CPG-based Locomotion Control of a Snake-like Robot

by

Norzalilah Binti Mohamad Nor

M.S. (Science University of Malaysia) 2011

A dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Department of Robotics

in the

GRADUATE SCHOOL OF SCIENCE AND ENGINEERING

of the

RITSUMEIKAN UNIVERSITY, BIWAKO-KUSATSU CAMPUS

Committee in charge:

Professor Shugen MA, Chair
Professor Sadao KAWAMURA
Professor Makoto NOKATA

Fall 2014

CPG-based Locomotion Control of a Snake-like Robot

by

Norzalilah Binti Mohamad Nor

Abstract

CPG-based Locomotion Control of a Snake-like Robot

by

Norzalilah Binti Mohamad Nor

Doctor of Philosophy in Department of Robotics

Ritsumeikan University, Biwako-Kusatsu Campus

Professor Shugen MA, Chair

Due to advantages of a natural snake, researchers have designed many control architectures to implement the unique movements into snake-like robots. In this study, we propose a novel control structure for a snake-like robot based on central pattern generator (CPG), which is inspired by a control mechanism in animal bodies. Based on the phase oscillator dynamical system, a simple network structure of coupled nonlinear oscillators or neural networks has been designed to produce rhythmic patterns. These rhythmic motions can be used to control the serpentine locomotion of a snake-like robot by manipulating control parameters of the phase oscillator model. Using the simplified CPG network, the locomotion pattern like forward movement, backward movement or alternating number of the S-shape is achievable. Feedback connections between oscillators and stability analysis for phase synchronization of the CPG network have also been analyzed. The proposed CPG network has a simple structure with less complexity, less mathematical computation, fast convergence speed and exhibits limit cycle behavior. Another uniqueness of our proposed control method is an easier parameter control based on only one parameter. With this parameter, all of the said movements can be realized in a snake-like robot.

Further analysis has been conducted for body shape control of a snake-like robot. The feature of body shape control is useful for adapting in various width of space such as tunnel, pipe and trench. This feature can be realized by modifying one of the CPG parameters. The control parameter, however, cannot be adjusted during the locomotion, because it results

in unstable movements of the snake-like robot. To overcome this issue, we have introduced an activation function so-called linear bipolar, which was developed from the relationship between the CPG parameter and the body shape control parameters. By adopting the activation function into the CPG model, online parameter modification can be achieved without resulting any discontinuous CPG output. Simulation and experiment results and torque analysis confirmed the effectiveness of incorporating the activation function into the CPG model. Another contribution of this thesis is a turning control of a snake-like robot for obstacle avoidance, which is more general and can be implemented for any kinds of snake-like robot's design.

For a snake-like robot to maintain its direction while moving in the hallway with different space width, we propose a straight-path control strategy of a snake-like robot based on our CPG model. Even though maintaining the straight locomotion of a snake-like robot can be realized by the control architecture like sine-based, model-based or other, a problem occurs when one or more of the control parameters are modified online, results in deviation of the trajectory of the snake-like robot from its initial path. This is not desirable when we need the snake-like robot to move in a hallway with different space width, where online modification of parameter is required to change its body shape. The proposed simple control strategy of the straight-path locomotion can keep its locomotion direction while the robot moves in a various width of the space. Computer simulation and experiment verified its validity and effectiveness.

Contents

List of Figures

List of Tables

Acknowledgements

First of all, I would like to bestow my gratefulness to Allah swt because without His permission I would not have completed my doctoral course as scheduled. I would like to dedicate my deep gratitude to my research supervisor, Prof. Shugen Ma, a professor in Department of Robotics at Ritsumeikan University, for allocating his precious time and effort to guide me throughout my doctoral course. Thank you for his sincere comments, ideas, guidance, wise advices and assistance.

My grateful thank is dedicated also to my sponsorship, Science University of Malaysia (USM) and Ministry of Higher Education (MOHE) Malaysia, during my three years of doctoral course in Japan.

Furthermore, thanks to all Professors in Department of Robotics at Ritsumeikan University, who have giving me useful and beneficial advices ever since I am in the first semester until my final semester: Prof. Makoto Nokata, Prof. Ryuta Ozawa and hirokazu Maeda. Thank you for your kind support and advices during the research seminar.

I would also like to extend deepest appreciation to all of my lab mates in Biomimetic Intelligent Mechatronics Laboratory, especially to Dr. Yi Sun, Mr. Dingxin Ge, Mrs. Yongchen Tang, Mr. Yang Yang, Mr. Chao Ren, Mr. Atsushi Kakogawa, Mr. Fabian Reyes, Mr. Wengbin Tang, Mr. Yang Tian for their kind advices during the doctoral seminar and their wise comments in proofreading of my journal and conference papers, to Mr. Chuanguo Li and Dr. Xiaodong Wu for their advices on the experimental setup, programming and control circuit design, to Mr. Xuan Zhou for his advice on the programming, and to others who have engaged with me directly or indirectly throughout completing my dissertation course. Thank you.

My thankfulness all of the fellows and officers in Robotics Department, in Office of Graduate Studies, and in Office of International Students at Ritsumeikan University for guiding me during my living in Japan. Without their help, I may not be able to fully focus

on my doctoral research. Thank you also to Ritsumeikan University for providing financial supports for participating in international academic conferences.

Finally, I would like to thank my beloved family and friends for their encouragement and understanding during my study. Not to forget, to my beloved akhawats, thank you for their sincere support and love. All lessons, experiences and the helping hands will be kept in my heart. May Allah bless us always.

Curriculum Vitæ

Norzalilah Binti Mohamad Nor

Education

1991-1996	Elementary School of Panglima Bukit Gantang (Ipoh, Perak, Malaysia) Elementary School Student
1997-1999	Tarcisian Convent Ipoh (Ipoh, Perak, Malaysia) Secondary School Student
2000-2001	High School of Teknik Kerian (Bagan Serai, Perak, Malaysia) Upper Secondary School Student
2002-2004	Matriculation Centre of International Islamic University Malaysia (Gombak, Selangor, Malaysia) A-level of Engineering
2004-2008	International Islamic University Malaysia (Gombak, Selangor, Malaysia) Bachelor Degree of Mechatronics Engineering
2010-2011	Science University of Malaysia (Penang, Malaysia) Master Degree of Science (Mathematics)
2011-2014	Ritsumeikan University, BKC (Kusatsu, Shiga, Japan) Doctoral Candidate, Robotics

Personal

Born	February 22, 1984, Perak, Malaysia.
Research Interests	Locomotion control, neural network, Intelligent control system.

Chapter 1

Research Background

Research in bio-inspired mobile robots has become attraction to many researchers due to their potential ability to replace human operations for searching, navigating and rescuing in hazardous environments such as natural disasters, explosions, mining and radiation areas. Several types of bio-inspired robots that have been widely developed are legged robot, wheeled robot, limbless robot, wall climbing robot and amphibious robot.

Each of the mobile robot has their own advantages and disadvantages depending on the situations in which it operates. For example, legged robot offers high maneuverability for obstacle avoidance in [14] [15] [13] which is very useful in unstructured environments. However, their stability control is complicated so they usually uses many motors to control the actuated joints. For searching purposes after a disaster or in a collapse building, some of the spaces are too small to be reached by human which will delay the rescuing of victims. Thus, mobile robots operations are highly demanded for fast action with ability to move in various space width such as in tunnel and pipe.

Motivated by the earthquake and tsunami that occurred in Japan on 11 March 2011, followed by the leaking of the nuclear power plant [16], leads to the necessity to develop mobile robots to navigate into the radiated area for detection and inspection. This helps to reduce human involvement and exposure to the radiation which harms the cells of the body and consequently causing sickness or even death.

Therefore, to overcome all of the above mentioned issues, high mobilities and advance controls of mobile robots are required. One of the biggest challenge is the control of the mobile robots. Simple control is preferable which promises a light computational cost, easy implementation, simple and understandable by human operators who may be not familiar with the robot's control system. Another interesting aspect of robot's control is its self-adaptive capabilities to the environment. This autonomous control requires interaction between the actuators and with the helps of sensors. The control approach is strongly depending on the mechanism and type of the mobile robots. Many researchers have designed various control architectures for achieving versatile locomotion of the mobile robots and the research is still ongoing.

In this thesis, locomotion control of a snake-like robot is studied due to its advantages over other types of mobile robots in term of their stability and traverse ability. Moreover, the unique locomotion of a biological snake in sneaking into a narrow space, brings great interest in mimicking it into a snake-like robot, which is useful for rescuing and searching tasks.

1.1 Biological Snakes

Snakes are one of the amazing creatures with its unique locomotion and body features in their lack of legs. Many researches have analyzed deeply concerning its anatomy including its mechanical interconnection, skeletal structure, skin, and scales [1] [17] [2]. By understanding the nature of the snakes, it helps to relevantly model, design and control a snake-like robot.

1.1.1 The Anatomy of Snakes

Skeletal structure of a snake is shown in Fig. 1.1 (a). The body shape of a snake is changed with the help of muscles that are arranged diagonally along each side of the snake [1]. The type of locomotion performed by the snake is determined by the contraction and relaxation of these muscles. For example, the body of the snake will bend if muscles on

one side of the snake are contracted at the same time as the equivalent muscles on the other side are relaxed. On the other hand, if the opposite sets of the muscles are simultaneously contracted or relaxed, then the body of the snake will be shortened or extended.

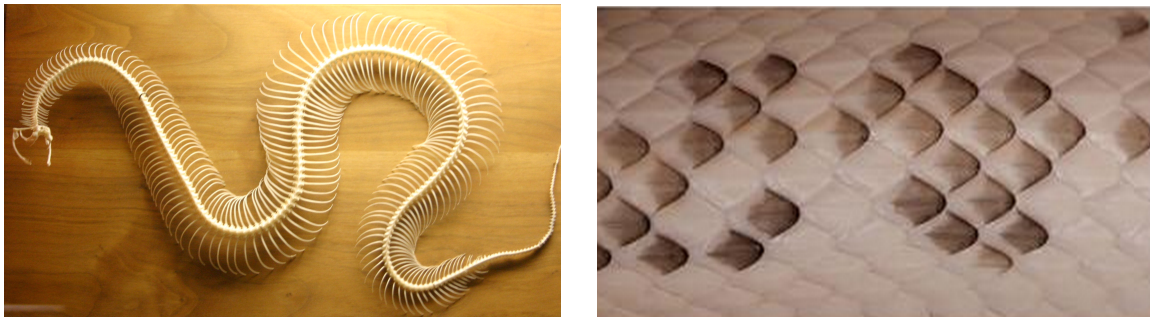


Figure 1.1. Anatomy of a snake: (a) skeleton, and (b) scales [1].

Studies of biological snakes have indicated that friction generated by the scales and redistribution of their weight are important for the forward gliding motion [1] [18]. These scales (Fig. 1.1 (b)) completely cover the skin of a snake and integrated with the skin. The areas of skin between the scales allow the snake to flex its body while maintaining a smooth coverage of the scales. Another important purpose of the scales is to protect the skin from wear and tear when the snake moves on rough surfaces. To realize the forward gliding motion, the scales provide anisotropic ground friction, that is, the scales give the snake a larger friction coefficient in the transversal direction than in the tangential direction of the snake body. Thus, in general, propulsion on the ground relies critically on the frictional anisotropy of the snake scales.

1.1.2 Locomotion of Biological Snakes

Locomotion of snakes can be classified into four categories namely serpentine, sidewinding, rectilinear, and concertina. The typical locomotion of snakes is the serpentine locomotion, which can be described as S-shape movement. These four unique modes are used for terrestrial locomotion and the selection of the locomotion modes a snake will use depends on several factors such as type of surface its travel and its speed.

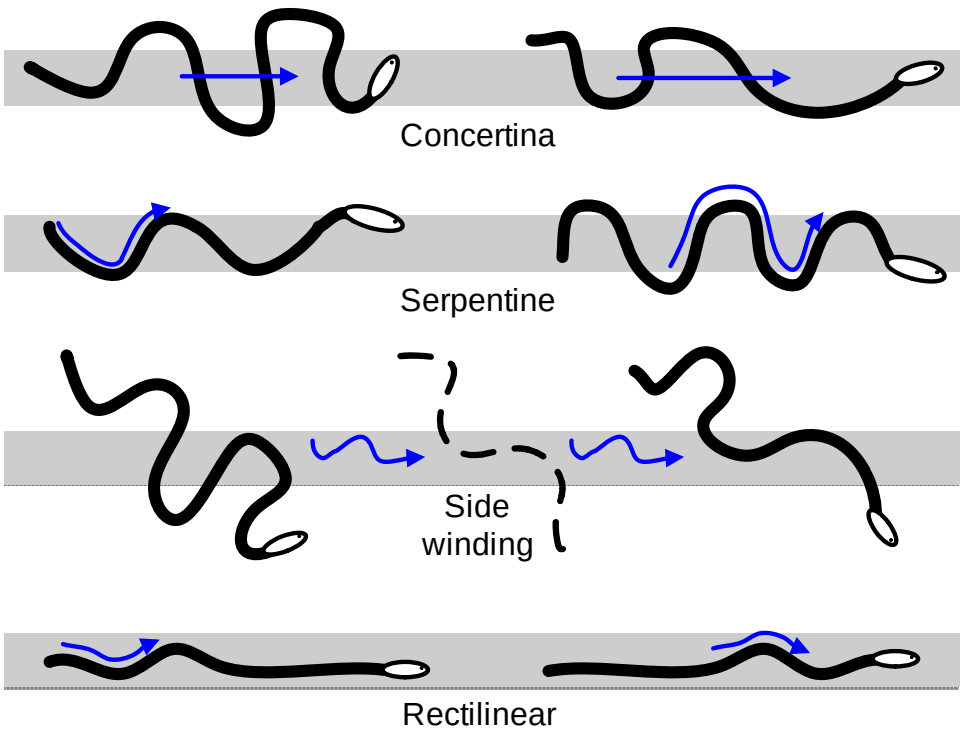


Figure 1.2. Four basic movements of snake.

Concertina locomotion utilizes stationary body parts as anchors to push or pull the remaining body forward. It involves alternately pulling up the body into bends and then straightening out the body forward from the bends. The front part of the body then comes to rest on the surface and the back part of the body is pulled up into bends again, and so forth. Static friction is critical to concertina locomotion because it is used as the reaction force for forward movement [19]. To keep on moving, the muscular-induced forces cannot exceed the static friction force. Thus, neuromuscular control is critical to concertina locomotion. Concertina locomotion is suitable in narrow passages such as crawling through tunnels, pipes or trench. It can also be used for climbing. However, concertina locomotion is difficult to be realized and is not suitable for snakes that are short and stout. The motion pattern is not very efficient in terms of energy consumption [19].

Lateral or serpentine undulation is the most common locomotion of snakes. This kind of locomotion mode is employed by various terrestrial species of snakes and usually used with the combination of other modes of locomotion. Early studies suggested that progression by

lateral undulation is most stable with at least three contact points [20]. However, recent studies have shown that this locomotion mode can also work with one or more contact points. The body of the snake have a higher tendency to slip sideways when there is just a single contact point. The snake will shift to a different mode of locomotion if it cannot find one or more stable contact points. In lateral undulation, the muscles are activated sequentially and continuous wave are propagated backwards along the body from head to tail. The muscles are active unilaterally in each bend, from the convex part of a bend forward to the straight or concave part of the bend. This form of locomotion is not suitable on slippery surfaces because to push the snake forward, the wave motion helps the side of the snake body to push against irregularities on the surface.

Sidewinding is employed for moving over flat, smooth or slippery surfaces. This form of locomotion can be employed on various surfaces regardless lesser or greater the frictional resistance. The motion resembles the lateral undulation in term of the bending pattern, but differs in the direction of motion. The snake travels roughly diagonally relative to the the track it forms on the ground. Sidewinding motion involves lifting the snake head off the ground and moving the anterior body forward, placing each point along the body in static friction with the substrate. The utilization of muscle activity is similar to the lateral undulation except that some muscles are active bilaterally in the regions of trunk lifting. Support of the snake mass can be transferred to its body segments that progressively contact the substrate during the sidewinding locomotion. Studies have confirmed that sidewinding is mechanically proficient and efficient in term of energy consumption. This form of locomotion is probably suitable to all midsized snakes and can form a characteristic track on sandy terrain.

Another common mode of snake locomotion is the rectilinear locomotion. Rectilinear locomotion refers to a straight line movement, with comparatively slow forward speed. The muscle activity is bilaterally symmetric, unlike the lateral undulation and sidewinding, which involve unilateral muscle activity that alternates from one side of the body to the other. The belly scales are alternately lifted slightly from the ground and pulled forward,

and then pulled downward and backward. The cycle repeats when the body has moved forward sufficiently to stretch the scales. This form of locomotion is often employed by large and heavy-bodied snakes. Due to the quiet and precise progression of the rectilinear motion, snakes will usually use this kind of crawling to stalk their prey to avoid alerting their target victim.

1.2 Introduction to Snake-like Robots

In this thesis, we focus on locomotion control of a snake-like robot. Due to the unique locomotion performed by the biological snake, researches on snake-like robots start to attract interests for robotic mobility in various environments Fig. 1.3 [21]. In 1940s, Gray [17] started analysis of the biological snake movement. After several decades, the first snake-like robot was developed by Hirose [2] in 1972 as shown in Fig. 1.4 (a). Since the first attempt of the development of snake-like robot, research on many kinds of designs, mechanisms and control approaches have increased enormously until now.

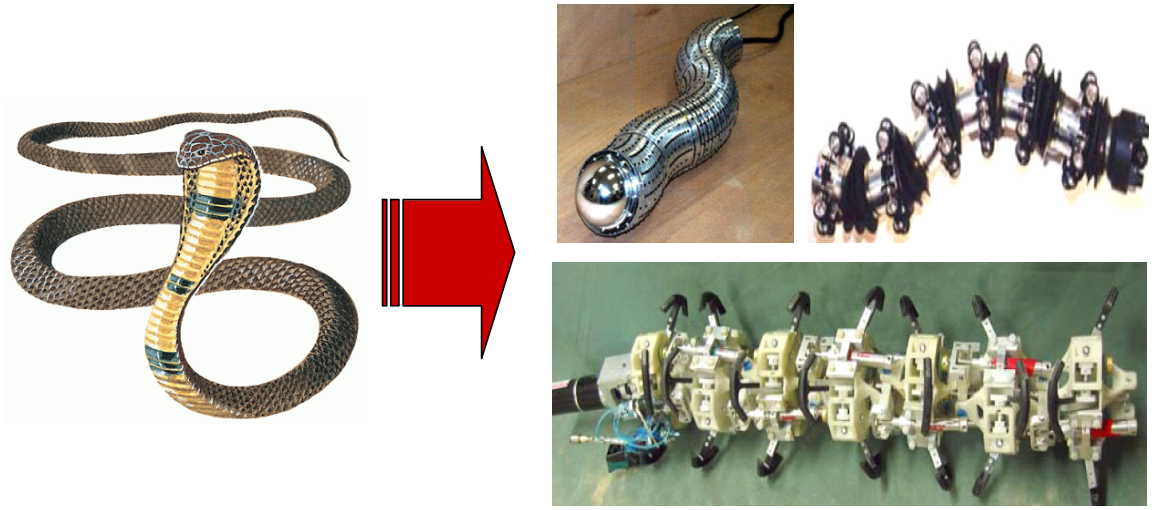


Figure 1.3. From biological snake to snake-like robots.

Typically, a series of rigid links connected by joints serve as the basic structure which mimics the body features of a biological snake. The biological snake locomotion therefore can be easily realized by swinging the joints from side to side. Besides, due to the design of

the serial connected joints, it allows the possibility for the snake-like robot to self assemble in the field, and to be easily repaired if a section is damaged. Although snake-like robots have many degrees of freedoms which increase complexity to control, but their stability and traverse ability [22] exceed the conventional wheeled, legged, and tracked mobile robots.

Furthermore, snake-like robots can distribute its mass over a large area for support so that it can still move effectively. Self-support between larger numbers of contact points enables continued operation and movement. The snake-like robots can disguise themselves very easily and move through the underbrush where is difficult for other robots. This ability could also be used for military purposes.

Another surplus of snake-like robots over other types of ground mobile robots is their ability to sneak into a narrow space. By changing the curvature of its body shape, it can adapt to different space widths which is very useful for inspection purposes especially in an unaccessible area that cannot be seen directly by the eye. Simple direct-view borescopes have proven useful, but articulated self-advancing devices which can follow complex paths could open many more applications [10]. To overcome the limitations of the traditional methods, self-propelled inspection devices, like snake-like robots are highly demanded.

Thus, two main reasons for the interest in the snake research; the unique and scientific interest in the mechanism of the snake's motion and the engineering interest for future applications of snake-like robots. Generally, a standard design of snake-like robots can be categorized into two types; active joint with passive wheels and active joint without passive wheels. In this thesis, we did not consider snake-like robots with active wheels [23] because we will focus on controlling only the joint motors.

1.2.1 Snake-like Robots with Passive Wheels

By installing passive wheels, it provides an advantage for locomotion on flat surface and for resistance to the lateral movement or side movement. Examples of snake-like robot with passive wheels include the world's first snake-like robot (ACM-III), ACM-R3, ACM-R5, and

AmphiBot (see Fig. 1.4). There are many more intense designs of snake-like robots with passive wheels which have not been listed here.

Active Cord Mechanism-III (ACM-III) consists of 20 connected joints that generate swinging motion around vertical axes [2]. Two small passive wheels were installed at the bottom of each snake's unit. Each of the joint is controlled actively by a motor. This snake-like robot can realize a smooth creeping motion by transmitting a control signal from joint to joint, from front to rear. It can also follow shape by adding tactile sensors on both side of the body. ACM-III also has a capability to automatically wrap around an object and autonomously driving inside a maze.

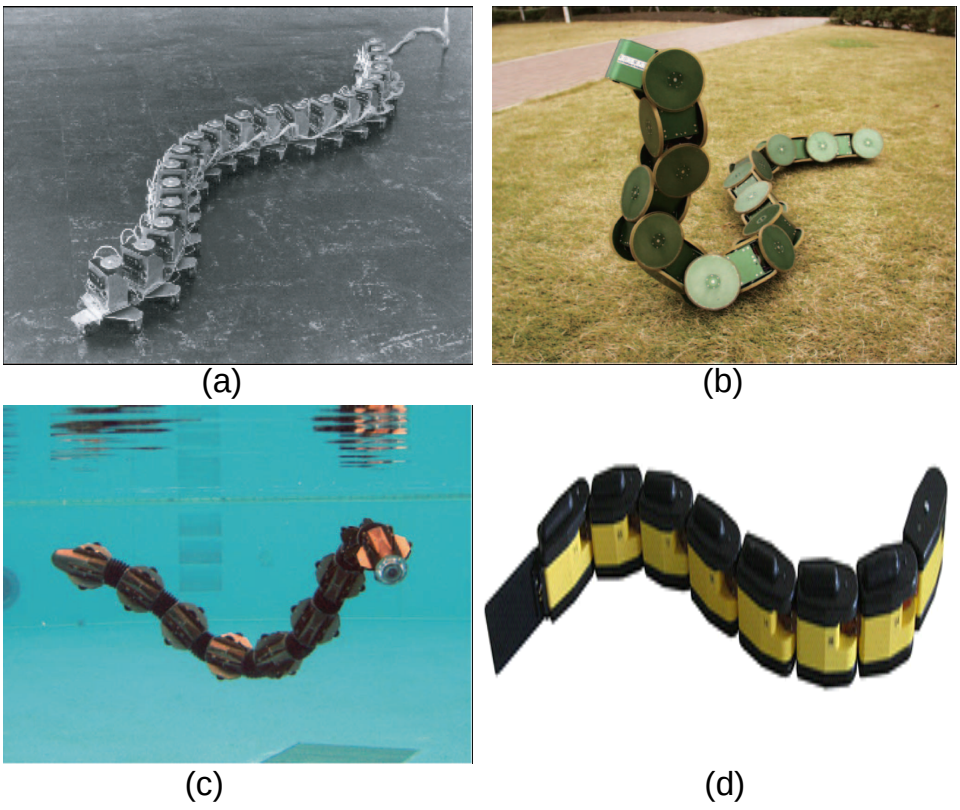


Figure 1.4. Snake-like robots with passive wheels: (a) ACM-III [2], (b) ACM-R3 [3], (c) ACM-R5 [4], and (d) Amphibot [5].

Another example of a snake-like robot with active joint and passive wheels is ACM-R3 [3]. Developed in 2001, ACM-R3 had a capability to move in 3D motion and had a large wheels on all sides of the body. These large passive wheels were added for additional

functionality where it can roll against contacted obstacles. The links were designed to contain all the components within a shell that had orthogonal axes of rotation on each end. The mechanism introduced in ACM-R3 can be used to realize serpentine locomotion, lift up its body to move over obstacles, lateral rolling and side winding gaits. This snake-like robot also had on-board power and could be radio-controlled which is useful for actual search and rescue application.

ACM-R5 [4] was designed as a waterproof and hermetic dust-proof snake-like robot. It was an amphibious design which can operate on ground and in water. The joint between each segment or module of the robot consisted of a universal joint and bellows. The universal joint acted as the bones and bellows acted as an integument, an enveloping layer (as a skin, membrane, or cuticle) of a snake. This robot includes paddles and passive wheels around its body on each segment, which can obtain resistance property on both ground and water. For realizing versatile locomotion, the ACM-R5 incorporated advanced control system where each segment had a CPU, battery and motors. Thus, each of the modular unit could operate independently. Communication between the modules was done to recognize its number from the head and how many units in operation.

AmphiBot I [5] was designed by Biologically Inspired Robotics Group at cole Polytechnique Fdrale de Lausanne. It is a modular amphibious snake-like robot, constructed out of several identical segments, known as elements. The aim in designing AmphiBot is to study the limbless body shape and neuronal control mechanisms. Each of the robots elements had one DOF and the elements were fixed such that all axes of rotation were aligned. The robot was designed to have distributed actuation, power and control; therefore, each element carried its own DC motor, battery, and microcontroller. For motion on a terrain, the robot could be equipped with removable sets of passive wheels. To support its amphibious role, each individual element was made waterproof, as opposed to having a covering skin over the entire chain of elements. The application of the robot was to serve as a test bed to support two research goals of the authors: (1) to take inspiration from snakes and elongated fish such as lampreys to produce a novel type of robot with dexterous locomotion abilities, and

(2) to use the robot to investigate hypotheses of how central nervous systems implement these abilities in animals [24] [25].

1.2.2 Snake-like Robots without Passive Wheels

Wheel-less mechanism of a snake-like robot is preferable to realize versatile locomotion for operating in unstructured environments [26]. One of the advantage of snake-like robot without wheels is their ability to move in different terrains, such as in sand or grass. For snake-like robots with wheels, the possibility to sink into sand is higher and the sand may get stuck in the rotational shaft of the wheels. However, frictional resistance of the wheel-less body on the ground is high, and large energy will be lost in the locomotion, if the normal serpentine motion is used. Some interesting designs of snake-like robots mechanisms are discussed here.

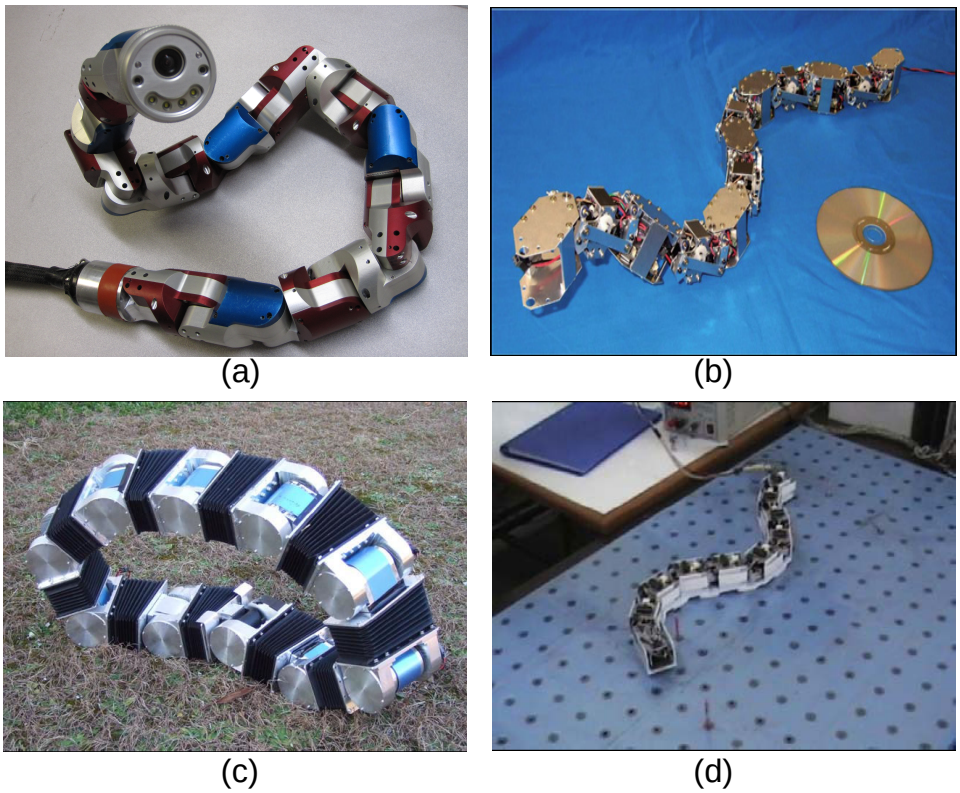


Figure 1.5. Snake-like robots without passive wheels [6] [7] [8] [9].

Fig. 1.5(a) shows one of a unique design of a snake-like robot which is called Uncle

Sam, developed at Carnegie Mellon University. This robot has a strong and compact joint mechanism and has an ability to climb trees by wrapping itself around a pole and climb vertically [6]. Although the demonstration shows only the ability to climb a straight trunk [27], the mechanism has a potential to perform branch to branch transition as long as the length of the robot is long enough to wraps between two branches. Moreover, it can also moves in a variety of different ways including rolling, wiggling, and side-winding. Due to its versatile design, it can be deployed to quickly climb a flagpole, structural support, gutter downspout, pipe, or tree and provide a view from a high vantage point.

In [7], the authors proposed a new 2-dof joint mechanism for a snake-like robot (see Fig. 1.5 (b)). The proposed mechanism has four rotation axes against two actuators which is different from a typical joint mechanism, where the typical joint mechanism has the same number of rotational axes as the actuators. In other words, the range of motion is twice as large as each universal joint and the power of rotation around yaw axis and pitch axis are the total of the power of two actuators. This joint mechanism has relatively large range of motion and it can improve the capability of snake-like robots to move in a narrow environment. Using the proposed mechanism, the snake-like robot can perform a side winding movement, a locomotion style of a natural snake in desert, which is suitable on a slippery surface.

ACM-7 (Fig. 1.5 (c)), developed by Ohashi et al [8], features a water-tight structure with a large range pitch joint of 90 degree and a high output-power actuator arrangement, based on the coupled drive concept. To avoid large energy lost in the locomotion, the ACM-R7 forms a loop shape and rolls like a wheel on the rim for locomotion efficiency of a snake-like robot. The unique mechanism of the looped ACM can create infinite spinning motion of the whole body. In addition, since there is no sliding motion between the body and the ground, the motion is fast and energy efficient. Another advantage of this mechanism is the capability to make turning and step climbing using a shape control method. The ACM-7 has successfully climbed a slope and move stably on a grassy surface.

Bayraktaroglu et al [9] designed a wheel-less snake-like robot (Fig. 1.5 (d)) based on

push-points in the reachable zone of the mechanism's modules. The propulsion of the mechanism is inspired by the one of natural snakes exerting the lateral undulation. The mechanism maintains a continuous lateral interaction with simultaneous push-points, which is only due to the internal action provided by joint actuators. The stability of the snake-like robot has been tested for different initial conditions and random perturbations. Beside the lateral undulation locomotion, the snake-like robot can perform inchworm-like motion. The snake-like robot can also pass through a constrained or narrow environment such as U-profile arc using the lateral undulation.

1.3 Control Architecture of a Snake-like Robot

Due to the enormous growth of versatile snake-like robots design, many control architectures have been proposed to efficiently control the movement of the snake-like robots based on their mechanical design and mechanisms. Some of the snake-like robots are complicated to be controlled depending on various locomotion they can perform, which often acquires many joints with higher degree of freedom. A simple control scheme is therefore preferable for making the snake-like robots system easier to handle and implement. Basically, the control architecture of a snake-like robot can be categorized into three: sine-based, model-based, and Central Pattern Generator (CPG)-based.

1.3.1 Sine-based Control

Sine-based control uses a simple time-indexed sine function to produce the traveling waves. For example, the world first's snake-like robot ACM-III [2] and [28] has been controlled using the sine-based control, which is a simple mathematical model. The parameters including amplitude, frequency and phase are directly related to the output, and can be modified according to the preferred kind of locomotion. This control method is useful for non expert operator to handle due to its easy control. However, because of the increasing of demand of versatility and mechanisms of the snake-like robots plus the existence of high

technology such as various sensor integrations, this method is not preferable due to the difficulty to integrate sensory feedback signals. In addition, the parameters of a sine-based function is not efficiently to be modified online. In other words, the snake-like robots need to be stopped (offline) before the parameters can be adjusted to achieve desired locomotion. This condition is not efficient to the snake-like robots for rescuing or searching purposes and causing operational downtime. Besides, the online modifications lead to unstable movements of the snake-like robots due to the discontinuous of the output signals. The unstable movements will cause damages to the gearboxes and motors. The problem can be overcomed by filtering the parameters or outputs, but the simplicity of this method will vanish. Moreover, it is not efficient for path following or path planning task using the sine-based approach.

1.3.2 Model-based Control

Model-based control uses kinematic and dynamic models of the robot to design control laws for gait generation. Some examples of robot's control that apply this kind of control can be referred to [29] [30] [31] [32]. The advantage of the the model-based approach is the identification of the fastest gaits of the robot using the kinematic constrains or approximations of the equations of motion. This is very useful in designing controllers that are compatible to the design and mechanisms of the robot. However, there are several limitations of this control method. First, the design controllers are not always suitable for interaction with human operator to modulate the parameters. Second, the performance of the controllers will deteriorate if the models become inaccurate, often when the robot is dealing with complex environments due to the fluctuation of the interaction forces. Another drawback is the control model need to be redesigned if the mechanism of the robot is changed. Furthermore, for a complicated robot, such as snake-like robots, to model its dynamic or kinematic is not an easy task due to large number of degree of freedom especially for robots with various functionality. Moreover, the design of the controller is only

suitable for specific robots mechanisms whose kinematic or dynamic is similar, which loses its generality.

1.3.3 CPG-based Control

CPG-based approach uses dynamical systems of coupled nonlinear oscillators or recurrent network to generate traveling waves for desired locomotion. Limit cycle behavior of the traveling waves are achieved by the differential equations integrated over time. The exhibition of this limit cycle behavior ensures the robustness of the CPG-based model against any random perturbation to the system, where the oscillatory patterns asymptotically return to the limit cycle. Furthermore, the limit cycle can usually be modulated by some parameters which offer the possibility to smoothly modulate the type of gaits produced. This control approach also can be easily integrated with sensory feedback signals in the differential equations, and show interesting properties such as entrainment by the mechanical body [33] [34] [35]. Moreover, some of the CPG models have explicit parameters such as amplitude and frequency which show clear relationship with the output signal. Depending on the applications, CPG-based control can be easily implemented and uses simple control which is understandable by readers who may not familiar with the control approach. CPG-based controller is very useful for smoothing gaits transition which depends on communication between modules. If just using CPG-signal without communication coordination, then there is no advantage in controlling modular robot compared with sine-based controller. However, one difficulty using CPG-based approach is the design of the CPG network to produce particular patterns that suit to the types of the robot. Also, since it is not based on the model, it cannot take friction, mass, and other physical properties in its formulation.

1.4 Central Pattern Generator

As mentioned in previous section, CPG-based approach is the alternative method for advance control of mobile robots. Due to many advantages offered by the CPG-based

method, this method is preferable in many robot applications such as legged robots, hexapod robots, and amphibious robots.

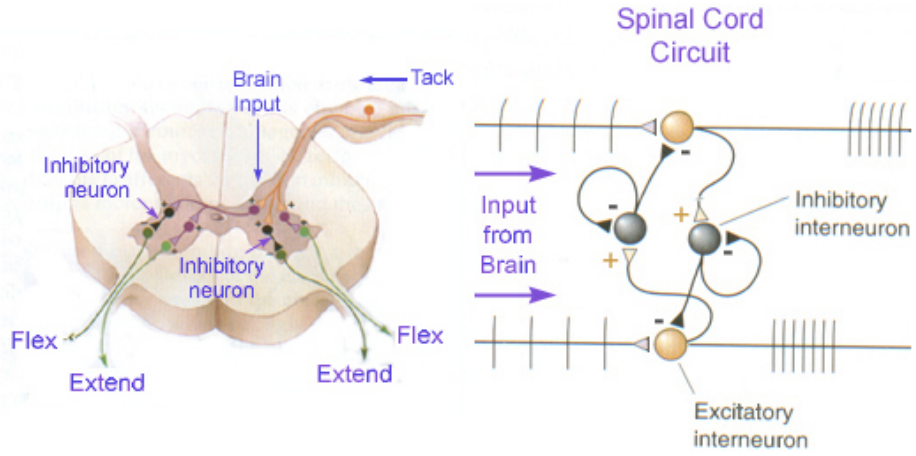
1.4.1 Introduction to Central Pattern Generator

The problem of locomotion control of various kinds of robots can be solved by taking inspiration from the neural control of locomotion in animals. The first pioneering studies reveal that the pattern of locomotor activity is due to a pattern of neural activity [36]. Despite the differences in the kind of locomotion and body forms of various animals, the different elements, which is hierarchically organized and common to the motor system can be identified. The key component of the motor system is called Central Pattern Generator (CPG).

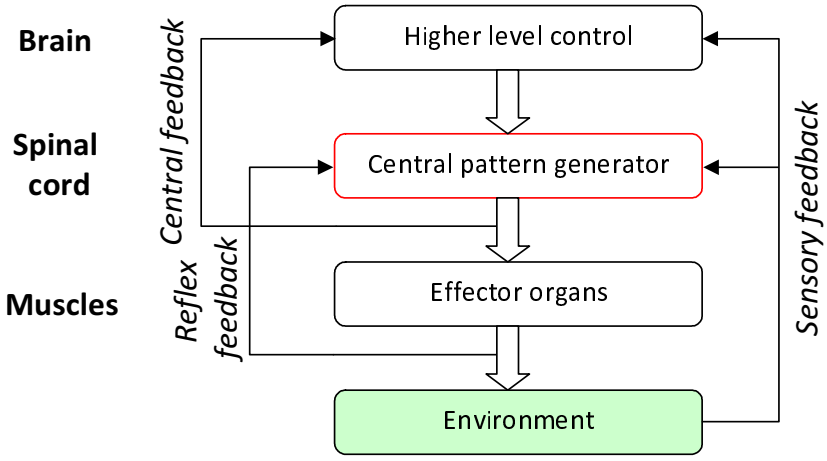
CPG is neural networks that exist in spinal cords of the living animals which are used as activation signal for muscle contractions. It is a neural circuit that can produce a rhythmic pattern without needing any inputs from sensory feedback signal or descending control. Even completely isolated CPGs in a petri dish can produce patterns of activity, called active locomotion, which are very similar to locomotion activated by simple electrical or chemical stimulation [37].

Signals from CPG directly control the effector organ such as cilia and legs. The signals receive from higher control control neurons are necessary only to initiate the locomotion, but not to generate the desired pattern of movements. Indeed, the rhythmic movements are generated at the neural level of the CPG. Thus, its functioning results from an interaction between central commands and local reflexes [38].

Fig. 1.6 (b) shows the main functional features of the motor system of animals, with presence different types of feedback namely central feedback which is directly input to the higher control neurons, reflex feedback from the motor output to the CPG, and sensory feedback from the environment to either the CPG or the higher level neurons. The presence of feedback from environment is not strictly necessary to generate a rhythmic pattern, which is called fictive locomotion [39]. However, the feedback signals help to achieve efficient



(a)



(b)

Figure 1.6. (a) Existence of neurons in spinal cord of living animals, and (b) Bionic control process [10].

performance of the whole control system in real environments by modifying the locomotion patterns. In addition, removing higher control signals will result in unstable movement of the animal. This situation shows the important of involvement of the higher level neurons for posture control.

1.4.2 CPG in Locomotion Control of Robots

From the engineering viewpoint, CPG-based approach uses dynamical systems of coupled nonlinear oscillators to produce rhythmic patterns. The main reason for the great interest in bio-inspired approaches is the fact that bio-robots provide suitable solutions for the design of efficient locomotion as they appear as simple solutions to hard problems. The challenge issue is how to realize the biological movement using the mechanical structures. For example, mimicking snakes motion. The movement of snakes is generated by the frictional force between its body and ground. Its geometric shape is used to control the frictional force and can change its speed by changing the shape of its body. Until now, the current technology is not sophisticated enough to resemble the muscles and joints of the animals. Thus, it is crucial to develop simple mechanisms and control that can generate similar biological motions such as CPG-based method.

CPG-based control has been utilized in various kind of bio-inspired robots. By manipulating the CPG parameters, different gaits or locomotion patterns of bio-inspired robots such as hexapod, salamander and others can be realized. Chunlin and Low [11] have designed locomotion control of a fish robot with fin propulsion. The CPG-based control method illustrates a way to switch motion patterns including swimming by flapping pins, turning by modulating phase relation of fins, and online transition of different motion by simply tuning corresponding parameters. In [13], the authors applied CPG-based approach to control locomotion of quadruped robots by proposing a control architecture with a 3-D workspace and a motion engine. By tuning the parameters of the CPG network, online adaptive workspace trajectories were generated using the proposed architecture. Seo et al [12] proposed CPG-based control strategy to control autonomous turtle-like underwater

vehicle by using contraction theory. They also implemented feedback of the actuator states to the coupled oscillators for environment adaptation. All of these papers [11] [13] [12] are some examples of various mobile robots that use CPG as the control approach. CPG oscillator has also been tested on a humanoid chewing robot by generating rhythmic and voluntary patterns of mastication [40].

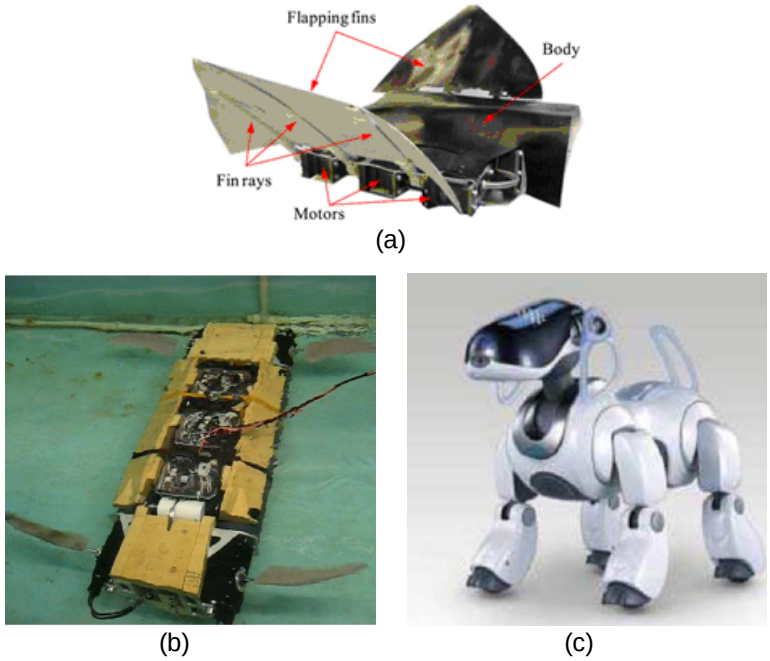


Figure 1.7. Examples of various mobile robots that uses CPG as their control method (a) Fish-like robot [11], (b) Turtle-like robot [12] and (c) Quadruped robot [13].

CPG-based control has been applied in many snake-like robots for various purposes. Crespi and Ijspeert [41] utilized CPG-locomotion control for online optimization of amphibious snake robot. The authors address problem in locomotion control of robots with multiple degrees of freedom mainly a snake-like robot by searching the fast swimming and crawling gaits to adapt to variety of environments using CPG-based approach. Lu et al [42] analyzed a snake-like robot controlled by cyclic inhibitory CPG model where all the parameters of the CPG are fixed during the locomotion. The authors present some necessary conditions for the CPG network to sustain a rhythmic output and verified the proposed control architecture into a real snake-like robot called "Perambulator". Wu and Ma [43] improved the CPG connection model proposed in [42] and analyzed the effect of CPG pa-

rameters to the CPG output without changing any parameters at any time instantly. They also proposed a new network with feedback connection which can generate uniform outputs without any adjustment. Tang et al [44], proposed a self-tuning multi phase CPG which enable a snake-like robot to adapt to environments. The proposed CPG control architecture can change the movement patterns and control parameters autonomously according to external information.

1.5 Outline of this Thesis

CPG-based approach has shown its flexibility in controlling many kinds of mobile robots with various desired motions. This control method can be applied in different terrains including aquatic and terrestrial environments. However, all of the previous and current designs of CPG network have either drawbacks: complicated CPG networks, high computational cost, and/or complex interneurons connection to produce rhythmic output.

The purpose of this research is to overcome the fundamental problem of the CPG-based control for a snake-like robot in term of the the body shape control for environment adaptability using a simple CPG network. A simple CPG network with combination of a simple mathematical model is preferable which can reduce the computational cost, easy to be implemented and handled by human operator who may not familiar with CPG. The critical issue of snake-like robots control, that is, online parameters modification of the CPG-based is also being analyzed further due to its significance in rescuing and searching tasks. Online modification of the body shape of snake-like robots is crucial for autonomously adapt to its environment which is fundamental to biological snakes motion.

The contents of this thesis are organized as follows.

Chapter 2 introduces the proposed CPG network with its mathematical model to control locomotion of a snake-like robot. The proposed CPG network is compared with other existing CPG network by other researches for controlling a snake-like robot in terms of the structure model and the convergence speed to phase locking state. The selection of

mathematical model, that is, the phase oscillator, to represent our CPG network is also described in details. Finally, the characteristics of the CPG parameters are discussed for its frequency control and feedback connection to attain the rhythmic output. Moreover, we also presents the stability analysis of the proposed CPG network. The stability analysis is important to prove the limit cycle behavior of the CPG network against any random perturbation. Based on the proposed CPG network, a new parameter is introduced to control the frequency and coupling weight between the oscillators to control deterioration of the output wave. Then, some examples of locomotion control of a snake-like robot are shown using the proposed control method. The basic locomotion of a snake-like robot, that is, the S-shape control and also the forward and backward movement are verified through simulations. Discussions of the overall analysis and results are presented at the end of this chapter.

Chapter 3 analyzes the fundamental problem of locomotion control of a snake-like robot. The main point addressed in this chapter is a method that produces a smooth transition of the body shape of a snake-like robot. By manipulating the phase difference of the CPG outputs instantly, it will result in a sharp point or discontinuity which leads to an unstable movement of the snake-like robot. To tackle the problem, a method of controlling the body shape is proposed: by incorporating activation function in the phase oscillator CPG model. The simplicity of the method promises an easy implementation and simple control. Simulation results and torque analysis confirm the effectiveness of the proposed control method and thus, can be used as a locomotion control in various potential applications of a snake-like robot.

Chapter 4 focuses on the versatility of the snake-like robot locomotion based on our CPG approach. This chapter is divided into two parts. The first part explains the straight path control of a snake-like robot. Forward straight path locomotion of a snake-like robot can be easily controlled using any control architecture such as sine-based, model-based or others. However, a problem occurs when one or more of the control parameters are modified online, where the trajectory of the snake-like robot will deviate from its original path. The

novelty of the approach is the online body shape transition of the snake-like robot which can be used to control the forward straight path of the snake-like robot. The second part describes the control of turning motion of a snake-like robot for obstacle avoidance using phase difference modulation. The proposed turning method is compared with amplitude modulation method by other researchers which shows advantages in the proposed method. Simulation results and torque analysis confirm both of the proposed straight-path control and turning methods can be used efficiently into a snake-like robot.

Chapter 5 presents the experimental verifications for the proposed CPG network in controlling various locomotion presented in each chapter. Details of the physical snake-like robots are explained. The control systems and the software programming used for the experiments are introduced and briefly explained in Appendix.

Chapter 6 concludes this thesis and discusses the possible works in the future.

Chapter 2

Analysis of CPGs with Phase Oscillator Model

Many efforts have been done in designing network architecture using CPG for locomotion control of mobile robots. One of the important aspect in designing a control approach is a simple network with less computation and easy to be implemented practically for searching and rescuing tasks in real environments. This requires not just a versatile design of robot, but should also integrate a simple control structure.

In this thesis, we focus on the CPG-based control for a snake-like robot to mimic the typical serpentine locomotion of snakes, which is the most efficient among all modes [45]. Previous works done for controlling locomotion of a snake-like robot utilizing CPG-based method have either complicated network or complex interneurons connections to produce rhythmic output, which requires high computational cost. Furthermore, the selection of mathematical model to represent the CPG oscillators is important for producing limit cycle behavior. The mathematical model should also has a clear relationship of their parameters and output for easy control and implementation into the robot. Also, CPG approach offers the possibility to integrate sensory feedback signals in the differential equation, which gives advantage among other control architectures.

This chapter firstly introduces different kinds of network structure of coupled oscillators and dynamical equations that are available for representing the the CPG oscillators. Each of the network structure and mathematical model can be used for producing rhythmic inputs with proper design depending on the applications and the kind of robots.

Then, we propose a simple unidirectional CPG structure with the mathematical model for controlling a snake-like robot locomotion. A comparison with previous work [41] which proposed a bidirectional structure is also presented. Finally, characteristics of the CPG parameters are discussed.

2.1 CPG Mathematical Model and Network

The coupling of two oscillators is fundamental for the construction of artificial CPGs. Depending on the network design and the selection of the mathematical model, the performance of the control architecture will differ. For instance, one directional coupling between oscillators is adequate for maintaining stable phase difference but the gait transition may be slow if large amount of oscillators are used in a CPG model. This problem can be solved by increasing the coupling strength. However, depending on the selection of the mathematical model, a large coupling strength will distort the shape of the output wave.

On the other hand, if we use mutual coupling scheme, it requires high computational cost than the one directional coupling, but the gait transition will become faster. Thus, it is crucial to determine the appropriate network structure that can be integrated with simple mathematical model to produce high locomotion performance of the robot.

2.1.1 CPG Mathematical Model

The motivation of modeling is to study on how the oscillator couplings affect the synchronization and the phase lags within the a populations of oscillators. The dynamics of population of oscillators depends mainly on the topology of network rather than on the local mechanisms of rhythmic generation [46] [47]. The ability of the CPG network to produce

and switch between multiple gait patterns is a model-independent phenomenon, that is, it does not depend upon the detailed dynamics of the component oscillators and/or the nature of the inter-oscillator coupling. The interaction of a CPG model with the environment offers the possibility to study the effect of sensory feedback on the CPG activity. In some cases, the CPG models have been coupled to biomechanical simulation of a body, in which case they are called neuromechanical models [48] [49].

There are common assumptions in the dynamical approach to CPG: the nonlinear oscillators are often assumed to be identical, the stepping movements of each joint are controlled by a single oscillator, while inter-joint coordination is provided by the connections between the oscillators. Moreover, the oscillator may represent either single motor-neuron or population of neurons controlling the movement of a joint or an inter-neuron.

One difficulty with CPG-based approaches is to determine how to design the CPG to produce a particular pattern. There are many types of oscillator models such as van der Pol, Stein, FitzHugh-Nagumo [50], Matsuoka oscillators [10] [51] [52] [53] and Hopf oscillator [5] [54]. Van der Pol oscillators have been used extensively in physiological modeling studies [55] and later modified by Bay and Hemami [56] for the coupling between the oscillators. Stein neuronal model is capable of producing oscillatory output, which has 3 sets of coupled nonlinear differential equations. Inhibitory coupling between the CPG oscillators was achieved by decreasing the magnitude of the inhibited oscillator's driving signal by an amount proportional to the inhibiting oscillator's signal. The FitzHugh-Nagumo model, was developed and analyzed by FitzHugh [57] and Nagumo et al [58]. The inhibitory coupling was implemented in a manner similar to that for the Stein CPG model, that is, the coupling served to decrease the driving signal acting on the inhibited oscillator. However these oscillators model (van der Pol, Stein, FitzHugh-Nagumo) are designed in a biological perspective which are structurally complicated and not suitable for purposive control or numerical analysis in engineering viewpoint [43]. Plus, the stability analysis of the oscillator models have not been proven.

Matsuoka oscillators provides several parameters that influence the CPG output. The

model of CPG neuron proposed by Matsuoka has the features of continuous-time and continuous-variable in its simple structure, and can thus be easily implemented into the control of a snake-like robot with a large number of DOFs. Stability analysis and mathematical proof have been done for generating rhythmic output [42]. However, realization of smooth rhythm waves depends on the inhibitory model and number of neurons. Furthermore, some of the parameters are interrelated to each other and need to follow some numerical conditions. It makes the robots control become complicated and difficult, which loses its simplicity. Meanwhile, for Hopf oscillator, the coupling terms between oscillators need to be further derived from the nonlinear differential equation. Most of the CPG models mentioned previously do not have explicit parameters defining quantities such as frequency, amplitude, and wavelength (for instance, a van der Pol oscillator does not have explicit frequency and amplitude parameters).

To overcome all of the above mentioned drawbacks by other oscillator models, phase oscillator has been adopted as the dynamic model representing our CPG coupled oscillators. An interesting aspect of the phase oscillator model is the combination of the simplicity of the sine-based approaches with the robustness of the CPG due to the possesses of limit cycle behavior by the phase oscillator. The parameters of the phase oscillator have explicit relationship to the output, where the frequency, phase, and amplitude are directly influence the CPG output, which provides easy control. Another advantage of using CPG-based model over the sine-based is the possibility to integrate sensory feedback signals in the differential equation.

2.1.2 CPG Network Model

Several topological structures of CPG exist when more oscillators are involved. The CPG structures can be mainly categorized into five types: the chain type [54] [59] [60], the ring type [60] [61] [62], the radial type [63], the fully connected type [64] [65], and the combinations of the three types aforementioned [66]. Besides, given one type of topological structure, the coupling profiles can be different. Fig. 2.1 shows the four fundamental

structures. The oscillators are not restricted to four, but can be added depending on the applications. Until now, no standard principles exist that determine which structure should be adopted. Basically, the selection of the topological structure depends on the actual situation of a robot application and the type of robot.

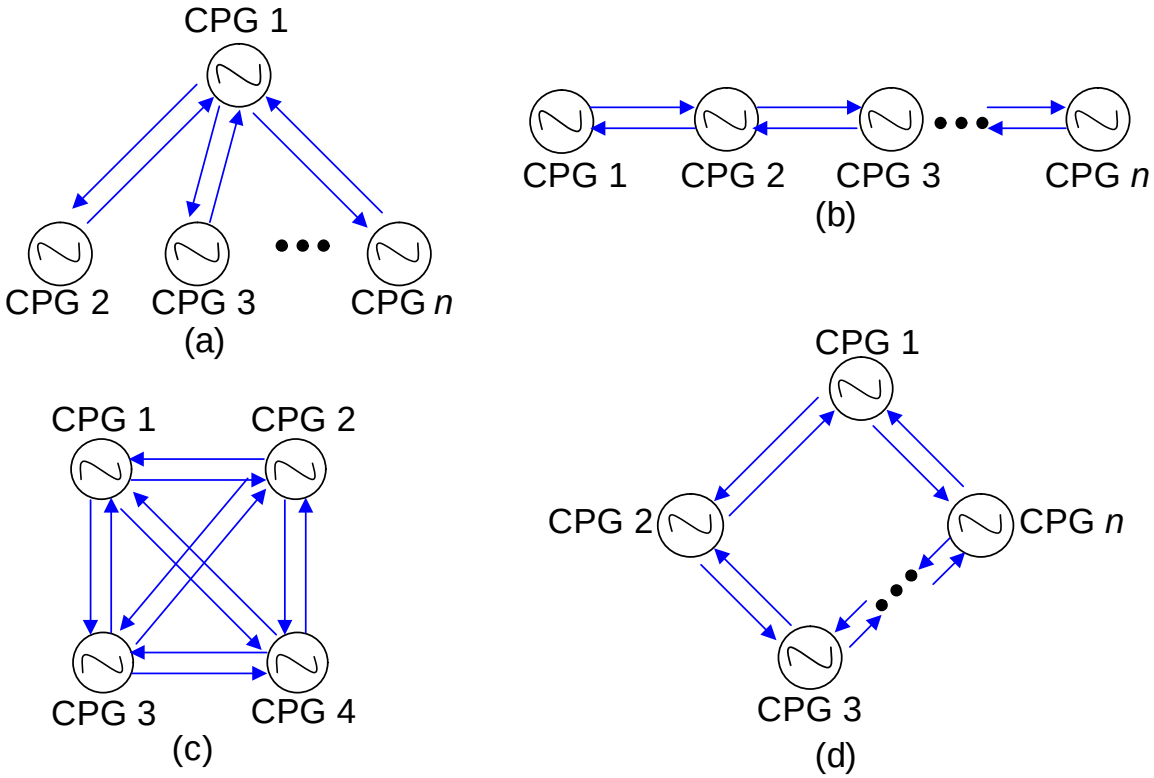


Figure 2.1. Common topology structures of coupled oscillators (a) radial type, (b) chain type, (c) fully connected type, and (d) ring type.

The radial type CPG structure is governed by central neural system which is called centralized control. The hierarchy of the structure is not restricted to the higher command of oscillator and its low level oscillators. The structure can be in multiple layers based on the robot applications. The higher level oscillator will always receive feedback from the low level command for synchronization of activity. Some applications of this kind of structure are fish-like robot [11] [63]. In [11], each fin of the fish-like robot is driven by three brushless servo motors and totally six oscillators are needed to generate the swimming gaits for six fin rays. Three oscillators are used to control one side fin. On each side, the oscillators are coupled with the first oscillator as a radial connection.

The fully connected type is usually used to control quadruped robots [50] [61] [67] [65]. In [61], four CPG oscillators were used to control four legs respectively. The oscillators need to cooperate to each other to make the quadruped hoping robot jump continuously by applying the same periodic force to each spring of the robot's leg. Hence, the fully connected type of CPG structure was adopted to obtain stable, continuous and rhythmical hopping performances. Due to the strong coupling of the parameters, the parameter analysis of the network is the major difficulty for the engineering application of the CPG-based control methods [65]. To overcome the problem, in [65], the authors added mapping signals from the fully connected CPG structure to modulate the CPG parameters. However, it only solves the difficulty of modulating the CPG parameters, but complex calculations and structure still remains.

The ring type resembles the fully connected except for their connection among the nearest neighbor oscillators. For fully connected type, the oscillators are connected to its neighbor oscillators and the crossover oscillators. In [60], the authors firstly use the chain type CPG structure to control the fins. However, the gait transition is slow and therefore modified the CPG structure to the ring type. All oscillators are synchronized to the first one with different stable phase lags depending on the position of the associated fin ray on the long fin. In [62], the authors adopted ring type CPG structure to match to the mechanical system of the robot. The CPG structure has a brain control and the oscillators are regulated by feedback information.

The chain type CPG structure is the simplest connection among others. This type of structure connected the oscillators to its nearest neighbor oscillator in an open loop form. This is the different to the ring type structure, where the oscillators are connected in a form of closed loop. Usually, due to its simplicity, this kind of CPG structure is adopted for various mobile robots such as snake-like robot [41], hexapod robot [54], fish-like robot [60] and other compliant robots [59]. In [60], the CPGs are connected in the way that the behavior of one CPG is perturbed by the previous one in order to maintain a stable phase difference between them, which forms a one-way chain structure. However,

the gait transition under this chain structure is slow. The reason is that each oscillator is synchronized with the previous one and the lag dynamics propagates and accumulates one by one. However, depending on the applications, the chain type CPG structure is more suitable in compliant robots such as snake-like robot and salamander-like robot [68].

Besides the selection of CPG structure, coupling way between oscillators should also be considered. Generally, coupling between oscillators can be categorized into two: unidirectional coupling and bidirectional coupling. Examples of bidirectional coupling can be found in [11] [50] [61] [65] [62] and unidirectional coupling in [43] [50] [60]. Oscillators coupled bidirectionally will send and receive information with each other. By applying a bidirectional coupling scheme, gait transition can be performed faster. The disadvantage of this scheme is that it requires more computing cost and the parameters cannot be easily modulated due to the strong coupling between the oscillator. On the other hand, unidirectional coupling is easy to be implemented in robot control with less computational cost. However, gait transition will be slow if large amount of oscillators is used in the CPG model. This problem can be reduced by increasing the coupling strength, but the shape of the output wave will distort.

Due to the advantages and disadvantages of each of the CPG structures and coupling scheme, a careful selection and analysis are required to match to the mechanical system of a particular robot. Thus, less complexity and computational cost, but with higher performance is preferable for controlling a mobile robot.

2.2 CPG Model for Generating Rhythmic Motion

This section describes the mathematical model and structure of our proposed CPG model. We propose a simple control structure, less computational cost with high efficiency for controlling a snake-like robot. Furthermore, we analyze our proposed CPG structure with bidirectional structure proposed by Crespi and Ijspeert [41] in terms of coupling complexity, mathematical computation and convergence speed to phase locking state. The

characteristics of the CPG parameters are discussed for its frequency control and feedback connection to attain the rhythmic output.

2.2.1 Phase Oscillator

Nonlinear model of phase oscillator has been utilized to control locomotion of amphibious snake robot and modular robot [41] [69]. The mathematical model describing a system of one phase oscillator is as follows:

$$\dot{\theta}_i = 2\pi v_i + \sum_{j\neq i}^{j=l} w_{ij} \sin(\theta_j - \theta_i) \tag{2.1}$$

Detailed explanations of the parameters in (2.1) can be referred in Table 2.1. This oscillator model spontaneously produces traveling waves with constant phase difference between neighboring segments along the body, and it is made of multiple oscillators connected as a double chain, and (2.1) has been modified [41] as follows:

$$\dot{\theta}_i = 2\pi v_i + \sum_{j\neq i}^{j=l} w_{ij} \sin(\theta_j - \theta_i - \phi_{ij}) \tag{2.2}$$

where ϕ_{ij} is the phase bias. Here, ϕ_{ij} is used to determine the phase lag between oscillators. Coupling term between the oscillators is defined by the weights w_{ij} and the phase bias ϕ_{ij}. The output for each oscillator is as follows:

$$x_i = A\cos(\theta_i) \tag{2.3}$$

Description of parameters in (2.2) and (2.3) are shown in Table 2.1:

The simplicity of the phase oscillator model reduces the computational cost. There is no looping process in the algorithm such as the optimization algorithm (for example: genetic algorithm) which requires high internal memory of processor. On the other hand, the CPG model is well suited to be programmed on a microcontroller on board of the robot

Table 2.1. Description of the parameters

Items	Details
θ_i	Phase of the ith oscillator
θ_j	Phase of the jth oscillator
v_i	Intrinsic frequency
A	Amplitude
w_{ij}	Coupling weights between oscillators
l	Number of oscillators
ϕ_{ij}	Phase bias
x_i	Rhythmic and positive output signal
$\dot{\theta}_i$	The time evolution of the phase θ_i

because of the direct relationship of the CPG parameters to the CPG output. The CPG mathematical model does not require many steps and is easily solved.

2.2.2 Structure of CPG Model

For the phase oscillator model, bidirectional coupling is usually adopted to control one actuator. Fig. 2.2 (a) illustrates the structure of multiple phase oscillator connected as a double chain. This type of coupling structure has been implemented by Crespi and Ijspeert [41]. Fig. 2.2 (b) shows our proposed CPG structure using the phase oscillator model, which is unidirectional coupling. Referring to Fig. 2.2 (b), for simplicity, we named these oscillators from left to right as: oscillator 1, oscillator 2, oscillator 3 and so on in ascending order. Referring to Fig. 2.2, the direction of arrow for each oscillator indicates the receiving θ_j (inward) from neighbor oscillator or transmitting θ_i (outward) to neighbor oscillator.

For bidirectional coupling, two neurons are adopted to control one output signal. For simplicity, we named the upper part as *[u]*, and the lower part as *[l]*. The mathematical model used to describe θ_i as in Fig. 2.2 (a) for *i=1, 2, 3* is as follows:

$$\begin{aligned}\dot{\theta}_{1[l]} =& 2\pi v_{1[l]} + w_{1[l]2[l]} \sin(\theta_{2[l]} - \theta_{1[l]} + \phi_{1[l]2[l]}) \\ &+ w_{1[l]1[u]} \sin(\theta_{1[u]} - \theta_{1[l]} - \phi_{1[l]1[u]})\end{aligned} \tag{2.4}$$

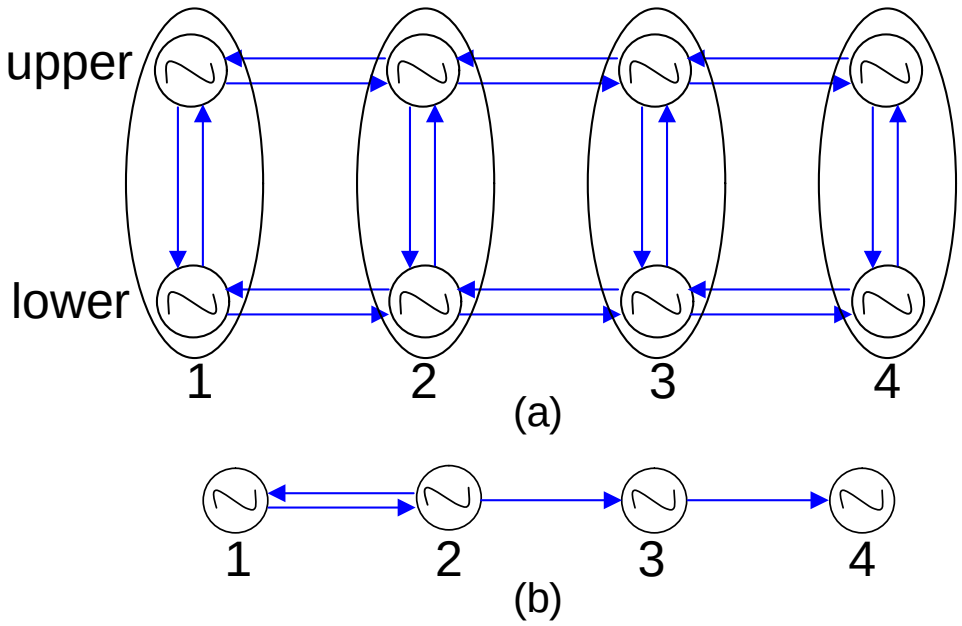


Figure 2.2. Structure of coupling oscillator: (a) bidirectional coupling, and (b) unidirectional coupling.

$$\begin{aligned}\dot{\theta}_{2[l]} =& 2\pi v_{2[l]} + w_{2[l]1[l]} \sin(\theta_{1[l]} - \theta_{2[l]} - \phi_{2[l]1[l]}) \\ &+ w_{2[l]3[l]} \sin(\theta_{3[l]} - \theta_{2[l]} + \phi_{2[l]3[l]}) \\ &+ w_{2[l]2[u]} \sin(\theta_{2[u]} - \theta_{2[l]} - \phi_{2[l]2[u]})\end{aligned} \tag{2.5}$$

$$\begin{aligned}\dot{\theta}_{3[l]} =& 2\pi v_{3[l]} + w_{3[l]2[l]} \sin(\theta_{2[l]} - \theta_{3[l]} - \phi_{3[l]2[l]}) \\ &+ w_{3[l]4[l]} \sin(\theta_{4[l]} - \theta_{3[l]} + \phi_{3[l]4[l]}) \\ &+ w_{3[l]3[u]} \sin(\theta_{3[u]} - \theta_{3[l]} - \phi_{3[l]3[u]})\end{aligned} \tag{2.6}$$

For simplicity but without loss of generality, the value of ϕ_{ij} can be respectively assigned to ϕ for descending connections and $-\phi$ for ascending connection. For upper and lower connections, the value of ϕ_{ij} is set to $-\phi$ for downward connections, and ϕ for upward connections. The CPG network proposed in [41] is structurally complicated and difficult to analyze numerically. Besides, it is not easy to change the phase difference because of the strong coupling between the oscillators. Furthermore, the number of connections in bidirectional network as in Fig. 2.2 (a) is three times more than our network which

increases the calculation time significantly. Due to the said drawbacks, we propose simple CPG coupling so-called unidirectional coupling for a snake-like robot locomotion control. As shown in Fig. 2.2 (b), only one neuron is adopted for controlling one output signal, and the mathematical model used to describe θ_i for *i=1, 2, 3* is as follows:

$$\dot{\theta}_1 = 2\pi v_1 + w_{12} \sin(\theta_2 - \theta_1 - \phi_{12}) \tag{2.7}$$

$$\dot{\theta}_2 = 2\pi v_2 + w_{21} \sin(\theta_1 - \theta_2 + \phi_{21}) \tag{2.8}$$

$$\dot{\theta}_3 = 2\pi v_3 + w_{32} \sin(\theta_2 - \theta_3 + \phi_{32}) \tag{2.9}$$

The design of our CPG network is rather simple in structure by connecting it into unidirectional CPGs network (see Fig. 2.2 (b)). In this model, the oscillators are connected in a direction from head to the tail of the snake-like robot. Referring to Fig. 2.2 (b), the first two oscillators are coupled bidirectionally because every phase oscillator needs to have input from other CPG in order to produce harmonic rhythmic output. The value of ϕ_{ij} is set to ϕ for descending connections and $-\phi$ for ascending connection. The desired joint angles for unidirectional coupling are described as:

$$joint\_angle[i] = x_i \tag{2.10}$$

The behavior of θ_i and desired joint angles of the unidirectional CPGs network are shown in Fig. 2.3. The parameters used are: $v_i = 0.1$, $w_{ij} = 10$, $A = 1$, and $\phi = \pm\pi/3$ for all CPGs. It is obvious that the output waves are homogeneously distributed with phase difference equal to ϕ. With unidirectional network, multi numbers of equations can be eliminated.

2.2.3 Convergence Speed to Phase Locking State

In previous subsection, we have already compared the complexity of the synapse connections between the two CPG models, that is, bidirectional and unidirectional structures. The

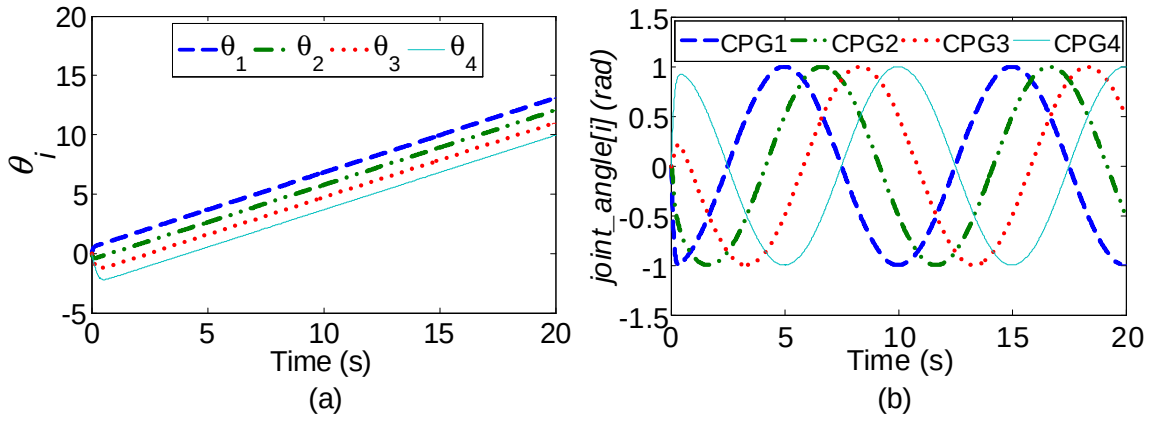


Figure 2.3. Unidirectional CPG network: (a) behavior of θ_i, and (b) joint angle.

results conclude that the proposed unidirectional structure has several advantages over the bidirectional structure. In this subsection, we further analyze our proposed CPG model by comparing the two CPG models, based on the convergence speed to the phase locking state. We compared both the convergence speed of the unidirectional and bidirectional structure using the same CPU frequency and internal memory of 2.80GHZ and 3.00GB respectively with 64-bit operating system. The parameters used for both CPG models are: $w_{ij} = 10$, and $A = 1$. We simulate the results only for four CPG couplings and the interpretation of the CPGs is similar as in Fig. 2.3.

In Table 2.2, we show results of the convergence speed during the initial state to the phase locking state for the two CPG models. We analyze the convergence speed by varying the value of $v_i = 0.1$ and 1.0, and value of $\phi_{ij} = \pi/2$, $\pi/3$, and $\pi/6$.

Table 2.2. Comparison of convergence speed

	Convergence speed (s)		
v_i	Unidirectional	Bidirectional	ϕ_{ij}
0.1	1.4	0.9	$\pi/2$
0.1	1.2	0.8	$\pi/3$
0.1	1.2	0.7	$\pi/6$
1.0	1.4	0.8	$\pi/2$
1.0	1.1	0.7	$\pi/3$
1.0	1.1	0.7	$\pi/6$

From the results in Table 2.2, the convergence speed for bidirectional network is a bit

faster than our proposed unidirectional coupling. But the difference of the convergence speed between these two models is only less than 0.5s, which we consider to be small and acceptable while comparing with motion time. It thus can be disregarded.

Based on our analysis, the convergence speed does not depend on the value of v_i. There are two factors that affect the convergence speed for the initial state to be at the phase locking state: 1) the convergence speed increases as the number of oscillator increase, and 2) the convergence speed increases slightly as the value of ϕ_{ij} increase.

2.3 Analysis of CPG Model with Unidirectional Coupling

In this section, we will show that our proposed structure is much simpler and provides the same stability as the bidirectional coupling. The control of phase difference of the unidirectional oscillator will be proved to be easier rather than the bidirectional coupling. Some modifications also have been done in (2.2), to control the smoothness of the CPG outputs as well as to control the speed of the snake-like robot.

2.3.1 Descriptions of the Phase Oscillator Model

Firstly, for further understanding, we describe the similarity of the terms of the phase oscillator model with the conventional sine-based model used for controlling the rhythmic output wave, as shown in Table 2.3.

Table 2.3. Similarity of the terms between sine-based and phase oscillator (CPG-based)

	Sine-based	CPG-based
Equation	$x_i = A\sin(\omega t + i\phi)$	$\dot{\theta}_i = 2\pi v_i + \sum_{j\neq i}^{j=l} w_{ij}\sin(\theta_j - \theta_i - \phi_{ij})$ $x_i = A\cos(\theta_i)$
Amplitude	A	A
Frequency	ω	$2\pi v_i$
Phase difference	ϕ	ϕ_{ij}
Output	x_i	x_i

The sine-based uses simple time-indexed sine-based functions, while the CPG-based uses

dynamical systems of coupled nonlinear oscillators to generate traveling waves. Even though both control architectures have similar terms, the sine-based method have two drawbacks: i) it is not efficient to modify its parameters online, and ii) it is not easy to integrate sensory feedback signals. Thus, it is clear that the parameters of the phase oscillator model have clear relationship with the output. With this advantage, it is easy to control and adjust the parameters accordingly.

For our proposed CPG-based model, the parameters for v_i, w_{ij}, and ϕ_{ij} are set to have the same value for all of the CPG oscillators in controlling the locomotion of our snake-like robot. In other word, we define $v_i = v$, $w_{ij} = w$, and $\phi_{ij} = \phi$ for all oscillators.

2.3.2 Feedback Connection of the CPGs

In this section, we analyze the feedback connection of the oscillator 1. For our proposed CPG structure (Fig. 2.2 (b)), the feedback connection is set from oscillator 2 to oscillator 1. With the set configuration, we can get homogeneous oscillations distribution between the oscillators output. The significant advantage of our proposed CPG structure with the feedback connection is that, the phase difference can be controlled by controlling parameter ϕ as shown in Fig. 2.3 (b). To further clarify the uniqueness of our CPG structure, we analyze the CPG outputs behavior when changing the feeding connection from other oscillators to oscillator 1. The parameters used are: $v = 1$, $w = 10$, and $A = 1$.

In Fig. 2.4, we show the behavior of the CPG output x_i in term of ϕ when we change the feedback connection to oscillator 1. The x-axis in Fig. 2.3 denotes oscillator 2, oscillator 3, and oscillator 4 while the y-axis denotes the value of ϕ with respect to the feedback connection. We analyze various values of ϕ (in degree) for $\phi = 90°$, $60°$, $45°$, and $30°$. From Fig. 2.3, we can conclude that there is a nonlinear change in value of ϕ as the feedback connection varies increasingly with the sequence of oscillator. Therefore, the feedback connection must be fixed from oscillator 2 to oscillator 1 to get the desired phase difference based on the set value of ϕ.

We further analyze the changing of the sign of ϕ for oscillator 1. In our proposed CPG

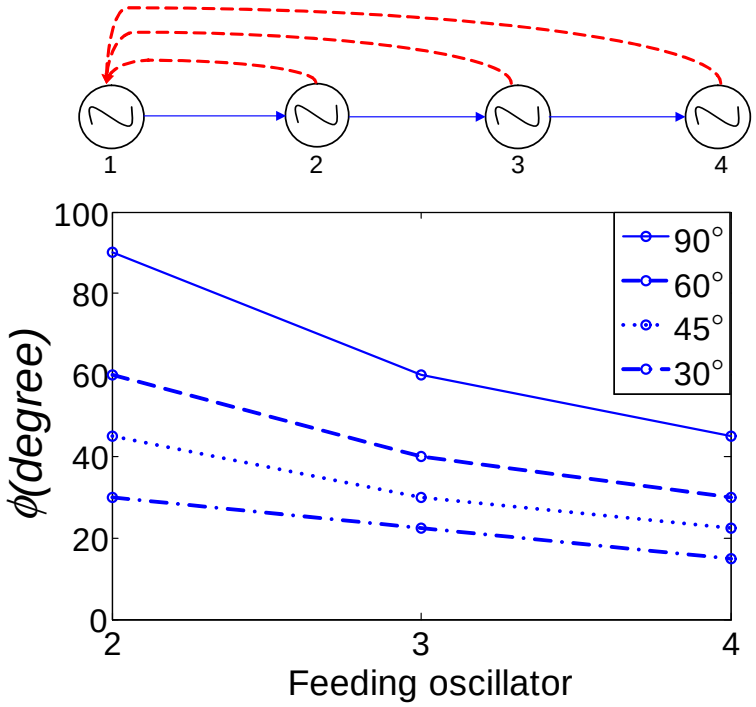


Figure 2.4. Behavior of ϕ when the feedback connection is changed.

structure, the sign of the first oscillator (oscillator 1) must be reversed with other oscillators. In Fig. 2.5, we can see that the outputs x_i are in ascending order when ϕ for oscillator 1 is negative, whereas in Fig. 2.5, the outputs x_i are in descending order when ϕ for oscillator 1 is positive.

Based on our analysis, to produce homogeneous waves we can conclude the following:

1) Oscillator 1 must always get feedback connection from oscillator 2;

2) as the feedback connection varies increasingly with the sequence of oscillator, the value of ϕ change with nonlinear decreasing;

3) the sign ($\pm$) for ϕ for oscillator 1 should always be opposite to the other oscillators;

4) if sign of the ϕ for oscillator 1 is positive (+), the CPG outputs will be in descending order;

5) if sign of the ϕ for oscillator 1 is negative (-), the CPG outputs will be in ascending order.

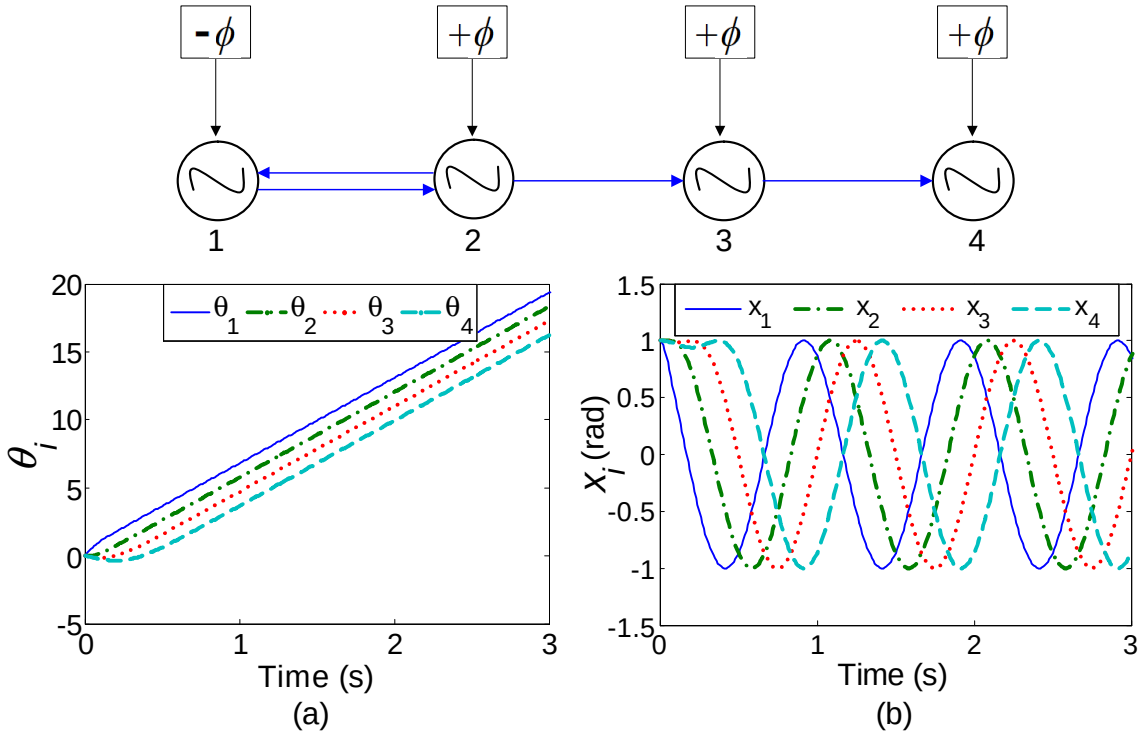


Figure 2.5. Behavior of CPG when ϕ for oscillator 1 is negative (-): (a) θ_i, and (b) output x_i.

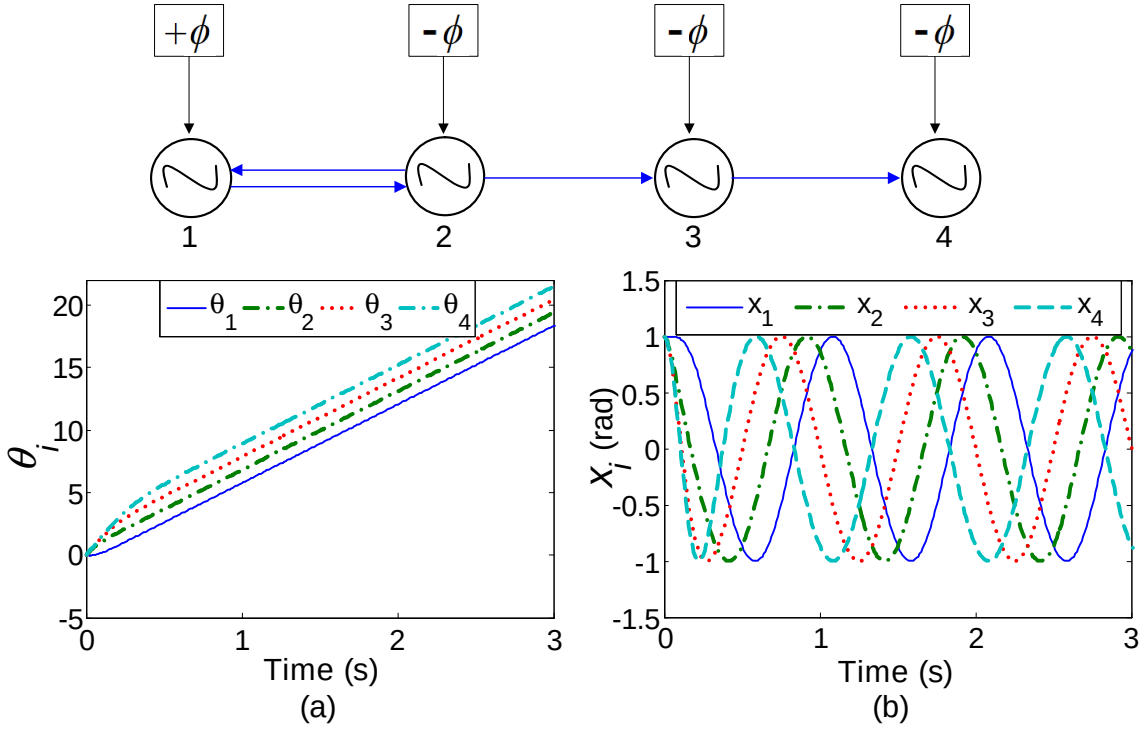


Figure 2.6. Behavior of CPG when ϕ for oscillator 1 is negative (+): (a) θ_i, and (b) output x_i.

2.3.3 Phase Difference Synchronization

Stability analysis for two bidirectional coupled oscillator for determining synchronization of the phase difference can be referred to [41]. Also, more in depth analysis of networks of phase oscillators can be found in [69]. In this section, we are interested to determine whether the unidirectional coupled oscillators (Fig. 2.2 (b)) will synchronize at constant phase difference.

The phase biases between the first two oscillators, oscillator 1 and oscillator 2 (refer Fig. 2.2 (b)) must always be in opposite sign to obtain the rhythmic output with constant phase difference. For other oscillators i.e., oscillator 3, oscillator 4 and so on, the sign of ϕ must follow the sign of ϕ for oscillator 2. From (2.9), consider the case for oscillator 3:

$$f(\theta_3) = \dot{\theta}_3 = 2\pi v + w\sin(\theta_2 - \theta_3 + \phi) \tag{2.11}$$

For the first oscillator, we use $+\phi$ and -ϕ for other oscillators. The equilibrium occurs when the right hand sides of the first order system, (2.11), simultaneously vanishes [70]:

$$2\pi v + w\sin(\theta_2 - \theta_3 + \phi) = 0 \tag{2.12}$$

Solving (2.12) for θ_3, we obtain:

$$\theta_3^{\infty} = \theta_2 + \phi + \sin^{-1}(2\pi v/w) \tag{2.13}$$

From (2.13), we can see that θ_3 will always evolve at constant phase difference of $\theta_2 + \phi + \epsilon$, where $\epsilon = sin^{-1}(2\pi v/w)$. In order to get the output of oscillator, x_i to converge to oscillations that are phase locked with a phase difference of ϕ, we must have:

$$v << w \tag{2.14}$$

So that ϵ will be relatively small and can be ignored. Therefore, the output will become:

$x_3^{\infty}(t) = \cos(\theta_2 + \phi)$. Here, we can see that ϕ is a phase difference between θ_i and θ_{i-1}, where:

$$\theta_3 = \theta_2 + \phi \tag{2.15}$$

Therefore:

$$\begin{aligned} \phi &= \theta_3 - \theta_2 \\ &= \theta_i - \theta_{i-1} = \Delta\theta_i \end{aligned} \tag{2.16}$$

If rule in (2.14) is obeyed, the solution will be asymptotically stable because:

$$\left.\frac{\partial f(\theta_3)}{\partial \theta_3}\right|_{\theta_3^{\infty}} = -w\cos(\theta_2 - \theta_3 + \phi) \tag{2.17}$$

Substitute (2.13) into (2.17), we obtain:

$$\frac{\partial f(\theta_3^{\infty})}{\partial \theta_3} = -w\cos(\sin^{-1}(2\pi v/w)) \tag{2.18}$$

(2.18) shows that the solution of $\partial f(\theta_3^{\infty})/\partial\theta_3 < 0$ at all value of ϕ. Note that (2.14) should be satisfied all times for every type of network connection because if value of $v \approx w$, the output wave will deteriorate.

2.3.4 Frequency Control

From the previous section, we can see that the output oscillator will evolve at constant phase difference $\pm\phi$. But, the effect of (2.14) to the output is that we need to control two parameters: v and w in order to produce smooth output. Parameter v can be used to control speed it determines the locomotion speed of the snake-like robot [41]. From our analysis by simulation, as we control the frequency, v, we need to adjust w as well. The CPG output will deteriorate if value of v is nearly to w, or if value of w is too large compared to v. To overcome this issue, we introduce a variable, τ as a single parameter to control both the frequency, v and weight, w of the oscillator. The modification of (2.2) is as follows:

$$\tau\dot{\theta}_i = 2\pi v + \sum_{j\neq i}^{j=l} w\sin(\theta_j - \theta_i - \phi) \tag{2.19}$$

The variable τ will not affect the phase locking of the output, where (2.13) can still be maintained even though we introduced a new variable in (2.19). Fig. 2.7 shows the effect of using τ where Fig. 2.7 (a) is the outputs without variable τ, and Fig. 2.7 (b) is the outputs with variable τ. For easy view, we bold one of the CPGs output. From Fig. 2.7, we can see that by introducing variable τ, the frequency of the output can be controlled without deteriorating the output or without needing to adjust the w , even the intrinsic frequency, v is small.

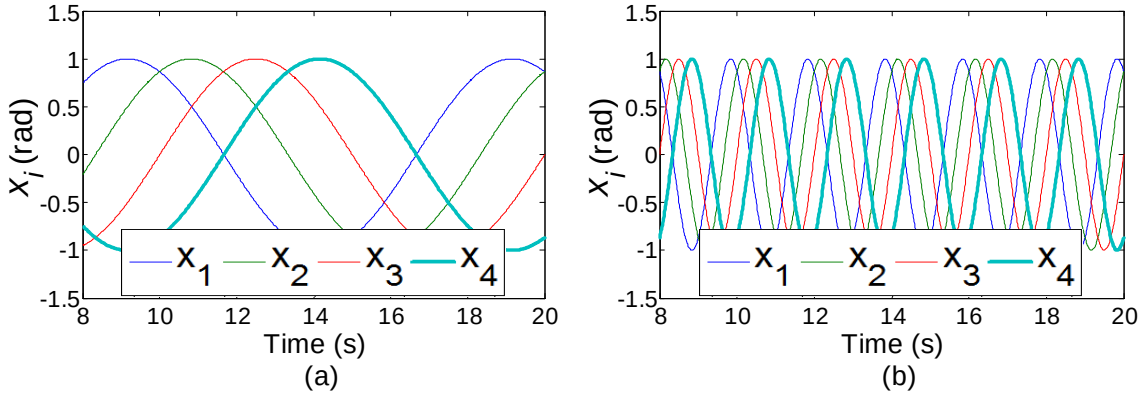


Figure 2.7. Effect of variable τ to the output, (a) without τ, and (b) with $\tau = 0.2$.

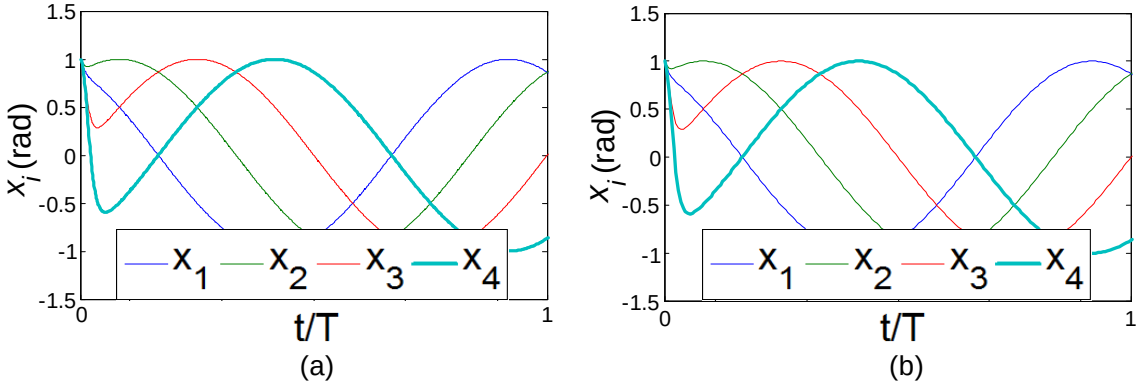


Figure 2.8. Convergence rate to phase locking state, (a) without τ, and (b) with $\tau = 0.2$.

In Fig. 2.8, we also analyze whether τ gives any effects of convergence rate to phase

locking state (i.e., stable state of the CPG output, x_i). Since the frequency is controlled by adjusting τ, we normalized the time by the output period, T in Fig. 2.7. We can see that the convergence rate is the same in both Fig. 2.8 (a) and Fig. 2.8 (b), which is less than 0.1s and thus, by introducing parameter τ, the convergence rate of the CPG output, x_i can still be maintained.

From (2.19), the value τ should be ≤ 1 to give effect to the output of CPG. If $\tau > 1$, the performance of the output will deteriorate.

2.3.5 Effects of Parameter τ to CPG Output Frequency

Fig. 2.9 shows how the parameter τ affects the output frequency. The value of τ is 1, 1/5, 1/10, 1/15, and 1/20 with constant intrinsic frequency, $v = 0.1$ and $w = 10$. As the value of τ decreases, the output frequency will increase. But as τ gets smaller, the tendency for the output frequency to increase is slower. Referring to the line connecting between points in Fig. 2.9, we can see that the slope is getting smaller as τ decreases.

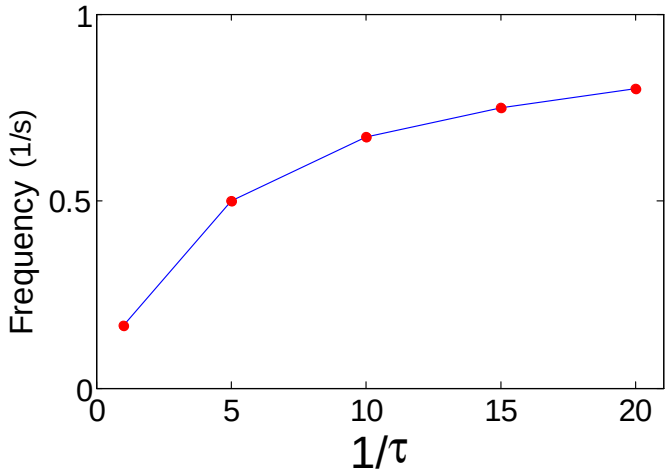


Figure 2.9. Effect of τ to the output frequency.

From Fig. 2.9, we can deduce that there is a limitation for adjusting the output frequency. The range of τ can be set from 1 to 1/15 to make the changes of frequency to be apparent. Also, if we set the value of τ to be much smaller, the output wave will tend to

deteriorate. For application to a snake-like robot, the frequency can be used as a control parameter of the robot's speed.

2.3.6 Frequency Transition

One of the requirements of high performance locomotion control of robots is a smooth transition between different gaits. This is to avoid jerky or discontinuous locomotion which may damage the gear box or actuator of the robots. To prove the proposed CPG model can rapidly adapt to parameter change, we analyzed the reaction of the new parameter τ, by switching its value at random time instantly, whilst other CPG parameters are kept constant for all period of time.

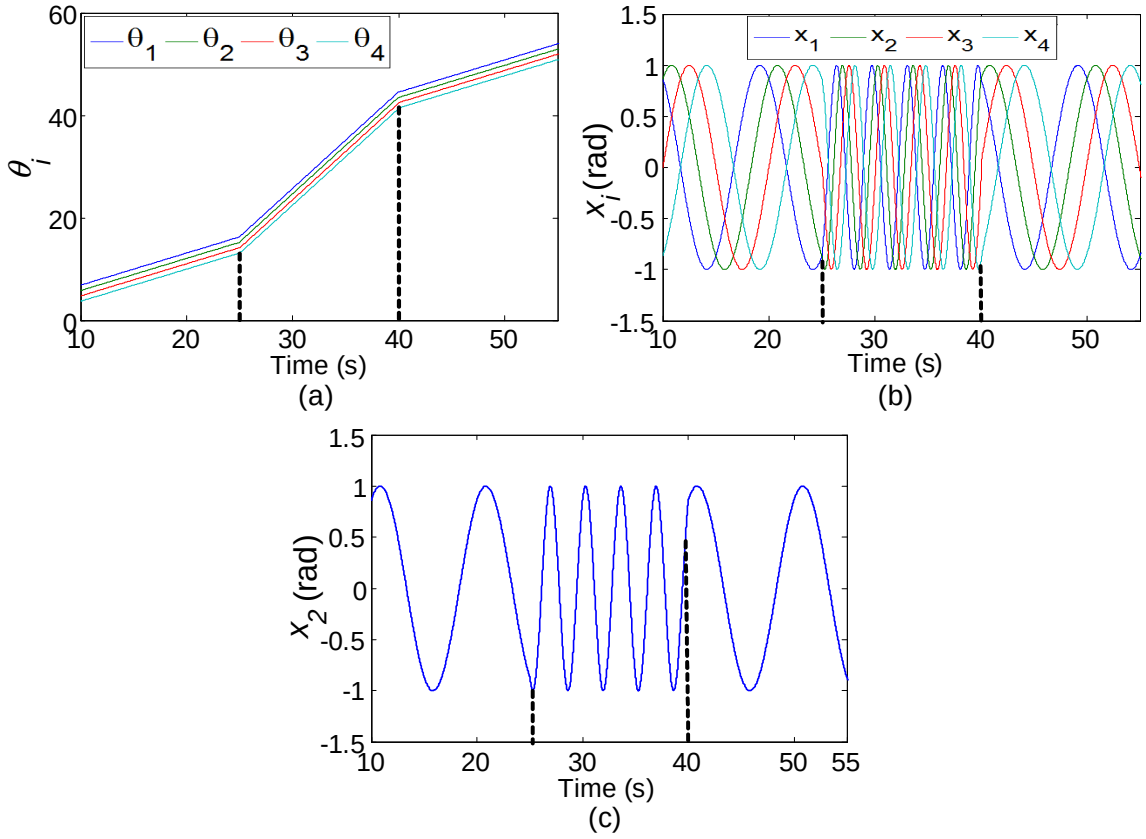


Figure 2.10. Behavior of CPGs when changing τ at random time of 25s and 40s: (a) θ_i with respect to time, (b) output x_i (rad), (c) example of output of one CPG.

Fig. 2.10 shows the behavior of our CPG model when changing the value of parameter τ at random time of 25s and 40s. From this figure, we can see that the output is stable during

the transition. The output converges to the desired traveling wave after a short transient period. We can deduce that the CPG is adaptive to the change of parameter τ.

2.4 Implementation into a Snake-like Robot

Snake-like robot is one of an example of a mobile robot which has large degree of freedoms. Many researches on controlling this kind of robot have been done based on interaction between mutually connected CPGs. For a snake-like robot, the phase difference should be adjusted continuously and must be easily controlled from head to tail.

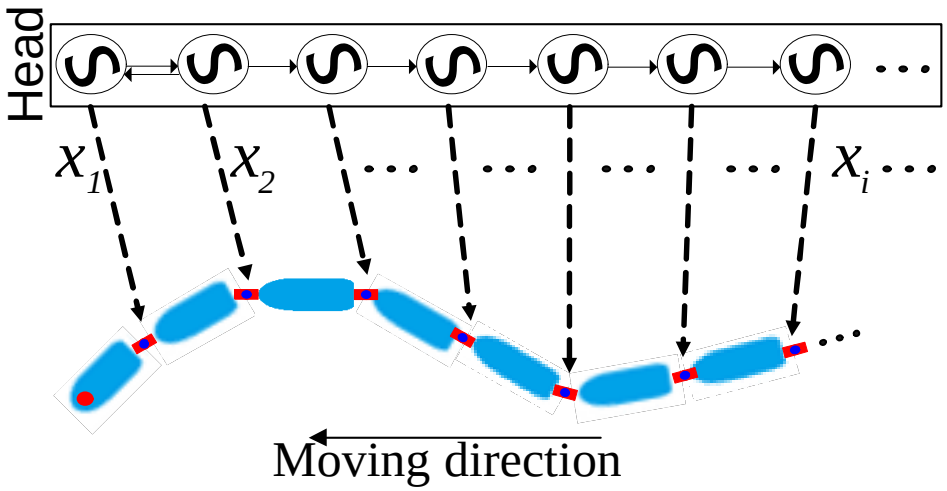


Figure 2.11. CPG network implemented to control snake-like robot.

Our CPG network can be implemented in a snake-like robot as shown in Fig. 2.11. One CPG will control one joint angle of the snake-like robot.

The propelling force of the serpentine motion on a snake-like robot comes from the interaction of the robot with the ground by swinging the joints from side to side [2]. Therefore, it is crucial to control the joint angle with constant phase difference.

Number of S-shape

Using the proposed structure of unidirectional coupling in Fig. 2.2 (b), the total phase difference, ϕ_{total} is given as follows:

$$\phi_{total} = n\phi \tag{2.20}$$

where n is the number of actuated joints from head to tail of the snake-like robot. To get one S-shape locomotion, the total phase difference should be equal to 2π. This is because, the CPG output of each of the joint actuator of the snake-like robot is in sequence with time at constant phase difference. Thus, the number of S-shape of the snake-like robot locomotion can be given as:

$$N = n\phi/2\pi \tag{2.21}$$

By changing the value of ϕ, we can obtain the desired serpentine locomotion of the snake-like robot, where (2.21) can be rearranged as follows:

$$\phi = 2\pi N/n \tag{2.22}$$

Hence, using (2.22), we can control the number of S-shape of the serpentine locomotion. Note that, the calculation of number of S-shape is the same for both unidirectional and bidirectional couplings.

2.5 Simulation Results

Simulation of a snake-like robot has been conducted to verify the proposed CPG-based control method using open dynamic environment (ODE). Physical features of the snake-like robot are listed in Table 2.4 where two passive wheels are adopted to realize the swinging movement from side to side with ground asymmetric friction where the normal friction

coefficient μ_N is larger than the tangential friction coefficient μ_T. Number of S_u can differ depending on the number of joint, n.

Table 2.4. Physical parameters of simulated snake-like robot

Items	Details
Snake units	$S_u = n$
Size of snake unit	0.05x0.08x0.14m
Weight of snake unit	$w = 0.1$kg
Radius of wheel	$r_w = 0.06$m
Thickness of wheel	$t_w = 0.015$m
Weight of wheel	$w_w = 0.01$kg
Friction coefficient	$\mu_N = 0.5$; $\mu_T = 0.01$

2.5.1 Control of Number of S-shape

The simulation results of the number of S-shape of the snake-like robot are shown in Fig. 2.12. Based on (2.22), we change the value of ϕ, to obtain different number of S-shape. For instance, with number of actuated joint, $n = 5$ (equal to s_u - 1), we can obtain desired number of S-shape by plugging different value of N. In our analysis, we used $N = 0.8$, and 1, where the value of $\phi = 8\pi/25$, and $2\pi/5$ respectively. Note that, the maximum or minimum number of S-shape is strictly depended on the physical features of the snake-like robot. Due to the limited numbers of the robot link, there will be a restriction in the number of S-shape that the snake-like robot can perform. In our case, the largest number of S-shape is $N = 1$. Thus, controlling the number of S-shape is one of the important criteria for the snake robot.

2.5.2 Forward and Backward Movement

Based on our analysis of the feedback connection of the CPGs, by changing the sign of ϕ of the first oscillator, we can change the output sequence of the CPGs. With that, using ODE simulation, we further analyze what is the effect to the locomotion of the snake-like robot. Since the CPGs output will be in reverse direction when we change the sign of ϕ, we then can control the movement direction the snake-like robot, as shown in Fig. 2.13.

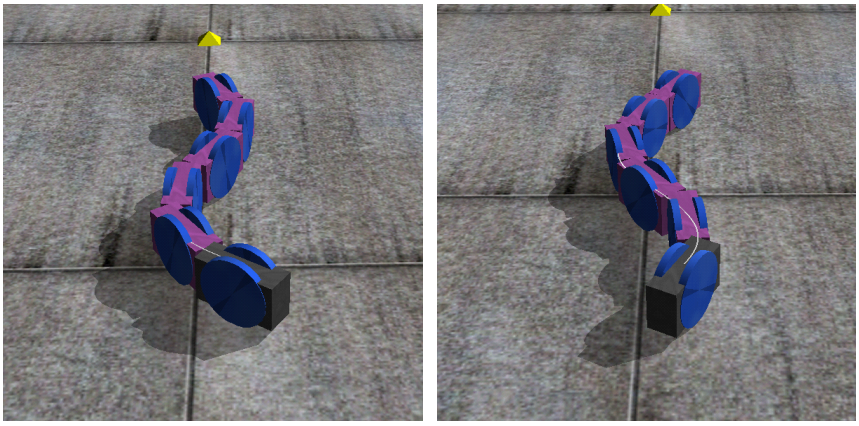


Figure 2.12. Different number of S-shape obtain by simulation (a) $\phi = 8\pi/25$ ($N = 0.8$), (b) $\phi = 2\pi/5$ ($N = 1$).

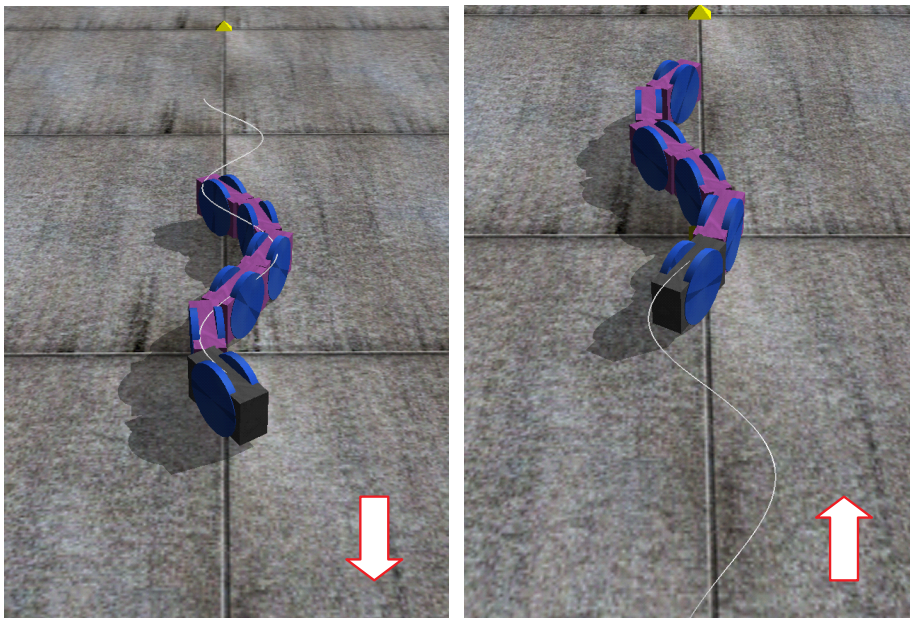


Figure 2.13. Movement of a snake-like robot by simulation (a) Forward movement (b) Backward movement.

The wave line in Fig. 2.13 is the trajectory of the center of mass (COM) of the head of the snake robot. In Fig. 2.13 (a), all the other units of the snake-like robot follow the head trajectory with forward motion. In Fig. 2.13 (b), the head trajectory line is at the back of the snake-like robot because of the backward movement. Therefore, the other snake units will not follow the head trajectory line. With this advantage, we can easily control the forward and backward movement of our snake-like robot just by changing the sign of the parameter ϕ.

2.6 Discussion

Based on the simulation results, we have verified the proposed unidirectional CPG structure can be used to control the locomotion of a snake-like robot. By changing parameter ϕ, we can control: 1) number of S-shape of the snake-like robot, and 2) forward and backward movement. However, the maximum number of S-shape should depend on the number of the snake-like robot link. The higher the link number, the higher number of S-shape can be performed. Also, there are specific procedures for the unidirectional CPG structure in order to realize the locomotion control of the snake-like robot as mentioned in Section 2.3.

Furthermore, we have compared our CPG structure with the bidirectional structure. With our proposed structure, we can achieve approximately the same convergence speed, less mathematical computation, and simplified structure. With just a mutual coupling between the first two CPGs, we can control our snake-like robot locomotion successfully with homogeneous distribution between the output of the oscillators.

Comparing with the proposed CPG network by Wu [10], the mathematical model of the CPG (Matsuoka oscillator) has indirect relationship between the parameters and its output. A strong coupling between the parameters and the output increases the difficulty in controlling the snake-like robot. Besides, there are limitations of the value of the parameters which diminish the simplicity of the parameters selection. Though a simple unidirectional coupling is used to control the snake-like robot, the feedback connection needs to be changed

to control the number of S-shape of the snake-like robot. This task is complicated for online modifications of the number of S-shape and is not practically achievable. Further comparison of the locomotion control is discussed in Section 4.1.4.

2.7 Summary

In this chapter, we have proposed an approach to design a CPG network with simplified structure to control the phase difference between coupled oscillators. Our proposed CPG network has been proven to have: 1) simple structure, 2) less computational cost, 3) fast convergence speed, and 4) converge to phase locking state. The phase difference can be used to control the serpentine locomotion using the phase oscillators as the CPG mathematical model. An additional parameter has been added to the phase oscillator to control the frequency and the smoothness of the CPG output simultaneously. With the proposed unidirectional network, the serpentine locomotion can be controlled by adjusting desired number of S-shape in one period.

Chapter 3

CPG-based Body Shape Control of a Snake-like Robot

One of the interesting features of a natural snake is its body shape transition. Natural snakes will change their body shape depending on the space it travels. Therefore, our goal is to mimic the body shape transition feature into to a snake-like robot, to make it capable traveling in different types of space and to avoid obstacles online. The target is to switch motion patterns by simply tuning a single parameter. The basic idea of control is shown in Fig. 3.1:

The wave line in Fig. 3.1 is the shape movement of the snake robot where the number of S-shape of the snake robot increases or decreases depending on the space it travels or avoiding an obstacle. The amplitude transition of the body shape of the snake-like robot hence needs to be controlled during locomotion especially for rescue or searching purposes in hazardous or dangerous environments where the operation space is beyond human reached.

Most of researchers focus on controlling the locomotion of mobile robots when moving forward/backward with rhythmic movement patterns using CPG-based control. However, switching locomotion mode of the mobile robots from one type to another online, has not yet being analyzed. The output signals show unstable or discontinuous waves if the CPG

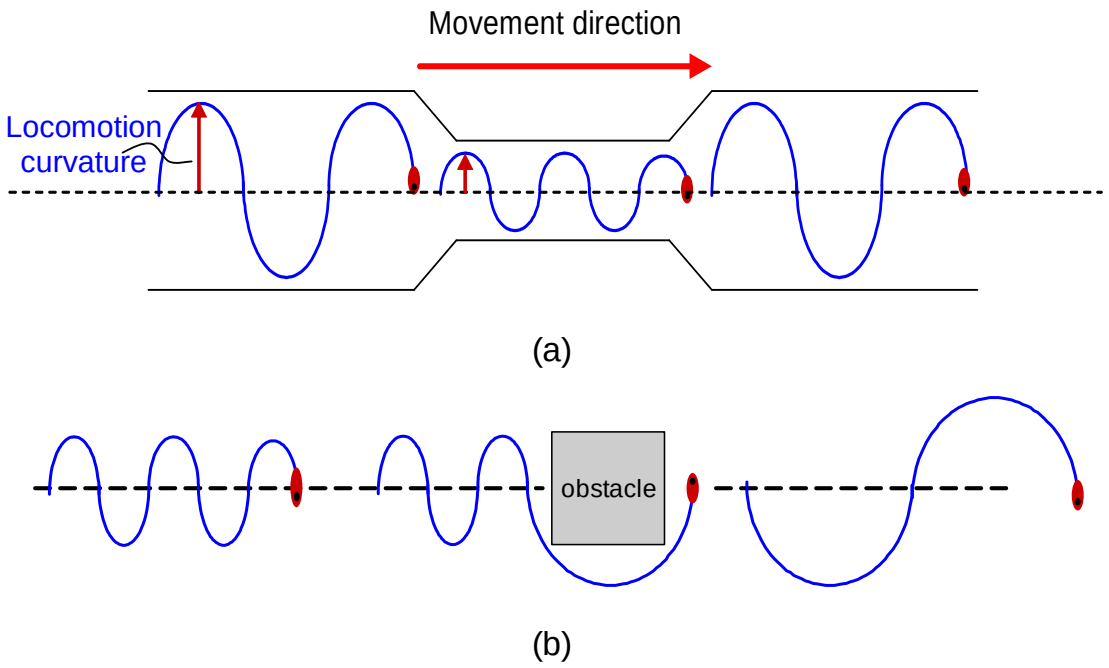


Figure 3.1. Illustration of snake movement: (a) in various dimensions of pipe, (b) avoiding obstacle.

parameters are modulated instantly and cannot be directly changed when the robot is online. To the extend of our knowledge, most of the previous works done by other researchers do not consider transition of gaits or locomotion modes online especially for the body shape transition of a snake-like robot.

Some papers do consider gait transitions, for instance [41] [68], but there is no analysis on online transition. The authors mentioned on how to control the CPG parameters for gait transitions, but the analysis is done separately. In other words, the authors modified the parameters offline. In [68], the authors provide the CPG outputs for the transition between swimming to walking of a salamander-like robot, but there is no simulation or experiment verification of the online transition of the locomotion modes. All of these papers [41] [54] [68], show deterioration in the CPG outputs when modifying the CPG parameters instantly, but did not further discuss about the phenomena. To the best of our knowledge, the discontinuity in the CPG outputs will affect the locomotion's stability of a robot.

Online transition of locomotion is a very significant requirement for a rescue robot. Thus, this chapter will mainly discuss the said problem and propose a way to solve the online transition issue focusing on body shape control of a snake-like robot.

3.1 Problem of Direct Phase Transition

Using (2.22), we can obtain the desired serpentine locomotion of the snake-like robot, by modifying the value of ϕ. To change body shape of a snake-like robot as shown in Fig. 3.1, we can directly input the desired number of S-shape, N and obtain the necessary value of ϕ. The obtained value of ϕ then will be input in the modified phase oscillator model ((2.19)) to achieve the required output CPG, x_i. Since the goal is to change the body shape of a snake-like robot online, first, we need to investigate the behavior of the CPGs output during direct ϕ transition, where the effect of manipulating the value of ϕ instantly at random time is analyzed. Fig. 3.2 shows the simulation results when ϕ is changed from $\pi/4$ to $\pi/2$. For easy view, we just show result of two outputs of the CPGs.

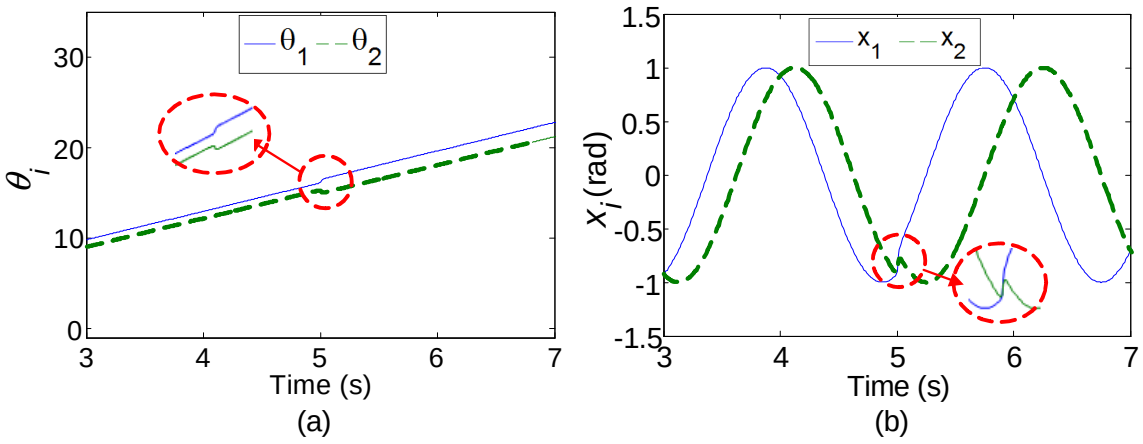


Figure 3.2. Behavior of CPGs when ϕ is changed from $\pi/4$ to $\pi/2$ at time 5s: (a) θ_i with respect to time, (b) effect of θ_i to the CPGs output x_i.

The outputs of CPGs produce sharp curve or discontinuity during the transition of ϕ from $\pi/4$ to $\pi/2$ at time $= 5s$. This non-smooth transition is due to the sudden change of ϕ. Thus, manipulating ϕ directly to the joint motor of a snake-like robot might cause unstable movements.

There are two main problems of the phase transition: 1) there is a sharp edge at the transition as shown in Fig. 3.2, and 2) there is more than one CPGs output to control the snake-like robot, which produces different behavior of curve at the transition point, as shown in Fig. 3.3. The transition point is defined as the point where the phase transition starts.

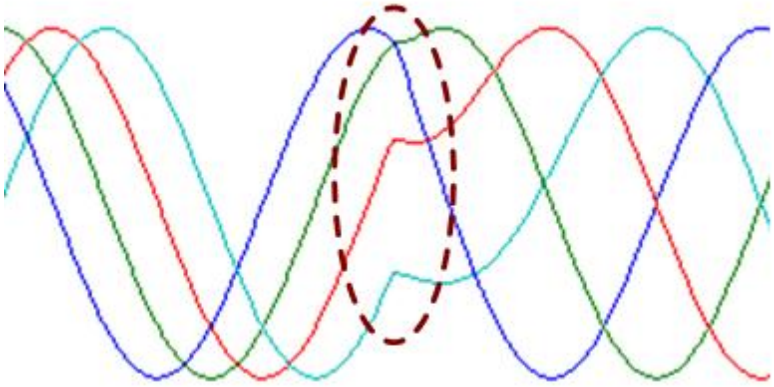


Figure 3.3. Different behavior of CPG outputs at the transition point.

In Fig. 3.3, as number of CPG outputs increase, the behavior of transition point for each output is different depending on the value of the output, x_i during the transition time. From analysis, as the output, x_i approaching its peak value, there is smoother phase transition at the transition point.

For further understanding, Fig. 3.4 (a) shows example of one of our CPG output with different value of ϕ. The value of ϕ for the left side curve in Fig. 3.4 (a) is $\pi/4$ and the right side curve is $\pi/2$. In Fig. 3.4 (b), the two curves are joined together at their end/start points, and clearly, it is continuous but it has a visible crease which is unacceptable for locomotion of a snake-like robot. This sharp join results in sudden change or jerky movement of the locomotion of the snake-like robot, which may damage the motor and gearbox of the snake-like robot.

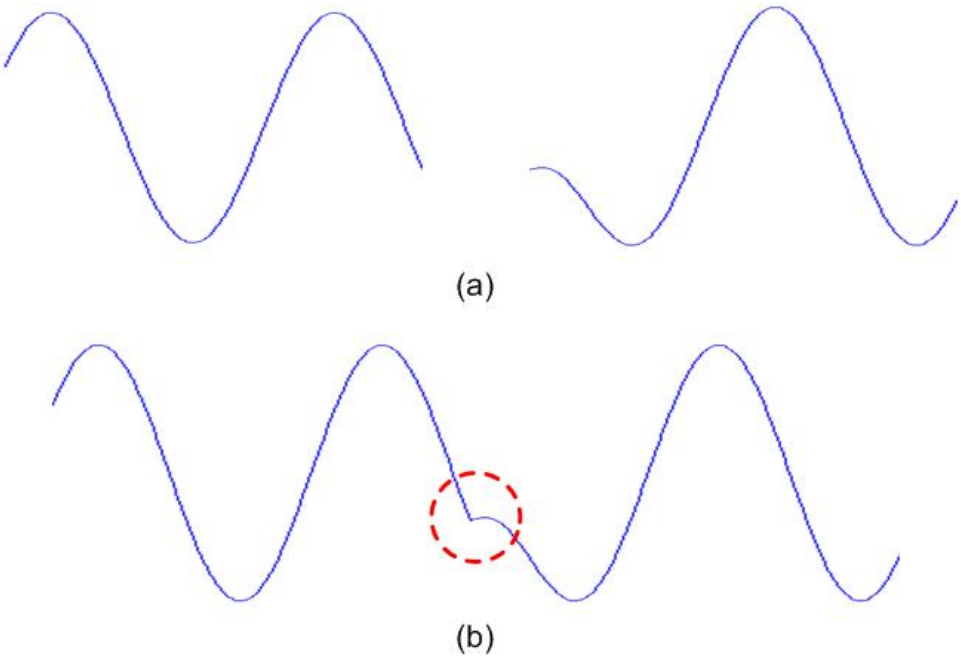


Figure 3.4. Illustration of transition point of CPG output between two different value of ϕ: (a) left side curve is $\pi/4$ and the right side curve is $\pi/2$, (b) the two curves are joined together at their end/begin points.

3.2 Criteria of Desired Continuity and Smoothness of Locomotion

This section will discuss the definition of continuity and smoothness of the desired CPG outputs curve. The measurement of continuity and smoothness of a curve is based on parametric continuity, C_n and geometric continuity, G_n. Geometric continuity requires the geometry to be continuous, while parametric continuity requires that the underlying parameterizations be continuous as well. Further explanation on the continuity can be found in [71]. In our case, the degree of smoothness that we desire for our snake-like robot locomotion is at least C_2 and G_2. It is easy to check that the CPG outputs are always 0^{th}-order continuity, C_0 and G_0. To analyze C_1 of the modified phase oscillator model (2.19), we need to calculate its derivative with respect to time, t. Thus, by using a chain rule, the derivative of (2.19) is obtained as follows:

$$\tau\ddot{\theta}_i = -w_{ij}[\dot{\theta}_i \cos(\theta_j - \theta_i - \phi_{ij})] \tag{3.1}$$

and we differentiate again (7) with respect to t, to analyze C_2:

$$\tau\dddot{\theta}_i = w_{ij}[[\dot{\theta}_i^2 \cos(\theta_j - \theta_i - \phi_{ij})] - \ddot{\theta} \cos(\theta_j - \theta_i - \phi_{ij})] \tag{3.2}$$

From (3.1) and (3.2), it can be directly figured out that the two equations do not imply C_1 and C_2 by substituting $\phi_{ij} = \phi_1$ before the phase transition and $\phi_{ij} = \phi_2$ after the phase transition.

Additionally, the curve of the CPG outputs produce a kink during the phase transition (refer Fig. 3.2 (b)). It does not imply G_1 and G_2 as well, because the tangent direction changes discontinuously at the kink. In Fig. 3.5, we define the criteria of a smooth curve based on the geometric continuity. In the next subsection, we will propose an approach to solve the addressed phase transition problem.

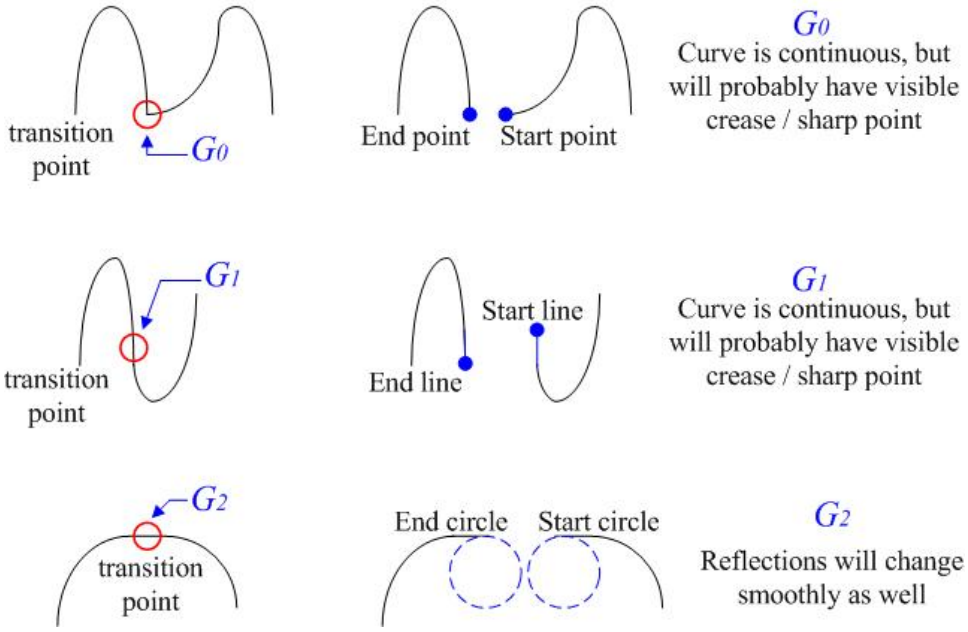


Figure 3.5. Description of geometric continuity for G_0, G_1 and G_2.

3.2.1 A method for Smooth Phase Transition - Introducing an Activation Function

In previous section, we have already shown the effect of manipulating parameter ϕ instantly. As a result of the sudden change of ϕ, the outputs of the CPG will not be smoothed, which can causes an unstable movement of the snake-like robot. Thus, we introduce a method to control the phase transition by introducing an activation function. For simplicity, we have selected linear bipolar as our activation function, where ϕ change linearly with respect to time during the phase transition.

The linear bipolar has been selected as activation function for modulating smooth phase transition because of its output behavior that is linearly increased with the independent variable. Fig. 3.6 illustrates the main idea of manipulating ϕ from ϕ_1 to ϕ_2 as time t increases from t_1 to t_2. ϕ can be either increased or decreased as time increases. We will derive a general mathematical model that describes both situations as depicted in Fig. 3.6 (a) and Fig. 3.6 (b).

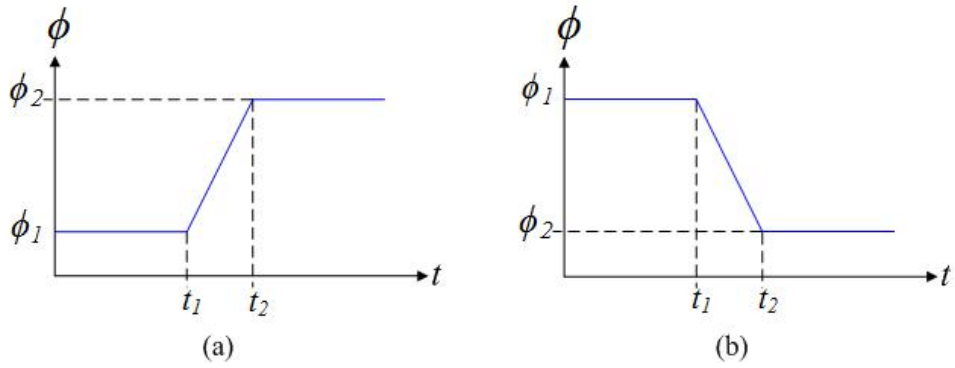


Figure 3.6. Linear bipolar: (a) ϕ linear increase, (b) ϕ linear decrease.

The mathematical model that describes relationship between ϕ and time, t so that at t_1, ϕ is ϕ_1 and at t_2, ϕ will be ϕ_2 is as follows:

$$\phi = \phi_1 - \alpha(t_1 - t) \tag{3.3}$$

$$\alpha = \phi_1(N_2/N_1)(1 - (N_1/N_2))/(t_2 - t_1) \tag{3.4}$$

where

$$\phi = \begin{cases} \phi_1 & t \leq t_1 \\ \phi_1 - \alpha(t_1 - t) & t_1 < t < t_2 \\ \phi_2 & t \geq t_2 \end{cases} \tag{3.5}$$

The details derivation of a mathematical model for (3.3) and (3.4) that describes relationship between ϕ and time, t so that at t_1, ϕ is ϕ_1 and at t_2, ϕ will be ϕ_2 are as follows:

Step 1: Derive the relation between ϕ_1 and ϕ_2 by manipulating (2.22): $n = 2\pi N \,/\, \phi$, we obtain:

$$\phi_1 = (N_1/N_2)\phi_2 \tag{3.6}$$

Step 2: Derive general equation for the linear bipolar

Let the general equation be (K and z are unknowns to be derived):

$$\phi = Kt + z \tag{3.7}$$

At t_1:

$$z = \phi_1 - Kt_1 \tag{3.8}$$

At t_2:

$$z = \phi_2 - Kt_2 \tag{3.9}$$

Substitute (3.6) into (3.8):

$$z = (N_1/N_2)\phi_2 - Kt_1 \tag{3.10}$$

Equate (3.9) and (3.10):

$$\phi_2 - Kt_2 = (N_1/N_2)\phi_2 - Kt_1 \tag{3.11}$$

Solving for K:

$$K = \phi_2(1 - (N_1/N_2))/(t_2 - t_1) \tag{3.12}$$

Substitute (3.12) into (3.8), and solve for z:

$$z = \phi_1 - [\phi_2(1 - (N_1/N_2))/(t_2 - t_1)]t_1 \tag{3.13}$$

Step 3: The linear bipolar function

let:

$$\alpha = \phi_1(N_2/N_1)(1 - (N_1/N_2))/(t_2 - t_1)$$

Finally we obtain:

$$\phi = \phi_1 - \alpha(t_1 - t)$$

From (3.3), the S-shape locomotion can be easily controlled by substituting the desired value of N_2. The parameters ϕ_1 and N_1 are all predefined constant values. $(t_2 - t_1)$ is the transition time of the phase difference ϕ_1 to ϕ_2. t_1 is the trigger time for the phase transition.

3.2.2 Analysis of Parametric Continuity

To begin, the analysis should be divided into two parts because it is a piecewise linear activation function. The first part is at $t = t_1$, while the second part is at $t = t_2$. To analyze C_1 properties, we refer to (3.1). For the first part of the analysis, we substitute ϕ_1 into (3.1):

$$\tau\ddot{\theta}_i = -w_{ij}[\dot{\theta}_i \cos(\theta_j - \theta_i - \phi_1)] \tag{3.14}$$

Then, we substitute (3.3) into (3.1):

$$\tau\ddot{\theta}_i = -w_{ij}[\dot{\theta}_i \cos(\theta_j - \theta_i - (\phi_1 - \alpha(t_1 - t)))] \tag{3.15}$$

at $t = t_1$, we obtain:

$$\tau\ddot{\theta}_i = -w_{ij}[\dot{\theta}_i \cos(\theta_j - \theta_i - \phi_1)] \tag{3.16}$$

For the second part of the analysis, we repeat the same procedure as the first part of the analysis at $t = t_2$, and obtain the following:

$$\tau\ddot{\theta}_i = -w_{ij}[\dot{\theta}_i \cos(\theta_j - \theta_i - \phi_2)] \tag{3.17}$$

for both left-hand side and right-hand side of t_2.

The analysis of the C_2 continuity is similar with the C_1 continuity, using (3.2), where we can directly obtain the following:

$$\tau\dddot{\theta}_i = w_{ij}[[\dot{\theta}_i^2 \cos(\theta_j - \theta_i - \phi_2)] - \ddot{\theta}\cos(\theta_j - \theta_i - \phi_2)] \tag{3.18}$$

for both left-hand side and right-hand side at both t_1 and t_2.

The results directly clarify the C_1 and C_2 continuity of the piecewise activation function.

3.2.3 Analysis of Geometric Continuity

Fig. 3.7 shows the results of the outputs of CPG when applying the derived linear bipolar function (3.3) into our CPG model. There is a smooth linear changed during the ϕ transition. In this simulation, we use $\phi_1 = \pi/4$, $\phi_2 = \pi/2$, $n = 8$, and $t_2 - t_1 = 1s$. Here, ϕ_1 corresponds to $N_1 = 1$, and ϕ_2 corresponds to $N_2 = 2$. In Fig. 3.8, we invert the value of ϕ_1 to ϕ_2 and vice versa while keeping the other parameters unchanged.

By incorporating our proposed activation function, we can obtain smooth linear change of the CPGs output. In Fig. 3.7, at transition time from $5s$ to $6s$, the output x_i changes smoothly with time.

In order to verify the desired geometric continuity, G_2 is achievable using our proposed method, we calculate the difference in slope of θ_i before and after the phase transition at

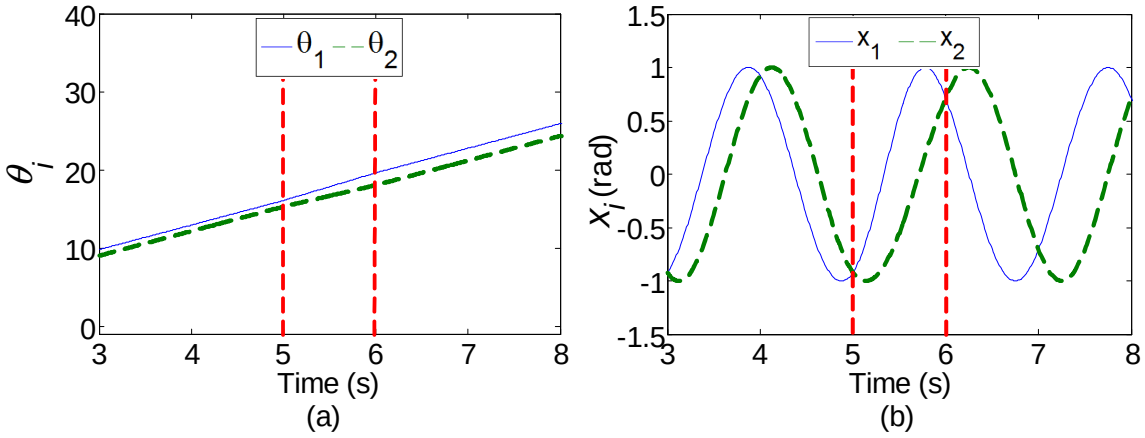


Figure 3.7. Behavior of CPGs when ϕ is changed from $\pi/4$ to $\pi/2$: (a) θ_i with respect to time (b) output x_i.

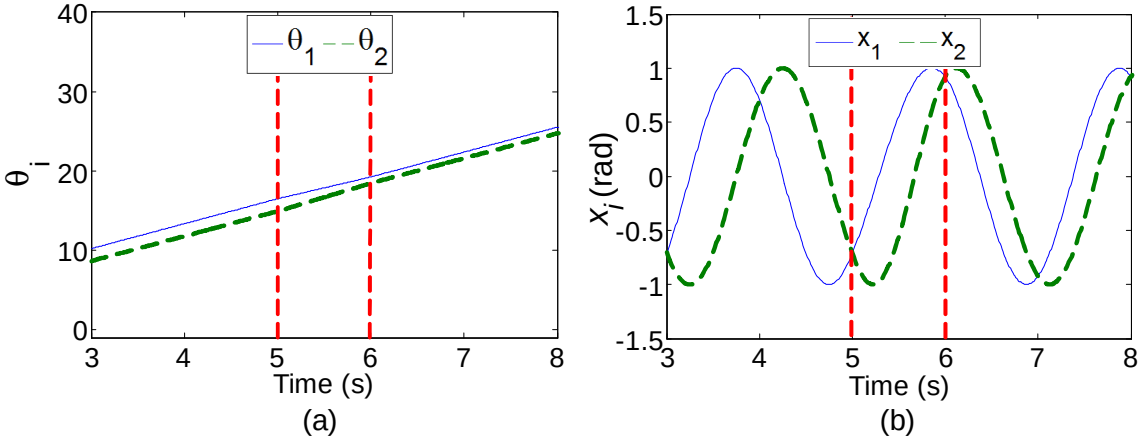


Figure 3.8. Behavior of CPGs when ϕ is changed from $\pi/2$ to $\pi/4$: (a) θ_i with respect to time (b) output x_i.

two transition points: 1) the starting transition point, and 2) the end of the transition point. The general calculation is given as follows:

$$|Slope_{before} - Slope_{after}| \leq 0.01 \tag{3.19}$$

The criterion of "0.01" is randomly chose based on the time step use in the simulation, where the error is small enough to be ignored.

Based on the calculation at the two transition points, it is found that the difference in slope before and after the phase transition at the two transition points is less than 0.01, which met the set criterion. Also, it is known that the parametric continuity of order n implies geometric continuity of order n, and the C_n continuity has already been proved in the previous subsection. Hence, it is clarified that the proposed method for smooth phase transition achieves the desired geometric continuity, G_2.

3.3 Simulation Results

For further verification, we show the simulation results of the proposed mathematical model. In the simulation, each joint is driven by an oscillator. The CPG outputs are used as angle inputs to the joints. The angle signal of the robot's joint, *joint_angle[i]* is calculated by

$$joint\_angle[i] = \beta x_i \tag{3.20}$$

where β is a gain from the control signal to the joint angle; and x_i is the output from i-th oscillator.

3.3.1 Movement of the Snake-like Robot

Fig. 3.9 shows the locomotion of simulated snake-like robot without applying the activation function.

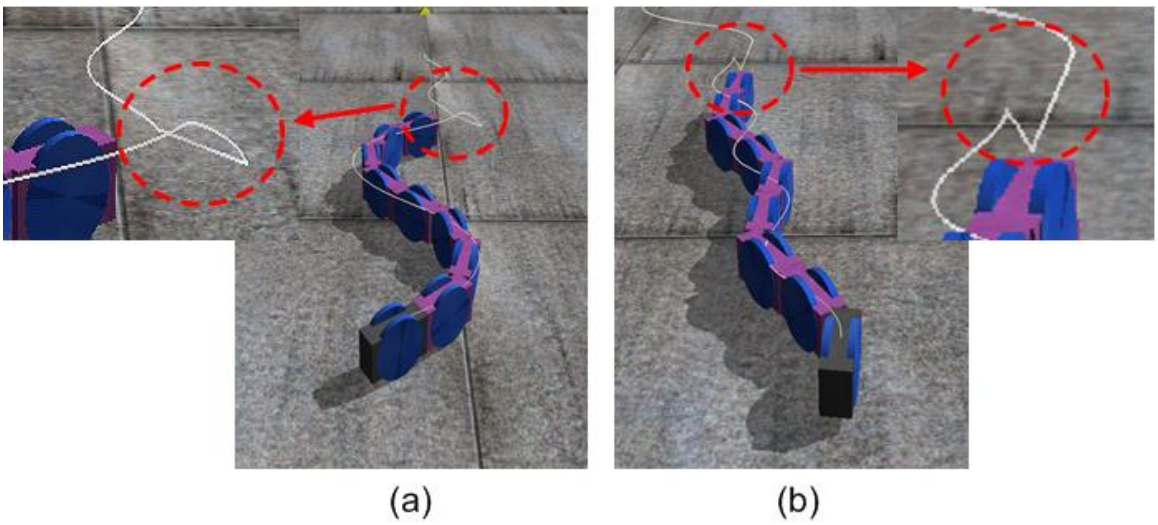


Figure 3.9. Locomotion of simulated snake-like robot with direct phase transition: (a) ϕ from $\pi/2$ to $\pi/4$, (b) ϕ from $\pi/4$ to $\pi/2$.

The trace line in Fig. 3.9 shows the trajectory of the COM of the head of the snake robot. The trace line in the round-dash line in Fig. 3.9 (a) and Fig. 3.4 (b) shows clearly the sharp edge during the phase transition. The robot's direction is uncertain due to the discontinuity in the forward movement of the robot, which vigorously confirms the drawback of the direct phase transition.

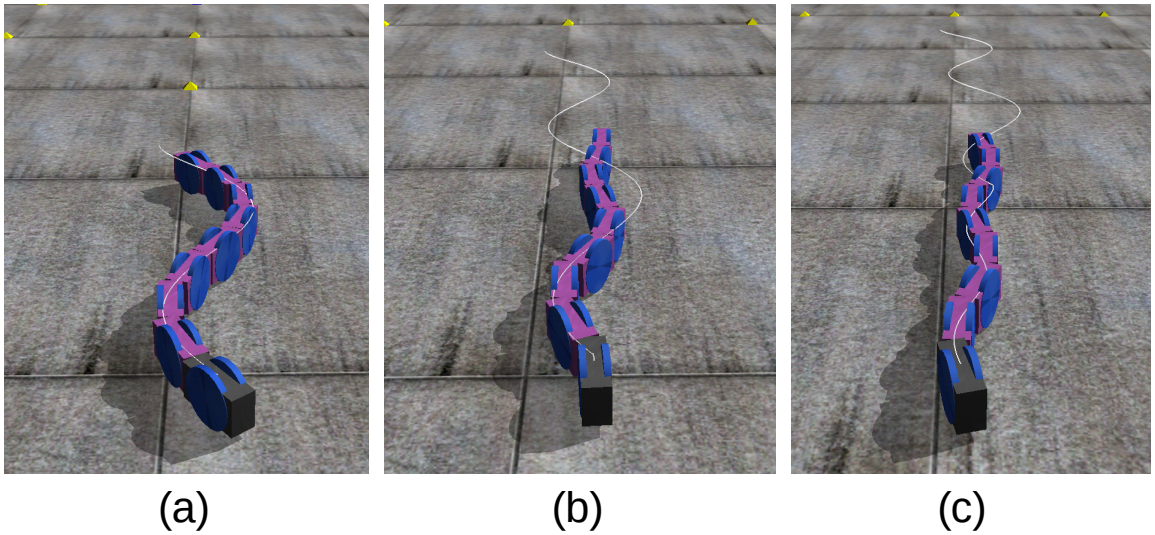


Figure 3.10. Simulation result of snake-like robot locomotion using activation function: (a) $\phi = \pi/4$ (b) transition of ϕ from $\pi/4$ to $\pi/2$ (c) $\phi = \pi/2$.

In Fig. 3.10, the body shape of the snake-like robot changes smoothly during the tran-

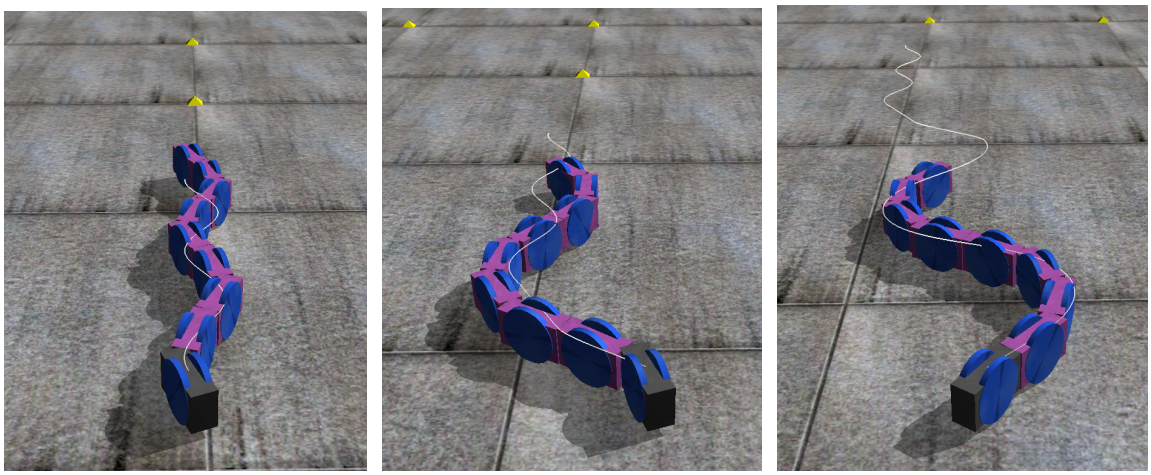


Figure 3.11. Simulation result of snake-like robot locomotion using activation function: (a) $\phi = \pi/2$ (b) transition of ϕ from $\pi/2$ to $\pi/4$ (c) $\phi = \pi/4$.

sition of ϕ from $\pi/4$ to $\pi/2$ by incorporating the activation function. In Fig. 3.11, we invert the value ϕ from $\pi/2$ to $\pi/4$. For easy view, in Fig. 3.12, we show the trajectory of the head of the simulated snake-like robot during transition of ϕ from $\pi/2$ to $\pi/4$.

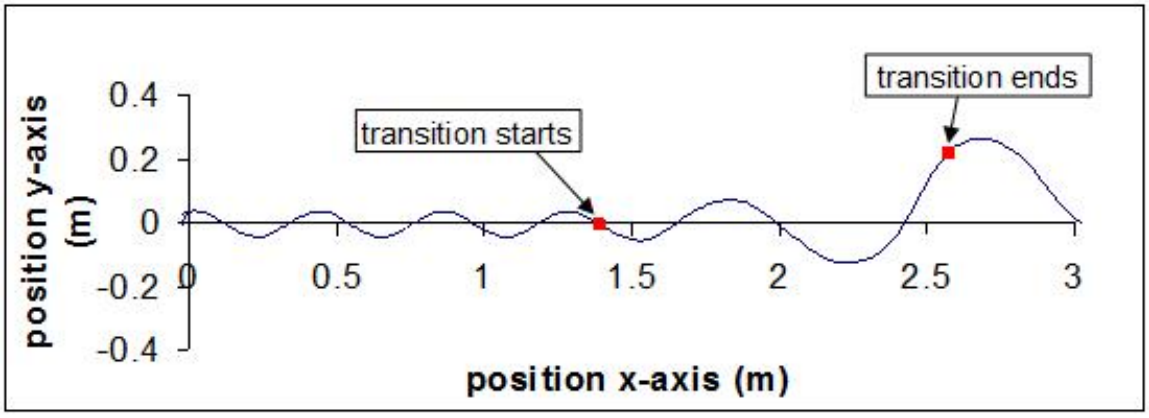


Figure 3.12. Simulation result of trajectory of snake-like robot locomotion: transition of ϕ from $\pi/2$ to $\pi/4$.

Based on the simulation results, we conclude that, it is essential to incorporate the activation function for phase transition to produce smooth locomotion of the snake-like robot.

3.3.2 Torque Analysis

The smoothness transition of the CPG outputs also affects the smoothness of the joint torque during the phase transition. The abrupt change of the phase will cause an abrupt change of the joint torque. We can get the information of the joint torque by applying the joint torque feedback function called *"dJointSetFeedback()"* and *"dJointGetFeedback()"* provided by the ODE. In Fig. 3.13, we present torque analysis for the first joint of the snake-like robot. Due to the symmetric gaits, it is expected that the torque results are almost the same for other joints as shown in Fig. 3.18 and Fig. 3.19.

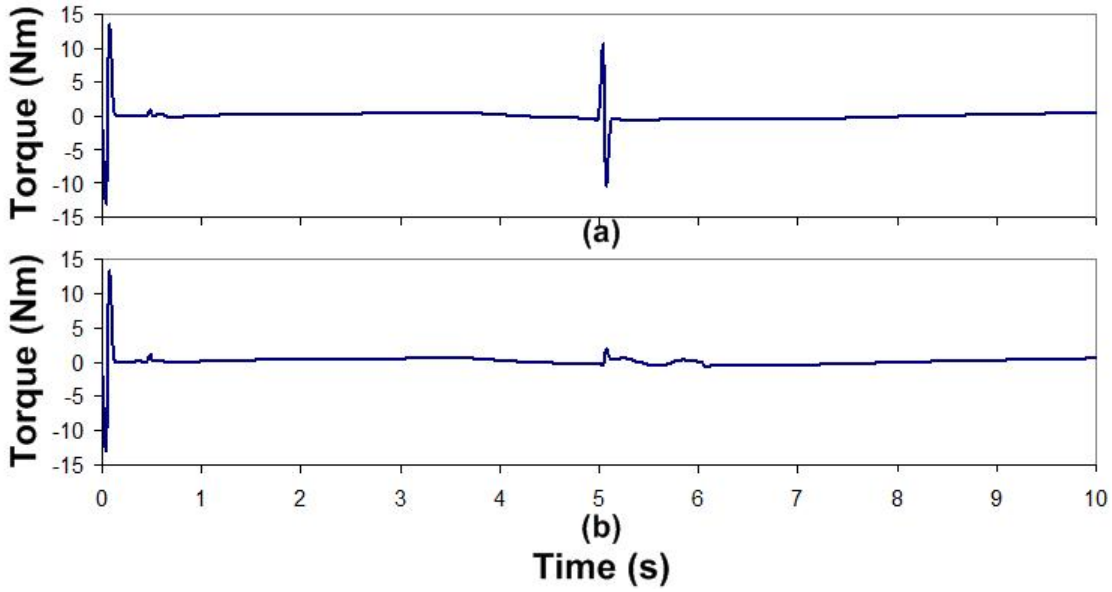


Figure 3.13. Torque of the first joint for transition of ϕ from $\pi/4$ to $\pi/2$: (a) direct transition (b) with activation function.

The total square-sum of joint torques for $joint_i$ (i= 1,..,n) is illustrated in Fig. 3.14 and figure Fig. 3.15. Fig. 3.14 shows the total square-sum of joint torques when changing ϕ from $\pi/4$ to $\pi/2$ (i.e., changing $N = 1$ to $N = 2$) whereas Fig. 3.15 shows the total square-sum of joint torques when changing ϕ from $\pi/2$ to $\pi/4$ (i.e., changing $N = 2$ to $N = 1$).

From our analysis, the torque will be affected by the transition of phase due to the stability of the CPG outputs before reaching the phase locking state. When the CPG outputs are perturbed, it needs time to converge to its steady state. In Fig. 3.13, at the

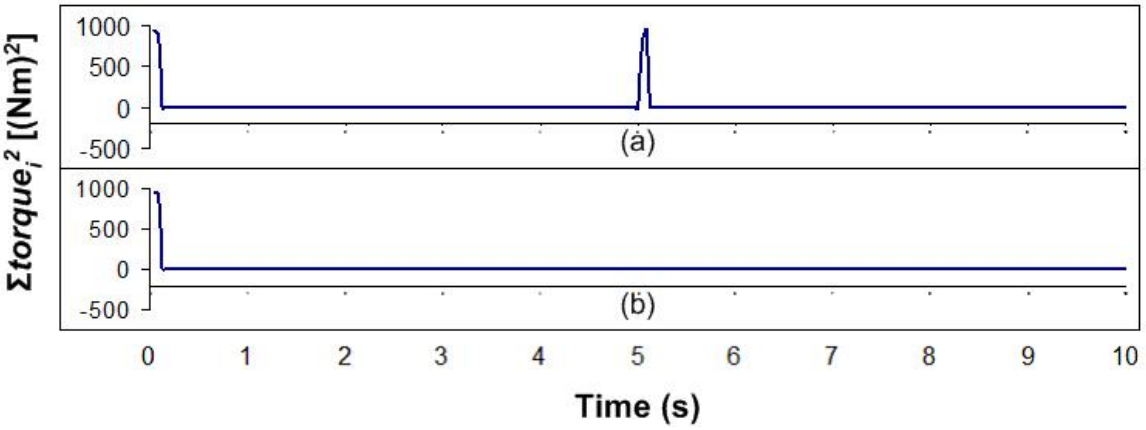


Figure 3.14. Total square-sum of joint torques for $joint_i$ ($i = 1,..,n$) when changing ϕ from $\pi/4$ to $\pi/2$: (a) direct transition (b) with activation function.

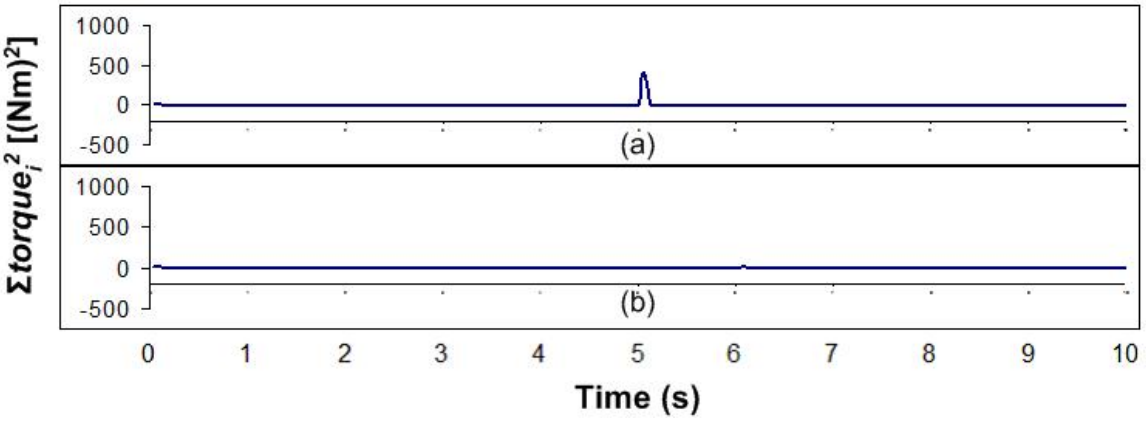


Figure 3.15. Total square-sum of joint torques for $joint_i$ ($i = 1,..,n$) when changing ϕ from $\pi/2$ to $\pi/4$: (a) direct transition (b) with activation function.

initial stage (time $< 0.2s$), there is a peak in the torque profile. This is because the CPG has yet reached its steady state (i.e., the CPG has yet converged to its limit cycle). During the transition of the phase (between time $5s$ to $6s$, Fig. 3.13 (b)), there is a small fluctuation due to the perturbation (change in the value of the CPG parameter). The small fluctuation is due to the linear parameter change of the ϕ with time (because the linear bipolar activation function is incorporated). Conversely, in Fig. 3.13 (a), the sudden change of the ϕ results a high peak of torque during the transition.

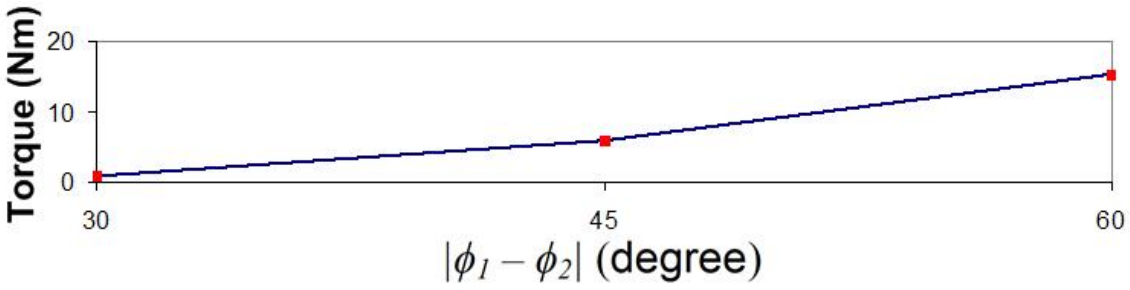


Figure 3.16. Maximum torque generated on the first joint for $|\phi_1$ - $\phi_2|$ without activation function (direct phase transition).

In Fig. 3.16, we show the maximum torque generated by the first joint for different $|\phi_1$ - $\phi_2|$ with direct phase transition, which results in nonlinear profile. As the difference between ϕ_1 and ϕ_2 gets larger, it produces higher peak of the torque. The reason behind this is due to the sudden and large change from the initial position to its new position. The larger the change of the slope of θ_i after the phase transition, the larger the peak of the torque produces during the phase transition.

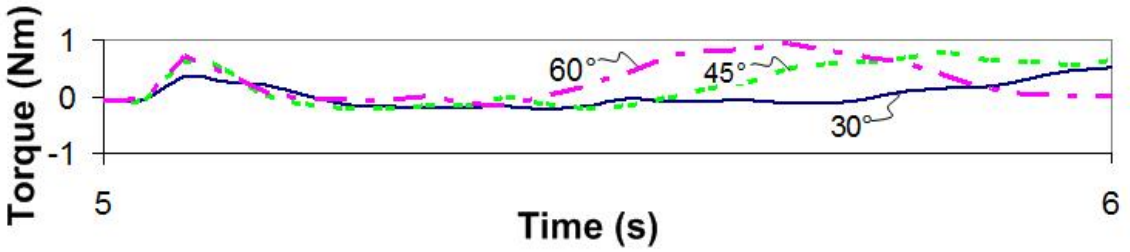


Figure 3.17. Behavior of torque of the first joint during phase transition for $|\phi_1$ - $\phi_2|$ with activation function, the transition time is between 5s to 6s.

On the contrary, by incorporating the activation function, the change of the torque is

nearly constant between 1 Nm to -1 Nm for various value of $|\phi_1 - \phi_2|$ (please refer Fig. 3.17). This is due to the linear change of the ϕ which directly affects the output of the CPG. By comparing the torque results between the direct phase transition and the activation function approach, we prove that the activation function approach gives nearly a constant torque during the phase transition.

Following Fig. 3.18 and Fig. 3.19 are the results of the middle joint torques. For simplicity, we only show the torque results for the third, fourth and fifth joints of the snake-like robot locomotion. The results of all of the joint torques are approximately the same and overlap with each other. Fig. 3.18 show the torque results when number of *S*-shape is changed from $N = 1$ to $N = 2$ (i.e., ϕ is changed from $\pi/4$ to $\pi/2$) and Fig. 3.19 shows the vice versa.

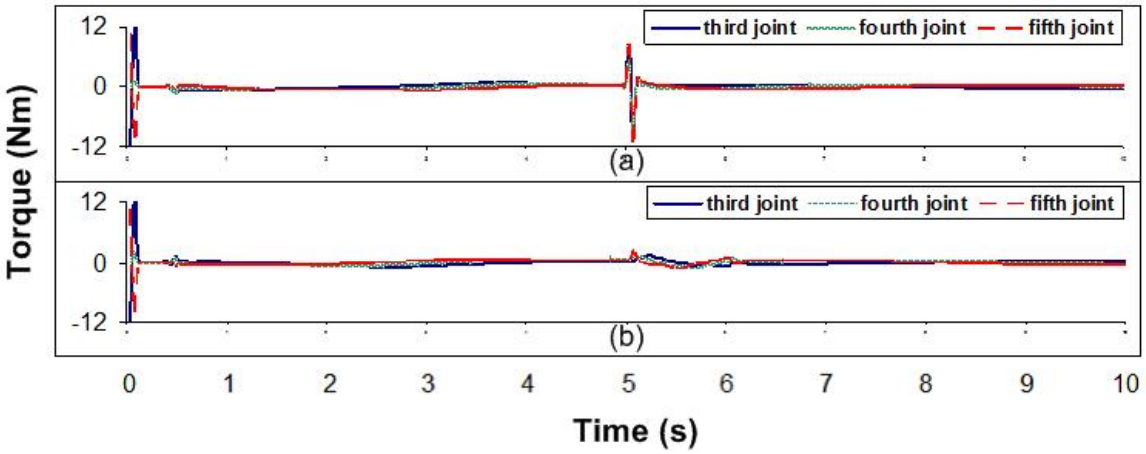


Figure 3.18. Torque of the middle joints when changing ϕ from $\pi/4$ to $\pi/2$: (a) direct transition (b) with activation function.

3.3.3 Future Application

Various potential and useful applications of the proposed body shape control can be used for snake-like robot locomotion. This section presents the preliminary works for the future applications. Fig. 3.20 shows an example of obstacle avoidance using our proposed control method.

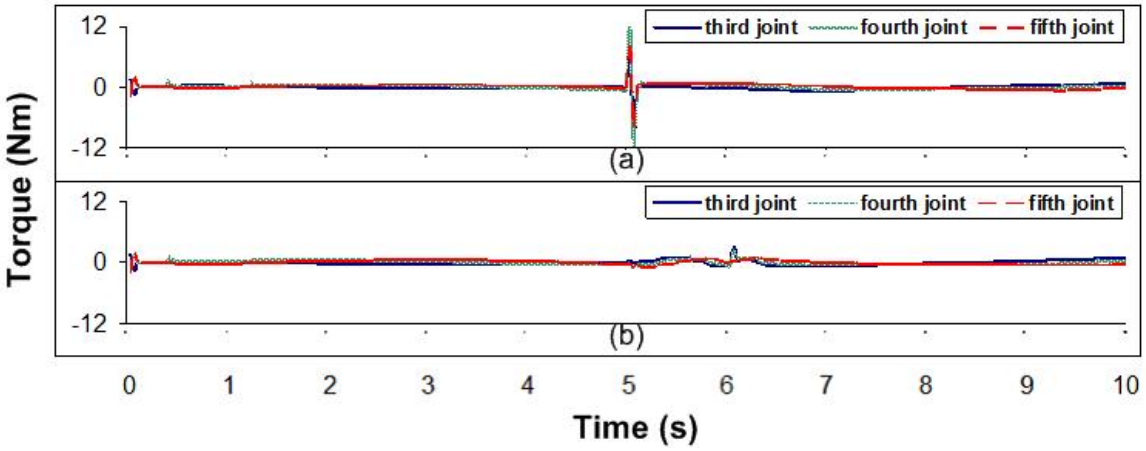


Figure 3.19. Torque of the middle joints when changing ϕ from $\pi/2$ to $\pi/4$: (a) direct transition (b) with activation function.

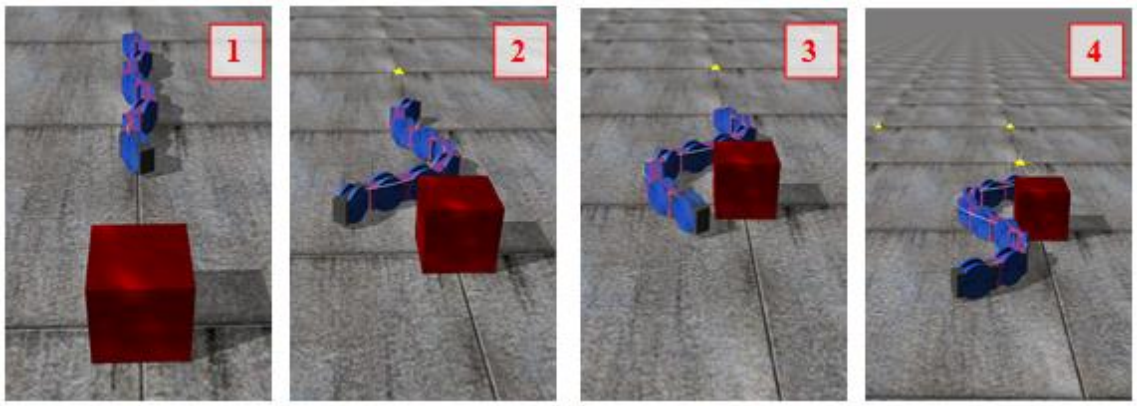


Figure 3.20. Potential application using our proposed body shape control: obstacle avoidance.

3.3.4 Control System Design

We can apply the control of the body shape transition into a snake-like robot to move into a different space width or to avoid an obstacle, as shown in Section 3.3.3. For instance, IR sensors can be installed at the head of the snake-like robot, or at the side of its body depending on the application. The trigger of the phase transition is t_1 with appropriate N_2.

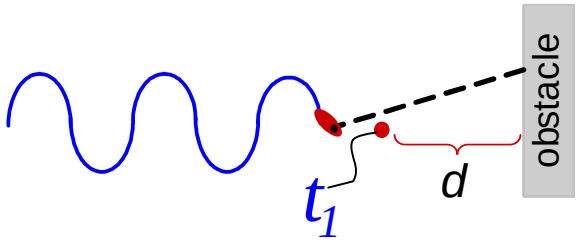


Figure 3.21. Schematic of sensor-based obstacle avoidance.

To set the parameter of (3.3), $(t_2 - t_1)$ is set to $1s$ (this value is selected based on in depth analysis), where the value is the minimum value to obtain a smooth phase transition without any jerky movement of the snake-like robot. In Fig. 3.21, d is the distance from the obstacle to the starting of the phase transition, t_1. The distance should be in appropriate length to ensure the snake-like robot is totally changed to its final S-shape, N_2. In the simulation environment (Fig. 3.20), the parameters t_1 and N_2 are calculated directly from the obstacle and the head of the snake-like robot. Fig. 3.22 shows the flow chart of the CPG-controlled process.

3.4 Discussion

Based on the simulation results, we conclude that by incorporating the activation function, smooth phase transition can be produced compared to direct transition at anytime instantly. Using the activation function, we can set any value of ϕ that corresponds to the desired number of S-shape. The advantage of this method is we can control the body shape of the snake-like robot at anytime by triggering t_1. With this advantage, we can plan whether to change the body shape before or during entering a different space width. This

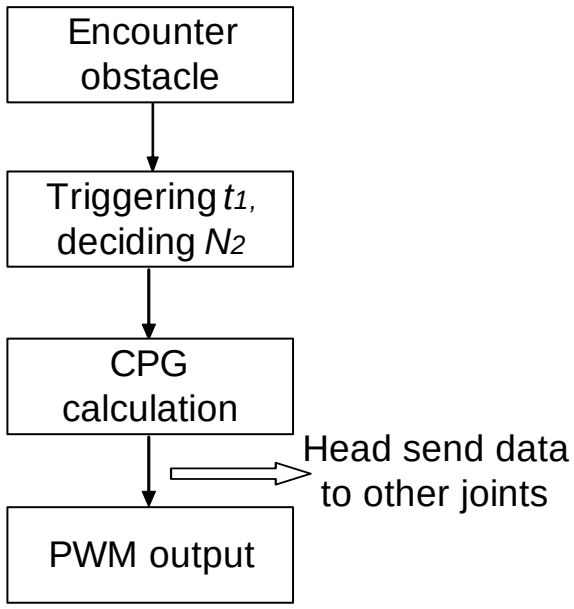


Figure 3.22. Flow chart of the CPG-controlled process.

is the superiority over [72], where the bias can only be added when the signals is at zero point to avoid sudden change or discontinuity of the output signal.

Furthermore, the activation function is a simple linear piece-wise function. The derivation is done to achieve the main goal, i.e., to control one of the CPG parameter, ϕ in relation with the number of S-shape of the snake-like robot. The simplicity in controlling the smooth body shape transition promises a light computational cost, easy implementation, simple control and understandable by readers who may not familiar with CPG-based control. This idea is a preliminary work for potential applications in wave-based locomotion systems.

As shown in Fig. 3.3, the behavior at the transition point of each CPG output is different, depending on the transition point during the start of the phase transition. The CPG outputs produce an apparent kink as the transition point is away from the peak signal. Thus, the activation function is formulated to tackle the worst case of the discontinuous change in the tangent direction at the kink, which is at the middle or between the zero point to the peak point. Based on our analysis, parametric continuity C_2 and geometric continuity, G_2 are sufficient to obtain smoothness of robot's locomotion during the phase transition.

In addition, the fluctuation of the joint torque is in a small range and nearly constant for every $|\phi_1$ - $\phi_2|$ by implementing the activation function during the phase transition. Compared with the direct phase transition, our proposed method produces good results of locomotion as well as the joint torque. Moreover, we found that the body shape curvature does not affect the torque except during the phase transition. With this, we know that the joint torque of the snake-like robot is not affected by the locomotion curvature. Therefore, the forward speed of the snake-like robot can be maintained during the locomotion.

However, right after the phase transition, we found that the direction of the snake-like robot deviates from its initial position (position before the phase transition starts). Based on initial analysis, the causes of the problems are due to: 1) slippery between the wheel and ground, 2) forward speed, 3) friction coefficient (surface parameter), 4) hardness of the ground, and 5) slip coefficients in tangential and normal direction. This is unacceptable for our snake-like robot locomotion during entering a narrow space because it may hit the wall or deviate from our desired path, but it is useful for obstacle avoidance.

3.5 Summary

In this chapter, we have proposed an approach to control the body shape of a snake-like robot by manipulating a single parameter, ϕ. By introducing an activation function during phase transition, we can get smooth movement of the snake-like robot. Simulation results confirmed the validity of our control method. There are two main conclusions: 1) the activation function can produce smooth phase transition, and 2) the body shape curvature does not effect the torque except during the phase transition.

Although we have successfully produced smooth change of the CPG outputs during phase transition, there is a problem that we encountered during our analysis. From our analysis by ODE simulations, we found that, after the phase transition, there is a small change of the direction of the snake-like robot. As our future work, we will investigate on how to encounter this problem so it would not effect our snake-like robot direction.

Chapter 4

Versatile Locomotion based on Body Shape Control for Obstacle Avoidance

Snakes are known to have the adaptability to transform its body shape depending on the environment they travel. This feature is useful to be applied into a snake-like robot to adapt to various barriers or avoiding obstacles, which is important for rescuing and searching tasks. However, for mimicking the fundamental locomotion of animals are difficult problems due to their complex body coordination and interactions with the environment. The efficiency of the animals moving adaptively to its surrounding is still beyond the reached of robots.

This chapter focuses on the motion control of a bio-inspired snake-like robot for avoiding an obstacle. Different types of bio-inspired mobile robots offer different ways of avoiding obstacles or barrier. For instance, biped hopping robot [73] can utilize its two legs to jump over a barrier. In [13] [74], the quadruped robot crosses an obstacle by lifting up their legs according to the obstacles height. While in [75], the amphibious robot uses its e-Paddle mechanism combining the wheel-legged gait to move in unstructured environment. All of these bio-inspired mobile robots provide high maneuverability for obstacle avoidance

but offer less locomotion stability than the snake-like robot. Furthermore, the ability of the snake-like robot to sneak in narrow space environments surpasses the mobility of the conventional wheeled, tracked, and legged robots.

We will mainly discuss the body shape control of a snake-like robot to move into variable width of path for two different purposes: 1) turning motion, and 2) straight-path motion. Based on the proposed model of the body shape transition, turning motion and straight-path motion can be controlled using one control parameter of the CPG.

Thus, this chapter is divided into two major parts: the first part is discussing the turning motion, and the second part is focusing on the straight-path motion of a snake-like robot.

4.1 CPG-based Turning Motion Control of a Snake-like Robot

This section presents a biomimetic approach based on central pattern generator (CPG), to control turning motion of a snake-like robot. One of the interesting features of a biological snake is its ability to avoid obstacles or a barrier by turning its whole body from its trajectory. This special obstacle avoidance motion is different from other types of animal, and thus, it is worth to be analyzed and realized into a snake-like robot.

In general, there are two ways for a snake to avoid obstacle; 1) by crawling over the obstacle, and 2) by turning its direction. It depends on the size and shape of the obstacle. For mimicking either methods, it is limited on the design of the snake-like robot. In this study, the focus will be on method 2 which is suitable for wheeled snake-like robot. Fig. 4.1 illustrates the turning motion of a snake-like robot when it encounters a barrier.

The main focuses of this section is to produce the turning motion of a snake-like robot based on our proposed CPGs network. The well-known control of turning motion for a snake-like robot is the amplitude modulation method (AMM) [10] [72]. A bias will be added to the output signal where an asymmetric swing of joint will lead to a change in the

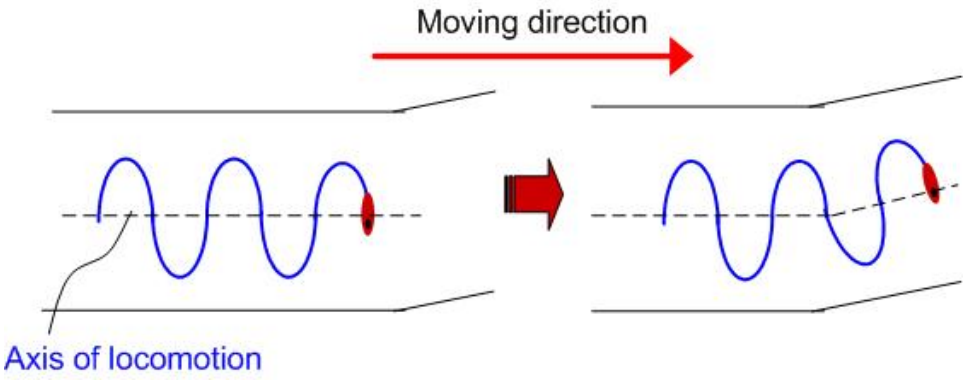


Figure 4.1. Turning motion of a snake-like robot when encounters a barrier.

motion direction. One of the major drawback of the AMM is that, the bias can only be added when the signals are at zero point to avoid sudden change or discontinuity of the output signal during the turning motion. Another turning method proposed by Crespi et al. [68], is by varying the offset and the amplitude bodys wave of a salamander-like robot. In [76], the authors exploit the obstacles to aid the locomotion of the snake-like robot, instead of avoiding it. This method shows a surplus of a snake-like robot locomotion over other types of mobile robot. However, it applies only for some limited situations such as scattered obstacles and round-shape obstacles.

All of the methods for turning motion mentioned previously utilize the amplitude as the control parameter. To the best of our knowledge, there are only few papers that utilize phase difference to control the turning motion: 1) for a fish-like robot [11] and 2) phase modulation method for a snake-like robot [77]. For bio-inspired robot which has independent left and right input control (different actuator) for example, flapping wings of a fish-like robot [11], the turning control is easy by just inputting different value of the control parameter for both sides. This is different for a snake-like robot, which utilize only one input signal for each joint and is dependent to its neighboring unit. In [77], phase modulation method is introduced to control turning motion of a snake-like robot, but results in discontinuity of the input angle. Thus, in this section, we introduce a simple way to control the turning motion of a snake-like robot which produces continuous input angle using the same control parameter that is, the phase difference.

4.1.1 Change in Locomotion's Direction

By using the phase transition to control the body shape of the snake-like robot, the direction of the snake-like robot deviates from its original position, that is, position before the phase transition starts. From our analysis by simulation, there is a small change in the direction of the snake-like robot after the phase transition and it is more apparent when changing ϕ from large to small i.e., decreasing the number of S-shape, N for example from 2 to 1. Fig. 3.10 and Fig. 3.11 show the simulation results by open dynamic engine (ODE).

As N is changed from bigger to smaller value (refer Fig. 3.11), the deviation from its original position becomes apparent. From our analysis, the parameter control of the deviation are: 1) control of frequency, v and 2) difference between value of ϕ_1 and ϕ_2. Frequency can be used for controlling locomotion speed of the snake-like robot [41], and thus contributes to the deviation of the trajectory.

Fig. 4.2 shows clearly the deviation of trajectory when we change ϕ from $\pi/2$ to $\pi/6$. The simulation result is the trajectory of the head of the simulated snake-like robot, and the body shape is modified from $N = 2$ to $N = 0.67$ at random time using the activation function in (3.3).

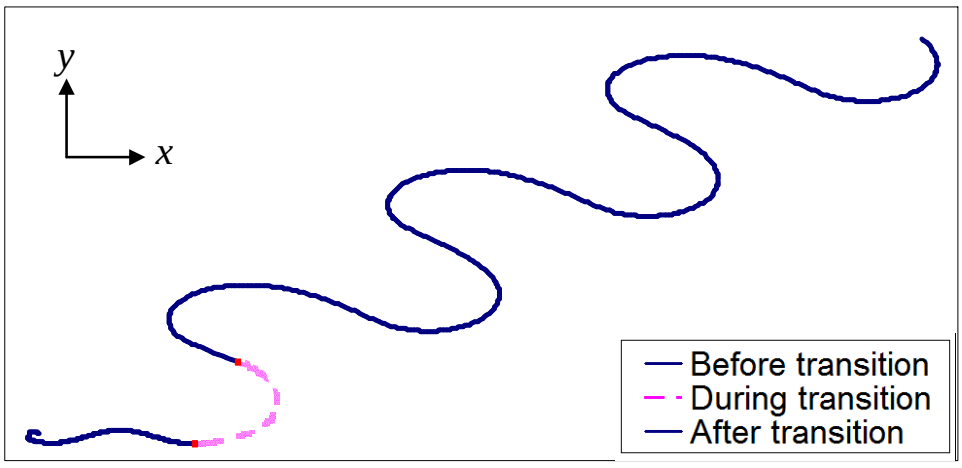


Figure 4.2. Simulation result of trajectory of a snake-like robot (position is in meter).

4.1.2 Turning Direction

The superiority of our proposed method for turning control over other methods (for example [72]) is the ability of the snake-like robot to maintain a smooth and continuous change of the joint angles at anytime during the locomotion. The turning motion can be implemented as at any desired time just by controlling the starting of the phase transition, t_1. The control of the direction of the turning can be easily controlled by the positive or negative value of the CPG output, x_i during the start of the phase transition. Thus, in cases where the direction of turning is important, the sign of x_i at t_1 needs to be identified. Fig. 4.3 shows an example of right and left turning motion with different starting point of phase transition. The small boxes in Fig. 4.3 are the starting point of the phase transition with the same value of t_1 for two trajectories. Different speed was used by controlling v to get different sign of the CPG output (positive and negative) at the same t_1.

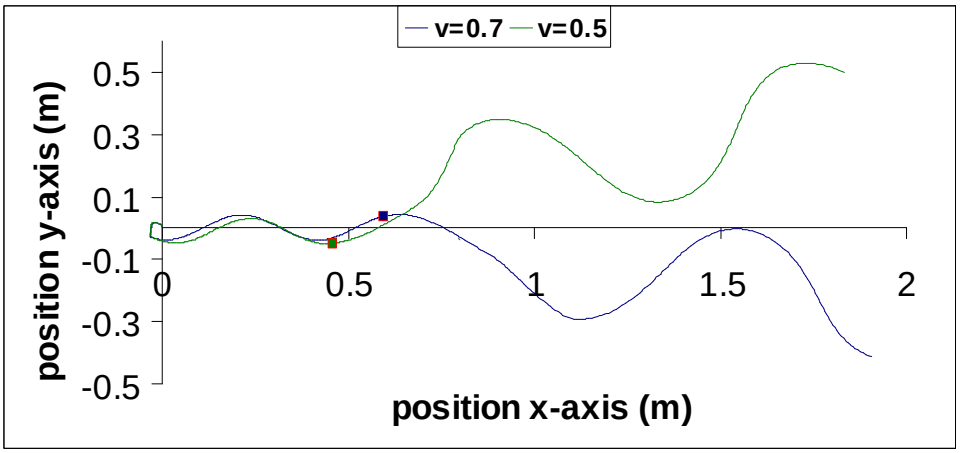


Figure 4.3. Direction of turning for left and right.

If the output signal is positive, the turning direction will be to the negative of the y-axis, and vice versa. If the right turning is defined at the negative signal of the y-axis, and the left turning is at the positive signal of the y-axis, therefore, it can be concluded that the turning direction of the snake-like robot can be controlled by the positive or negative value of the CPG output, x_i during the start of the phase transition, as follows:

$$\text{Turning direction} = \begin{cases} \text{Right, if } x_i \text{ is positive} \\ \text{Left, if } x_i \text{ is negative} \end{cases} \tag{4.1}$$

4.1.3 Control of the Deviation Angle

The following Fig. 4.4 and Fig. 4.5 show how the deviation are affected by the two control parameters i.e., frequency and difference between value of ϕ_1 and ϕ_2. In these two simulations, the results of the locomotion trajectory are shown by changing value of ϕ from large to small. In Fig. 5, the value of ϕ_1 is $\pi/2$ and the value of ϕ_2 is $\pi/4$. It can be found that different value of frequency, v gives little effect to the deviation angles. The deviation angles are almost the same, which is not suitable to control the turning motion of the snake-like robot.

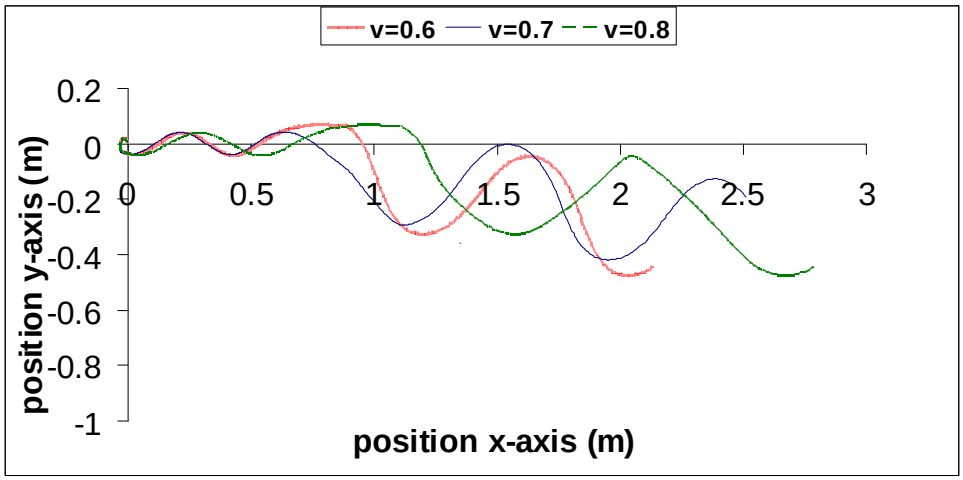


Figure 4.4. Analysis of trajectory's deviation with different value of v(rad/s).

Fig. 4.5, shows the trajectories of the deviation angle while keeping the value of ϕ_1 constant i.e., $\pi/2$ and varying the value of ϕ_2 with $v = 0.5$, and $A = 1.0$. The small box in Fig. 4.5 marks the start of the transition from ϕ_1 to ϕ_2. It shows that the deviation angle becomes more apparent as the difference between (ϕ_1 - ϕ_2) becomes larger. Fig. 4.6 summarizes the relationship between deviation angle with respect to (ϕ_1 - ϕ_2) where the

deviation angle is increasing nonlinearly with the increase of (ϕ_1 - ϕ_2). Thus, the value of (ϕ_1 - ϕ_2) can be increased for producing larger turning angle.

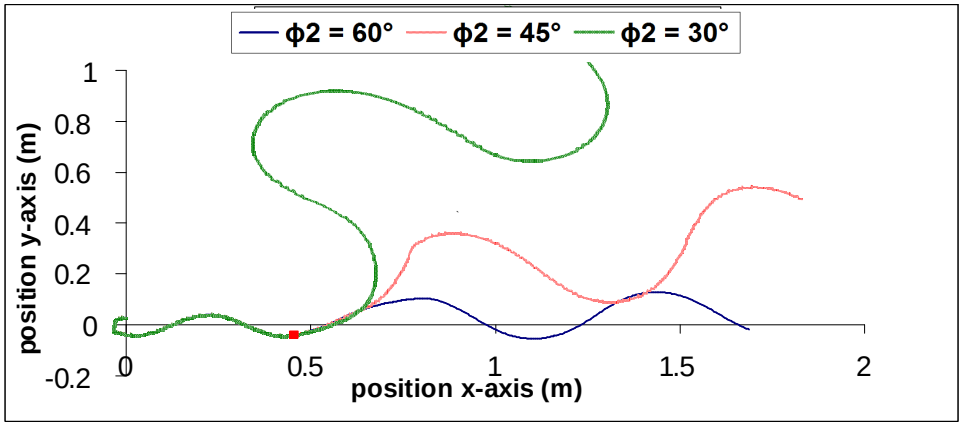


Figure 4.5. Analysis of trajectory's deviation with different value of ϕ.

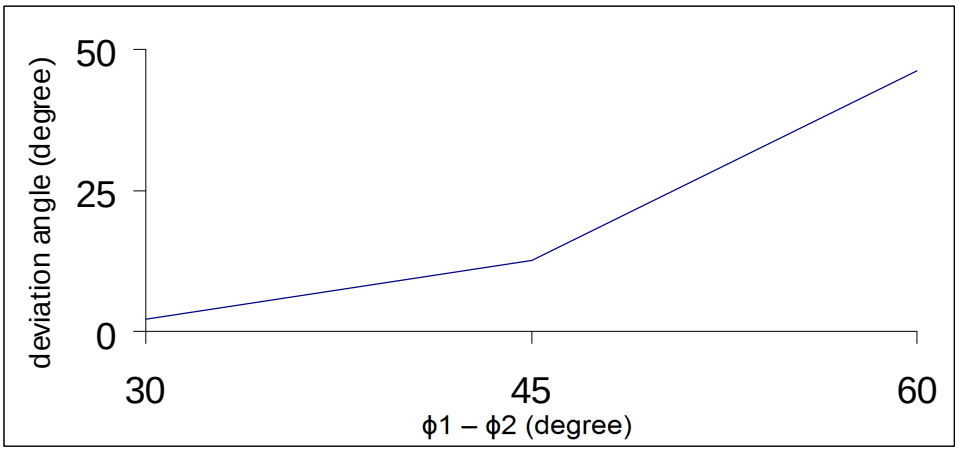


Figure 4.6. Deviation angle with respect to (ϕ_1 - ϕ_2).

4.1.4 Motion Optimization

Another advantage of our proposed phase transition method is the motion optimization. The propellant force of a snake-like robot can be optimized by controlling the locomotion curvature. Detailed analysis about the interaction between a snake-like robot and ground can be found in [78]. As shown in Fig. 4.7, increasing the rotation angle, ψ_p will increase

the forward force. Adopting Coulomb friction model, the relationship between the forward force and ψ_p are described as follows:

$$\begin{aligned} f_p^t &= -\mu_T m_p g \text{sign}(\delta^p r^t) \\ f_p^n &= -\mu_N m_p g \text{sign}(\delta^p r^n) \end{aligned} \tag{4.2}$$

where f_p^t and f_p^n are the components of friction in tangential and normal directions, respectively, exerted on the pth ($p = 1, 2,...$) module of a snake-like robot. μ_T and μ_N are the friction coefficients; m_p is the weight of the pth module; $\delta^p r^t$ and $\delta^p r^n$ are the tangential and normal displacements of the pth at the contacting point, respectively.

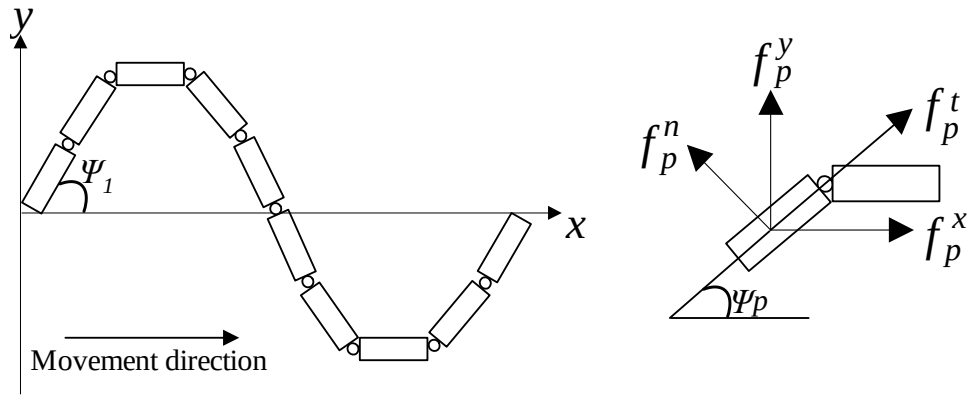


Figure 4.7. Relationship between forward force and locomotion curvature.

Based on the coordinate system shown in Fig. 4.7, where x-axis is set as the forward direction of the snake-like robot, the resultant friction force can be obtained as follows:

$$\begin{aligned} f_p^x &= f_p^t \cos\psi_p - f_p^n \sin\psi_p \\ f_p^y &= f_p^t \sin\psi_p + f_p^n \cos\psi_p \end{aligned} \tag{4.3}$$

where f_p^x and f_p^y are the resultant force in x and y direction, respectively. To avoid backward slippage, sufficient friction needs to be generated along the x direction. There are two ways in obtaining necessary resultant force in the forward direction: 1) anisotropic friction with μ_N larger than μ_T, and 2) with the increasing of ψ_p. The second option can be controlled by our proposed phase transition, where the ψ_p can be increased or decreased by changing the

locomotion curvature. Not all cases need to have a large locomotion curvature, for instance, climbing a slope [79], where the most efficient locomotion is when the number of S-shape, N is 2.

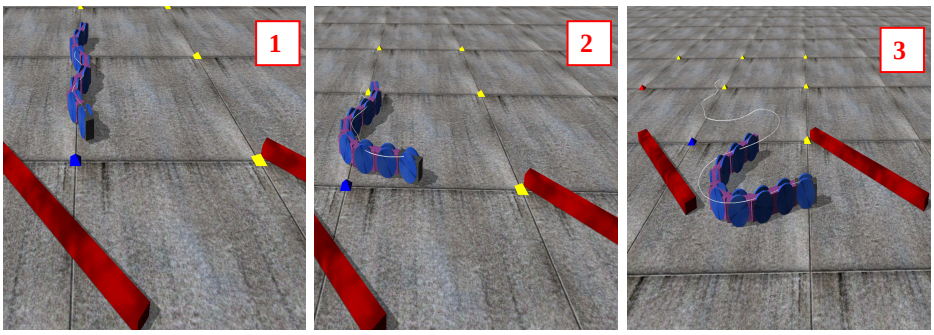


Figure 4.8. Locomotion optimization in large space.

Another vital point is the body shape adaptation to the surrounding space. This feature of body shape transition is important especially for searching and rescuing tasks. The spaces for instance, after a disaster will be differ with many obstacles surround, thus, an ability of turning motion with body shape transition of a snake-like robot is highly required.

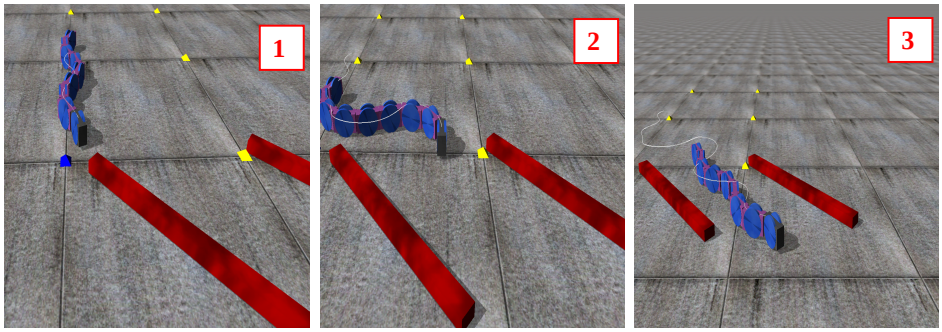


Figure 4.9. Body shape adaptivity with limited space.

There are two ways in controlling the amplitude of the curvature of a snake-like robot, as shown in Fig. 4.10: 1) amplitude-controlled, or 2) number of S-shape.

In Fig. 4.10 (a), we can easily control the amplitude using parameter, A. This work has been done in [10] [72] but using different mathematical model. The superiority of our proposed method of controlling the number of S-shape (Fig. 4.10 (b)) for adaptive

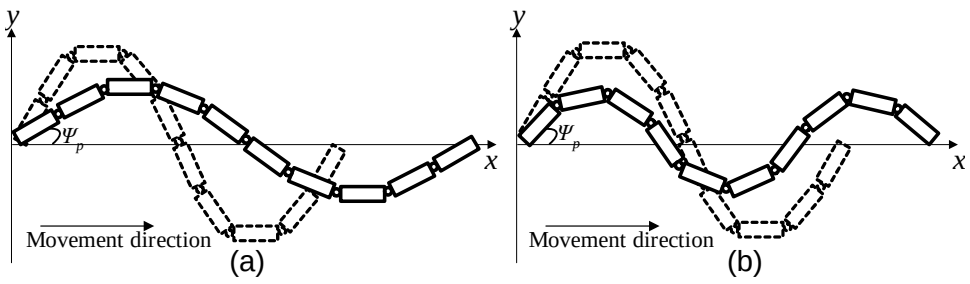


Figure 4.10. Control of locomotion curvature: (a) amplitude-controlled, (b) number of *S*-shape.

locomotion curvature as compared to the amplitude-controlled are: 1) the joint torques is lower and uniformed for all joints (refer Fig. 4.11), 2) the phase transition controlled can be modified online without needing to stop the robot, and 3) the number of S-shape transition resembles closely the biological snakes when adapting to different space width.

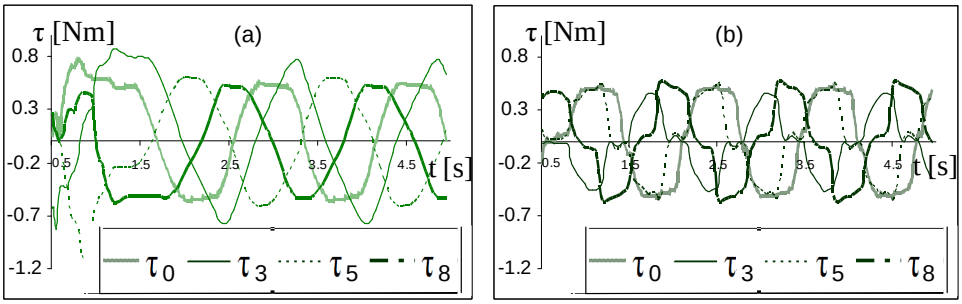


Figure 4.11. Joint torques for a snake-like robot: (a) for amplitude-controlled, (b) for number of *S*-shape.

The simulation results in Fig. 4.11 show examples of joint torques for joint 1, joint 3, joint 5 and joint 8. We can see that there is a fluctuation of the torque for the amplitude-controlled method. Inversely, for the number of S-shape method, the torque profile shows uniform range of torque between $\pm 0.6 Nm$.

Beside energy consumption, another important criteria needs to investigate is the efficiency of forward distance traveled. Based on our analysis through the simulation experiment, the amplitude-controlled method give an advantage of larger forward distance traveled as compared to the number of S-shape method, but the difference is not significant. The total forward distance traveled in one period for the amplitude controlled method

is approximately $1.101m$, while the traveling distance for the number of S-shape method in one period is approximately $1.052m$. In the simulation, the same setup is used for both methods.

4.2 CPG-based Straight-path Motion Control of a Snake-like Robot

Maintaining a straight-path for locomotion of a snake-like robot can be easily controlled using any control architecture such as sine-based, model-based or others. For locomotion efficiency, some control parameters need to be adjusted for the snake-like robot to adapt to the surrounding, for example, the speed, the body shape, the turning motion and others. However, a problem occurs when one or more of the control parameters are modified online, where the trajectory of the snake-like robot will deviate from its original path.

This section focuses on a straight-path control of a snake-like robot for body shape transition. Here, the definition of straight-path is maintaining the moving direction of the snake-like robot with respect to the axis of locomotion. Fig. 4.12 illustrates the body shape transition of a snake-like robot avoiding an obstacle and adapting to various space width. For rescuing and searching tasks, online modification of the control parameters is crucial for realizing the versatility of the robot's locomotion.

Most of the papers focusing on a snake-like robot locomotion did not consider or discuss in details the deviation of trajectory after modifying control parameters online. In [68], the authors successfully controlled the locomotion of a salamander-like robot including its speed, turning and and swimming. However, details of discussion regarding online parameter modification were not presented. Furthermore, in [11] [41] [68] [72], all of the authors present a control strategy for turning motion by modifying one of the control parameter. One can clearly observe that by adjusting any control parameter online, it will cause the robot to deviate from its initial trajectory, and thus, perform a turning motion by appropriate control. To the best of our knowledge, the issue of online modification while maintaining

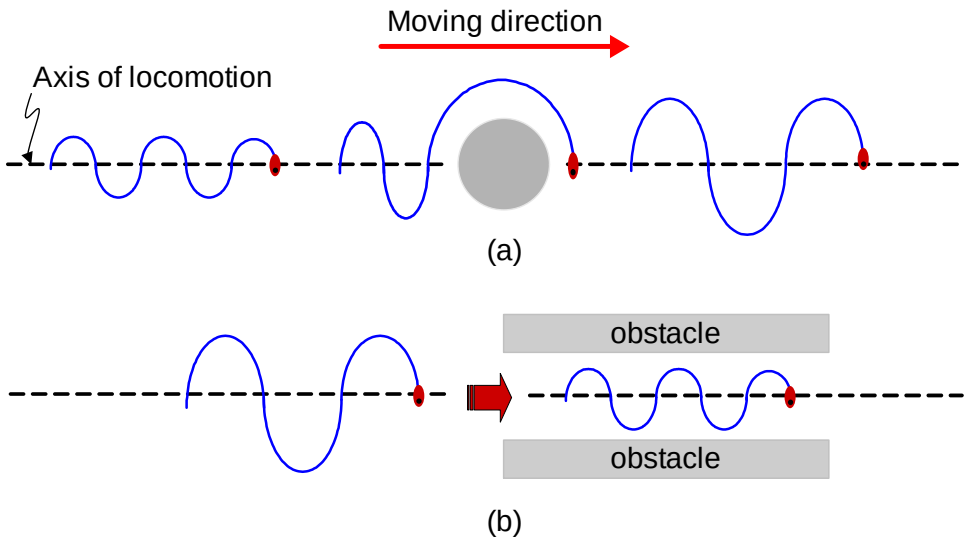


Figure 4.12. Various locomotion of a Snake-like robot: (a) avoiding obstacle, (b) moving in a different space width.

straight-path locomotion especially on a snake-like robot has not yet been discussed in detail by other researchers. In this paper, the main goal is to minimize the deviation of the trajectory after adjusting the control parameter online. It should be noted that, the deviation of trajectory is also due to several factors such as slippage between the wheel and the ground (friction), surface parameter and other factors that are beyond human control and beyond a non model-based control.

In [80], it is mentioned that the trajectory of the snake-like robot will deviate after the phase transition. Although the body shape control has successfully achieved without any unstable behavior of the snake-like robot, the deviation of the trajectory should also be considered. The deviation of trajectory is useful for turning motion [81], however some applications need a straight-path forward locomotion. For example, when a snake-like robot moving from a large space into a narrow tunnel as illustrated in Fig. 4.12 (b). If the straight-path locomotion cannot be controlled, the snake-like robot may hit the wall.

To overcome this problem, two control strategies for straight path locomotion of a snake-like robot will be proposed using the derived linear bipolar function ((3.3)). Two control parameters have been analyzed namely: 1) the transition time of the phase transition, $(t_2 - t_1)$ and 2) the trigger time, t_1.

4.2.1 Transition Time of the Phase Transition

In this section, two transition times of the phase transition, $(t_2 - t_1)$ are compared with different value of frequency, v. The reason of using different value of v is because v can be adjusted as a control parameter for robot's speed. Note that, speed of a wheeled mobile robot affects the slippage between the wheel and the ground. The higher the speed, the higher the slippage. In this paper, the main focus is on the parameters in (3.3), therefore, the effects of the CPG parameters in (2.19) will not be discussed in detail. Another vital point to mention is the deviation of trajectory cannot be fully eliminated due to slippage, and thus, the main goal is to minimize the deviation of the trajectory of the snake-like robot before and after body shape transition.

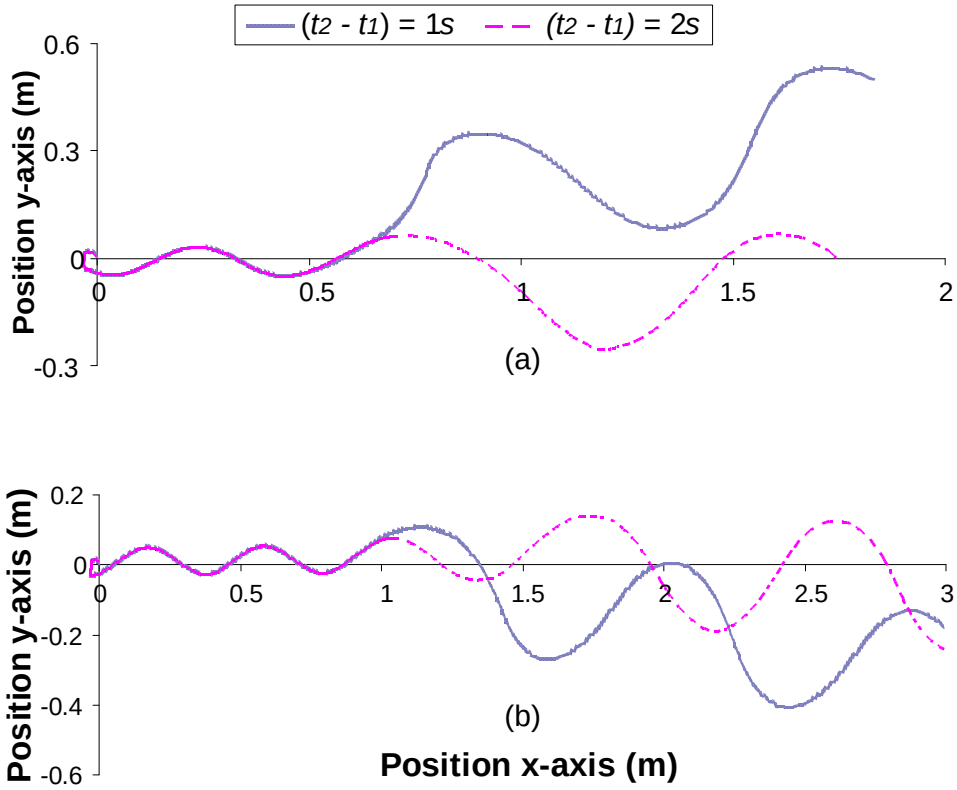


Figure 4.13. Simulation result of the trajectory of a simulated snake-like robot: transition of ϕ from $\pi/2$ to $\pi/4$ (a) $v = 0.5$ (rad/s) (b) $v = 1.0$ (rad/s).

Fig. 4.13 (a) and Fig. 4.13 (b) show the trajectory of the simulated snake-like robot when v is 0.5 and 1.0 respectively. Two different values of $(t_2 - t_1)$ are compared. Two conclusions

can be made from the results: 1) as $(t_2 - t_1)$ gets higher, the deviation of trajectory can be minimized to its original axis, and 2) as the speed increases, the longer time $(t_2 - t_1)$ is needed to minimize the deviation of trajectory. However, the snake-like robot needs longer time to change its body shape completely, which means a longer distance, d is needed to trigger the phase transition from the obstacle, as shown in Fig. 4.14.

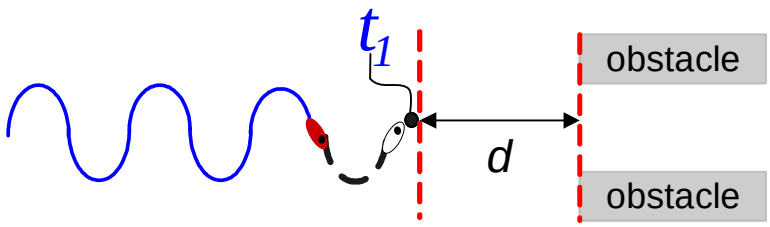


Figure 4.14. Schematic of a snake-like robot when encounters obstacles.

For further clarification, torque profiles of the first joint of the simulated snake-like robot are presented in Fig. 4.15. It shows that the torque increases linearly before and after the transition starts as $(t_2 - t_1)$ gets higher, whereas, with smaller $(t_2 - t_1)$, the torque profile shows a bent or crook before and after the transition (please focus on the box). Furthermore, the torque fluctuates more frequent during the phase transition for smaller $(t_2 - t_1)$. This situation contributes to the deviation of the snake-like robot's trajectory. Thus, with lower rate of change of ϕ, the fluctuation of torque can be minimized.

4.2.2 Trigger Time of the Phase Transition

Another control parameter that can be used to control the straight-path locomotion of a snake-like robot is the trigger time, t_1. In [72], the authors mentioned that the CPG output can only be adjusted when the signals is at zero point, to avoid any sudden or discontinuous change of joint angle and results in turning motion. It is common to add a bias for turning motion of a mobile robot so the balance is broken and the motion direction is changed [11] [41] [68] [72]. However, using the proposed linear bipolar activation (3.3), the straight path motion can be controlled by adding a bias when the signals is at zero point. This is the novelty of the approach which is different from other previous researches.

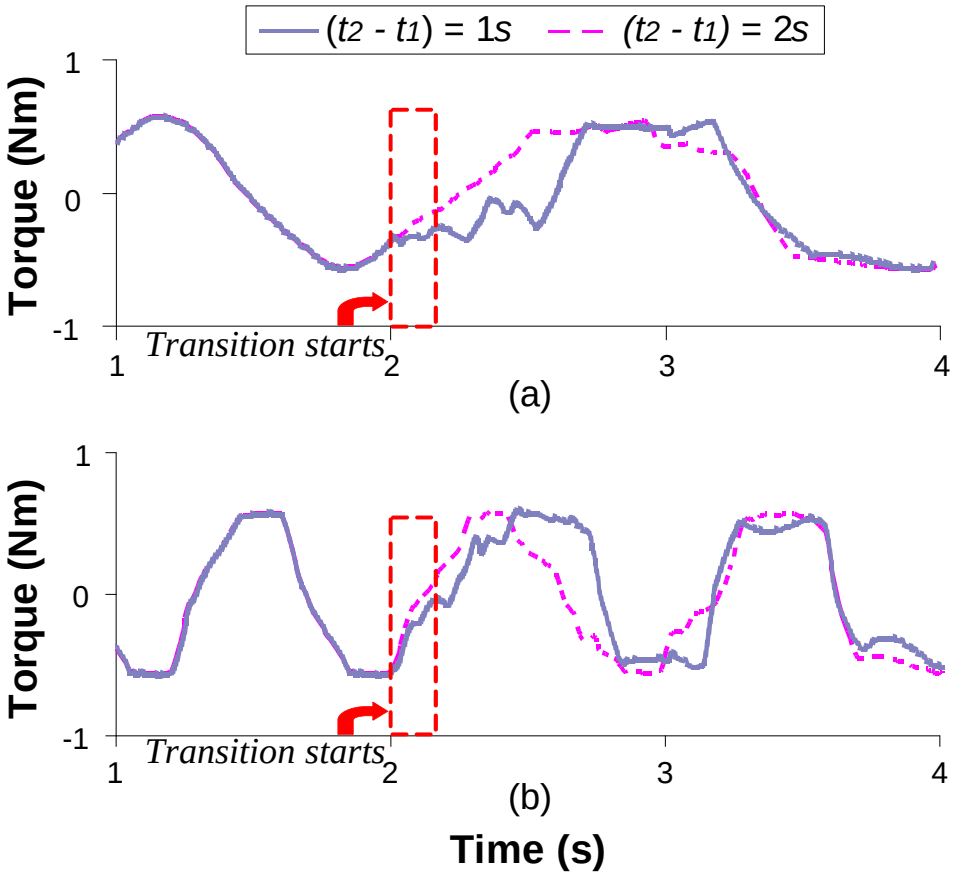


Figure 4.15. Torque profile of the first joint of a simulated snake-like robot: transition of ϕ from $\pi/2$ to $\pi/4$ (a) $v = 0.5$ (rad/s) (b) $v = 1.0$ (rad/s).

In (4.4), t_p is the consecutive time taken between each of the CPG output to be at zero point as shown in Fig. 4.16. ϕ_1 is the initial value inputs by the user.

$$t_p = \phi_1 / 2\pi \tag{4.4}$$

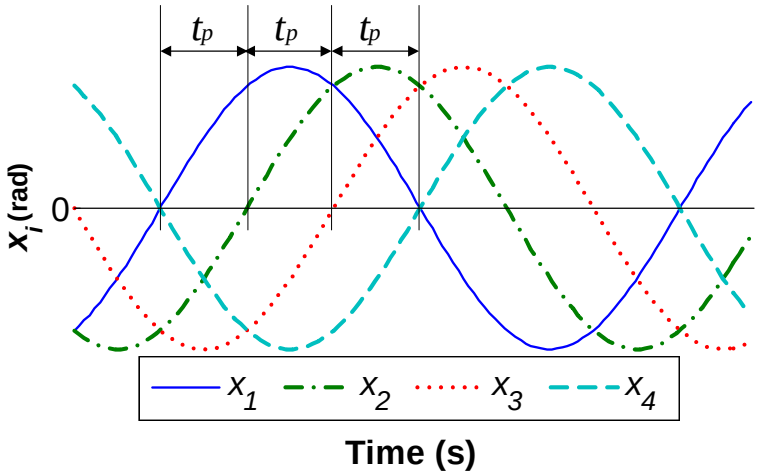


Figure 4.16. Illustration of t_p calculation for four CPG outputs.

Fig. 4.17 shows the trajectory of the head of the simulated snake-like robot with different t_1 and $(t_2 - t_1) = 1$s . The result shows the position of the snake-like robot in y-axis with respect to time because the goal is to maintain the direction of the snake-like robot with respect to the axis of locomotion (x-axis) regardless the forward path it travels. The value of t_1 is set randomly to show clearly the effect of the start point of the phase transition to the direction of the snake-like robot. As the phase transition starts, the point which is nearest to zero in the y-axis produces less deviation of the snake-like robot trajectory after the transition (see $t_1 = 3$s). Conversely, at $t_1 = 2$s, where the start point of the transition is at the peak point approximately, the trajectory results in large deviation after the phase transition.

One drawback of this method is the phase transition needs to start at zero point, which diminish the advantage of incorporating the linear bipolar activation function, as the goal is to start the phase transition at any time instantly.

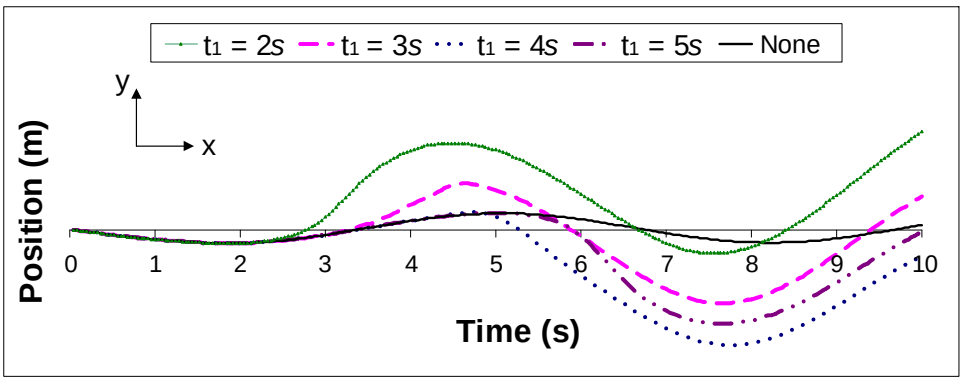


Figure 4.17. Trajectory of the simulated snake-like robot with different t_1.

In other words, the phase transition needs to be started at zero point to maintain the straight-path of the snake-like robot.

4.2.3 Control Strategy

Fig. 4.18 shows a control strategy for both methods and it has been verified through the ODE simulation platform.

Herein, method 1 and method 2 mean manipulating $(t_2 - t_1)$ and t_1 as control parameter, respectively. After selecting a motion and a method that is best suit to the situation, the appropriate value of the control parameters is decided. Then, the CPG outputs will be calculated and input to the PWM motor as joint angles. Finally, the first unit (head) of the snake-like robot will transmit the data to the other joints consecutively.

The proposed control strategy is realized into a simulated snake-like robot, which is shown in Fig. 4.19. The application for both methods is similar i.e., to maintain a straight-path when moving in a different space width by utilizing body shape transition, and thus, the simulation result is shown only for method 1. The parameters are set as follows: $(t_2 - t_1)$ = 2s, $n = 8$, and $v = 0.5$. ϕ_1 corresponds to $N_1 = 2$, and ϕ_2 corresponds to $N_2 = 1$. The value of ϕ is $\pi/2$ and $\pi/4$, depending on the situation of the snake-like robot as depicted in Fig. 4.19 (1).

Fig. 4.20 (a) and Fig. 4.10 (b) show the trajectory of the head (first unit) of the simulated

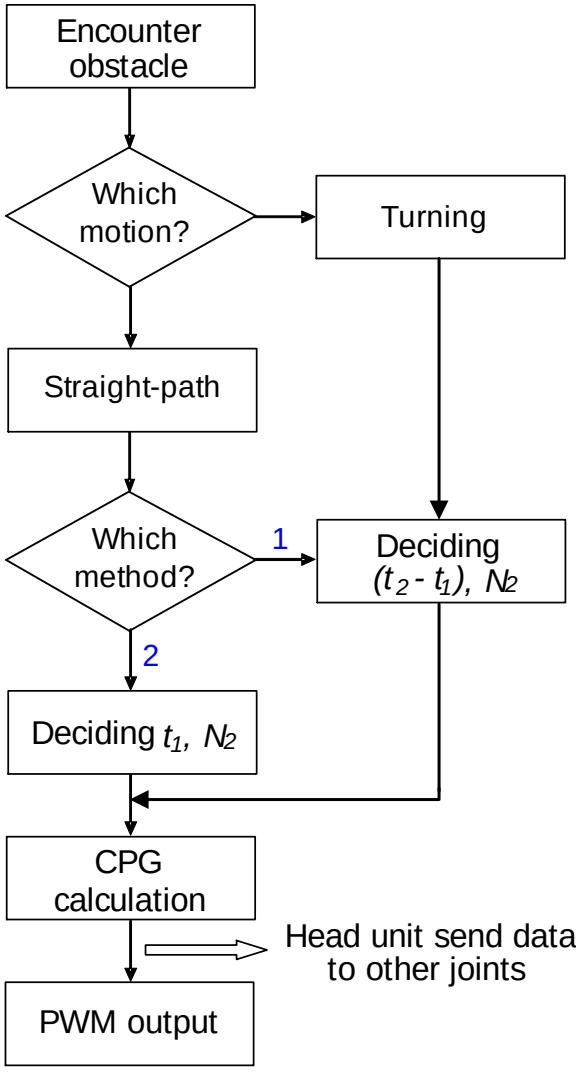


Figure 4.18. Flow chart of the control strategy.

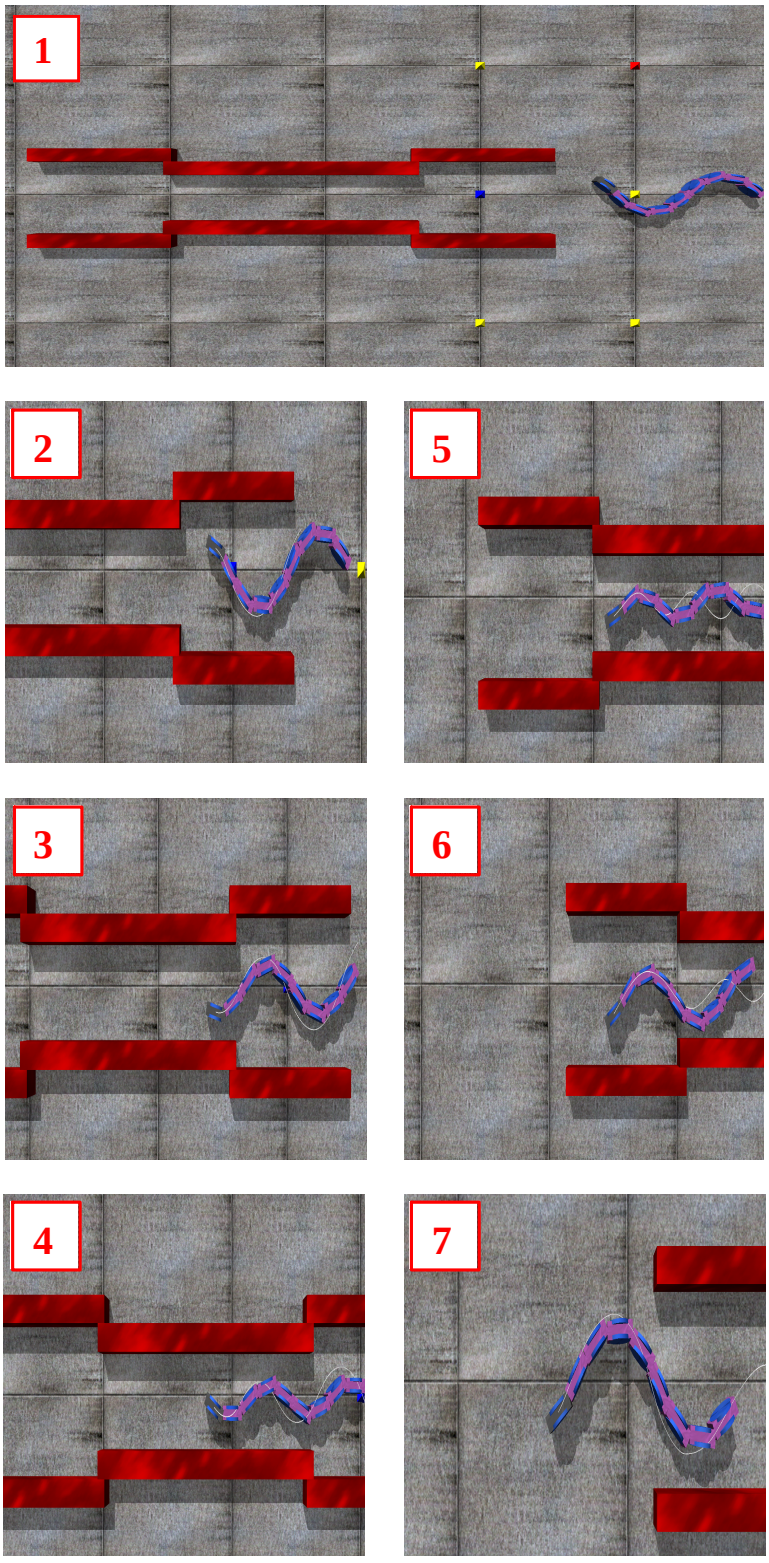


Figure 4.19. Locomotion of a snake-like robot when moving in different space width.

snake-like robot using method 1 and method 2 respectively. For further clarity, the body shape of the snake-like robot is changed twice, from large curvature to small curvature, and vice versa. No discontinuity or feasible cease occurs during the body shape transition as well as any obvious deviation of the trajectory before and after the transition.

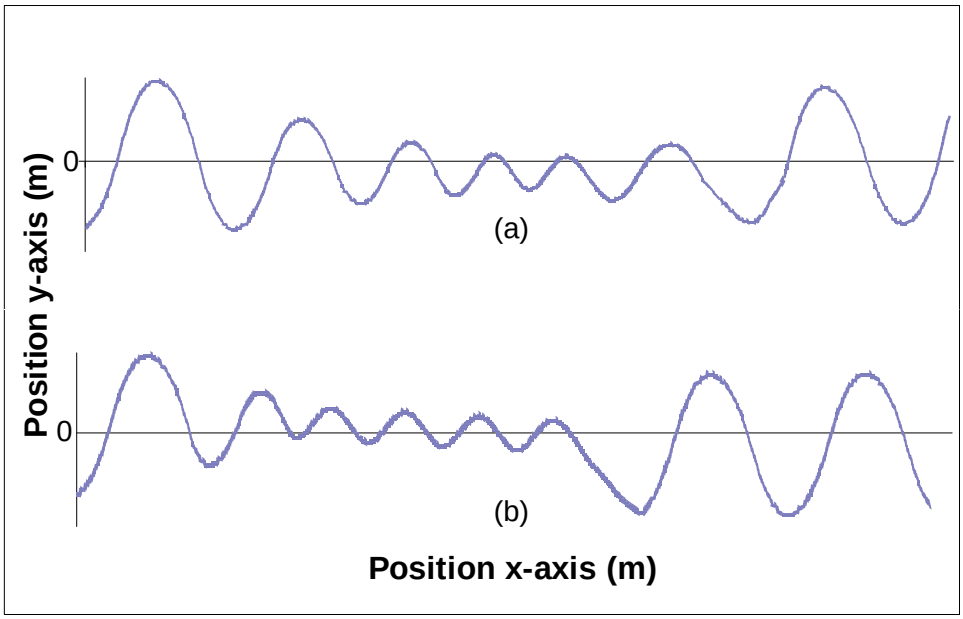


Figure 4.20. Trajectory of the head of the snake-like robot: (a) method 1, and (b) method 2

4.3 Parameters Selection for Locomotion Control of a Snake-like Robot

Due to the same control parameter in (3.3) is used for achieving both motion of a snake-like robot: 1) straight-path motion, and 2) turning motion, appropriate value of the parameter should be analyzed carefully. In this section, further analysis has been conducted for selection of parameters to control both motions.

Following Fig. 4.21 shows nonlinear behavior between $(\phi_2 - \phi_1)$ and deviation angle for various value of $(t_2 - t_1)$. As $(t_2 - t_1)$ gets higher, straight-path locomotion can be maintained and the deviation angle can be minimized. For turning motion, we can select lower value of $(t_2 - t_1)$ to get higher turning angle.

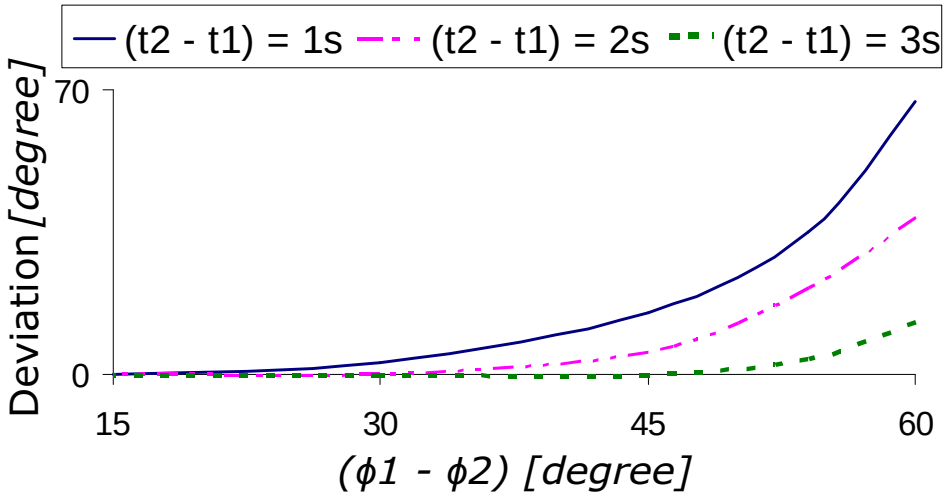


Figure 4.21. Different value of $(t_2 - t_1)$ with respect to $(\phi_2 - \phi_1)$.

From these results, desired locomotion control of a snake-like robot can be achieved by selecting $(t_2 - t_1)$ with respect to $(\phi_2 - \phi_1)$.

In Fig. 4.22, the simulation results of the trajectory of the head of a snake-like robot are presented with different $(\phi_2 - \phi_1)$ when $(t_2 - t_1) = 1$s. As $(\phi_2 - \phi_1)$ gets larger, the deviation of trajectory becomes apparent. Thus, two important parameters need to be controlled rigorously, that is, $(t_2 - t_1)$ with respect to $(\phi_2 - \phi_1)$.

For further clarification on the behavior of the snake-like robot locomotion during the body shape transition, torque analysis has been done for various combinations of $(t_2 - t_1)$ and $(\phi_2 - \phi_1)$ (Fig. 4.24). Three torque profiles for $(t_2 - t_1) = 1$s and 3s are shown, respectively, when : 1) $(\phi_2 - \phi_1) = \pi/12$, 2) $(\phi_2 - \phi_1) = \pi/4$, and 3) $(\phi_2 - \phi_1) = \pi/3$. The starting point of the phase transition is marked with dashed line. For easy view, the results of the torque profile are shown only for the first joint of the snake-like robot.

From the torque profile, it can be deduced that the change of the robot's position results in the change of the joint torque. The main influence of the deviation is the fluctuation of torque at the starting of the phase transition. Fluctuation of torque becomes apparent with larger $(\phi_2 - \phi_1)$. The larger the $(\phi_2 - \phi_1)$, the frequent the torque fluctuates after the phase

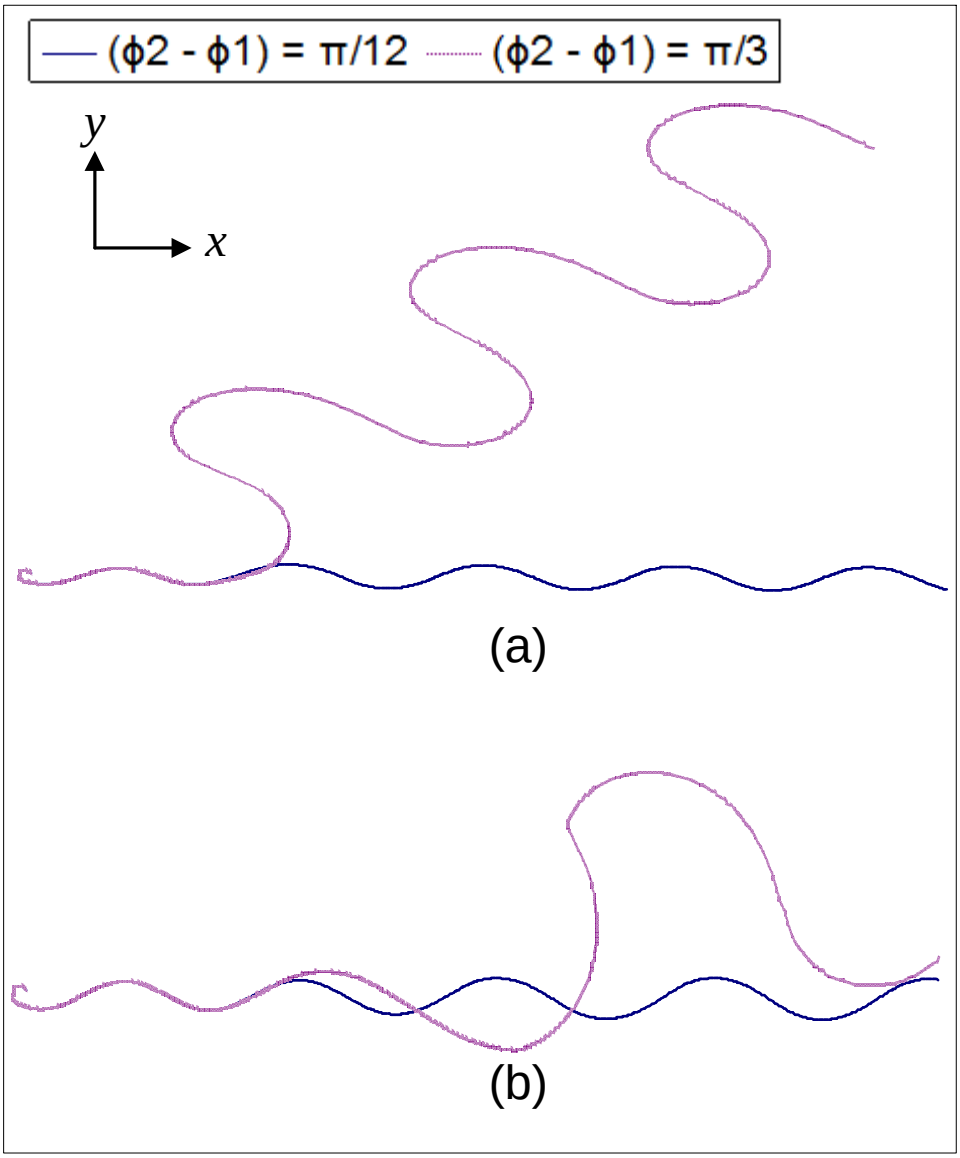


Figure 4.22. Simulation results of trajectory of a snake-like robot (position is in meter) when $(t_2 - t_1) = 1$s for different value of $(\phi_2 - \phi_1)$.

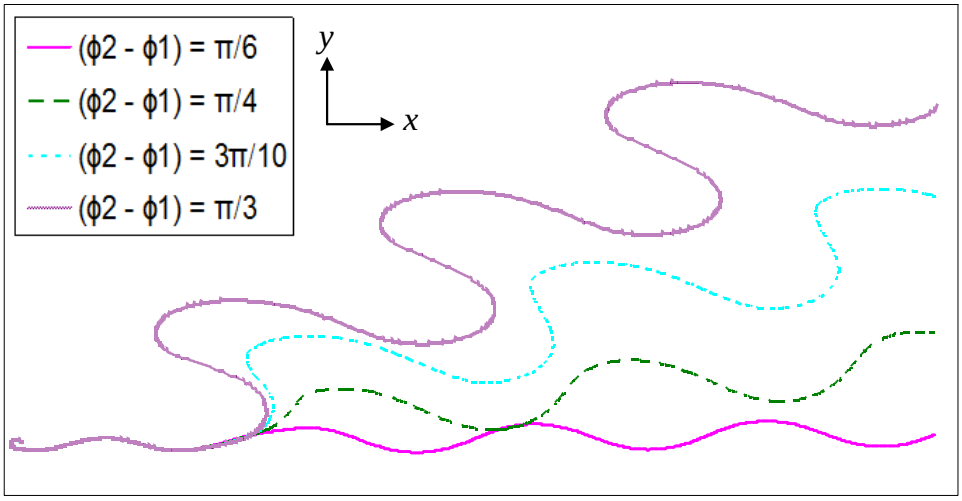


Figure 4.23. Simulation results of trajectory of a snake-like robot (position is in meter) for different value of $(\phi_2 - \phi_1)$ when $(t_2 - t_1)$ = (a) 1s, and (b) 3s.

transition, and fluctuates inconsistently. This condition results in unstable locomotion of the snake-like robot, thus, results in deviation of its trajectory or direction.

4.3.1 Discussion

An ODE simulator has been adopted for testing the feasibility of the proposed turning motion and straight-path motion for a snake-like robot. The simulation results show a promising approach in controlling smooth and continuous turning motion which can be applied for obstacle avoidance. Even though experimental verifications are not presented yet in this chapter, the simulation results can still be reliable [77] [78].

For turning motion, the amplitude modulation method proposed in [72] [77] shows several disadvantages such as: 1) the turning radius is big, turning angle is confined to the amplitude of serpenoid curve, and 3) the amplitude can only be modulated when the input angle is at zero. Nevertheless, the advantages of the amplitude modulation are its continuous input angle value when it changes from forward movement to turning movement and its control is simple. Meanwhile, for the phase modulation method proposed in [77], it also addresses a limitation of the method, where the input angle is discontinuous during the transition from forward motion to turning motion. This results in large sliding of

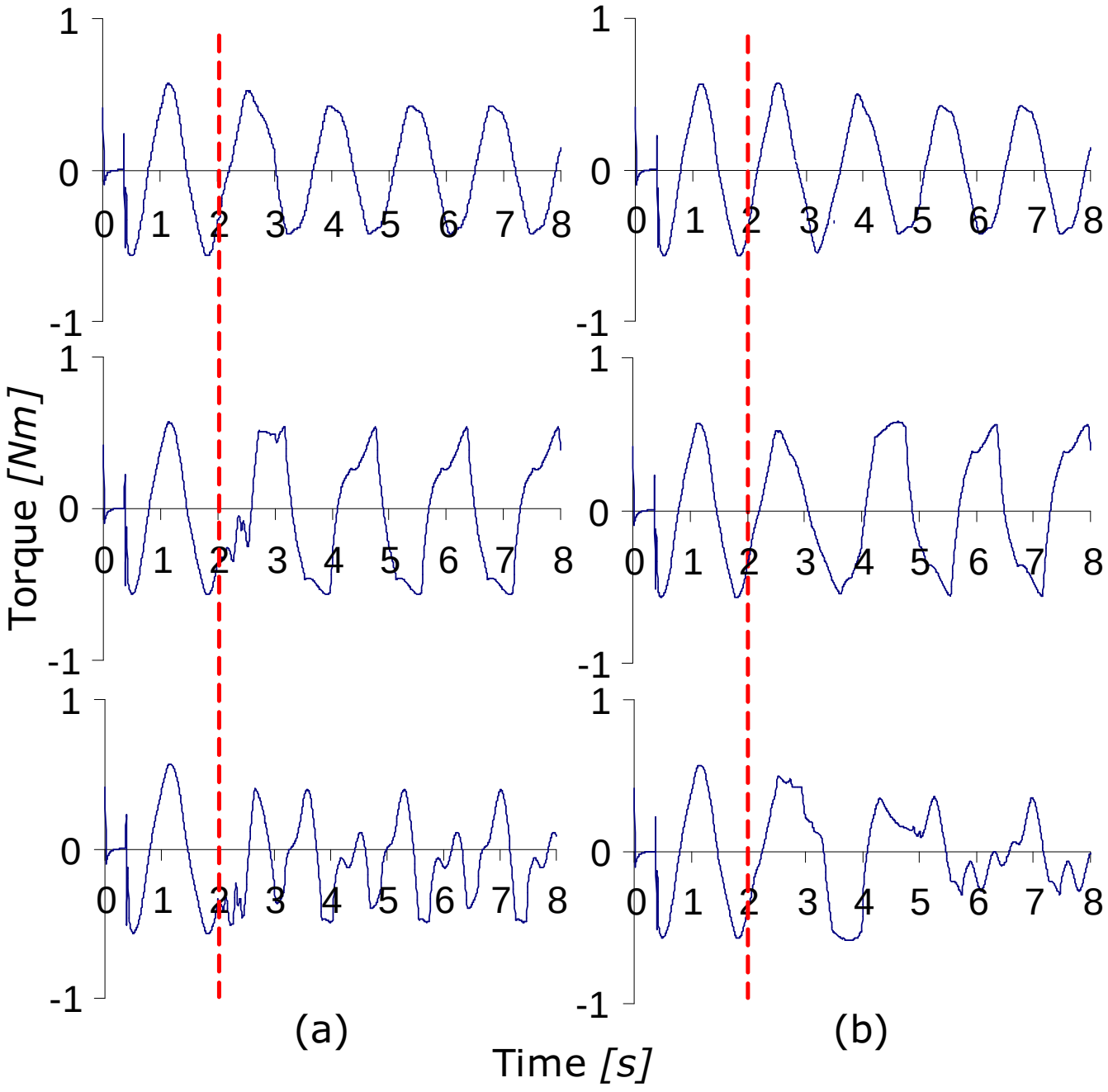


Figure 4.24. Torque profile of a snake-like robot for $(\phi_2 - \phi_1) = \pi/12$ [top], $(\phi_2 - \phi_1) = \pi/4$ [middle], and $(\phi_2 - \phi_1) = \pi/3$ [bottom], when $(t_2 - t_1) =$ (a) 1s, and (b) 3s.

the snake-like robot's motion. By applying our proposed activation function for controlling phase transition, smooth and continuous locomotion of the snake-like robot can be obtained. All of the said drawbacks for amplitude modulation and phase modulation methods have been overcome by our simple method. In addition, our simplified CPG structure with phase oscillator model promises an easy way of integrating feedback information. Another advantage of the phase transition method for turning motion is the ability to control the number of S-shape to adapt to the surrounding.

For straight-path motion, two methods have been proposed for controlling straight-path locomotion of a snake-like robot by manipulating $(t_2 - t_1)$ and t_1 as control parameter, respectively. The results prove the efficiency of both methods without producing any unstable movement and the ability of minimizing deviation of trajectory before and after the body shape transition of the snake-like robot. The advantage of using $(t_2 - t_1)$ as control parameter is that the transition can be initiated at any time, whereas for method 2, using t_1 as the control parameter will reduce the transition time, consequently reduce the distance, d (refer Fig. 4.16) needed for the snake-like robot to fully transform to its new body shape. However, both methods have their own drawback. As for method 1, longer distance, d, is required for fully transform due to the longer transition time. For method 2, the body shape transition needs to be started at zero point, which restricts the advantage of incorporating the linear bipolar function. Thus, either methods should be selected based on the suitability of the environment and situation.

The novelty of the approaches presented in this chapter is the parameter used for both turning and straight-path motion control is independent from the parameters of the CPG mathematical model. Previous studies focus on the CPG parameters to control a snake-like robot locomotion such as speed and amplitude. By incorporating the derived linear bipolar function (3.3), a smooth change of the body shape as well as the deviation of trajectory can be controlled. It is no doubt that speed of the robot contributes to the slippage when adding a bias to the control signal, which results in deviation from the robot's original path,

but the key point is to minimize the slippage regardless of its speed by manipulating the linear bipolar parameters.

However, due to the influence of external parameters such as slippage between the wheel and the ground (friction) and hardness of the ground and surface parameter, which are beyond control, the exact turning angle may not be accurate. Moreover, to get perfect straight-path motion is also impossible due to the external factors. Thus, experimental results may be different from the results in the simulation environment.

4.3.2 Summary

The proposed body shape control for a snake-like robot can be used for turning motion and straight-path motion. The advantage of the phase transition method is that the bias to the output signal can be altered at any random time for online modification. Using the phase transition method, a smooth body shape transition and continuous input angle can be achieved, with turning and straight-path control ability to adapt to the surrounding. For straight-path motion, two methods have been proposed for the locomotion control: by manipulating $(t_2 - t_1)$ or t_1.

Based on the relationship between parameter $(t_2 - t_1)$ with respect to $(\phi_2 - \phi_1)$, desired turning or straight-path motion of a snake-like robot can be achieved for adapting to the environment.

Chapter 5

Experimental Verifications

One of the advantage of the CPG-based controller is the smoothing gaits transition which depends on communication between modules. For the control system of our snake-like robot, we implement a distributed controller where each of the snake joint is responsible for calculating its own CPG output to perform the desired joint angle. The distribute system is fast, because there is no need to communicate with a central controller especially for large number of joints. Each of the snake joint receives information from its neighbor joints to produce the required joint position. Detail of the control system and the communication protocol are provided in the Appendix.

In this section, we verify our proposed locomotion control into our snake-like robot called SR-I. Its physical properties are described in Table 5.1. Fig. 5.1 shows the image of our snake-like robot with different number of joint. Structure of one link of the SR-I is shown in Fig. 5.1. Note that, we can increase or decrease the number of snake units, n depending on our objective. For this experiments, we will use only 4 units to verify our CPG model. In the future, we will increase the number of snake units for larger mobility.

For the experiment, each joint of the snake-like robot is controlled by one MCU where each MCU controls and sends the CPG output signal, x_i to the motors. Communication between each joint is done using Inter-Integrated Circuit (I^2C). The control system design of the snake-like robot is shown in Fig. 5.3.

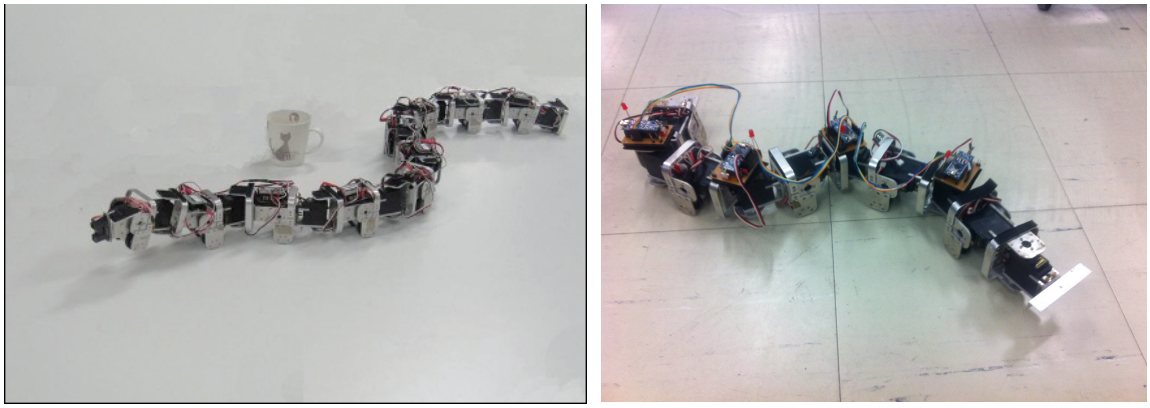


Figure 5.1. Snake-like robot for experimental analysis with different number of joints: (a) 8 joints, and (b) 4 joints.

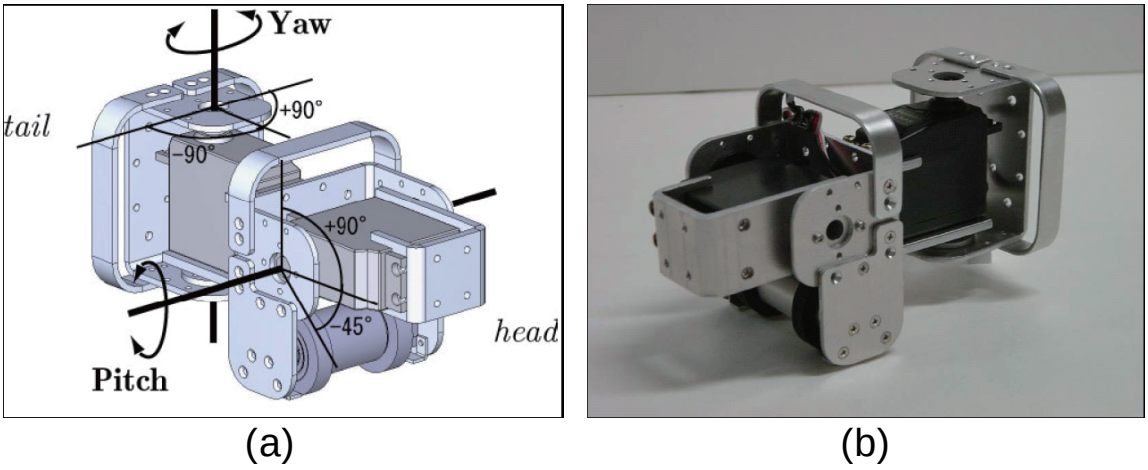


Figure 5.2. Structure of one link in SR-I.

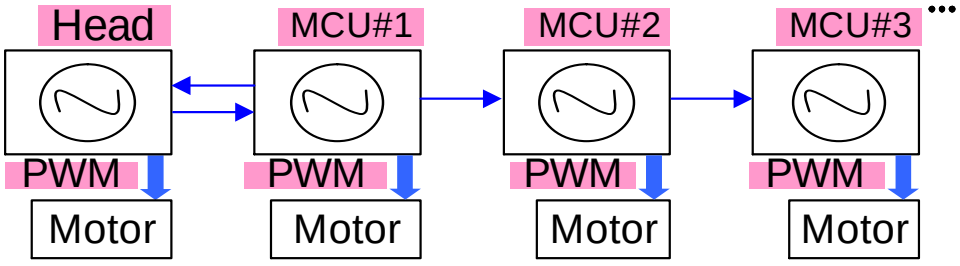


Figure 5.3. Control system design.

Table 5.1. Description of the parameters

Items	Details
Snake unit	n
Size of each unit [mm^3]	130x62x77
Weight of each unit [kg]	0.28
Motion range of yaw angle [deg]	[-90,+90]
Actuator	RC servo motor

5.1 Analysis of CPGs with Phase Oscillator Model

This section shows the verifications of the locomotion of the snake-like robot for different number of S-shape, and, forward and backward motions based on the proposed unidirectional CPG network in Section 2.2.2. The CPG parameters used in this experiment are as follows: $v = 0.1$, $w_{ij} = 10$, and $\tau = 1$.

5.1.1 Control of Number of S-shape

To verify the number of S-shape, N, we can observe locomotion curvature of the snake-like robot. For instance, in Fig. 5.4, we illustrate on how we define number of S-shape of a snake-like robot. Two ways to recognize the number of S-shape: 1) the amplitude of the locomotion curvature will become smaller as N gets higher, and 2) number of peaks of the locomotion's wave.

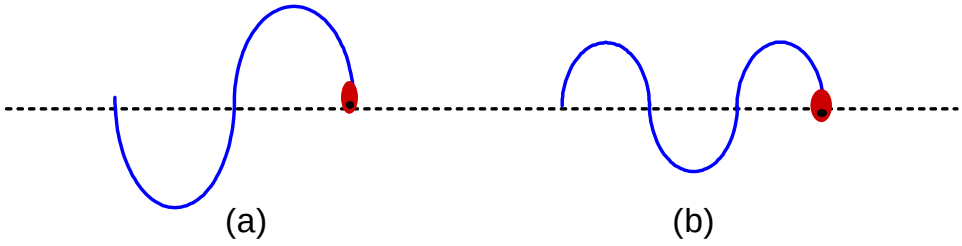


Figure 5.4. Definition of number of S-shape (a) $N = 1.0$, (b) $N = 1.5$.

In this experiment, due to lower number of joints ($n = 4$), we only show two different number of S-shape for $\phi = \pi/2$ ($N = 1.0$), and (b) $\phi = 3\pi/4$ ($N = 1.5$). For $N = 1.0$ (Fig. 5.5 (a)), the locomotion curvature is symmetric as one period of wave. Meanwhile, for $N = 1.5$, we can observe that the locomotion curvature has three peaks (see Fig. 5.5 (b)),

which are identical to Fig. 5.4 (b). The number of S-shape can be controlled by manipulating the value of parameter ϕ using (2.22). The desired value of ϕ is inserted in (2.19) and the calculated CPG output, x_i with a gain inputs to the PWM to drive the joint motor.

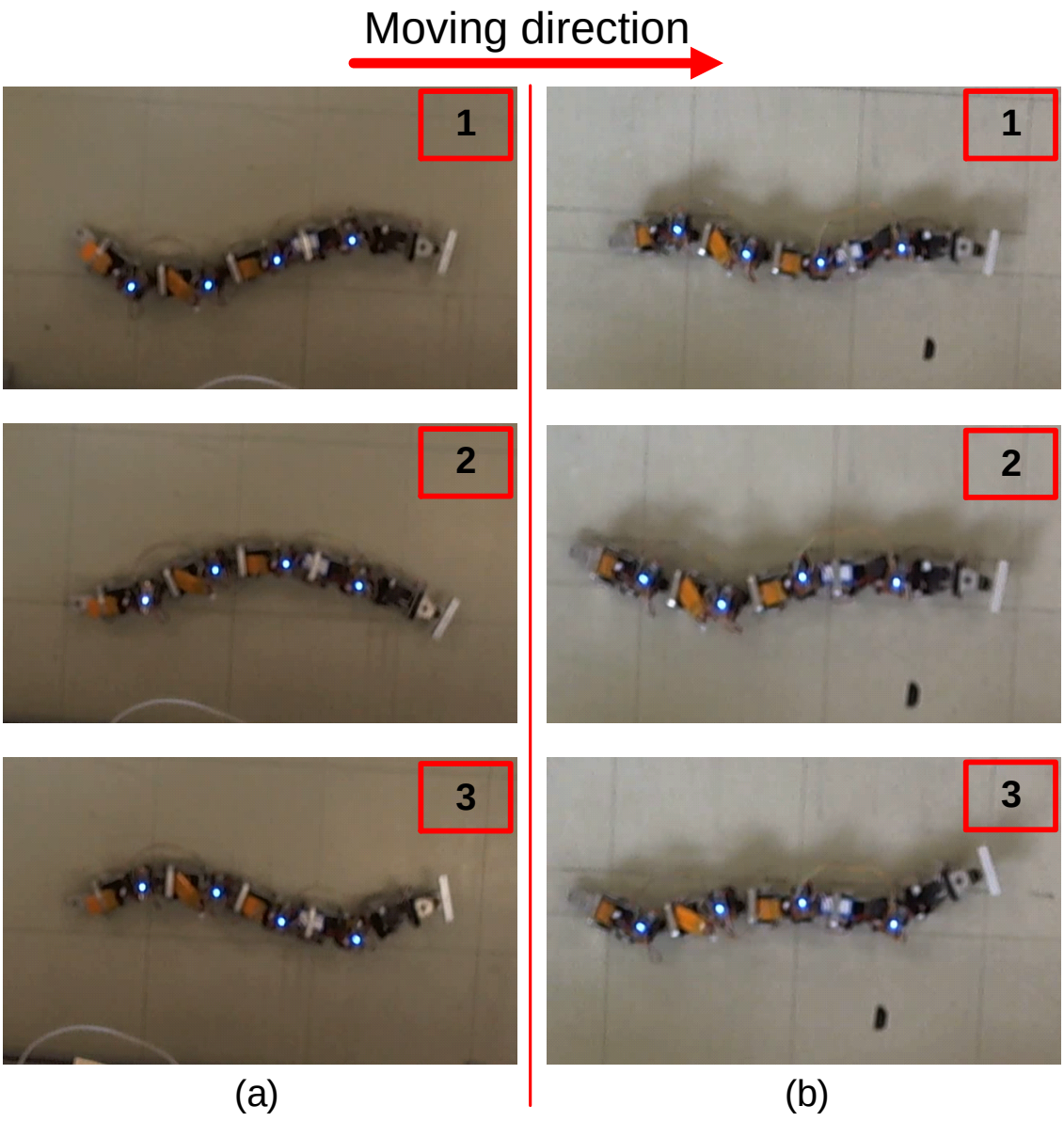


Figure 5.5. Different number of S-shape obtain by experiment (a) $\phi = \pi/2$ ($N = 1.0$), (b) $\phi = 3\pi/4$ ($N = 1.5$).

In Fig. 5.5, the upper figures show the locomotion of the snake-like robot performing the desired number of S-shape. The middle figures show the transition of the locomotion

curvature before performing the motion shown in the bottom figures, which is the mirrored version of the locomotion curvature shown in the upper figures.

5.1.2 Forward and Backward Movement

For forward and backward movement, the only change of the CPG parameters is the sign of the ϕ, where we can control the output sequence of the CPGs. Referring to Section 2.3.2, the sign ($\pm$) of parameter ϕ for oscillator 1 should always be opposite to the other oscillators. For forward movement, the sign of ϕ for oscillator 1 is negative (-), which results in ascending order of the CPG outputs, whereas for backward movement, the sign of the ϕ is reversed. For the experimental verification, the setting of CPG parameters is similar as the simulation, and the value of ϕ is selected arbitrarily. Fig. 5.6 shows the control of forward movement of the snake-like robot, while Fig. 5.7 shows the control of the backward movement.

5.2 Body Shape Control of the Snake-like Robot

In Chapter 3, we showed that by changing the value of ϕ, we can change the S-shape of the snake-like robot. However, if we modify ϕ instantly online, the snake-like robot movement becomes unstable. This motion is not desired as it will damage the gear box or the joint motor. Therefore, in this section, we experimentally verify our proposed linear bipolar activation function (3.3) for smooth body shape transition of the snake-like robot. The CPG parameters used in this experiment are as follows: $v = 0.1$, $w_{ij} = 10$, and $\tau = 1$. Value of ϕ differs from the simulation setup because less actuated joints were used in the experiment.

Fig. 5.8 (a) shows a body shape transition of the snake-like robot from $\phi = \pi/2$ ($N = 1.0$) to $\phi = 3\pi/4$ ($N = 1.5$) when $(t_2 - t_1) = 1$s. The top figure in Fig. 5.8 (a) shows the initial body shape ($N = 1.0$), while the middle figure shows the body shape transition from $N = 1.0$ to $N = 1.5$, and the bottom figure shows the final body shape ($N = 1.5$).

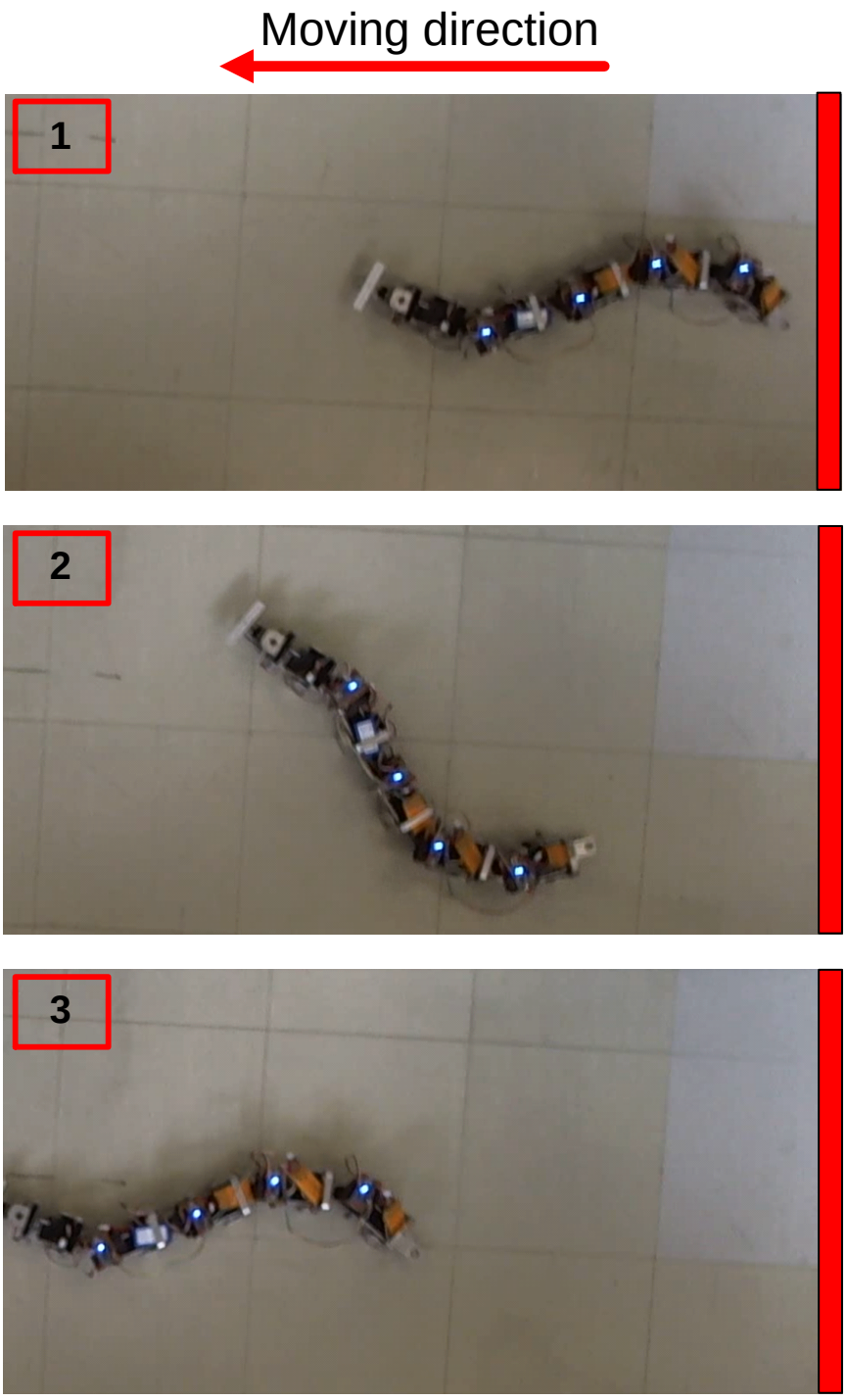


Figure 5.6. Forward movement of a snake-like robot.

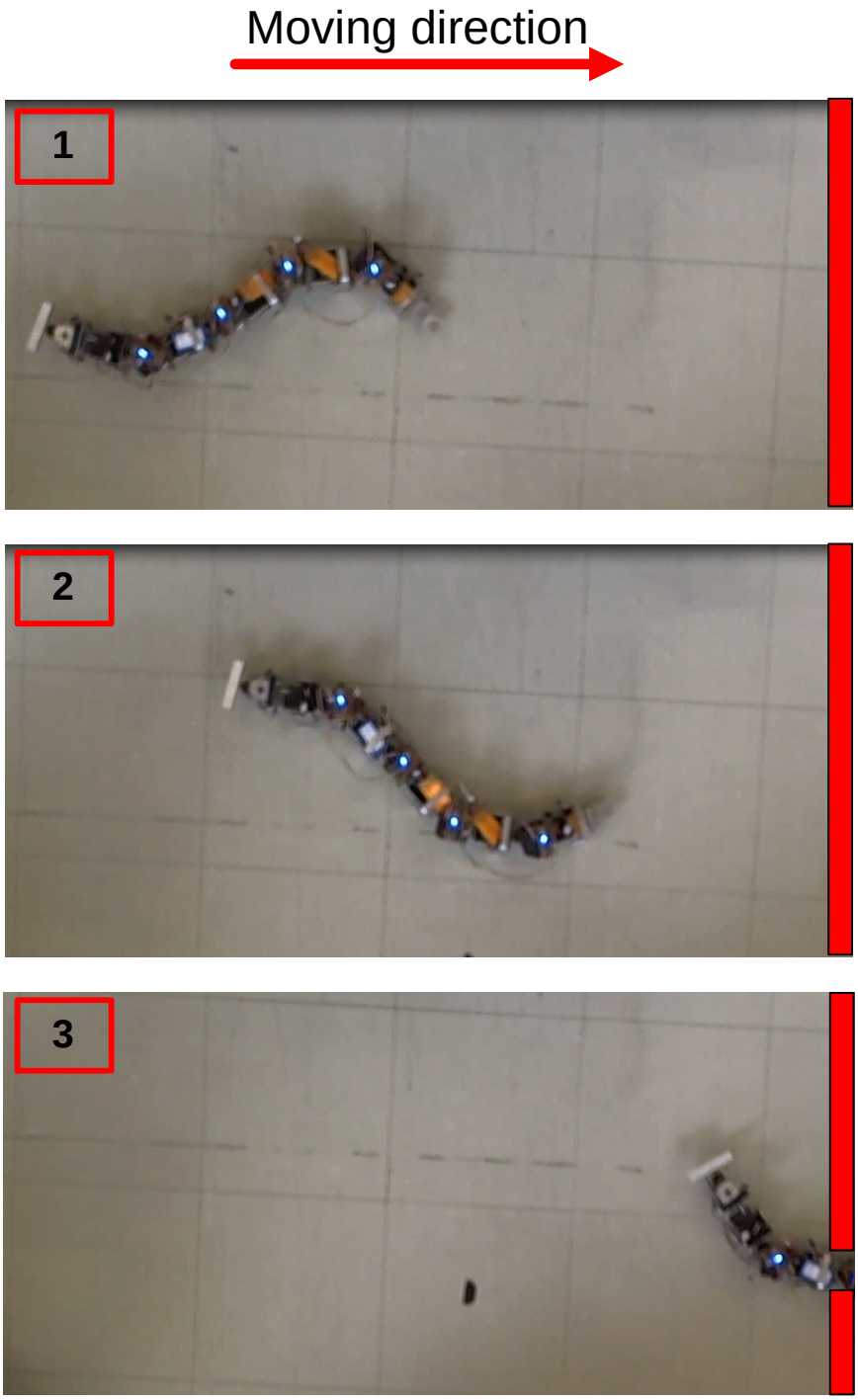


Figure 5.7. Backward movement of a snake-like robot.

Similar with the simulation results in Fig. 3.10, the direction of the snake-like robot results in less deviation if we change the value of ϕ from small to large. In the next section, we will show the effect of the snake-like robot's direction when we change ϕ from large to small. Thus, with the experimental verification in Fig. 5.8 (a), we had successfully achieved body shape transition using our proposed linear bipolar activation (3.3), without any unstable movement of the snake-like robot occurs during the experiment.

5.3 Versatile Locomotion based on Body Shape Control for Obstacle Avoidance

Using the body shape control model in the previous section, locomotion of a snake-like robot can be achieved in two ways: 1) straight-path motion, and 2) turning motion. This section shows the experimental verifications on a real snake-like robot for different these two different motions. The CPG parameters used in this experiment are as follows: $v = 0.1$, $w_{ij} = 10$, and $\tau = 1$. Small value of v is selected to lower the speed of the snake-like robot.

5.3.1 Turning Motion

Turning motion of a snake-like robot can be realized by modifying value of ϕ from large to small. In Fig. 5.8 (b), the body shape is changed from $\phi = 3\pi/4$ ($N = 1.5$) to $\phi = \pi/2$ ($N = 1.0$) for $(t_2 - t_1) = 1$s. With small $(t_2 - t_1)$, the rate of the phase transition becomes higher, which causes a large deviation of the snake-like robot's direction. Due to the limited number of joints, we only show one experimental verification for the turning motion.

The experimental result in Fig. 5.8 proves that as the value of ϕ changes from large to small, the deviation of trajectory of the snake-like robot becomes apparent. For simulation results please refer to Fig. 3.10 and Fig. 3.11.

In Fig. 5.9, the experimental result shows the locomotion of the snake-like on rough

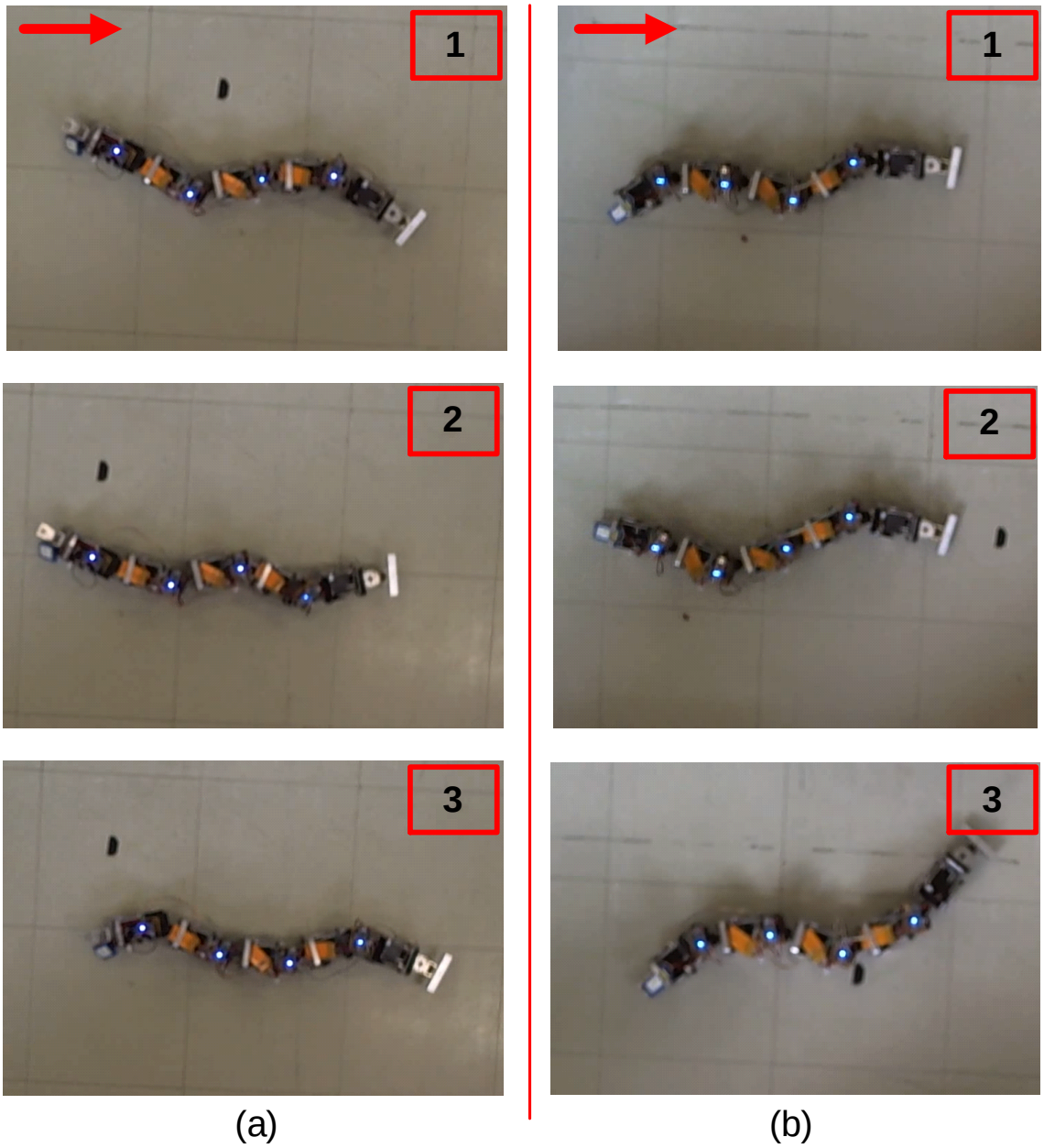


Figure 5.8. Body shape transition on smooth surface when $(t_2 - t_1) = 1$s from: (a) $\phi = \pi/2$ ($N = 1.0$) to $\phi = 3\pi/4$ ($N = 1.5$), and (b) $\phi = 3\pi/4$ ($N = 1.5$) to $\phi = \pi/2$ ($N = 1.0$).

surface. The deviation angle maybe different depending on the surface properties, however, this result shows that deviation angle becomes apparent when N is changed from small to large regardless the surface parameters.

5.3.2 Straight-path Motion

Some situations need a straight-path motion of a snake-like robot, for instance, when moving from a large space into a narrow tunnel. In Section 4.2, we have already analyzed two control strategies: 1) the transition time of the phase transition, $(t_2 - t_1)$ (method 1) and 2) the trigger time, t_1 (method 2), for controlling a straight-path motion of a snake-like robot and showed the simulation results. In this section, we further verify the straight-path motion into a snake-like robot.

Since the application for both control strategies is similar i.e., to maintain a straight-path, the experiment is done only for method 1. The parameters are set as follows: $(t_2 - t_1)$ = 2s and 3s, $v = 0.1$, $w_{ij} = 10$, and $\tau = 1$.

As the roughness of the surface increases, the deviation of the trajectory can be minimized further, as shown in Fig. 5.10 for smooth surface and Fig. 5.11 for rough surface, due to less slippage between the robot's wheels and the surface.

5.4 Summary

Our experimental results verify the efficiency of our proposed CPG model. However, due to the influences of external parameters such as slippage between the wheel and the ground (friction) and hardness of the ground and surface parameter, which are beyond our control, it is difficult to achieve the same results as the simulation. The exact turning angle may not be accurate and a perfect straight-path motion is also impossible. Another factor of the discrepancies is the control of the servo motor which sometimes differs in their home position, which results in difficulty to maintain symmetric body shape and straight forward motion.

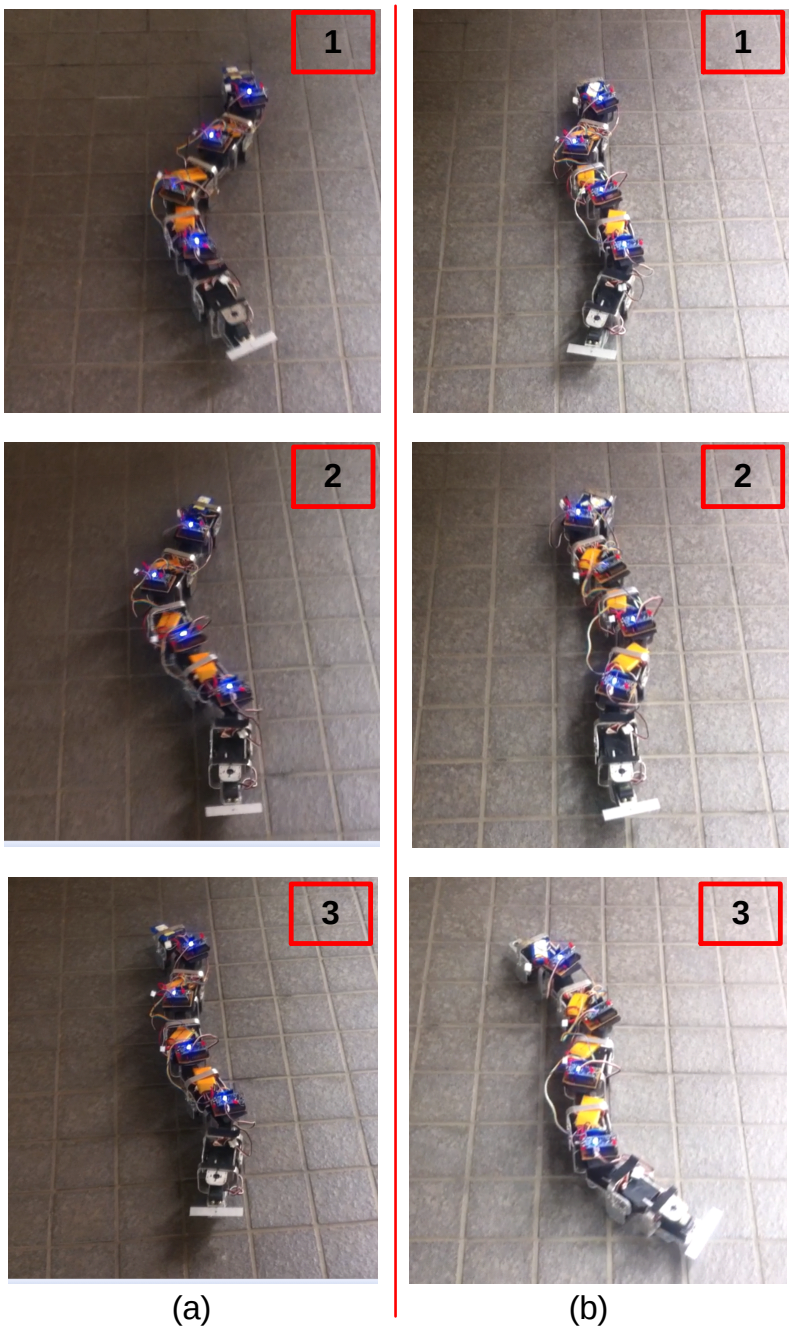


Figure 5.9. Body shape transition on rough surface when $(t_2 - t_1) = 1$s from: (a) $\phi = \pi/2$ ($N = 1.0$) to $\phi = 3\pi/4$ ($N = 1.5$), and (b) $\phi = 3\pi/4$ ($N = 1.5$) to $\phi = \pi/2$ ($N = 1.0$).

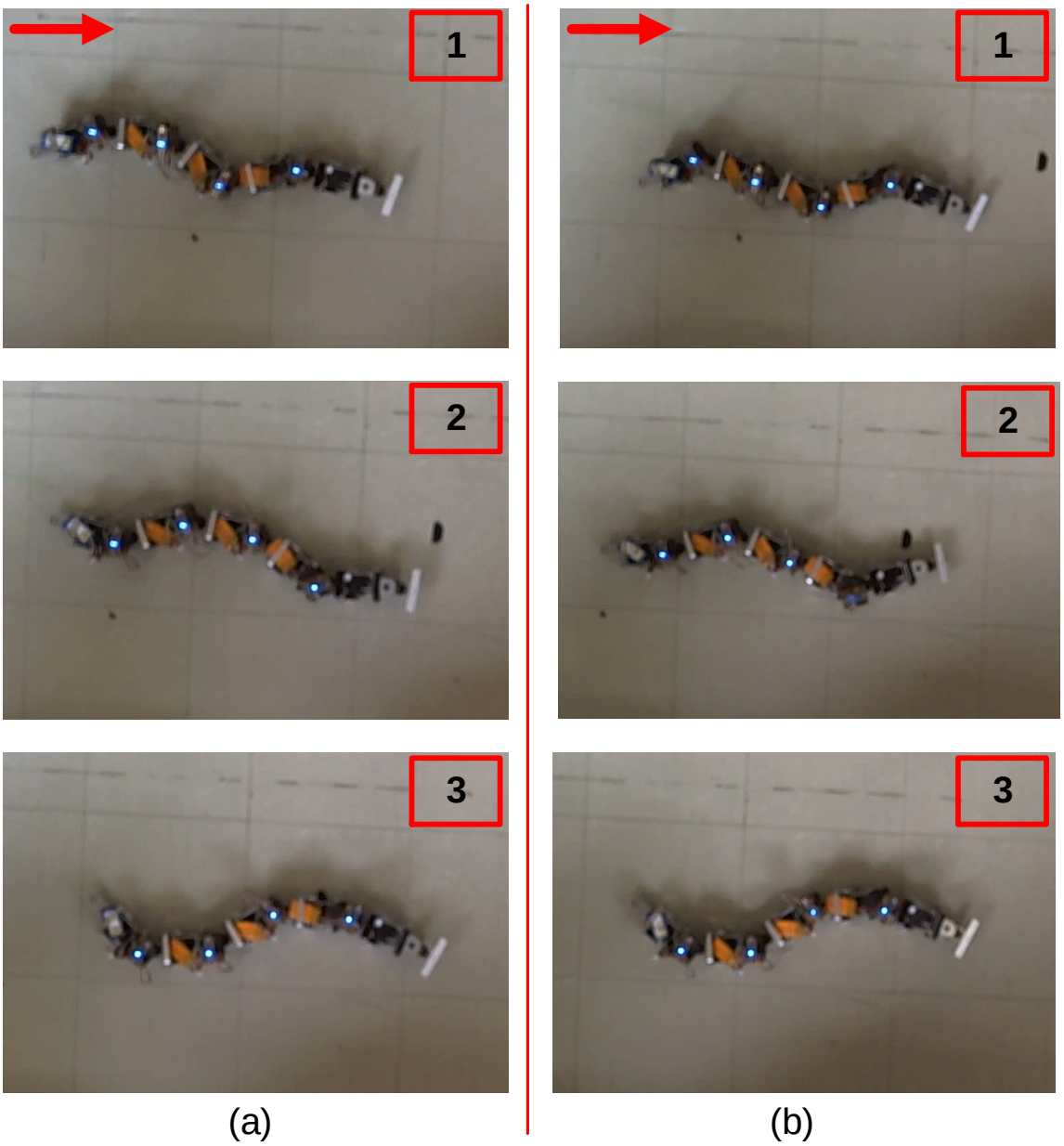


Figure 5.10. Body shape transition from $\phi = 3\pi/4$ ($N = 1.5$) to $\phi = \pi/2$ ($N = 1.0$) when: (a) $(t_2 - t_1) = 2$s, and (b) $(t_2 - t_1) = 3$s.

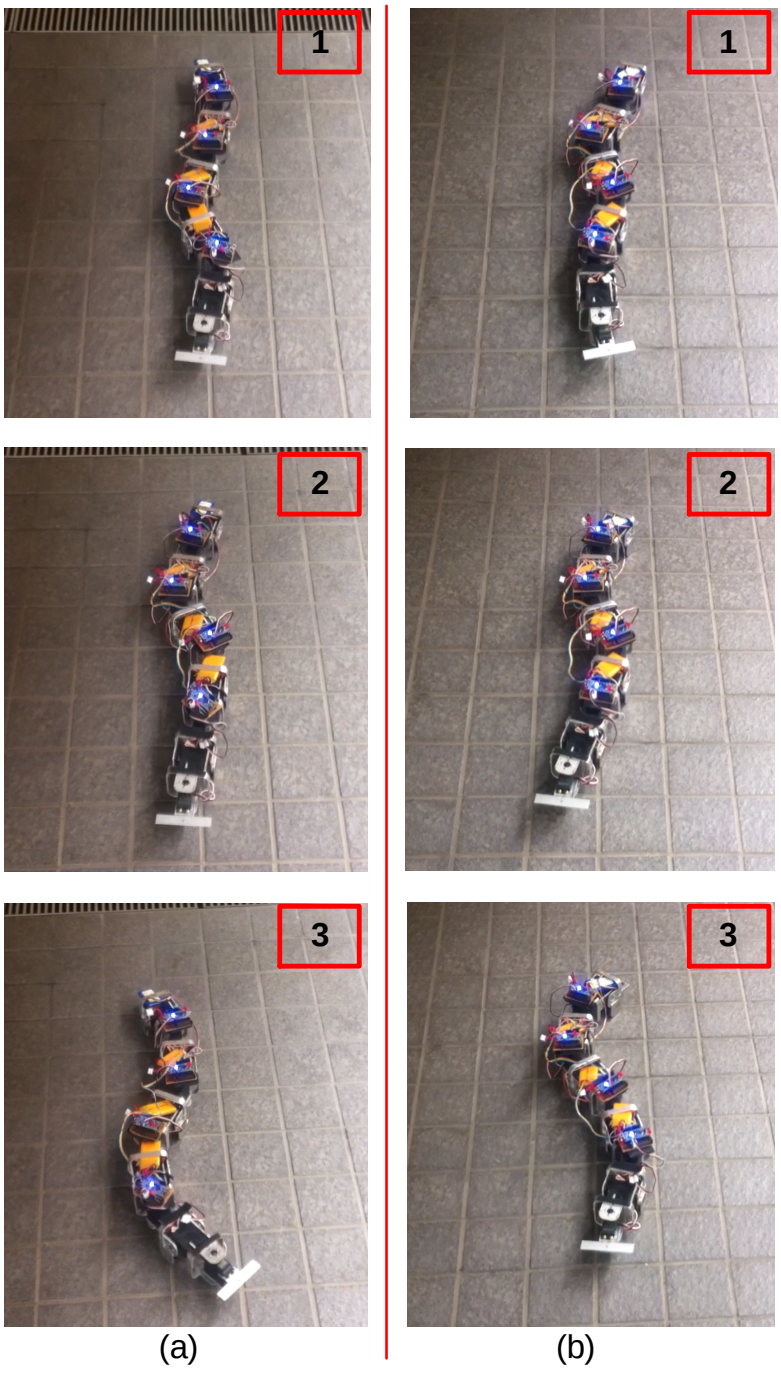


Figure 5.11. Body shape transition on rough surface from $\phi = 3\pi/4$ ($N = 1.5$) to $\phi = \pi/2$ ($N = 1.0$) when: (a) $(t_2 - t_1) = 2$s, and (b) $(t_2 - t_1) = 3$s.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

CPG-based approach can be used to control various locomotion of a snake-like robot. The phase oscillator network has been selected to represent the dynamic model of the CPG. Using the phase oscillator model, we have developed a simple unidirectional CPG network to control the phase difference between coupled oscillators. This proposed unique CPG structure has been compared with the existence bidirectional coupling in term of their convergence speed and computational cost. It has been proven that our proposed CPG structure is simple, less computational cost, fast convergence speed and converge to phase locking state. Furthermore, we have also verified the network stability which can produce distributed wave form with the control of phase difference.

By controlling the phase difference, we can mimic the serpentine locomotion of a biological snake with symmetric S-shape. However, there is a restriction on the selection of the frequency and the weight of the oscillator. This restriction has been solved by adding a parameter to control both frequency and weight in an appropriate range. With the proposed unidirectional network, we can also realize the forward and backward movement by only changing the sign of the phase difference control parameter.

Due to the significant feature of online parameter modification, we have proposed an

activation function for controlling the body shape of a snake-like robot, which can be modified online at any random time. This novel approach of body shape transition has been analyzed thoroughly in term of the smoothness of the output signal and torque profile of the snake-like robot's joint. A simple linear bipolar function has been developed to control the phase transition, which results in the modifying of the number of S-shape. Thus, various applications can be achieved due to the online modification ability of the locomotion of the snake-like robot.

Using the developed activation function, we proposed a control strategy for two different kinds of motion: 1) turning motion, and 2) straight-path motion.The same parameter has been used to control both motions with appropriate selection of the parameter value. This is the superiority of our proposed control method where same parameter can be utilized to produce two different motions. The turning motion can be easily produced by giving bias to the output signal, which has been proposed by various researches. The difference between the turning method is the selection of the control parameter to produce bias to the output signal. However, to achieve a straight-path motion when parameter is modified online is not an easy task. Up till now, there is no discussion on the straight-motion control. But we have solved this problem by proposing a parameter to control both turning and straight-path motions.

However, due to the influence of external parameters such as slippage between the wheel and the ground (friction) and hardness of the ground and surface parameter, which are beyond control, the exact turning angle may not be accurate. Moreover, to get perfect straight-path motion is also impossible due to the external factors, but the key point is to minimize the deviation of trajectory by manipulating the linear bipolar parameters. Thus, experimental results may be different from the results in the simulation environment.

In conclusion, the novel CPG structure and the developed activation function for controlling body shape online of a snake-like robot can be used to realize the serpentine locomotion. A single control parameter of the phase oscillator model can produce various kinds

of versatile locomotion: S-shape motion, forward and backward motion, turning motion, and straight-path motion.

6.2 Future Work

Although various locomotion control of snake-like robot has been proposed, there is a large number of problems still remaining for realization on practical snake-like robots, both in software and hardware. For our future works, we will continue investigating the appropriate control system of the joint motor, sensors and the software programming to realize the proposed locomotion. Moreover, for our ongoing work, we will analyze the followings:

1) various activation functions to produce smooth parameter transition, which can be used generally in other kinds of mobile robot.

2) the radius of the curvature with respect to ϕ. If we can develop a quantitative analysis of the radius of curvature with respect to the parameter, we can easily control the appropriate body shape of the snake-like robot to adapt to obstacles of different size.

3) implementing sensory feedback information into the CPG model for adaptive and autonomous control. Easy integration of the sensor to obtain feedback information from the environment is one of the advantage of the CPG. Thus, it should be implemented in our future works.

4) further implementation of the control strategies on the real snake-like robot. Verifications on real field experiment is useful to show the effectiveness of the proposed control.

Bibliography

[1] P. Liljeback, K. Y. Pettersen, O. Stavdahl, and J. T. Gravdahl, *Snake Robots: Modelling, Mechatronics, and Control*. Oxford: Springer, 2013.

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

[3] M. Mori and S. Hirose, "Three-dimensional serpentine motion and lateral rollingby active cord mechanism acm-r3," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2002, pp. 829–834.

[4] H. Yamada, S. Chigisaki, M. Mori, K. Takita, K. Ogami, and S. Hirose, "Development of amphibious snake-like robot acm-r5," in *Proc. 36th Int. Symp. Robotics*, 2005.

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

[6] C. Wright, A. Johnson, A. Peck, Z. McCord, A. Naaktgeboren, P. Gianfortoni, M. Gonzalez-Rivero, R. Hatton, and H. Choset, "Design of a modular snake robot," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2007, pp. 2609–2614.

[7] H. Yamada and S. Hirose, "Study of a 2-dof joint for the small active cord mechanism," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2009, pp. 3827–3832.

[8] H. Ohashi, H. Yamada, and S. Hirose, "Loop forming snake-like robot acm-r7 and its serpenoid oval control," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2010, pp. 413–418.

[9] Z. Y. Bayraktaroglu, A. Kilicarslan, A. Kuzucu, V. Hugel, and P. Blazevic, "Design and control of biologically inspired wheel-less snake-like robot," in *Proc. IEEE / RAS-EMBS international conference on biomedical robotics and biomechatronics*, 2006, pp. 1001–1006.

[10] X. Wu, "Cpg-based neural controller for serpentine locomotion of a snake-like robot," Ph.D. dissertation, Science and Engineering, Ritsumeikan University, Japan, 2011.

[11] C. Zhou and K. H. Low, "Design and locomotion control of a biomimetic underwater vehicle with fin propulsion," *IEEE/ASME Transactions on Mechatronics*, vol. 17, no. 1, pp. 25–35, 2012.

[12] K. Seo, S. J. Chung, and J. J. E. Slotine, "Cpg-based control of a turtle-like underwater vehicle," *Autonomous Robot*, vol. 28, pp. 247–269, 2010.

[13] C. Liu, Q. Chen, and D. Wang, "Cpg-inspired workspace trajectory generation and adaptive locomotion control for quadraped robots," *IEEE Transactions on Systems, Man, and Cybernatics*, vol. 41, no. 3, pp. 867–880, 2011.

[14] T. Wang, W. Guo, M. Li, F. Zha, and L. Sun, "Cpg control for biped hopping robot in unpredictable environment," *Journal of Bionic Engineering*, vol. 9, no. 1, pp. 29–38, 2012.

[15] B. Li, Y. Li, and X. Rong, "Gait generation and transitions of quadruped robot based on wilson-cowan weakly neural networks," in *Proceedings of the 2010 International Conference on Robotics and Biomimetics*, Tianjin, China, 2010, pp. 19–24.

[16] D. K. Nanto, W. H. Cooper, J. M. Donnelly, and R. Johnson. (2011, April) Japans 2011 earthquake and tsunami: Economic effects and implications for the united states. [Online]. Available: http://www.crs.gov

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

[18] D. L. Hu, J. Nirody, T. Scott, and M. J. Shelley, "The mechanics of slithering locomotion," in *Proceedings of the national academy of sciences*, 2009, p. 1008110085.

[19] M. Walton, B. C. Jayne, and A. F. Bennet, "The energetic cost of limbless locomotion," *Science*, vol. 249, no. 4968, pp. 524–527, 1990.

[20] H. B. Lillywhite, *How Snakes Work: Structure, Function and Behavior of the World's Snakes*. Oxford: Oxford University Press, 2014.

[21] K. J. Dowling, "Limbless locomotion.learning to crawl with a snake robot," Ph.D. dissertation, Carnegie Mellon University, 1997.

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

[23] J. Borenstein and A. Borrell, "The omnitread ot-4 serpentine robot," in *Proc. IEEE Int. Conf. Robotics and Automation*, Pasadena, CA, USA, 2008, pp. 1766–1767.

[24] M. Frasca, P. Arena, and L. Fortuna, *Bio-inspired Emergent Control of Locomotion Systems*. World Science, 2004.

[25] A. J. Ijspeert, "Central pattern generators for locomotion control in animals and robots: a review," *Neural Networks*, vol. 21, no. 4, pp. 642–653, 2008.

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

[27] D. Rollinson and H. Choset, "Virtual chassis for snake robots," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2011, pp. 221–226.

[28] D. P. Tsakiris, M. Sfakiotakis, A. Menciassi, G. La Spina, and P. Dario, "Polychaete-likeundulatory robotic locomotion," in *Proc. IEEE Int. Conf. Robotics and Automation*, Barcelona, Spain, 2005, pp. 3029–3034.

[29] J. Ostrowski and J. Burdick, "Gait kinematics for a serpentine robot," in *Proc. IEEE Int. Conf. Robotics and Automation*, 1996, pp. 1294–1299.

[30] F. Matsuno and K. Suenaga, "Control of redundant 3d snake robot based on kinematic model," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2003, pp. 2061–2066.

[31] P. Prautsch and T. Mita, "Control and analysis of the gait of snake robots," in *Proc. IEEE Int. Conf. Control Applications*, 1999, pp. 502–507.

[32] H. Date, Y. Hoshi, M. Sampei, and N. Shigeki, "Locomotion control of a snake robot with constraint force attenuation," in *Proceedings of the American Control Conference*, 2001, pp. 113–118.

[33] J. W. Ayers, J. L. Davis, and A. Rudolph, "Neurotechnology for biomimetic robots," 2002.

[34] G. Taga, "A model of the neuro-musculo skeletal system for anticipatory adjustment of human locomotion during obstacle avoidance," *Biological Cybernetics*, vol. 78, no. 1, pp. 9–17, 1998.

[35] R. C. Arkin, *Behavior-Based Robotics*. The MIT Press, 1998.

[36] G. M. Sheperd, *Neurobiology*. Oxford: Oxford University Press, 1997.

[37] S. Grillner, *Feedback and Motor Control in Invertebrates and Vertebrates*. Croom Helm, 1985.

[38] R. Heliot and B. Espiau, "Multisensor input for cpg-based sensory-motor coordination," *IEEE Transactions on Robotics*, vol. 24, no. 1, pp. 191–195, 2008.

[39] A. H. Cohen and P. Wallen, "The neuronal correlate of locomotion in fish," *Experimental Brain Research*, vol. 41, no. 1, pp. 11–18, 1980.

[40] W. L. Xu, F. Clara Fang, J. Bronlund, and J. Potgieter, "Generation of rhythmic and voluntary patterns of mastication using matsuoka oscillator for a humanoid chewing robot," *Mechatronics*, vol. 19, no. 2, pp. 205–217, 2009.

[41] A. Crespi and A. J. Ijspeert, "Online optimization of swimming and crawling in an amphibious snake robot," *IEEE Transactions on Robotics*, vol. 24, no. 1, pp. 75–87, 2008.

[42] Z. Lu, S. Ma, B. Li, and Y. Wang, "3d locomotion of a snake-like robot controlled by cyclic inhibitory cpg model," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Beijing, China, 2006, pp. 3897–3902.

[43] X. Wu and S. Ma, "Cpg-based control of serpentine locomotion of a snake-like robot," in *Proceedings of the 9th International IFAC symposium on robot control*, Gifu, Japan, 2009, pp. 871–876.

[44] C. Tang and S. Ma, "A self-tuning multi-phase cpg enabling the snake robot to adapt to environments," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, San Francisco, USA, 2011, pp. 1869–1874.

[45] Z. Lu, S. Ma, B. Li, and Y. Wang, "Serpentine locomotion of a snake-like robot controlled by musical theory," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Edmonton, Canada, 2005, pp. 102–107.

[46] Y. Kuramoto, *The handbook of brain theory and neural networks.* The MIT Press, 2003.

[47] M. Golubitsky, I. Stewart, P. L. Buono, and J. J. Collins, "Symmetry in locomotor central pattern generators and animal gaits," *Nature*, vol. 401, pp. 693–695, 1999.

[48] S. Ekeberg, "A combined neuronal and mechanical model of fish swimming," *Biological Cybernetics*, vol. 69, no. 5-6, pp. 363–374, 1993.

[49] T. L. Williams, *Locomotor neural mechanisms in arthropods and vertebrates.* Manchester University Press, 1991.

[50] J. Collins and S. Richmond, "Hard-wired central pattern generators for quadrupedal locomotion," *Biological Cybernetics*, vol. 71, no. 1, pp. 375–385, 1994.

[51] H. Kimura, S. Akiyama, and K. Sakurama, "Realization of dynamic walking and running of the quadruped using neural oscillators," *Autonomous Robots*, vol. 7, no. 3, pp. 247–258, 1999.

[52] K. Matsuoka, "Sustained oscillations generated by mutually inhibiting neurons with adaptation," *Biological Cybernetics*, vol. 52, no. 6, pp. 367–376, 1985.

[53] G. Taga, Y. Yamaguchi, and H. Shimizu, "Selforganized control of bipedal locomotion by neural oscillators in unpredictable environment," *Biological Cybernetics*, vol. 65, pp. 147–159, 1991.

[54] W. Chen, G. Ren, J. Zhang, and J. Wang, "Smooth transition between different gaits of a hexapod robot via a central pattern generators algorithm," *J. Intell. Robot. Syst*, vol. 67, pp. 255–270, 2012.

[55] A. T. Winfree, *The geometry of biological time.* Springer, 1990.

[56] J. S. Bay and H. Hemami, "Modeling of a neural pattern generator with coupled nonlinear oscillators," *IEEE Trans Biomed Eng*, vol. 34, pp. 297–306, 1987.

[57] R. FitzHugh, "Impulses and physiological states in theoretical models of nerve membrane," *Biophys J*, vol. 1, pp. 445–466, 1961.

[58] J. Nagumo, S. Arimoto, and S. Yoshizawa, "An active pulse transmission line simulating nerve axon," pp. 2061–2070, 1962.

[59] J. Buchli, "Engineering limit cycle systems: Adaptive frequency oscillators and applications to adaptive locomotion control of compliant robots," Ph.D. dissertation, Inst. Bioeng., Swiss Fed. Inst. Technol., Lausanne, Switzerland, 2007.

[60] C. Zhou and K. H. Low, "Kinematic modeling framework for biomimetic undulatory fin motion based on coupled nonlinear oscillators," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Taipei, Taiwan, 2010, pp. 934–939.

[61] B. M. K. Anuar and T. Yasuno, "Moving control of quadruped hopping robot using adaptive cpg networks," in *Proc. IEEE Conf. Robot., Autom. Mechatron.*, Singapore, 2010, pp. 581–586.

[62] J. Yu, R. Ding, Q. Yang, M. Tan, W. Wang, and J. Zhang, "On a bio-inspired amphibious robot capable of multimodal motion," *IEEE Transactions on Mechatronics*, vol. 17, no. 5, pp. 847–856, 2012.

[63] D. Zhang, D. Hu, L. Shen, and H. Xie, "Design of an artificial bionic neural network to control fish-robot's locomotion," in *4th International Symposium on Neural Networks*, 2008, pp. 648–654.

[64] J. Buchli and A. J. Ijspeert, "Distributed central pattern generator model for robotics application based on phase sensitivity analysis," in *Proc. 1st Int. Workshop Biol. Inspired Approaches Adv. Inf. Technol*, Lausanne, Switzerland, 2004, pp. 333–349.

[65] C. Liu, Q. Chen, and G. Wang, "Adaptive walking control of quadruped robots based on central pattern generator (cpg) and reflex," *J Control Theory Appl*, vol. 11, no. 3, pp. 386–392, 2013.

[66] F. D. Libera, T. Minato, H. Hishiguro, and E. Menegatti, "Direct programming of a central pattern generator for periodic motions by touching," pp. 847–854, 2010.

[67] K. Kondo, T. Yasuno, and H. Harada, "Generation of jumping motion patterns for quadruped hopping robot using cpg network," *Journal of Signal Processing*, vol. 11, no. 11, pp. 321–324, 2007.

[68] A. Crespi, K. Karakasiliotis, A. Guignard, and A. J. Ijspeert, "Salamandra robotica ii: An amphibious robot to study salamander-like swimming and walking gaits," *IEEE Transactions on Robotics*, vol. 29, no. 2, pp. 308–320, 2013.

[69] A. H. Cohen, P. J. Holmes, and R. H. Rand, "The nature of coupling between segmental oscillators of the lamprey spinal generator for locomotion: a mathematical model," *J. Math. Biol.*, vol. 13, pp. 345–369, 1982.

[70] P. Olver. Nonlinear ordinary differential equation. [Online]. Available: http://www.math.umn.edu/~olver/am_/odz.pdf

[71] D. F. Rogers, *An Introduction to NURBS: With Historical Perspective*. Morgan Kaufmann, 2000.

[72] X. Wu and S. Ma, "Autonomous collision-free behavior of a snake-like robot," in *Proceedings of the 2010 IEEE International Conference on Robotics and Biomimetics*, Tianjin, China, 2010, pp. 1490–1495.

[73] T. Wang, W. Guo, M. Li, F. Zha, and L. Sun, "Cpg control for biped hopping robot in unpredictable environment," *Journal of Bionic Engineering*, vol. 9, no. 1, pp. 19–24, 2012.

[74] B. Li, Y. Li, and X. Rong, "Gait generation and transitions of quadruped robot based on wilson-cowan weakly neural networks," in *Proceedings of the 2010 IEEE International Conference on Robotics and Biomimetics*, Tianjin, China, 2010, pp. 19–24.

[75] Y. Sun and S. Ma, "epaddle mechanism: Towards the development of a versatile amphibious locomotion mechanism," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, San Francisco, CA, USA, 2011, pp. 5035–5040.

[76] P. Liljebck, K. Y. Pettersen, O. Stavdahl, and J. T. Gravdahl, "Hybrid modelling and control of obstacle-aided snake robot locomotion," *IEEE Transactions on Robotics*, vol. 26, no. 5, pp. 781–799, 2010.

[77] C. Ye, S. Ma, B. Li, and Y. Wang, "Turning and side motion of snake-like robot," in *Proc. IEEE Int. Conf. Robotics and Automation*, New Orleans, LA, 2004, pp. 5075–5080.

[78] S. Ma, "Analysis of creeping locomotion of a snake-like robot," *Advanced Robotics*, vol. 15, no. 2, pp. 205–224, 2001.

[79] S. Ma and N. Tadokoro, "Analysis of creeping locomotion of a snake-like robot on a slope," *Autonomous Robots*, vol. 20, no. 1, pp. 15–23, 2006.

[80] N. M. Nor and S. Ma, "Smooth transition for cpg-based body shape control of a snake-like robot," *Bioinspir. Biomim.*, vol. 9, 2014.

[81] N. M. Norz and S. Ma, "Cpg-based locomotion control of a snake-like robot for obstacle avoidance," in *Proc. IEEE Int. Conf. Robotics and Automation*, Hong Kong, 2014.

Appendix A

Control System

Following circuit board shown in Fig. A.1 is used to drive one servo motor. Each of the snake-like robot's joint is controlled by one servo motor, and communicate through Inter-Integrated Circuit (I^2C) bus protocol. To perform different experiments for various locomotion control of the snake-like robot, we can just modify the program code inside the micro controller without needing to modify the circuit design.

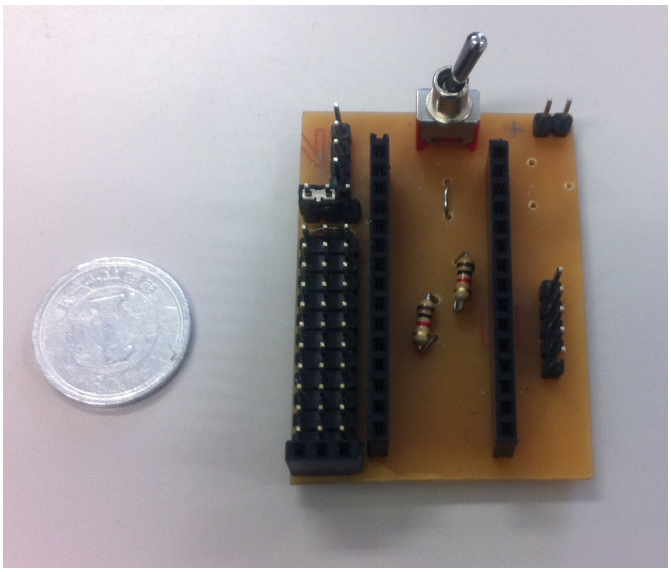


Figure A.1. Circuit board for controlling the servo motor.

The schematic drawing of the circuit board and the layout of the print-circuit-board (PCB) are shown in Fig. A.2 and Fig. A.3, respectively.

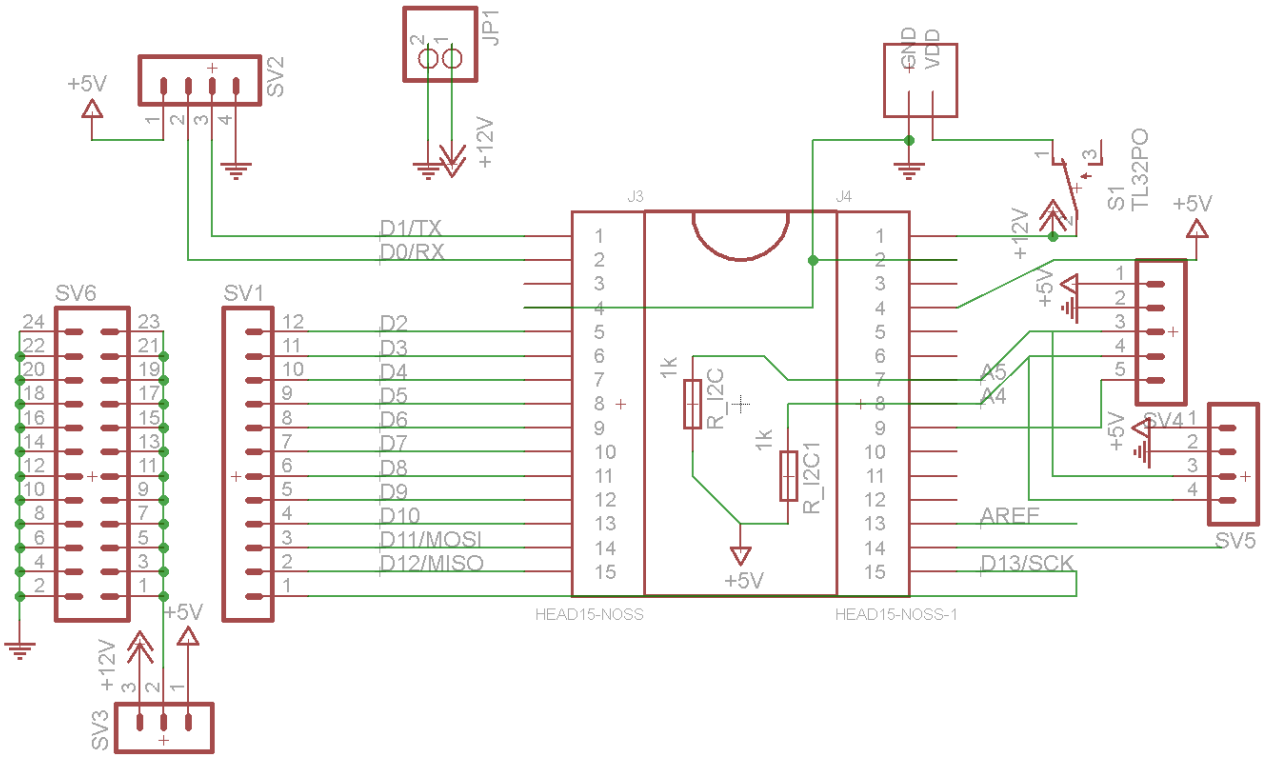


Figure A.2. Schematic of the circuit board.

The circuit board is designed based on the Arduino Nano controller as shown in Fig. A.4. We use Arduino Nano due to its lower cost, easy implementation and simple to write the program code. The software for the Arduino is an open source which has many libraries to be used for controlling mobile robots. For our experiments, we have used two libraries to control the servo motor and to communicate between joints i.e., *servo.h* and *wire.h*, respectively. This safe our time in programming a large code and building a simple circuit board.

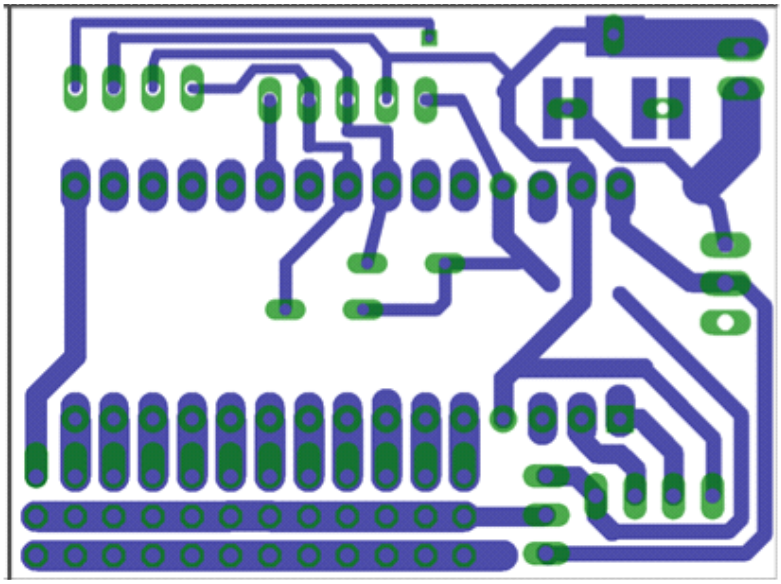


Figure A.3. Layout of the PCB.

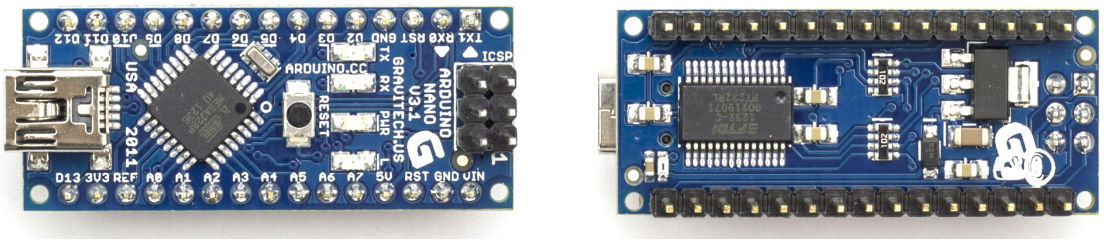


Figure A.4. Arduino Nano: front (left), and rear (right).

Appendix B

Communication Protocol

Fig. B.1 shows the architecture of the control system for realizing the locomotion of the snake-like robot. I^2C communication between the Arduino Nano is shown in Fig. B.1 (a), where SDA and SCL pins are used for data line and clock line respectively. The MCUs have to share the ground pin. Data transfer between CPGs is shown in Fig. B.1 (b). Each of the MCU is assigned the task to calculate and send the CPG output x_i to the servo, and transfer θ_i to the other oscillator through bus communication. For our CPG network, the first joint (master) and the second joint communicates bidirectionally, whereas the other joints (third joint and other) only receive input from the previous joint. The programming is done using the open-source Arduino environment version *Arduino 1.0.6-r2*.

Fig. B.2 shows the overview of the communication protocol between the controllers. The general code of the protocol is as following:

Master:

1	for $i = 1$	
2	request data from second controller	data = θ_2
3	convert received bytes from integer to float	
4	calculates θ_1	refer (2.19)
5	send data to second controller	data = θ_1
6	calculates x_1	refer (2.3)
7	write x_1 to servo	

Slave (second controller):

1	for $i = 2$	
2	request data from master	data = θ_1
3	convert received bytes from integer to float	
4	calculates θ_2	refer (2.19)
5	send data to master and third controller	data = θ_2
6	calculates x_2	refer (2.3)
7	write x_2 to servo	

Slave (other controllers):

1	for $j = i+1$	$i=2$ to n
2	request data from $j-1$ controller	data = θ_{j-1}
3	convert received bytes from integer to float	
4	calculates θ_j	refer (2.19)
5	send data to $j+1$	data = θ_j
6	calculates x_j	refer (2.3)
7	write x_j to servo	

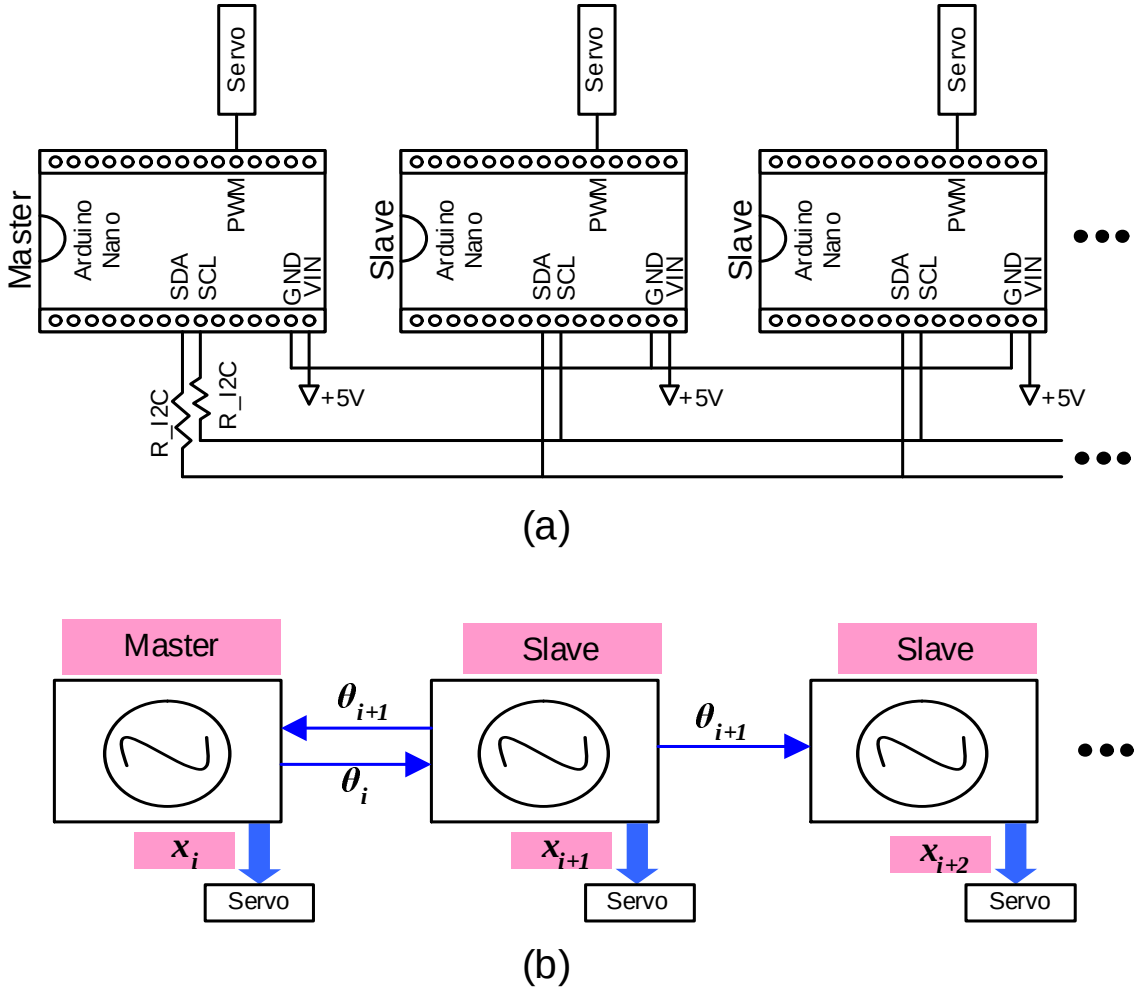


Figure B.1. Control system for the snake-like robot: (a) Communication design between the MCUs, and (b) the CPG network

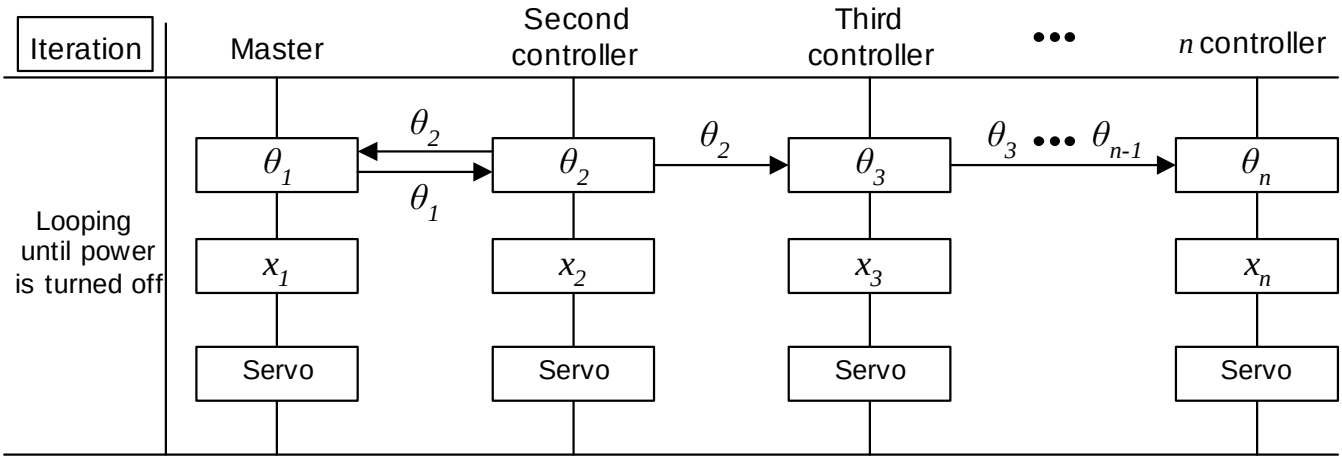


Figure B.2. Communication protocol between the controllers

Published Papers During Doctoral Course

Journal Papers:

1. **Norzalilah Mohamad Nor** and Shugen Ma, A simplified CPGs network with phase oscillator model for locomotion control of a snake-like robot, ***J Intell Robot Syst***, vol.75, no.1, pp. 71–86, 2014.

2. **Norzalilah Mohamad Nor** and Shugen Ma, Smooth transition for cpg-based body shape control of a snake-like robot, ***Bioinspir. Biomim.***, vol.9, 2014.

3. **Norzalilah Mohamad Nor** and Shugen Ma, CPG-based versatile locomotion control of a snake-like robot for obstacle avoidance, ***IEEE Transactions on Robotics***, 2014. [Submitted]

Conference Papers:

1. **Norzalilah Mohamad Nor** and Shugen Ma, A Simplified CPGs Network with Phase Oscillator Model for Locomotion Control of Snake-like Robot, In *Proc. of the 2012 IEEE Int. Conf. on Robotics and Biomimetics (**ROBIO'12**)*, Guangzhou, China, pp. 1299–1304, Dec, 2012.

2. **Norzalilah Mohamad Nor** and Shugen Ma, Body Shape Control of a Snake-like Robot Based on Phase Oscillator Network, In *Proc. of the 2013 IEEE Int. Conf. on Robotics and Biomimetics (**ROBIO'13**)*, Shenzhen, China, Dec, 2013.

3. **Norzalilah Mohamad Nor** and Shugen Ma, Cpg-based locomotion control of a snake-like robot for obstacle avoidance, In *Proc. of the 2014 IEEE Int. Conf. on Robotics and Automation (**ICRA'14**)*, Hong Kong, May, 2014.

4. **Norzalilah Mohamad Nor** and Shugen Ma, CPG-based Locomotion Control of a Snake-like Robot for Passing through a Variable Width of Path, In *Proc. of the 2014 JSME Robotics and Mechatronics Conference (**ROBOMEC'14**)*, Toyama, Japan, May, 2014.

5. **Norzalilah Mohamad Nor** and Shugen Ma, CPG-based Straight Path Control of a Snake-like Robot for Moving in Various Space Width, In *The 11th World Congress on Intelligent Control and Automation (**WCICA'14**)*, Shenyang, China, June, 2014. [Accepted]