/s
2.00 Hz	$\frac{11}{12}\pi$	0.300 BL/s	0.243 BL/s	0.123 BL/s	0.107 BL/s

the ground, giving the body an S-shape with two points of contact with the ground. These contact points serve to push the snake in a constant direction. In this gait, the snake as a whole moves in a diagonal motion which behaves helical shape from head to tail. This gait has the advantage of high efficiency because of the fewer frictional losses associated with serpentine locomotion. To realize this locomotion in the soft robotic snake, the travelling wave it forms that each segment follows is shown in Equation 2.2 with the notions explained in Table 2.2.

A series of experiments, ranging ω from $\frac{3}{2}\pi$ to 2π , ranging phase difference β from $\frac{1}{3}\pi$ to $\frac{2}{3}\pi$, and number of modules applied to assemble the soft snake robot from 3 to 5 on four different surfaces are conducted, as shown in Table 5.5, Table 5.6, and Table 5.4. Different parameter combinations also have a large impact on the sidewinding movement of the soft robotic snake. As shown in Table 5.5, Table 5.6, and Table 5.4, the number of segments plays an important role in the speed of sidewinding. Similar to Figure 5.1, Figure 5.4 shows the fastest sidewinding trajectory for snakes of differing numbers of segments. During sidewinding locomotion, the snake itself need

Table 5.4: Undulation locomotion velocity by body length per second (BL/s), 5 segments (total length 616 mm). The fastest result is marked red.

Gait Frequency ($\frac{2\pi}{\omega}$)	Phase Difference (β)	Velocity on Rubber	Velocity on Paper	Velocity on Carpet	Velocity on Grass
1.50 Hz	$\frac{2}{3}\pi$	0.218 BL/s	0.228 BL/s	0.148 BL/s	0.054 BL/s
1.50 Hz	$\frac{3}{4}\pi$	0.224 BL/s	0.247 BL/s	0.142 BL/s	0.049 BL/s
1.50 Hz	$\frac{5}{6}\pi$	0.236 BL/s	0.247 BL/s	0.152 BL/s	0.048 BL/s
1.50 Hz	$\frac{11}{12}\pi$	0.224 BL/s	0.232 BL/s	0.142 BL/s	0.051 BL/s
1.75 Hz	$\frac{2}{3}\pi$	0.232 BL/s	0.251 BL/s	0.133 BL/s	0.052 BL/s
1.75 Hz	$\frac{3}{4}\pi$	0.234 BL/s	0.258 BL/s	0.138 BL/s	0.051 BL/s
1.75 Hz	$\frac{5}{6}\pi$	0.225 BL/s	0.253 BL/s	0.127 BL/s	0.051 BL/s
1.75 Hz	$\frac{11}{12}\pi$	0.226 BL/s	0.242 BL/s	0.116 BL/s	0.058 BL/s
2.00 Hz	$\frac{2}{3}\pi$	0.241 BL/s	0.226 BL/s	0.128 BL/s	0.056 BL/s
2.00 Hz	$\frac{3}{4}\pi$	0.238 BL/s	0.269 BL/s	0.131 BL/s	0.068 BL/s
2.00 Hz	$\frac{5}{6}\pi$	0.236 BL/s	0.255 BL/s	0.134 BL/s	0.062 BL/s
2.00 Hz	$\frac{11}{12}\pi$	0.240 BL/s	0.253 BL/s	0.137 BL/s	0.066 BL/s

to lift its body to move, meaning that snake weight is more of a factor than in undulation. We found that maximum speeds occurred at medium frequencies. Due to the slow response of the soft actuator, especially when lifting its body up in the air, a lower frequency allowed the snake to more fully complete the sidewinding motion. However, it can't be too low as a low frequency results in a lower speed.

In finding the optimal phase delay, we experimented with delays from $\frac{1}{3}\pi$ to $\frac{5}{6}\pi$ with an interval of $\frac{1}{6}\pi$ to find the optimal parameter, and it is around $\frac{1}{2}\pi$. As shown in Figure 5.5, both the experimental result and simulation result shows that the optimal phase delay for highest speed is around 60 to 90 degrees, and the speed decreases as the phase delay becomes larger. For deviation, however, there is no obvious rules we can observe. Another factor that influenced the speed a lot is the test surface selection, which have different friction. The results in Figure 5.6 show that the friction of the surface does not affect the speed for sidewinding as much as in undulation gait because the movement does not depend on the rolling friction (and its difference with the sliding

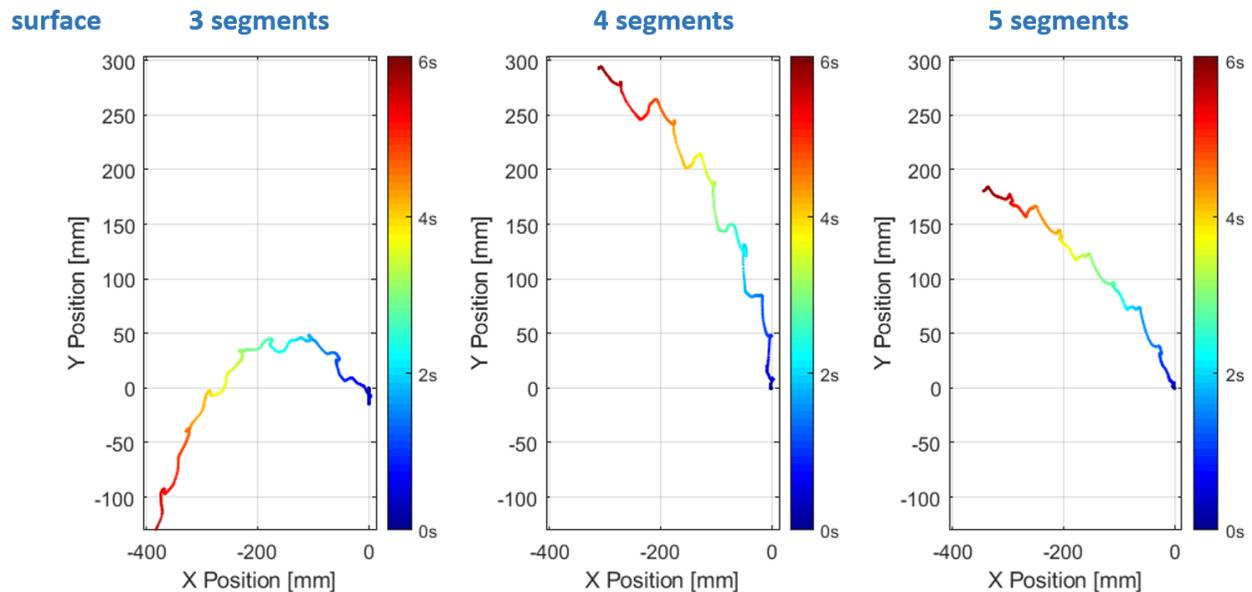


Figure 5.4: Trajectories highlighting the maximum undulation speed (in body-lengths per second) for each number of modules in the Snake. Different frequencies, offsets, and surface materials were tested, and the fastest shown here. For three segment robot, we choose the parameter: rubber surface, 0.875 Hz frequency, $\frac{1}{3}\pi$ phase difference; for the four segment robot, we choose the parameter: rubber surface, 1.00 Hz frequency, $\frac{1}{2}\pi$ phase difference; for the five segment robot, we choose the parameter: rubber surface, 0.875 Hz frequency, $\frac{1}{2}\pi$ phase difference.

friction). However, lower friction can result in more deviation, which is the same as with the lateral undulation gait.

5.2 Recent Research about Flexible Mobile Robots

Mobile robot with flexible structure are developed utilizing inchworm locomotion, snake-like locomotion, and other custom locomotion. Here we list some basic information including of these research in Table 5.8. For each research, the actuation method, maximum velocity, power autonomy, controller, and locomotion gait is recorded and compared. For more detailed description, please check our previously published review paper (Sun et al. (2021)).

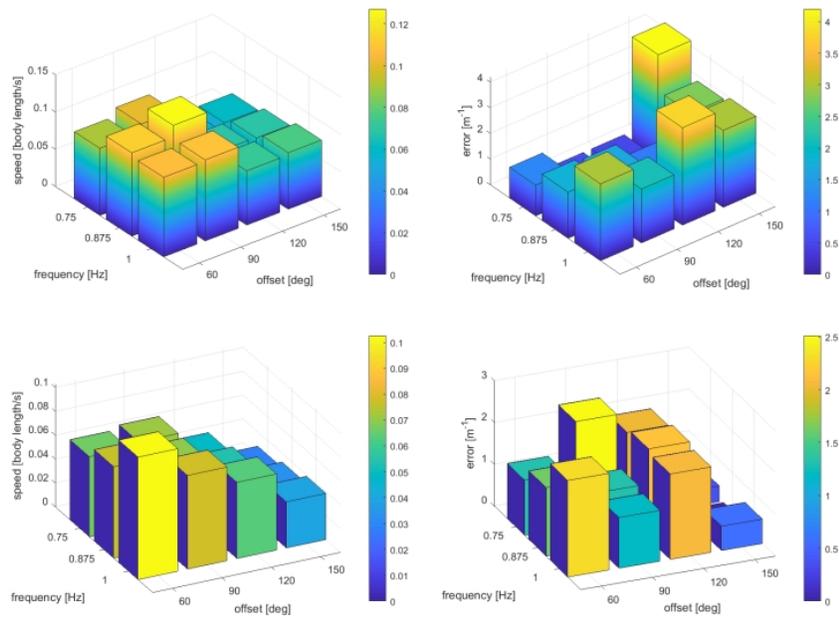


Figure 5.5: Speed and deviation measured with curvature of the trajectory for five-segment soft snakes performing sidewinding locomotion on a paper surface with different undulation frequencies and segment offsets tested in real world experiments and simulation. (top-left): Speed in experiments; (top-right): Curvature of the trajectory in experiment; (bottom-left): Speed in simulation; (bottom-right): Curvature of the trajectory in simulation.

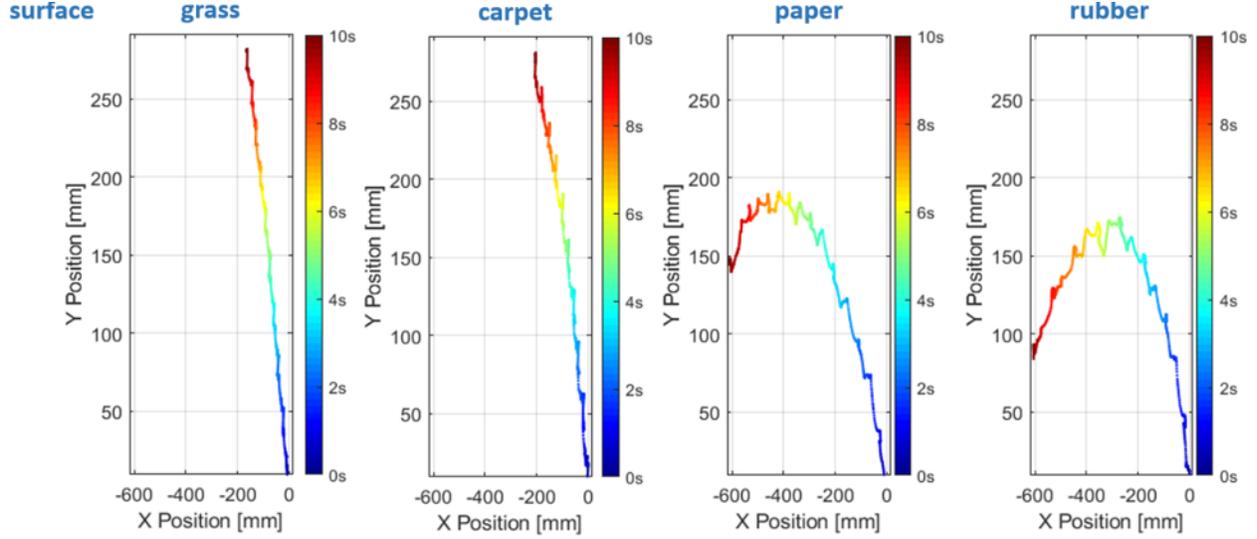


Figure 5.6: Trajectories of the snake performing sidewinding locomotion on different surfaces. For each surface chosen, we choose the parameters which maximized the speed. On grass, we choose parameters: 1.00 Hz frequency, $\frac{5}{6}\pi$ phase difference; on carpet, we choose parameter: 1.00 Hz frequency, $\frac{5}{6}\pi$ phase difference; on paper, we choose parameter: 0.875 Hz frequency, $\frac{1}{2}\pi$ phase difference; on rubber, we choose parameter: 0.875 Hz frequency, $\frac{1}{2}\pi$ phase difference.

Table 5.5: Sidewinding locomotion velocity by body length per second (BL/s), 3 segments (total length 404 mm). The fastest result is marked red.

Gait Frequency ($\frac{2\pi}{\omega}$)	Phase Difference (β)	Velocity on Rubber	Velocity on Paper	Velocity on Carpet	Velocity on Grass
1.000 Hz	$\frac{1}{3}\pi$	0.169 BL/s	0.155 BL/s	0.088 BL/s	0.093 BL/s
1.000 Hz	$\frac{1}{2}\pi$	0.183 BL/s	0.168 BL/s	0.087 BL/s	0.087 BL/s
1.000 Hz	$\frac{2}{3}\pi$	0.084 BL/s	0.085 BL/s	0.056 BL/s	0.052 BL/s
1.000 Hz	$\frac{5}{6}\pi$	0.086 BL/s	0.090 BL/s	0.059 BL/s	0.052 BL/s
0.750 Hz	$\frac{1}{3}\pi$	0.179 BL/s	0.148 BL/s	0.065 BL/s	0.065 BL/s
0.750 Hz	$\frac{1}{2}\pi$	0.182 BL/s	0.143 BL/s	0.060 BL/s	0.057 BL/s
0.750 Hz	$\frac{2}{3}\pi$	0.062 BL/s	0.063 BL/s	0.041 BL/s	0.036 BL/s
0.750 Hz	$\frac{5}{6}\pi$	0.064 BL/s	0.067 BL/s	0.041 BL/s	0.040 BL/s
0.875 Hz	$\frac{1}{3}\pi$	0.210 BL/s	0.184 BL/s	0.073 BL/s	0.097 BL/s
0.875 Hz	$\frac{1}{2}\pi$	0.184 BL/s	0.179 BL/s	0.069 BL/s	0.095 BL/s
0.875 Hz	$\frac{2}{3}\pi$	0.073 BL/s	0.076 BL/s	0.052 BL/s	0.047 BL/s
0.875 Hz	$\frac{5}{6}\pi$	0.079 BL/s	0.079 BL/s	0.057 BL/s	0.043 BL/s

Table 5.6: Sidewinding locomotion velocity by body length per second (BL/s), 4 segments (total length 510 mm). The fastest result is marked red.

Gait Frequency ($\frac{2\pi}{\omega}$)	Phase Difference (β)	Velocity on Rubber	Velocity on Paper	Velocity on Carpet	Velocity on Grass
1.000 Hz	$\frac{1}{3}\pi$	0.154 BL/s	0.139 BL/s	0.065 BL/s	0.072 BL/s
1.000 Hz	$\frac{1}{2}\pi$	0.158 BL/s	0.144 BL/s	0.059 BL/s	0.070 BL/s
1.000 Hz	$\frac{2}{3}\pi$	0.088 BL/s	0.089 BL/s	0.051 BL/s	0.041 BL/s
1.000 Hz	$\frac{5}{6}\pi$	0.084 BL/s	0.092 BL/s	0.046 BL/s	0.039 BL/s
0.750 Hz	$\frac{1}{3}\pi$	0.104 BL/s	0.105 BL/s	0.041 BL/s	0.069 BL/s
0.750 Hz	$\frac{1}{2}\pi$	0.099 BL/s	0.080 BL/s	0.042 BL/s	0.065 BL/s
0.750 Hz	$\frac{2}{3}\pi$	0.058 BL/s	0.064 BL/s	0.041 BL/s	0.037 BL/s
0.750 Hz	$\frac{5}{6}\pi$	0.045 BL/s	0.062 BL/s	0.040 BL/s	0.039 BL/s
0.875 Hz	$\frac{1}{3}\pi$	0.0123 BL/s	0.134 BL/s	0.049 BL/s	0.073 BL/s
0.875 Hz	$\frac{1}{2}\pi$	0.084 BL/s	0.150 BL/s	0.048 BL/s	0.085 BL/s
0.875 Hz	$\frac{2}{3}\pi$	0.079 BL/s	0.077 BL/s	0.042 BL/s	0.040 BL/s
0.875 Hz	$\frac{5}{6}\pi$	0.068 BL/s	0.077 BL/s	0.047 BL/s	0.033 BL/s

Table 5.7: Sidewinding locomotion velocity by body length per second (BL/s), 5 segments (total length 616 mm). The fastest result is marked red.

Gait Frequency ($\frac{2\pi}{\omega}$)	Phase Difference (β)	Velocity on Rubber	Velocity on Paper	Velocity on Carpet	Velocity on Grass
1.000 Hz	$\frac{1}{3}\pi$	0.105 BL/s	0.094 BL/s	0.036 BL/s	0.039 BL/s
1.000 Hz	$\frac{1}{2}\pi$	0.082 BL/s	0.081 BL/s	0.037 BL/s	0.039 BL/s
1.000 Hz	$\frac{2}{3}\pi$	0.071 BL/s	0.073 BL/s	0.057 BL/s	0.042 BL/s
1.000 Hz	$\frac{5}{6}\pi$	0.068 BL/s	0.074 BL/s	0.058 BL/s	0.048 BL/s
0.750 Hz	$\frac{1}{3}\pi$	0.084 BL/s	0.078 BL/s	0.025 BL/s	0.028 BL/s
0.750 Hz	$\frac{1}{2}\pi$	0.091 BL/s	0.087 BL/s	0.028 BL/s	0.036 BL/s
0.750 Hz	$\frac{2}{3}\pi$	0.057 BL/s	0.057 BL/s	0.045 BL/s	0.031 BL/s
0.750 Hz	$\frac{5}{6}\pi$	0.054 BL/s	0.056 BL/s	0.049 BL/s	0.031 BL/s
0.875 Hz	$\frac{1}{3}\pi$	0.099 BL/s	0.093 BL/s	0.023 BL/s	0.034 BL/s
0.875 Hz	$\frac{1}{2}\pi$	0.106 BL/s	0.105 BL/s	0.034 BL/s	0.043 BL/s
0.875 Hz	$\frac{2}{3}\pi$	0.072 BL/s	0.072 BL/s	0.054 BL/s	0.044 BL/s
0.875 Hz	$\frac{5}{6}\pi$	0.069 BL/s	0.068 BL/s	0.052 BL/s	0.045 BL/s

Table 5.8: Taxonomy of recent flexible mobile robot research.

Research	Actuation	Speed	Power Autonomy	Control	Locomotion Gait
Resilientbot (Tolley et al. (2014))	Pneumatic	0.5 mm/s	Untethered	Open loop	Ambulatory
Shepherd2011 (Shepherd et al. (2011))	Pneumatic	/	Tethered	Open loop	ambulatory
Duggan2019 (Duggan et al. (2019))	Pneumatic	0.28 mm/s	Untethered	Open loop	Inchworm
Kirigami (Liu et al. (2019))	Pneumatic	33.33 mm/s	Tethered	Open loop	Inchworm
Das2020 (Das et al. (2020))	Pneumatic	26.2 mm/s	Tethered	Open loop	Inchworm
Wu2018 (Wu et al. (2018))	Pneumatic	9.85 mm/s	Tethered	Close loop	Inchworm
Zou2018 (Zou et al. (2018))	Pneumatic	5.14 mm/s	Tethered	Open loop	Inchworm
Tang2018 (Tang et al. (2018))	Pneumatic	4.77 mm/s	Tethered	Open loop	Inchworm
Qin2019 (Qin et al. (2019))	Pneumatic	16.29 mm/s 0.12 BL/s	Tethered	Open loop	Inchworm
Booth2018 (Booth et al. (2018))	Pneumatic	25 mm/s	Tethered	Open loop	Inchworm
Rothemund2018 (Rothemund et al. (2018))	Pneumatic	1.4 mm/s	Tethered	Open loop	Inchworm
Kim2019 (Kim et al. (2019))	Pneumatic	/	Tethered	Open loop	Inchworm
Singh2019 (Singh et al. (2019))	Pneumatic	4.2 mm/s	Tethered	Open loop	Inchworm
Xie2018 (Xie et al. (2018))	Pneumatic	6.67 mm/s	Tethered	Close loop	Inchworm
Rafsanjani2018 (Rafsanjani et al. (2018))	Pneumatic	/	Tethered	Open loop	Inchworm
Ito2019 (Ito et al. (2019))	Pneumatic	17.5 mm/s	Tethered	Open loop	Inchworm
Luo2018 (Luo et al. (2018b))	Pneumatic	/	Tethered	Open loop	Inchworm
Henke2017 (Henke et al. (2017))	DEA	0.83 mm/s	Tethered	Open loop	Inchworm
Li2019 (Li et al. (2019))	DEA	161 mm/s 4 BL/s	Untethered	Open loop	Inchworm
Guo2020 (Guo et al. (2020))	DEA	0.005 mm/s	Tethered	Open loop	Inchworm
Jung2007 (Jung et al. (2007))	DEA	2.5 mm/s	Tethered	Open loop	Inchworm
Gu2018 (Gu et al. (2018))	DEA	88.46 mm/s 1.04 BL/s	Tethered	Open loop	Inchworm
Goldberg2019 (Goldberg et al. (2019))	SMA	10 mm/s	Untethered	Open loop	Inchworm
Huang2018 (Huang et al. (2018))	SMA	74 mm/s	Tethered	Open loop	Inchworm
Trimmer2012 (Trimmer et al. (2012))	SMA	0.5 mm/s	Tethered	Open loop	Inchworm
Seok2012 (Seok et al. (2012))	SMA	5.25 mm/s	Untethered	ILC	Inchworm
Menciassi2004 (Menciassi et al. (2004))	SMA	0.22 mm/s	Tethered	Open loop	Inchworm
Yuk2011 (Yuk et al. (2011))	SMA	0.21 mm/s	Tethered	Open loop	Inchworm
Umedachi2018 (Umedachi and Kawahara (2018))	SMA	4 mm/s	Tethered	Open loop	Inchworm
Umedachi2018 (Umedachi and Kawahara (2018))	SMA	4 mm/s	Tethered	Open loop	Inchworm
Onal2012 (Onal et al. (2012))	SMA	0.31 mm/s	Untethered	Open loop	Inchworm
Umedachi2016 (Umedachi et al. (2016))	Cable-driven	112 mm/s 0.56 BL/s	Tethered	Open loop	Inchworm
Manwell2018 (Manwell et al. (2018))	Cable-driven	1.43 mm/s	Tethered	Open loop	Inchworm
Ta2018 (Ta et al. (2018))	Motor	17.5 mm/s	Tethered	Open loop	Inchworm
Horchler2015 (Horchler et al. (2015))	Motor	4.3 mm/s	Tethered	Open loop	Inchworm
Umedachi2019 (Umedachi et al. (2019b))	Motor	/	Tethered	Model based PD	Inchworm
Shin2018 (Shin et al. (2018))	Hygroscopical	6 mm/s 0.24 BL/s	Untethered	Open loop	Inchworm
Electromag-worm (Nemitz et al. (2016))	Electromagnetic	/	Tethered & Untethered	Open loop	Inchworm
Joyce2019 (Joyce and Pan (2019))	Magnetic	26.2 mm/s	Tethered	Open loop	Inchworm
Hybrid Starfish (Stokes et al. (2014))	Pneumatic	1.8 mm/s 0.02 BL/s	Tethered	Open loop	Starfish
Lee2020 (Lee et al. (2020))	Cable-driven	/	Untethered	Open loop	Starfish
Mao2013 (Mao et al. (2013))	SMA	/	Tethered	Open loop	Starfish
Snake (Luo et al. (2014))	Pneumatic	40 mm/s	Tethered	Open loop	Snake
Onal2013 (Onal and Rus (2013))	Pneumatic	19 mm/s	Untethered	Open loop	Snake
Jia2020 (Jia and Ma (2020))	Pneumatic	/	Tethered	Bayesian	Snake
Branyan2018 (Branyan and Mençüç (2018))	Pneumatic	2.2 mm/s	Tethered	Open loop	Snake
3Dsnake (Qin et al. (2018))	Pneumatic	131.6 mm/s	Tethered	Open loop	Snake
Liao2020 (Liao et al. (2020))	Pneumatic	30.85 mm/s 0.19 BL/s	Tethered	Open loop	Snake climbing
Calisti2011 (Calisti et al. (2011))	Cable-driven	3.99 mm/s	Tethered	Open loop	Octopus
Verma2018 (Verma et al. (2018))	Pneumatic	4 mm/s	Tethered	Open loop	Custom
Rieffel2018 (Rieffel and Mouret (2018))	Vibration Motor	150 mm/s 1.15 BL/s	Tethered	Learning Algorithm	Custom
Kanada2019 (Kanada et al. (2019))	Motor	20 mm/s	Untethered	Open loop	Custom
Usevitch2020 (Usevitch et al. (2020))	Motor	60 mm/s 0.04 BL/s	Untethered	PID	Custom
Yamada2018 (Yamada and Nakamura (2018))	Motor	1000 mm/s	Untethered	Open loop	Custom
Digumarti2018 (Digumarti et al. (2018))	DEA	12 mm/s	Tethered	Open loop	Custom
Wang2019 (Wang et al. (2019))	Pneumatic	60 mm/s	Tethered	Open loop	Custom
Umedachi2019 (Umedachi et al. (2019a))	Cable-driven	20 mm/s	Tethered	Open loop	Custom

Bibliography

- Arachchige, D. D., Chen, Y., and Godage, I. S.: Soft Robotic Snake Locomotion: Modeling and Experimental Assessment, in: 2021 IEEE 17th International Conference on Automation Science and Engineering (CASE), pp. 805–810, IEEE, 2021.
- Azadi, S., Jafari, A., and Samadian, M.: Effect of parameters on roll dynamic response of an articulated vehicle carrying liquids, *Journal of Mechanical Science and Technology*, 28, 837–848, 2014.
- Backman, A., Bodin, K., Lacoursière, C., et al.: Democratizing CAE with Interactive Multiphysics Simulation and Simulators, in: NAFEMS NORDIC Conference: Engineering Simulation: Best Practices, New Developments, Future Trends, 22-23 May 2012, Gothenburg, Sweden, 2012.
- Banerjee, H., Pusalkar, N., and Ren, H.: Single-motor controlled tendon-driven peristaltic soft origami robot, *Journal of Mechanisms and Robotics*, 10, 064 501, 2018.
- Booth, J. W., Shah, D., Case, J. C., White, E. L., Yuen, M. C., Cyr-Choiniere, O., and Kramer-Bottiglio, R.: OmniSkins: Robotic Skins That Turn Inanimate Objects into Multifunctional Robots, *Science Robotics*, 3, <https://doi.org/10.1126/scirobotics.aat1853>, 2018.
- Bouteldja, M. and Cerezo, V.: Low rate error modeling of articulated heavy vehicle dynamics and experimental validation, *International Journal of Control, Automation and Systems*, 15, 2203–2212, 2017.

- Branyan, C. and Mengüç, Y.: Soft snake robots: Investigating the effects of gait parameters on locomotion in complex terrains, in: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1–9, IEEE, 2018.
- Branyan, C., Hatton, R. L., and Mengüç, Y.: Snake-inspired kirigami skin for lateral undulation of a soft snake robot, *IEEE Robotics and Automation Letters*, 5, 1728–1733, 2020.
- Byravan, A., Boots, B., Srinivasa, S. S., and Fox, D.: Space-time functional gradient optimization for motion planning, in: 2014 IEEE International Conference on Robotics and Automation (ICRA), pp. 6499–6506, IEEE, 2014.
- Calisti, M., Giorelli, M., Levy, G., Mazzolai, B., Hochner, B., Laschi, C., and Dario, P.: An octopus-bioinspired solution to movement and manipulation for soft robots, *Bioinspiration & biomimetics*, 6, 036 002, 2011.
- Coad, M. M., Blumenschein, L. H., Cutler, S., Reyna Zepeda, J. A., Naclerio, N. D., El-Hussieny, H., Mehmood, U., Ryu, J.-H., Hawkes, E. W., and Okamura, A. M.: Vine Robots: Design, Teleoperation, and Deployment for Navigation and Exploration, *IEEE Robotics Automation Magazine*, 27, 120–132, <https://doi.org/10.1109/MRA.2019.2947538>, conference Name: IEEE Robotics Automation Magazine, 2020.
- Crespi, A. and Ijspeert, A. J.: 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), CONF, pp. 19–27, 2006.
- Das, R., Babu, S. P. M., Palagi, S., and Mazzolai, B.: Soft Robotic Locomotion by Peristaltic Waves in Granular Media, in: 2020 3rd IEEE International Conference on Soft Robotics (RoboSoft), pp. 223–228, IEEE, 2020.
- Digumarti, K. M., Cao, C., Guo, J., Conn, A. T., and Rossiter, J.: Multi-directional crawling robot with soft actuators and electroadhesive grippers, in: 2018 IEEE International Conference on Soft Robotics (RoboSoft), pp. 303–308, IEEE, 2018.

- Dong, X., Palmer, D., Axinte, D., and Kell, J.: In-situ repair/maintenance with a continuum robotic machine tool in confined space, *Journal of Manufacturing Processes*, 38, 313–318, 2019.
- Duggan, T., Horowitz, L., Ulug, A., Baker, E., and Petersen, K.: Inchworm-inspired locomotion in untethered soft robots, in: 2019 2nd IEEE International Conference on Soft Robotics (RoboSoft), pp. 200–205, IEEE, 2019.
- Duriez, C.: Control of elastic soft robots based on real-time finite element method, in: 2013 IEEE international conference on robotics and automation, pp. 3982–3987, IEEE, 2013.
- Fathi, J., Oude Vrielink, T. J., Runciman, M. S., and Mylonas, G. P.: A Deployable Soft Robotic Arm with Stiffness Modulation for Assistive Living Applications, in: 2019 International Conference on Robotics and Automation (ICRA), pp. 1479–1485, <https://doi.org/10.1109/ICRA.2019.8793670>, iSSN: 2577-087X, 2019.
- Fossen, T. I., Breivik, M., and Skjetne, R.: Line-of-sight path following of underactuated marine craft, *IFAC proceedings volumes*, 36, 211–216, 2003.
- Gart, S. W., Mitchel, T. W., and Li, C.: Snakes partition their body to traverse large steps stably, *Journal of Experimental Biology*, 222, jeb185 991, 2019.
- Gasoto, R., Macklin, M., Liu, X., et al.: A validated physical model for real-time simulation of soft robotic snakes, in: 2019 International Conference on Robotics and Automation (ICRA), pp. 6272–6279, IEEE, 2019.
- Godage, I. S., Webster, R. J., and Walker, I. D.: Center-of-Gravity-Based Approach for Modeling Dynamics of Multisection Continuum Arms, *IEEE Transactions on Robotics*, 35, 1097–1108, <https://doi.org/10.1109/TRO.2019.2921153>, conference Name: IEEE Transactions on Robotics, 2019.
- Goldberg, N. N., Huang, X., Majidi, C., Novelia, A., O’Reilly, O. M., Paley, D. A., and Scott, W. L.: On planar discrete elastic rod models for the locomotion of soft robots, *Soft robotics*, 6, 595–610, 2019.

- Greer, J. D., Morimoto, T. K., Okamura, A. M., and Hawkes, E. W.: A soft, steerable continuum robot that grows via tip extension, *Soft robotics*, 6, 95–108, 2019.
- Gu, G., Zou, J., Zhao, R., Zhao, X., and Zhu, X.: Soft Wall-Climbing Robots, *Science Robotics*, 3, <https://doi.org/10.1126/scirobotics.aat2874>, 2018.
- Guo, J., Xiang, C., Conn, A., and Rossiter, J.: All-soft skin-like structures for robotic locomotion and transportation, *Soft robotics*, 7, 309–320, 2020.
- Henke, E.-F. M., Schlatter, S., and Anderson, I. A.: Soft dielectric elastomer oscillators driving bioinspired robots, *Soft robotics*, 4, 353–366, 2017.
- Hirose, S. and Yamada, H.: Snake-like robots [tutorial], *IEEE Robotics & Automation Magazine*, 16, 88–98, 2009.
- Horchler, A. D., Kandhari, A., Daltorio, K. A., Moses, K. C., Ryan, J. C., Stultz, K. A., Kanu, E. N., Andersen, K. B., Kershaw, J. A., Bachmann, R. J., et al.: Peristaltic locomotion of a modular mesh-based worm robot: precision, compliance, and friction, *Soft Robotics*, 2, 135–145, 2015.
- Huang, W., Huang, X., Majidi, C., et al.: Dynamic simulation of articulated soft robots, *Nature communications*, 11, 1–9, 2020.
- Huang, X., Kumar, K., Jawed, M. K., Nasab, A. M., Ye, Z., Shan, W., and Majidi, C.: Chasing Biomimetic Locomotion Speeds: Creating Untethered Soft Robots with Shape Memory Alloy Actuators, *Science Robotics*, 3, <https://doi.org/10.1126/scirobotics.aau7557>, 2018.
- Ijspeert, A. J. and Crespi, A.: 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*, pp. 262–268, IEEE, 2007.
- Ito, F., Kawaguchi, T., Kamata, M., Yamada, Y., and Nakamura, T.: Proposal of a Peristaltic Motion Type Duct Cleaning Robot for Traveling in a Flexible Pipe, in: *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 6614–6621, IEEE, 2019.

- Jia, Y. and Ma, S.: A Bayesian-Based Controller for Snake Robot Locomotion in Unstructured Environments, in: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 7752–7757, IEEE, 2020.
- Jones, B. A. and Walker, I. D.: Kinematics for multisection continuum robots, *IEEE Transactions on Robotics*, 22, 43–55, 2006.
- Joyee, E. B. and Pan, Y.: A fully three-dimensional printed inchworm-inspired soft robot with magnetic actuation, *Soft robotics*, 6, 333–345, 2019.
- Jung, K., Koo, J. C., Lee, Y. K., Choi, H. R., et al.: Artificial annelid robot driven by soft actuators, *Bioinspiration & biomimetics*, 2, S42, 2007.
- Kalakrishnan, M., Chitta, S., Theodorou, E., Pastor, P., and Schaal, S.: STOMP: Stochastic trajectory optimization for motion planning, in: 2011 IEEE international conference on robotics and automation, pp. 4569–4574, IEEE, 2011.
- Kanada, A., Giardina, F., Howison, T., Mashimo, T., and Iida, F.: Reachability improvement of a climbing robot based on large deformations induced by tri-tube soft actuators, *Soft robotics*, 6, 483–494, 2019.
- Kano, T., Yoshizawa, R., and Ishiguro, A.: Tegotae-based decentralised control scheme for autonomous gait transition of snake-like robots, *Bioinspiration & biomimetics*, 12, 046 009, 2017.
- Kim, W., Byun, J., Kim, J.-K., Choi, W.-Y., Jakobsen, K., Jakobsen, J., Lee, D.-Y., and Cho, K.-J.: Bioinspired Dual-Morphing Stretchable Origami, *Science Robotics*, 4, <https://doi.org/10.1126/scirobotics.aay3493>, 2019.
- Lee, H., Jang, Y., Choe, J. K., Lee, S., Song, H., Lee, J. P., Lone, N., and Kim, J.: 3D-Printed Programmable Tensegrity for Soft Robotics, *Science Robotics*, 5, <https://doi.org/10.1126/scirobotics.aay9024>, 2020.
- Li, T., Zou, Z., Mao, G., Yang, X., Liang, Y., Li, C., Qu, S., Suo, Z., and Yang, W.: Agile and resilient insect-scale robot, *Soft robotics*, 6, 133–141, 2019.

- Liao, B., Zang, H., Chen, M., Wang, Y., Lang, X., Zhu, N., Yang, Z., and Yi, Y.: Soft Rod-Climbing Robot Inspired by Winding Locomotion of Snake, *Soft Robotics*, 2020.
- Liljeback, P., Stavadahl, O., and Beitnes, A.: SnakeFighter-development of a water hydraulic fire fighting snake robot, in: 2006 9th International Conference on Control, Automation, Robotics and Vision, pp. 1–6, IEEE, 2006.
- Liu, B., Ozkan-Aydin, Y., Goldman, D. I., and Hammond, F. L.: Kirigami skin improves soft earthworm robot anchoring and locomotion under cohesive soil, in: 2019 2nd IEEE International Conference on Soft Robotics (RoboSoft), pp. 828–833, IEEE, 2019.
- Liu, X., Gasoto, R., Jiang, Z., Onal, C., and Fu, J.: 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), pp. 7758–7765, IEEE, 2020.
- Luo, M., Agheli, M., and Onal, C. D.: Theoretical modeling and experimental analysis of a pressure-operated soft robotic snake, *Soft Robotics*, 1, 136–146, 2014.
- Luo, M., Pan, Y., Skorina, E. H., et al.: Slithering towards autonomy: a self-contained soft robotic snake platform with integrated curvature sensing, *Bioinspiration & biomimetics*, 10, 055 001, 2015.
- Luo, M., Skorina, E. H., Tao, W., Chen, F., Ozel, S., Sun, Y., and Onal, C. D.: Toward modular soft robotics: Proprioceptive curvature sensing and sliding-mode control of soft bidirectional bending modules, *Soft robotics*, 4, 117–125, 2017.
- Luo, M., Yan, R., Wan, Z., Qin, Y., Santoso, J., Skorina, E. H., and Onal, C. D.: OriSnake: Design, Fabrication, and Experimental Analysis of a 3-D Origami Snake Robot, *IEEE Robotics and Automation Letters*, 3, 1993–1999, <https://doi.org/10.1109/LRA.2018.2800112>, conference Name: IEEE Robotics and Automation Letters, 2018a.
- Luo, M., Wan, Z., Sun, Y., et al.: Motion Planning and Iterative Learning Control of a Modular Soft Robotic Snake, *Frontiers in Robotics and AI*, 7, 191, 2020.

- Luo, Y., Zhao, N., Kim, K. J., Yi, J., and Shen, Y.: Inchworm Locomotion Mechanism Inspired Self-Deformable Capsule-Like Robot: Design, Modeling, and Experimental Validation, in: 2018 IEEE International Conference on Robotics and Automation (ICRA), pp. 6800–6805, IEEE, 2018b.
- Manwell, T., Guo, B., Back, J., and Liu, H.: Bioinspired setae for soft worm robot locomotion, in: 2018 IEEE International Conference on Soft Robotics (RoboSoft), pp. 54–59, IEEE, 2018.
- Mao, S., Dong, E., Zhang, S., Xu, M., and Yang, J.: A new soft bionic starfish robot with multi-gaits, in: 2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, pp. 1312–1317, IEEE, 2013.
- Marchese, A. D., Katzschmann, R. K., and Rus, D.: Whole arm planning for a soft and highly compliant 2d robotic manipulator, in: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 554–560, IEEE, 2014.
- Marvi, H., Gong, C., Gravish, N., et al.: Sidewinding with minimal slip: Snake and robot ascent of sandy slopes, *Science*, 346, 224–229, 2014.
- McKenna, J. C., Anhalt, D. J., Bronson, F. M., et al.: Toroidal skin drive for snake robot locomotion, in: 2008 IEEE International Conference on Robotics and Automation, pp. 1150–1155, IEEE, 2008.
- Menciassi, A., Gorini, S., Pernorio, G., Weiting, L., Valvo, F., and Dario, P.: Design, fabrication and performances of a biomimetic robotic earthworm, in: 2004 IEEE International Conference on Robotics and Biomimetics, pp. 274–278, IEEE, 2004.
- Mukadam, M., Yan, X., and Boots, B.: Gaussian process motion planning, in: 2016 IEEE international conference on robotics and automation (ICRA), pp. 9–15, IEEE, 2016.
- Nayl, T., Nikolakopoulos, G., and Gustafsson, T.: Effect of kinematic parameters on MPC based on-line motion planning for an articulated vehicle, *Robotics and Autonomous Systems*, 70, 16–24, 2015.

- Nemitz, M. P., Mihaylov, P., Barraclough, T. W., Ross, D., and Stokes, A. A.: Using voice coils to actuate modular soft robots: wormbot, an example, *Soft robotics*, 3, 198–204, 2016.
- Onal, C. D. and Rus, D.: Autonomous undulatory serpentine locomotion utilizing body dynamics of a fluidic soft robot, *Bioinspiration & biomimetics*, 8, 026 003, 2013.
- Onal, C. D., Wood, R. J., and Rus, D.: An origami-inspired approach to worm robots, *IEEE/ASME Transactions on Mechatronics*, 18, 430–438, 2012.
- Onal, C. D., Tolley, M. T., Wood, R. J., and Rus, D.: Origami-inspired printed robots, *IEEE/ASME transactions on mechatronics*, 20, 2214–2221, 2014.
- Pankert, J. and Hutter, M.: Perceptive model predictive control for continuous mobile manipulation, *IEEE Robotics and Automation Letters*, 5, 6177–6184, 2020.
- Qi, F., Ju, F., Bai, D., Wang, Y., and Chen, B.: Motion modelling and error compensation of a cable-driven continuum robot for applications to minimally invasive surgery, *The international journal of medical robotics and computer assisted surgery*, 14, e1932, 2018.
- Qin, L., Liang, X., Huang, H., Chui, C. K., Yeow, R. C.-H., and Zhu, J.: A versatile soft crawling robot with rapid locomotion, *Soft robotics*, 6, 455–467, 2019.
- Qin, Y., Wan, Z., Sun, Y., et al.: Design, fabrication and experimental analysis of a 3-D soft robotic snake, in: *2018 IEEE International Conference on Soft Robotics (RoboSoft)*, pp. 77–82, IEEE, 2018.
- Rafsanjani, A., Zhang, Y., Liu, B., Rubinstein, S. M., and Bertoldi, K.: Kirigami Skins Make a Simple Soft Actuator Crawl, *Science Robotics*, 3, <https://doi.org/10.1126/scirobotics.aar7555>, 2018.
- Ratliff, N., Zucker, M., Bagnell, J. A., and Srinivasa, S.: CHOMP: Gradient optimization techniques for efficient motion planning, in: *2009 IEEE International Conference on Robotics and Automation*, pp. 489–494, IEEE, 2009.

- Rezapour, E., Hofmann, A., Pettersen, K. Y., Mohammadi, A., and Maggiore, M.: Virtual holo-
nomic constraint based direction following control of planar snake robots described by a simpli-
fied model, in: 2014 IEEE Conference on Control Applications (CCA), pp. 1064–1071, IEEE,
2014.
- Rieffel, J. and Mouret, J.-B.: Adaptive and resilient soft tensegrity robots, *Soft robotics*, 5, 318–
329, 2018.
- Rothmund, P., Ainla, A., Belding, L., Preston, D. J., Kurihara, S., Suo, Z., and Whitesides,
G. M.: A Soft, Bistable Valve for Autonomous Control of Soft Actuators, *Science Robotics*,
3, <https://doi.org/10.1126/scirobotics.aar7986>, 2018.
- Santoso, J., Skorina, E. H., Luo, M., Yan, R., and Onal, C. D.: Design and analysis of an origami
continuum manipulation module with torsional strength, in: 2017 IEEE/RSJ International Con-
ference on Intelligent Robots and Systems (IROS), pp. 2098–2104, IEEE, 2017.
- Schoels, T., Rutquist, P., Palmieri, L., Zanelli, A., Arras, K. O., and Diehl, M.: CIAO*: MPC-
based Safe Motion Planning in Predictable Dynamic Environments, *IFAC-PapersOnLine*, 53,
6555–6562, 2020.
- Seok, S., Onal, C. D., Cho, K.-J., Wood, R. J., Rus, D., and Kim, S.: Meshworm: a peristaltic soft
robot with antagonistic nickel titanium coil actuators, *IEEE/ASME Transactions on mechatron-
ics*, 18, 1485–1497, 2012.
- Shepherd, R. F., Iliovski, F., Choi, W., Morin, S. A., Stokes, A. A., Mazzeo, A. D., Chen, X.,
Wang, M., and Whitesides, G. M.: Multigait soft robot, *Proceedings of the national academy of
sciences*, 108, 20400–20403, 2011.
- Shin, B., Ha, J., Lee, M., Park, K., Park, G. H., Choi, T. H., Cho, K.-J., and Kim, H.-Y.: Hygrobot:
A Self-Locomotive Ratcheted Actuator Powered by Environmental Humidity, *Science Robotics*,
3, <https://doi.org/10.1126/scirobotics.aar2629>, 2018.

- Singh, G., Patiballa, S., Zhang, X., and Krishnan, G.: A pipe-climbing soft robot, in: 2019 International Conference on Robotics and Automation (ICRA), pp. 8450–8456, IEEE, 2019.
- Skorina, E. H. and Onal, C. D.: Soft Hybrid Wave Spring Actuators, *Advanced Intelligent Systems*, 2, 1900097, 2020.
- Skorina, E. H., Luo, M., Oo, W. Y., et al.: Reverse pneumatic artificial muscles (rPAMs): Modeling, integration, and control, *PloS one*, 13, e0204637, 2018.
- Stewart, D. E. and Trinkle, J. C.: An implicit time-stepping scheme for rigid body dynamics with inelastic collisions and coulomb friction, *International Journal for Numerical Methods in Engineering*, 39, 2673–2691, 1996.
- Stokes, A. A., Shepherd, R. F., Morin, S. A., Ilievski, F., and Whitesides, G. M.: A hybrid combining hard and soft robots, *Soft Robotics*, 1, 70–74, 2014.
- Sun, Y., Jiang, Y., Yang, H., Walter, L.-C., Santoso, J., Skorina, E. H., and Onal, C.: Salamanderbot: A soft-rigid composite continuum mobile robot to traverse complex environments, in: 2020 IEEE International Conference on Robotics and Automation (ICRA), pp. 2953–2959, <https://doi.org/10.1109/ICRA40945.2020.9196790>, iSSN: 2577-087X, 2020.
- Sun, Y., Abudula, A., Yang, H., Chiang, S.-S., Wan, Z., Ozel, S., Hall, R., Skorina, E., Luo, M., and Onal, C. D.: Soft Mobile Robots: a Review of Soft Robotic Locomotion Modes, *Current Robotics Reports*, pp. 1–27, 2021.
- Ta, T. D., Umedachi, T., and Kawahara, Y.: Design of frictional 2D-anisotropy surface for wriggle locomotion of printable soft-bodied robots, in: 2018 IEEE International Conference on Robotics and Automation (ICRA), pp. 6779–6785, IEEE, 2018.
- Tang, Y., Zhang, Q., Lin, G., and Yin, J.: Switchable adhesion actuator for amphibious climbing soft robot, *Soft robotics*, 5, 592–600, 2018.
- Tolley, M. T., Shepherd, R. F., Mosadegh, B., Galloway, K. C., Wehner, M., Karpelson, M., Wood, R. J., and Whitesides, G. M.: A resilient, untethered soft robot, *Soft robotics*, 1, 213–223, 2014.

- Transth, A. A., Liljeback, P., and Pettersen, K. Y.: Snake robot obstacle aided locomotion: An experimental validation of a non-smooth modeling approach, in: 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2582–2589, IEEE, 2007.
- Transth, A. A., Leine, R. I., Glocker, C., et al.: 3-D snake robot motion: nonsmooth modeling, simulations, and experiments, *IEEE transactions on robotics*, 24, 361–376, 2008a.
- Transth, A. A., Leine, R. I., Glocker, C., et al.: Snake robot obstacle-aided locomotion: Modeling, simulations, and experiments, *IEEE Transactions on Robotics*, 24, 88–104, 2008b.
- Trimmer, B. A., Lin, H.-T., Baryshyan, A., Leisk, G. G., and Kaplan, D. L.: Towards a biomorphic soft robot: design constraints and solutions, in: 2012 4th IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob), pp. 599–605, IEEE, 2012.
- Umedachi, T. and Kawahara, Y.: Caterpillar-inspired crawling robot on a stick using active-release and passive-grip elastic legs, in: 2018 IEEE International Conference on Soft Robotics (RoboSoft), pp. 461–466, IEEE, 2018.
- Umedachi, T., Vikas, V., and Trimmer, B.: Softworms: the design and control of non-pneumatic, 3D-printed, deformable robots, *Bioinspiration & biomimetics*, 11, 025 001, 2016.
- Umedachi, T., Shimizu, M., and Kawahara, Y.: Actuation Frequency-dependent Automatic Behavioral Switching on Caterpillar-inspired Crawling Robot, in: 2019 2nd IEEE International Conference on Soft Robotics (RoboSoft), pp. 167–171, IEEE, 2019a.
- Umedachi, T., Shimizu, M., and Kawahara, Y.: Caterpillar-inspired crawling robot using both compression and bending deformations, *IEEE Robotics and Automation Letters*, 4, 670–676, 2019b.
- Usevitch, N. S., Hammond, Z. M., Schwager, M., Okamura, A. M., Hawkes, E. W., and Follmer, S.: An Untethered Isoperimetric Soft Robot, *Science Robotics*, 5, <https://doi.org/10.1126/scirobotics.aaz0492>, 2020.

- Verma, M. S., Ainla, A., Yang, D., Harburg, D., and Whitesides, G. M.: A soft tube-climbing robot, *Soft robotics*, 5, 133–137, 2018.
- Virgala, I., Kelemen, M., Prada, E., Sukop, M., Kot, T., Bobovský, Z., Varga, M., and Ferenčík, P.: A snake robot for locomotion in a pipe using trapezium-like travelling wave, *Mechanism and Machine Theory*, 158, 104 221, 2021.
- Wang, J., Fei, Y., and Liu, Z.: FifoBots: Foldable Soft Robots for Flipping Locomotion, *Soft robotics*, 6, 532–559, 2019.
- Webster III, R. J. and Jones, B. A.: Design and kinematic modeling of constant curvature continuum robots: A review, *The International Journal of Robotics Research*, 29, 1661–1683, 2010.
- Wooten, M., Frazelle, C., Walker, I. D., Kapadia, A., and Lee, J. H.: Exploration and Inspection with Vine-Inspired Continuum Robots, in: 2018 IEEE International Conference on Robotics and Automation (ICRA), pp. 5526–5533, <https://doi.org/10.1109/ICRA.2018.8461132>, iSSN: 2577-087X, 2018.
- Wright, C., Johnson, A., Peck, A., et al.: Design of a modular snake robot, in: 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2609–2614, IEEE, 2007.
- Wright, C., Buchan, A., Brown, B., et al.: Design and architecture of the unified modular snake robot, in: 2012 IEEE International Conference on Robotics and Automation, pp. 4347–4354, IEEE, 2012.
- Wu, P., Jiangbei, W., and Yanqiong, F.: The structure, design, and closed-loop motion control of a differential drive soft robot, *Soft robotics*, 5, 71–80, 2018.
- Xavier, M. S., Fleming, A. J., and Yong, Y. K.: Experimental Characterisation of Hydraulic Fiber-Reinforced Soft Actuators for Worm-Like Robots, in: 2019 7th International Conference on Control, Mechatronics and Automation (ICCMA), pp. 204–209, <https://doi.org/10.1109/ICCMA46720.2019.8988691>, 2019.

- Xie, R., Su, M., Zhang, Y., Li, M., Zhu, H., and Guan, Y.: PISRob: A pneumatic soft robot for locomoting like an inchworm, in: 2018 IEEE International Conference on Robotics and Automation (ICRA), pp. 3448–3453, IEEE, 2018.
- Yamada, H.: S. development of amphibious snake-like robot acm-r5, in: the 36th International Symposium on Robotics (ISR 2005), Tokyo, 2005.
- Yamada, Y. and Nakamura, T.: Blade-type crawler capable of running on the surface of water as bio-inspired by a basilisk lizard, in: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1–9, IEEE, 2018.
- Yeshmukhametov, A., Koganezawa, K., and Yamamoto, Y.: A Novel Discrete Wire-Driven Continuum Robot Arm with Passive Sliding Disc: Design, Kinematics and Passive Tension Control, *Robotics*, 8, 51, <https://doi.org/10.3390/robotics8030051>, <https://www.mdpi.com/2218-6581/8/3/51>, number: 3 Publisher: Multidisciplinary Digital Publishing Institute, 2019.
- Yuk, H., Kim, D., Lee, H., Jo, S., and Shin, J. H.: Shape memory alloy-based small crawling robots inspired by *C. elegans*, *Bioinspiration & biomimetics*, 6, 046 002, 2011.
- Zhang, K., Qiu, C., and Dai, J. S.: An extensible continuum robot with integrated origami parallel modules, *Journal of Mechanisms and Robotics*, 8, 031 010, 2016.
- Zou, J., Lin, Y., Ji, C., and Yang, H.: A reconfigurable omnidirectional soft robot based on caterpillar locomotion, *Soft robotics*, 5, 164–174, 2018.
- Zucker, M., Ratliff, N., Dragan, A. D., Pivtoraiko, M., Klingensmith, M., Dellin, C. M., Bagnell, J. A., and Srinivasa, S. S.: Chomp: Covariant hamiltonian optimization for motion planning, *The International Journal of Robotics Research*, 32, 1164–1193, 2013.