EXPLOITING THE INHERENT COORDINATION OF CENTRAL PATTERN GENERATOR IN THE CONTROL OF HUMANOID ROBOT WALKING

HUANG WEIWEI

NATIONAL UNIVERSITY OF SINGAPORE

2010

Exploiting the Inherent Coordination of Central Pattern Generator in the Control of Humanoid Robot Walking

HUANG WEIWEI

(***B.Eng, USTC***)

A THESIS SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF MECHANICAL ENGINEERING

NATIONAL UNIVERSITY OF SINGAPORE

2010

Acknowledgments

First and foremost, I would like to express my most sincere gratitude to my supervisor, Assoc. Prof. Chew Chee-Meng, for his valuable supervision, incisive insight, enthusiastic encouragements and personal concerns in both academically and socially. These five years study has been fun and rewarding due to the freedom, support, and respect Chew has given to me.

I want to thank my supervisor Assoc. Prof. Hong Geok-Soon, who has given me constructive suggestions for this research.

I wish to specifically thank all the thesis reviewers and oral deference examiners. Your comments enlighten me to a deeper lever of understanding about my research work.

Thanks to all the thesis proofreaders: Albertus, James, Samuel, Huan and Chanaka who helped to point out many errors in the thesis.

My gratitude is also extended to all the members of the Mechatronics and Control Lab who have supported me and become friends over the years.

Finally, my deepest thanks go to my parents, my family, and specially to my wife Wenting for their great support during my study.

Table of Contents

Abstract

In this work, a bio-inspired central pattern generator (CPG) controller is developed to achieve an adaptive and robust walking control. CPG is an approach which tries to model the local control system of bipedal animals through a neural oscillator based network structure. This work includes designing a coordination connection between oscillators in the CPG; classifying the sensory feedback to the CPG; building a humanoid robot for the real implementation and controlling the robot with the proposed CPG controller.

Coordination among oscillators in the CPG is critical and important for the adaptive walking control. A CPG is usually composed of many coupled oscillators which output rhythm trajectories. These oscillators need to coordinate with other oscillators when there are external perturbations. By using the entrainment property of the neural oscillator, we develop a coordination connection between oscillators. With this connection, the main oscillator can adjust the phase of other oscillators for the coordination purpose. With this coordination connection, a CPG controller is developed to control the walking of a 2D bipedal robot. The simulation results show that the coordination connection enables the CPG controller to maintain the phase relationship among oscillators after the

push applied on the robot. This helps the robot to maintain the stability after the pushes are applied.

Another topic studied in the thesis is sensory feedback classification. The sensory feedbacks modulate the output of the oscillator and enable an adaptive behavior to the environment changes. Based on the way of modification to the oscillator output, we classify the sensory inputs into three types: inhibition input, triggering input and modification input. The purpose of this classification is to make the feedback design easier. With these three types of sensory inputs, the CPG controller can generate the reference trajectories for the 3D dynamic walking. In the simulation, the CPG controller is used to control a 3D stepping motion first. The sensory feedbacks modify the output of the oscillators to balance the robot motion when pushes are applied. After the stepping experiments, a stable 3D level ground walking is achieved by adding the forward motion trajectories.

To further test the controller, we implement it to control our physical humanoid robot NUSBIP-III ASLAN. ASLAN is a newly developed robot which serves as a platform to test different walking algorithms. It is a fully autonomous humanoid robot which has an approximate height of 120cm and an approximate weight of 60kg. It has 23 DoFs it total with two arms, two legs and one head. We have successfully implemented the CPG controller on ASLAN for stepping and walking motion. The robot shows a stable walking behavior with the CPG controller.

List of Tables

List of Figures

Acronyms

BLM	Blushless Motor
BM	Blush Motor
CMAC	Cerebellar model articulation controller
CoM	Center of mass
CoP	Center of pressure
CPG	Central pattern generator
DoF	Degree of freedom
GF	Ground reaction force
GH	Gear Head
HD	Harmonic Drive
IK	Inverse kinematics
LIPM	Linear inverted pendulum model
PB	Pulley Belt
PWM	Pulse Width Modulation
ZMP	Zero moment point
4R_K	Fourth order Runge-Kutta equation

Nomenclature

a	Strength of inhibition connection between neurons
$Ankle^{l(r)}_{r(p)}$	Angle value of left(right) ankle roll(pitch) joint
$Arm^{l(r)}_{y(r,p)}$	Angle value of left(right) arm yaw(roll, pitch) joint
β	Parameter that indicates the effect of adaptation
c	Constant stimulus to the oscillator
E_{in}	External input to the neuron
$Elbow^{l(r)}_{p}$	Angle value of left(right) elbow pitch joint
$Hip^{l(r)}_{y(r,p)}$	Angle value of left(right) hip yaw(roll, pitch) joint
$Hipx(y)\_v$	The actual hip velocity in x(y) direction in the simulation
H_{CoM}	Height of center of mass
$lift\_s$	a variable which trigger the activity of foot lifting
$Knee^{l(r)}_{p}$	Angle value of left(right) knee pitch joint
$Neck_{p(r)}$	Angle value of neck pitch(roll) joint

$Os\_hipx(y)$	Oscillator which gives reference trajectory for hip in x(y) direction
Os_hipx_v	Oscillator which gives reference velocity trajectory for hip in x direction
$Os\_hipx(y)v\_ad$	Oscillator which gives additional motion trajectory for hip in x(y) direction when a strong push is applied
$Os\_r(l)footx(y,z)$	Oscillator which gives reference trajectory for right(left) foot in x(y,z) direction
$Os\_r(l)footx\_v$	Oscillator which gives reference velocity trajectory for right(left) foot in x direction
$Os\_r(l)footx(y)v\_ad$	Oscillator which gives additional motion trajectory for right(left) foot in x(y) direction when a strong push is applied
$\tau_{1(2)}$	Time constant of neuron 1(2)
$u_{1(2)}$	State of the neuron 1(2)
$v_{1(2)}$	State of neural adaptation 1(2)
$Wrist_{r(p)}^{l(r)}$	Angle value of left(right) wrist roll(pitch) joint
$x(y)_{ZMP}$	ZMP location in x(y) direction

Chapter 1

Introduction

1.1 Background

A humanoid robot working with human in human environment is the dream of researchers in humanoid robot area. Human environment is especially designed for human to live and work, while humanoid robot is a robot that has similar physical characteristic with human. Therefore, this kind of robot has an advantage to operate in human environment and avoids the need to alter the environment for the robots. Humanoid robot and people could potentially collaborate with each another in the same working place using the same tools. In addition, compared with other kinds of robots, a humanoid robot has a smaller foot print and some mobility advantages in terms of walking up and down stairs. All these advantages make humanoid research an important area among all robotics research works.

Generally, the applications of humanoid robot can be divided into four areas. First, humanoid robot have been developed as a test-bed for a better understanding of human. This understanding can aid the development of better leg prostheses and help people who lose their leg to walk again as shown in Fig. 1.1 a. Second, humanoid robot can be a team worker with us. Human tools match human dexterity while humanoid robot can potentially take advantage of these same accommodations. Fig. 1.1 c1 and c2 show examples where humanoid robots use different types of tool to work in human environment. Humanoid robot has a similar mobility pattern as human which enable it to move in human environment freely. In Fig. 1.1 c3, ASIMO walks down from a staircase which is a common in the human environment. Third, humanoid robot can be used to provide assistance for children and elder care. Humanoid robot always has a human-like look (examples can be seen in Fig. 1.1 b1 and b2). Children or elder will feel more comfortable when facing a humanoid robot instead of other types of robot. Also, humanoid robots are inherently appropriate for the entertainment of humans. For example, many traditional forms of entertainment, such as playing music and dancing, require a similar structure of human as shown in Fig. 1.1 d1, d2, d3 and d4.

In humanoid research, walking is one of the most important and challenging area to study. Bipedal walking is the key advantage of humanoid robot compared with wheel robots. Currently, great effort has been put in this area to enable the humanoid robot to have the similar mobility as human. Until now, bipedal walking is still a challenging area to achieve human-like walking motion. The followings are some of the reasons. First, walking is a highly non-linear and discontinuous system. To model and analyze it

Figure 1.1: Applications of humanoid research; a:robot leg for disable people [2]; b(1,2):human friendly robot Actroid [1] DER2and HRP-4 [4]; c(1-3): working robots HRP-2 [4], HRP-3 [4] and ASIMO [3]; d(1-4): entertainment robots: Toyota robot [8], QRIO [7], Murata boy [5] and HRP-2 [4]

thoroughly is very challenging. Secondly, a humanoid robot has multiple joints, which increase the number of control variables. Thirdly, fast/dynamic walking is statically unstable which requires a good control algorithm to make it dynamically stable.

To achieve a robust walking behavior, various walking algorithms have been proposed. These algorithms can be divided into three main categories [16]: 1) model-based; 2) learning based; and 3) biologically inspired. Model based approaches develop control law through dynamic analysis. In the analysis, a mathematical model of the bipedal walking derived from physics is used for the control algorithm synthesis. The learning approaches are motivated from the observations of how children achieve walking. These observations show that walking is a learning process. The idea of biologically inspired approach is from the analysis of animal's walking behaviors. The finding shows that certain legged animals seem to centrally coordinate the muscles by a local control system in the spinal cord called central pattern generator (CPG) instead of the brain. Biologically inspired approaches model this CPG structure and use it to control the human walking. Besides these three main categories, passive walker is another interesting area which has been widely studied. It mainly focuses on the study of energy efficiency and limit cycle behavior of bipedal walking. A more detail review of walking algorithm will be presented in Chapter 2. Overall, the goal of these walking researches is to enable the humanoid robot to have a similar walking ability as human. Currently, achieving a robust and adaptive walking behavior like human is still a very challenging task.

1.2 Motivation

There are over 20 billion bipedal walking animals in the world today [34], and how their walk is still not fully understood yet. The mechanics of walking and muscle firing patterns are widely studied. This understanding can greatly help to develop a robust walking controller for humanoid robot.

Many walking algorithms have been developed for humanoid robot to achieve a human like walking. However, no current method achieves this target yet. Most of the methods are based on a dynamic analysis of a simplified walking model, while the dynamic model of human walking is very complex. It is very challenging to fully analyze the dynamic model and develop a control method. On the other side, this complex dynamic does not have any problem for human. Walking is an easy and basic locomotion for human. Without a detail dynamic calculation, human can control the walking easily. With additional training, human can perform a more difficult job such as stilt-walking. It is very interesting to study how human do the control during the walking. Therefore, this thesis focuses on a bio-inspired approach of walking control.

CPG is a bio-inspired approach which focuses on the modeling of local control system of bipedal animals through a neural network structure. It has been widely used in robotics control. In general, CPG is composed of many coupled oscillators which provide reference trajectories for the control. CPG research can be divided into many areas. Many of the research works focus on oscillator model design to get a closer model as bipedal animals. However, getting a good model is still very difficult due to a lack of

knowledge about the natural neuron activities. In this thesis, instead of finding a good oscillator model, we focus on the connection among oscillators.

Coordination among oscillators in CPG is an important area to analyze. A CPG model is normally composed of many oscillators to generator control trajectories. These rhythmic trajectories are the result of coordination among oscillators inside the CPG. The relationship among oscillators will greatly affect the output of CPG and robot motion. Therefore, coordination among oscillators is an important issue to be solved in CPG research.

Also, sensory feedback is another important issue to analyze. Sensory feedback provides robot the information of environment such as external perturbation. An ideal feedback will make the oscillator generate a correct output to balance the walking under perturbation. Therefore, designing a correct feedback pathway is also an important issue in CPG research.

1.3 Objectives and Scope

CPG based controller has been widely applied in robot locomotion control. Different approaches have been proposed to improve the controllers. However, there are still many challenges which require a further study:

- CPG controller is usually composed of many coupled oscillators. There is no complete method to achieve coordination connection among oscillators for an adaptive

motion. Much effort is required to manually tune the parameters of the oscillators in order to achieve the coordination connection among oscillators.

- CPG generates reference trajectories to control the robot. The coupling issue between CPG and the mechanical body is still unclear.

- Currently, most of the research works are still based on the simulation analysis. Few research works applies the CPG on controlling a physical 3D humanoid robot.

The main aim of this thesis is to propose a coordination structure among oscillators in a CPG which enables a behavior that can adapt to the environment. The coordination connection among oscillators enables the CPG controller to handle external perturbation. A detail description of the focuses is listed as following:

- A study of coordination connection among oscillators in CPG: the purpose of this study is to investigate the possibility of coordination connection among oscillators. A new type of connection among oscillators is investigated. This connection is designed for the problem of external perturbation to the CPG controller. When an external perturbation is added on the robot, the phase relationship among oscillators is always changed due to the effect of perturbation. With the coordination connection, it can guide the oscillators' output back to normal phase relationship. This enables the robot return to normal locomotion pattern after perturbation.

- Applying the CPG controller to control the walking of a bipedal robot: this part is to design a CPG controller for bipedal walking. A way of oscillators' arrangement

with coordination connection is developed. With this controller, the bipedal robot can perform basic locomotion task such as level ground walking.

- Sensory feedback coordination for the external perturbation: this part is to analyze the sensory feedback design under perturbation to improve the performance of CPG controller. Sensory feedback provides the external perturbation information for the robot. Based on the feedback, oscillators change their output to balance the robot. A correct sensory feedback design enables the oscillators to response correctly to the perturbation.

- Applying the CPG controller on a physical humanoid robot: with a successful implementation in the simulation, the goal of this thesis also includes the application of controller on a physical humanoid robot. For this purpose, a humanoid robot which serves as a platform to test different walking algorithms is developed. The proposed CPG controller will be verified through the physical implementation.

It is understood that locomotion includes many scenarios of walking behaviors. The scope of this research is restricted to bipedal walking on level ground along a linear path. Other behaviors of bipeds like turning, jumping, and running are not studied. In this thesis, the oscillator model used in CPG is the neural oscillator model proposed by Matsuoka [47]. The focus of this research is on the coordination among oscillators and sensory feedback and how it may contribute to the robust walking behavior. Other studies like oscillator model design, parameters optimization, and stability analysis of CPG will not be analyzed in this thesis.

1.4 Thesis Contribution

The results of this study contributes to a better understanding of how human walks. Although the study is mainly based on neural oscillator model proposed by Matsuoka, the concept of coordination structure could also be applied to other oscillator-model-based CPG approaches.

Specifically, the following works are carried out:

- Explore the importance of coordination in the CPG research (described in Chapter 3).
- Develop a way of coordination among oscillators in CPG and show the improvement of walking behavior with proposed method (described in Chapter 3).
- Propose a new method of oscillator arrangement in CPG design (described in Chapter 4).
- Give a brief classification of sensory feedback and analyze the effect of several feedback pathways to achieve a stable 3D walking behavior (described in Chapter 4).
- Describe the development of a fully autonomous humanoid robot (described in Appendix I).
- Successfully implement the CPG based walking algorithm on the humanoid robot (described in Chapter 5).

1.5 Simulation Tools

Before implementing the CPG controller on the real hardware, the proposed walking algorithms are verified in a dynamic simulation environment. A 2D dynamic walking simulation is first tested in a simulation software namely Yobotics. Then, a 3D dynamic walking simulation is tested in a simulation software namely Webots. Yobotics (seen in Fig. 1.2) and Webots (seen in Fig. 1.3) are both designed based on the open dynamics engine (ODE). In Yobotics environment, it is very easy to get the position, velocity and acceleration value of an object. Also, adding an external force with a desired value on the robot is very simple and straightforward in Yobotics. On the other hand, Webots software has a great interface with CAD software. This makes it possible to design a complex structure of the simulated robot and the walking environment. The 3D dynamic simulation is conducted in Webots environment.

1.6 Experiment Robot

A humanoid robot is an efficient platform to verify and improve a walking algorithm. A new humanoid robot prototype NUSBIP-III ASLAN depicted in Fig. 1.4 is developed in National University of Singapore since 2008. The robot has an approximate height of 120cm and an approximate weight of 60Kg. It consists of 23 actuated rotational joints, 2 cameras and onboard computing. Thirteen of the joints are the most relevant for walking: six in each leg in the standard configuration for 6 DoFs humanoid robot legs, and one in the waist for yaw motion of the waist. Motion of the 4 DoFs arms

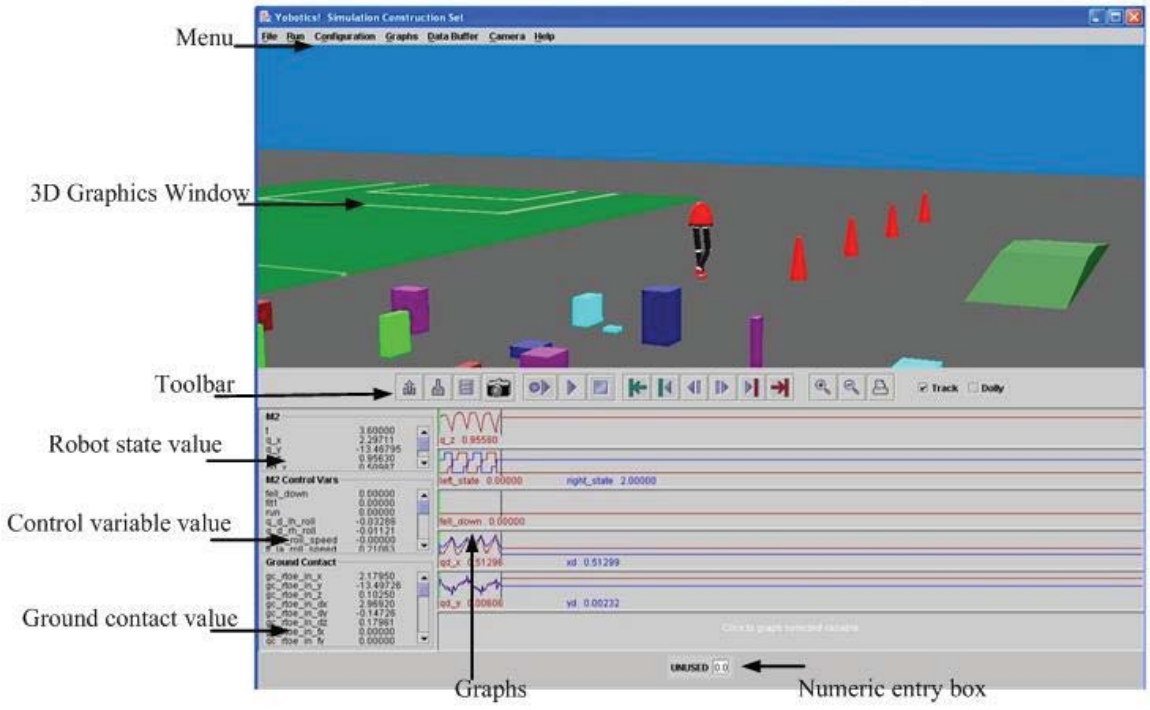


Figure 1.2: Yobotics simulation construction set GUI window

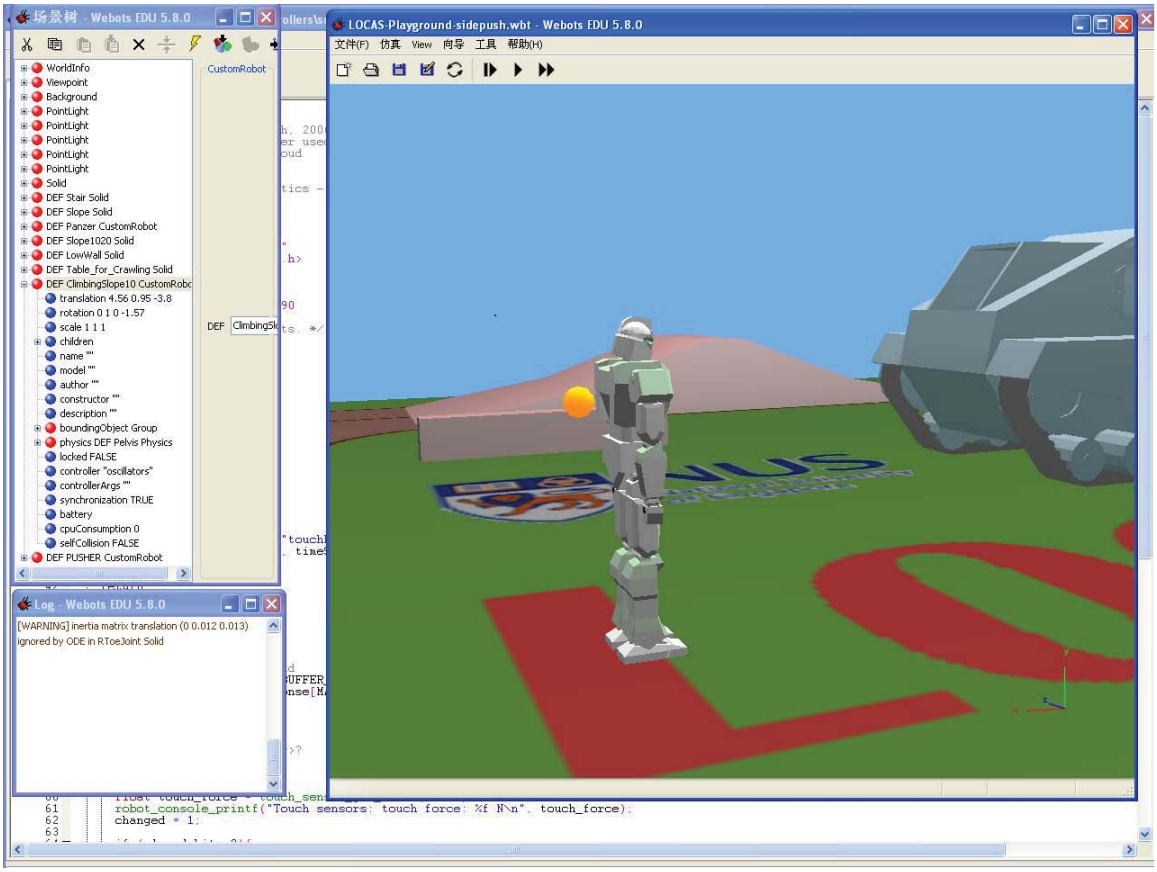


Figure 1.3: Webots simulation construction set GUI window

can be used for stabilization of the locomotion. The neck has 2 DoFs. A more detailed description of the ASLAN is given in Appendix I.

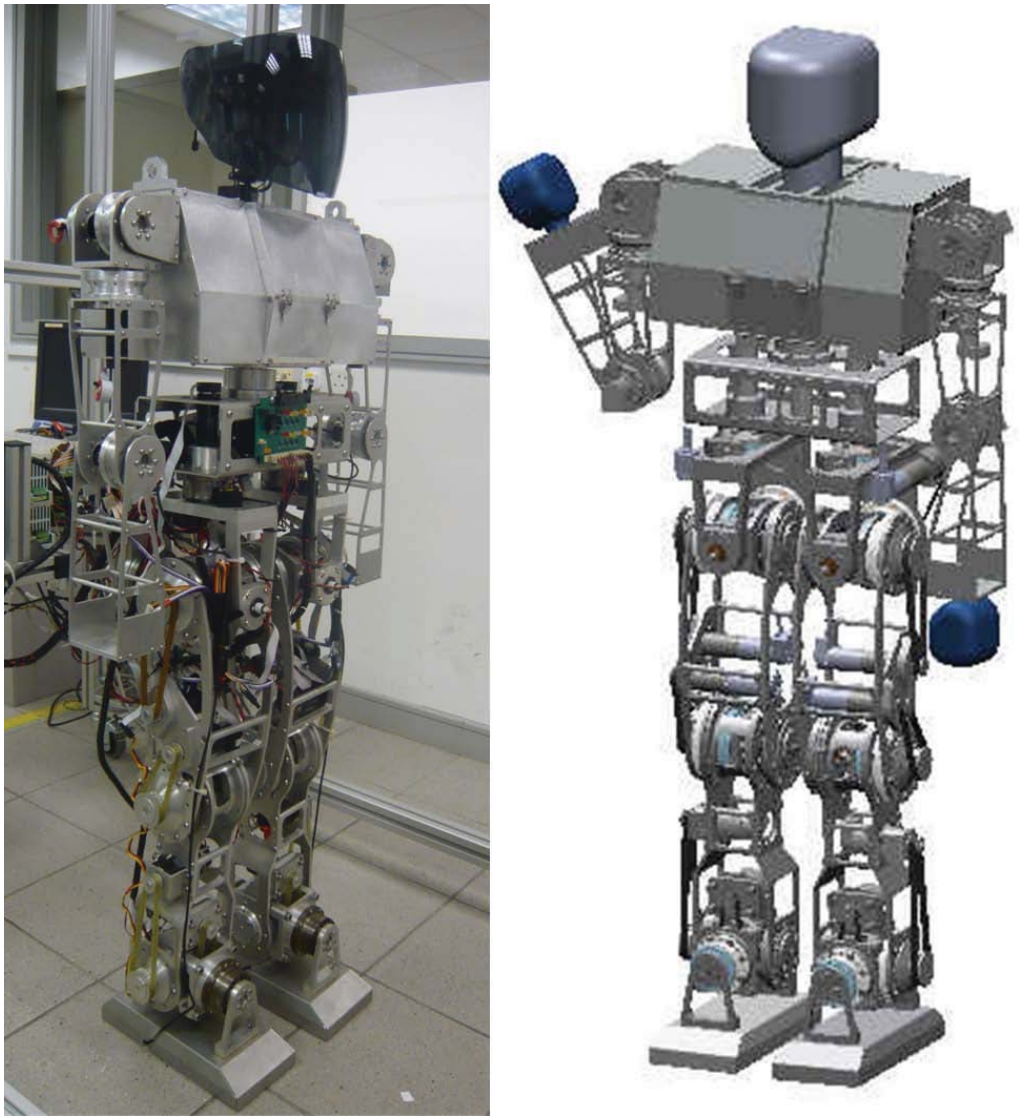


Figure 1.4: Humanoid robot ASLAN in SolidWorks design and actual robot

1.7 Organization of the Thesis

This thesis is organized as follows:

Chapter 2 gives a literature review of humanoid research and different approaches of walking algorithm. Then, a detail review of CPG based walking algorithm is presented. In CPG approach, the coordination issue is discussed.

Chapter 3 presents the proposed coordination method based on Matsuoka's neural oscillator. Properties of this oscillator are presented first. Then, a detail description of the coordination method is given. To verify the method, 2D walking simulation experiments are tested. The robot controlled by CPG shows a robust walking even with external pushes.

Chapter 4 addresses the issue of feedback design in CPG. Three types of sensory feedback pathways are discussed. The sensory feedback is designed to improve the robustness of CPG controller. Several 3D dynamic simulation experiments are tested with the proposed feedback designs.

Chapter 5 presents the result of implementing CPG controller on a physical robot ASLAN. The result of CPG based walking control is given. The robot shows a robust level ground walking.

Chapter 6 summarizes the contributions in this thesis and outlines directions for future research.

Chapter 2

Literature Review

2.1 Overview of the Powered Humanoid Robot

Throughout history, artists, engineers, and scientists dream of designing a human like machine. In early 1495, Leonardo da Vinci designed a humanoid automaton that looked like an armored knight, known as Leonardo's robot. The pioneering works on legged robots were done around 1970 by two famous researchers: Kato and Vukobratovic. Kato [33] demonstrated the first anthropomorphic robot, WABOT I, at Waseda University. In the same period, M.Vukobratovic [66] introduced the concept of zero-moment point (ZMP) for the analysis of locomotion stability which has since been widely used. The next breakthrough in legged robots was brought about by Raibert who launched MIT LegLab [54]. Raibert designed a sequence of active hopping robots, with one, two and four legs with impressive results. In the early 1990s, McGeer was the pioneer to study

passive biped systems. He introduced the poincare map to analyze orbital stability for passive dynamic systems [48]. Towards the end of the 1990s, industrial breakthroughs finally made the building of a real humanoid robot possible. ASIMO and HRP [30, 40] are major examples today realized by industrial companies.

ASIMO [45], a product of Honda, is not only can walk stably on level ground and stair, but also can run at speed of 6km/h. ASIMO has a total of 34 DoFs with 6 DoFs in each leg. It has servomotors with harmonic gear drive, 6-axis force/torque sensors, gyroscopes and accelerometers. The detailed walking algorithm of ASIMO has not yet been disclosed. The only known knowledge is that the walking algorithm of ASIMO is designed based on the ZMP criteria. Since it has a stable locomotion behavior, many high level locomotion behaviors such as obstacle avoidance, vision guided walking are tested on the robot.

HRP [40], designed by AIST, also has a stable walking ability on level ground. A typical property of the HRP series robot is that the hip joint of the robot has a cantilever type structure. The cantilever type structure reduces the chance of collision between the thigh-links. It also enables the robot to walk on a narrow path by cross-legged walk. The walking algorithm of HRP is a combination between linear inverted pendulum model and ZMP criteria. A predictive control is used to pre-plan walking trajectories.

Some research groups do not focus on a full body humanoid research. Instead, they focus on one particular part, such as walking or manipulating. PETMAN [6], a bipedal robot developed by Boston Dynamics, has demonstrated a natural human-like walking style. The heel to toe walking gait is similar to the human's. The robot can reach a

respectable speed of 5 km/h. It is also capable of balancing itself when hit from the side while walking. Unfortunately, a detail description of its walking algorithm has not been disclosed yet. JUSTIN [68], developed by the University of Napoli and the German Space Agency DLR, can carry out complex manipulation tasks such as picking up a small soft straw for drinking or unscrewing a can.

2.2 Overview of the Walking Algorithm

With technology advancement, designing a sophisticated humanoid robot becomes possible. In this case, a robust walking algorithm which enables a robot to walk in a human environment is necessary. Walking is easy for a human, but it is the toughest problem for humanoid robot. In general, walking is a cyclic motion. Studying the cyclic walking properties can help in designing a better walking algorithm. Based on the literature search, the walking algorithm can generally be divided into three types: model-based, learning based and biologically inspired.

To have a better understand of walking, the cyclic property of bipedal walking is widely studied. Normally, bipedal walking is a periodical motion where the steady-state behavior is characterized by a cycle in the phase plane. Raibert [53] proved that the control of such a system could be split into three separate components: the first component controls the height by providing a push during each cycle; the second part controls the forward velocity of the whole system by assigning a forward step; and the last one controls the body attitude by controlling the hip angle during the stance phase. The concept

of cyclic walking is widely applied in passive walker. McGeer [48] introduced a simple passive walking system: a planar compass walking on an inclined plane. Stable walking results from the balance between the energy gained from gravity and the energy loss due to impact. The stability of this system can be analyzed in terms of orbital stability. For a general nonlinear system, the proof of the existence of a limit cycle, the analysis of its local orbital stability, and the procedure to compute the cycle and its basin of attraction are often difficult. Nevertheless, it is possible to test the local stability of a limit cycle. One method to determine the stability of the robot gait is through the numerical computation of its Poincare map [48]. It shows that if a periodic system can return to its starting states after a motion cycle, then the system is stable.

Model-based approach is a method which uses a simplified model of bipedal walking for the control algorithm synthesis. A well-known and widely used model is the linear inverted pendulum model (LIPM) proposed by Kajita [39]. In this model, the robot is approximated as a ”hip” point mass which is maintained at a constant height. Based on LIPM, Kajita proposed another model namely cart-table model, which use a preview control of the ZMP location [37]. In this model, a cart with mass at its center runs on a pedestal table whose mass is negligible. This model is effective when solving the stepping-stones problem [37]. In both of the above models, the angular momentum of the body mass is not considered. One approach which considers the angular momentum is by J. Pratt [52]. He proposed a flywheel model adding rotational inertia which enables the humanoid to control its centroidal angular momentum. Another model namely Acrobot model [26] takes the leg inertia into consideration. It is a double pendulum model

with no actuation between the ground and base link.

The learning-based approach is often used to develop biological behaviors. The evidence comes from observing the process of walking learning by a child. Learning is often applied to systems which are hard to model. In the learning based approach, one or more key parameters are acquired through a learning process. Miller et. al [49] adopted a simple gait oscillator which generated the trajectory of the swing leg for a simulated planar bipedal robot. The input to the oscillator was the step time and desired step length based on a fixed model. A cerebellar model articulation controller (CMAC) network was used to achieve a desired step length based on past experience. Chew [17] proved that a robot can maintain a stable walking motion with a proper swing time and step length. Reinforcement learning, in particular, Q-learning with CMAC as function approximator, is used to learn these key parameters. The robot achieves an adaptive walking on level ground and slope terrain.

The biologically inspired approach stems from the medical experiment of human and animal walking. Walking is a fundamental task of human and bipedal animals. Neurophysiological studies have revealed that animal walking is generated by the Central Pattern Generator (CPG) [27, 57, 28], found in the spinal cord, which produces rhythmic motor patterns to activate their limbs. It is known that during locomotion the CPG can generate rhythmic excitation signals to the muscles even without input from sensory feedback or from brain signals. Most evidence of the existence of CPG in vertebrates comes from lamprey [27] and cats [60, 9]. The concept of CPG is widely used in humanoid robot research to achieve a human-like walking. CPG is usually modeled by

many coupled nonlinear oscillators. An interesting property of CPG is that the basic pattern produced by the intrinsic oscillation can interact with sensory feedback. The intrinsic oscillation of CPG synchronizes the oscillation of sensory feedback signals, which is known as ”entrainment” phenomenon [21]. CPG modulates rhythmic patterns in response to sensory information, resulting in adaptive behaviors.

While reviewing the history and the present state of art, humanoid research has made a great progress in the last 20 years. A stable level ground walking has been achieved by many robots. At this moment, achieving adaptive walking in an unknown environment or with external perturbation is the new challenge. A robust way to achieve this behavior remains to be demonstrated.

2.3 CPG Based Approach

Achieving a robust walking behavior is a challenging problem in humanoid robot research. Many approaches have been proposed to solve this problem. Among these approaches, CPG based approach seems very interesting. Inspired by animal and human locomotion, CPG based approach has been proposed to meet the challenge of walking. In this section, several key areas of CPG based locomotion control are reviewed. The areas include the oscillator model design, the application of CPG in locomotion control, and coordination in CPG research.

2.3.1 Model Design

In CPG research, the oscillator model design is the main research interest. The oscillator is the basic component of CPG. It determines the performance of the CPG controller. In CPG based approach, oscillator models are mathematical models of coupled nonlinear oscillators [18, 46, 55]. Currently, many types of oscillators are extensively being used in robot control. In these applications, three types of models are mainly used. They are bio-inspired based, oscillating equation based and learning based models.

The bio-inspired model is designed based on neural activities. A widely used model is the neural oscillator model proposed by Matsuoka which is inspired form the behavior of neuron activity [47]. This oscillator model has an important property, namely the entrainment property. When a periodic input is given to the oscillator, the oscillator's output can entrains to this input. This property can be used to design the structure of adaptive walking controller. Based on this oscillator model, many successful research works have been done in the control of robot locomotion [41, 47, 65]. However, the challenge with the Matsuoka oscillator is that the mathematical equation of this model is very complex. It is very difficult to fully analyze the behavior of the model.

Oscillating equation based model is usually based on some mathematics equations which have oscillating property. The phase oscillator model is derived from a phase equation. The output of the phase oscillator is affected by two key parameters: amplitude and phase. An interesting property of this oscillator is that the desired phase relationship among oscillators can be specified in a straightforward manner. Applications of coupled

phase oscillators have been explored in the gait control of multi-legged robots [36, 43]. A disadvantage of this phase oscillators is that it has a fixed waveform for a given frequency. This property restricts the application of this phase oscillator. Van der Pol model is also derived from the nonlinear mathematical equations that have oscillatory output. This model is simpler than the previous two models. Many studies have used it to analyze the simplified model of the robots [13, 73]. A problem of this model is that the physical meaning of the oscillator's parameters is not clear. Designing an effective feedback pathway and tuning the parameters for a desired trajectory will be challenging.

Learning-based model usually uses a lot of oscillators to generate a desired trajectory. Righetti et. al [55] designed a programmable central pattern generator based on many Hopf oscillators with different frequencies. In this design, each oscillator is tuned to learn a frequency component of the input signal. By a combination of enough oscillators, the input signal can be produced by these oscillators. An interesting property of this design is that the learning is embedded in the network and does not require external optimization tools. Jiang et. al [58, 71] propose a CPG controller which is constructed by a combination of many neural circuits that are modeled by recurrent neural networks. The CPG circuit is composed of different groups of sub circuit. These recurrent neural networks also learn the desired trajectories and the output of the control signal. This CPG structure has been successfully implemented to control a HOAP robot walking up and down stairs. After training, the learning-based model can output the desired trajectory for the motion. A problem of this model is that, the learning process is usually done off-line without sensory feedback from real environment. The effeteness of

the learned trajectories in real environment is unclear. Also, a new learning process is always needed for a new walking pattern or new environment.

2.3.2 Applications in Robotics

Based on the coupled oscillators, CPG models are increasingly used in the robotics community. Models of CPG have been used to control a variety of different types of robots and different modes of locomotion. For instance, CPG models have been used with hexapod and octopod robots inspired by insect locomotion [42, 12]. Related work in simulation was undertaken by Randall Beer and his colleagues in the early 1990s [14]. Cruse et al. [20] study showed that reflexive controllers could also lead to robust insect locomotion.

CPG has also been used for controlling swimming robots. Anguilliform swimming is one kind of swimming. Its forward motion is obtained by a traveling undulation of the elongated body from head to tail. Inspired by this motion, CPG models using coupled oscillators are used to control a lamprey swimming robot [11, 19].

Quadruped walking control using CPG has been extensively explored by Hiroshi Kimura and his colleagues [41, 23]. They explored different means of integrating sensory feedback in locomotion control and found that sensory feedback, which modulates CPG activity, tends to lead to the most stable locomotion in complex terrain (as opposed to feedback which is independent of the CPG activity).

CPG based controllers are also increasingly used for the bipedal locomotion control.

The motivation is straightforward which is to study human waking and implement the knowledge in the control of bipedal walking. Examples of CPG-based bipedal locomotion control include Aoi and Tsuchiya [10], Endo et al. [22], Heliot and Espiau [29], Morimoto et al. [51], and Righetti and Ijspeert [56]. In these applications, the sensory feedback plays an important role for CPG to adapt to the real world environment.

2.3.3 Coordination

We have reviewed different types of oscillator models in CPG and their application in robotics control, but we have not yet discussed how CPG is built by these oscillators. Currently, there is no established method to design the CPG structure. Many approaches have been proposed such as hand-coding, dynamical theory based and learning based.

When designing a CPG controller, Ijspeert proposed five items to be considered [34]:

- "The general architecture of the CPG". It includes the type and number of oscillators. When controlling a robot, it also includes the control strategy: position control or torque control, joint space or task space.
- "The type and topology of couplings". The trajectories generated by the CPG result from the combined output of many oscillators. The coupling between oscillators directly affect the adaptation of the CPG controller.
- "The waveforms". These waveforms determine the trajectories to control the robot.

- "The effect of input signals". This will specify how the control input can modulate the oscillator's output such as the frequency, amplitude, phase delay and overall waveform.

- "The effect of feedback signals". This specifies how feedback from the robot will affect the activity of the CPG.

When designing a CPG controller, these five items are strongly coupled together which make the problem even more complicated. For example, when considering the effect of input and feedback to the CPG, the coupling effect of the oscillators needs to be considered.

Coupling among oscillators is an important issue in the CPG design. The motivation comes from the fact that the overall performance of the oscillatory system mainly depends on the type and topology of couplings rather than on the local mechanisms of rhythm generator. This is well established in dynamical systems theory [25]. In the CPG based approach, the main advantages are the ability to couple to a mechanical body and the ability to produce different gaits. This enables the robot to adapt to the environment. To have this adaptation property, a coordinated connection among oscillators in CPG is needed.

In CPG research, different oscillator model may have different ways of connection. For the bio-inspired model and oscillating equation based model, inhibition is the commonly used method of connection. The main oscillator gives the inhibited input to the other oscillators [64]. One limitation of this connection is that the scaling factors of

the connection are very difficult to tune. Parameter optimization tools are used to get the required connection parameters. In the learning based model, several oscillators are combined to generate a desired trajectory. The connection parameters among these oscillators are trained to generated the desired trajectory. After training, the oscillator system can generate the trajectory. However, it is not clear how each oscillator works with each other. As such, designing the feedback pathway to adjust the oscillator's output is very difficult and a retraining of the connection parameters has to be done for the robot to walk in a new environment. With all these complications, there is a real need to develop a simple way to connect and coordinate among oscillators in CPG.

2.4 Summary

In this chapter, we first give a general review of the humanoid robot. Humanoid research has a wide application in our society. Over the years, many advanced humanoid robots have been developed such as ASIMO, HRP and HUBO. Although, they are still not able to interact effectively with people, there is much progress in humanoid development over the last 20 years. Bipedal walking is one of the most challenging areas in humanoid research. With a better control algorithm, the humanoid robot can achieve a more natural locomotion behavior.

Different walking algorithms have been reviewed. A well-established walking algorithm for the humanoid robot has to be developed. Each algorithm has its advantages and disadvantages. Ideally, the model-based approach is a good solution if the model

is not too complex to compute and not too simple to be unable to capture nonlinear behaviors. One disadvantage is that additional modifications and tuning may be required when the theoretical model is applied to the walking control. The learning approach is very interesting since no complex dynamic analysis is needed. Without knowing the dynamic model, the learning algorithm can adjust the control parameters to achieve robust walking. However, the learning process requires a long time to learn and is also dependent on the environment. A new learning process will always be required for each new environment.

The CPG approach seems promising. It tries to model a similar locomotion controller as bipedal animals. The model is designed to have an adaptation property. Currently, the CPG research is still in its developing stage. Most research works focus on the oscillator model design. There are still many open research topics which need to be addressed, such as how the CPG can be coupled to a mechanical body, how the CPG can produce multiple gaits and how the inner oscillators can be connected to generate an adaptive trajectory.

In this thesis, we focus on the coordination among oscillators in the CPG. The oscillator model adopted here is the neural oscillator model proposed by Matsuoka. This model was inspired from the behavior of biological neurons. It can output rhythmic signals without external input and has an entrainment property. Our target is to design a connection among neural oscillators to achieve an adaptive walking behavior. In the next chapter, we will present our coordination method among neural oscillators. This connection is simple in design and can coordinate multiple oscillators. A CPG controller

with this coordination connection is developed to control the walking of a bipedal robot.

Chapter 3

Coordination between Oscillators

3.1 Introduction

Human beings are able to produce a large variety of motions that involve the coordination of upper and/or lower limbs. The motions of the limbs are coordinated with each other based on the desired task such as walking, running and swimming. These motions are characterized by synchronized movements of limb pairs. Neuronal research indicates that the spinal cord of neonatal rat can produce rhythmic motor output [61][15] closely resembling the activation of hindlimb muscles during locomotion. Although the prevailing viewpoint attributes the coordination to the human brain, more recent evidence indicates a distributed network that governs the processes of neural synchronization and desynchronization. When the spinal cord is isolated from the higher neural systems, animal still shows coordination behavior among limbs[63]. It shows that the

CPG, which is defined as a neural structure in spinal cord, governs the locomotor coordination among limbs.

It is clear that there exists a coordination mechanism in our neural system for the control of limb motion. In this mechanism, relative phase among oscillators is considered as a key variable to characterize the coordination mode [63]. Inspired by this idea, the coordination connection in this thesis is considered as phase adjustment between oscillators. Walking is a cyclical motion. A given walking pattern corresponds to a pattern of limbs' movement with certain phase relationship. This phase relationship can be disturbed by external perturbations. When the phase of the oscillator cannot be adjusted, the desired phase relationship among oscillators cannot be recovered which may cause the walking motion to fail. Therefore, the phase adjustment among oscillators is needed.

In CPG research, a common way of connection between oscillators is through inhibition connection [65, 64]. The method is inspired by the biological neuron connection. A disadvantage of this method is that achieving phase adjustment is very challenging by this connection. To achieve phase adjustment, the weights of connection between oscillators should be carefully designed. Many intelligent tools such as generic algorithm [44] and reinforcement learning method [24] are often used to tune the weights. Therefore, an advanced connection between neural oscillators, which can achieve coordination purpose, is needed.

On the other side, there are many coordination examples in our real world. A typical example is the performance of orchestra. An orchestra usually contains several instrument groups such as string, brass, woodwind, and percussion instruments. A generally ac-

cepted hierarchy structure exists among these instrument groups and within each group of instruments. A principal is generally responsible for leading his/her instrumental group. All the principal of the instrumental groups coordinate with the conductor or principal violin if there is no conductor. In this case, the conductor provides the reference pace for the principal players. The principal players then guide their groups to keep in pace. With this hierarchical coordination structure, the orchestra offers audiences a harmonious music.

In this chapter, we propose a new coordination connection in CPG which is inspired from the orchestra hierarchical structure. In this CPG design, a "conductor" oscillator provides the reference pace to some "principal" oscillators and the "principal" oscillators guide their own group of oscillators to keep in pace with the whole oscillators in CPG structure. The schematic of the CPG structure is shown in Fig 3.1. Here, the phase of the oscillator is considered as the pace in the orchestra performance. Relative phase is considered as a key issue in the coordination connection. A general analysis of coordination between oscillators by Klavins et al. [43] shows that the phase relationship of oscillators determines the sequence of each sub-motion controlled by the oscillators and therefore the behavior of the robot. Therefore, achieving phase adjustment between oscillators is a key issue to achieve coordination connection in CPG controller.

In the rest of this chapter, we will address the problem of phase adjustment between oscillators. First, neural oscillator, the basic component of CPG controller, will be introduced. Secondly, the problem of phase adjustment between oscillators will be discussed. Then, we will propose our new phase adjustment method for the coordination

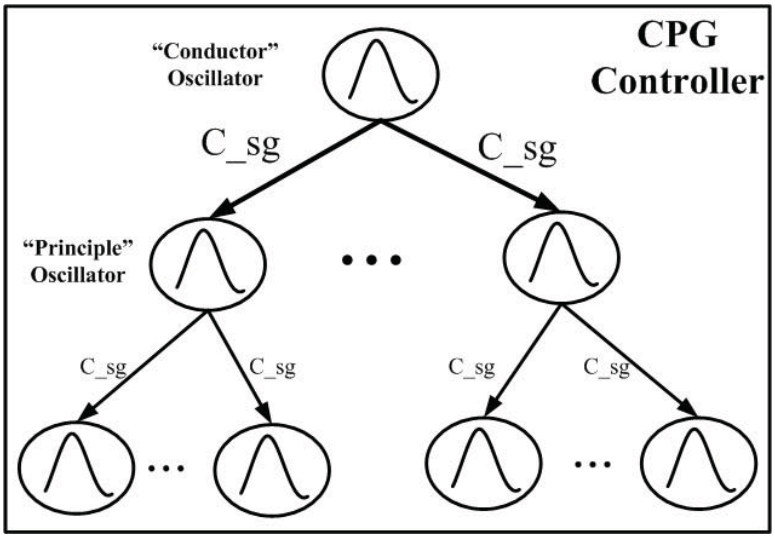


Figure 3.1: A hierarchical CPG structure with coordination connection between oscillators.

connection. After that, several experiments of 2D dynamic simulation controlled by CPG controller with coordination connection are given. A summary on the coordination method and simulation results is provided in the last part of the chapter.

3.2 Neural Oscillator Description

3.2.1 Neural Oscillator Model

The oscillator is a fundamental component of a CPG structure. Matsuoka [47] has proposed a neural oscillator model which is inspired from the behavior of biological neuron and commonly used in the CPG research. The oscillator model consists of two simulated neurons arranged in mutual inhibition as shown in Fig. 3.2. It can produce rhythmic signal without external input and possesses the limit cycle property [28]. The mathematical

model is described by the following equations [47]:

$$\tau_1 \dot{u}_1 = c - u_1 - \beta v_1 - a[u_2]^+ - K[E_{in}]^+ \quad (3.1)$$

$$\tau_2 \dot{v}_1 = [u_1]^+ - v_1 \quad (3.2)$$

$$\tau_1 \dot{u}_2 = c - u_2 - \beta v_2 - a[u_1]^+ + K[E_{in}]^- \quad (3.3)$$

$$\tau_2 \dot{v}_2 = [u_2]^+ - v_2 \quad (3.4)$$

$$[u_1]^+ = \max(0, u_1) \quad [u_1]^- = \min(0, u_1) \quad (3.5)$$

$$Y = [u_1]^+ - [u_2]^+ \quad (3.6)$$

Here $u_{1(2)}$ is the state of the neuron; $v_{1(2)}$ is the degree of neural adaptation; c is the constant stimulus; τ_1 and τ_2 are the time constants; β is the parameter that indicates the effect of adaptation; a represents the strength of inhibition connection between neurons; E_{in} is the external input to oscillator; K is a factor to adjust the input; Y is the oscillator output. The tonic excitation c determines the amplitude of the oscillator, which is proportional to c. The two time constants τ_1 and τ_2 determine the frequency of the oscillator and the parameter β affects the shape of the oscillator output.

Limit cycle behavior is an important property of oscillator in the CPG based control. To prove the limit cycle property, we have conducted a mathematical analysis on the oscillator to derive the conditions for the limit cycle behavior. According to the Poincaré-Bendixson theorem, the neural oscillator model described by equations (3.1)-(3.6) has a

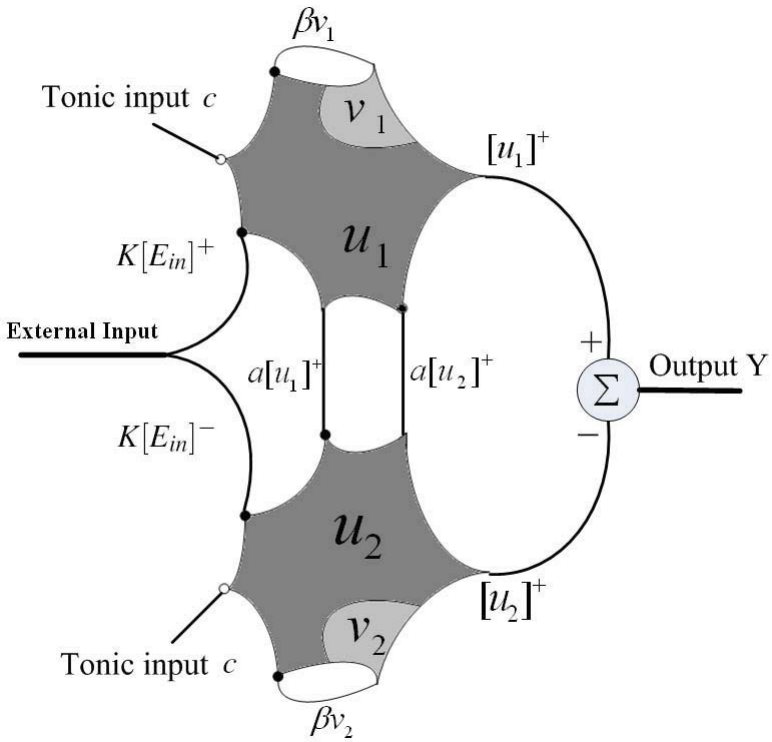


Figure 3.2: Scheme of Matsuoka's neural oscillator model; white and black cycles represent excitation and inhibition, respectively

unique limit cycle behavior if and only if

$$a - 1 - \frac{\tau_1}{\tau_2} > 0 \tag{3.7}$$

$$a - 1 - \beta < 0 \tag{3.8}$$

A detail procedure of the proof can be found in Appendix II. When designing the CPG controller, these two conditions must be satisfied to have a limit cycle property.

When there is no external input to the oscillator, the frequency and amplitude of the oscillator are only dependent on oscillator parameters. For the neural oscillator described

by equations (3.1)-(3.6), the frequency is determined by [72]:

$$F = \frac{1}{2\pi\tau_1}\sqrt{b(1+(\frac{\beta}{a}-1)(1+b))} \tag{3.9}$$

where $b = \frac{\tau_1}{\tau_2}$ is usually taken to be a constant. The approximate amplitude is given by the following equation. A detail proof the amplitude calculation is given in Appendix III.

$$A = \frac{2c}{1+\beta+a} \tag{3.10}$$

Equations (3.9) and (3.10) give a relationship between oscillator parameters and oscillator output. The same relationship has been numerically explored by Williamson [67]. In this thesis, these two equations will be used to obtain the values of the oscillator's parameter for a desired trajectory.

3.2.2 Entrainment Property

Entrainment property is commonly observed in neural oscillator behavior. Williamson revealed that when the oscillator receives a periodical external input, it entrains the input by falling into the frequency of input [67]. When the input signal is a large constant, then the oscillator output will be suppressed and becomes a constant. Fig. 3.3 (top) shows an example where the oscillator output locks to a sinusoidal input, and Fig. 3.3 (bottom)

illustrates an example that the oscillator output is suppressed by a large constant input.

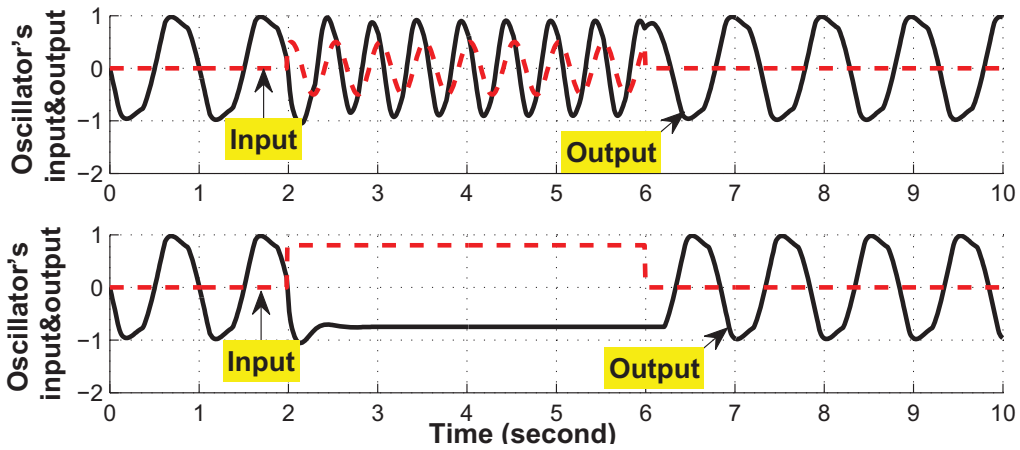


Figure 3.3: Properties of neural oscillator; top: entrainment property, bottom: suppression of oscillator by a large constant input

3.3 Coordination between Neural Oscillators

3.3.1 Phase Adjustment

In the previous section, we have described the neural oscillator model, the basic component of the CPG controller. A CPG controller is usually composed of many neural oscillators which are connected together. Among these oscillators, the inner parameters which dictate the property of the oscillator may not be the same. Due to this difference, oscillators have different response to an external perturbation. When the external perturbation is given to this system, it can result in different shift of phase values among oscillators. The error of phase relationship among oscillators will cumulate. This cumulated error can cause the task to fail in the later control. To achieve a robust and adaptive behavior, a phase adjustment connection is necessary for the coordination between these

oscillators.

Unlike the phase oscillator [51], in which the phase value is clearly indicated by a phase variable, the phase state of neural oscillator is not clear. As for the neural oscillator model described by Equation (3.1)-(3.6), there is no variable directly related with the phase state. Since the output of the neural oscillator is similar to a sinusoidal wave, we adopt the phase calculation method of sinusoidal wave to calculate the phase of the neural oscillator. The relationship between the oscillator output and oscillator phase is described in the following function:

$$\phi = \begin{cases} mod(\textbf{Real}(\arcsin(\frac{Y}{OS_{Amp}})), 2\pi) & s = 1 \\ \pi - \textbf{Real}(\arcsin(\frac{Y}{OS_{Amp}})) & s = -1 \end{cases} \tag{3.11}$$

where Y is the output of the oscillator; OS_{Amp} is the amplitude of the oscillator without any external input; ϕ is the phase of the oscillator. To convert the phase range to $[0, 2\pi]$, we define a parameter s, which indicates the direction of change of the oscillator: $s = 1$ indicates that the oscillator value is increasing, while $s = -1$ indicates that the oscillator value is decreasing. The oscillator's output can be larger than its amplitude due to the effect of external inputs. In this case, we only take the real part value of arcsin function as the phase value.

Generally, phase adjustment can be achieved by modifying the phase variable in the oscillator. However, there is no phase variable in the neural oscillator model. To build a relationship between oscillator phase and oscillator states, a possible solution is to design a lookup table mapping from the neuron states to the phase value. This lookup

table can be built according to the relationship among oscillator output, oscillator state and oscillator phase. Fig. 3.4 shows an example of lookup table from the neuron states to the phase value. In this example, a single cycle of oscillating without any external input is captured from the oscillator's output. In this cycle, a set of state variable u_1, u_2, v_1, v_2 generates an output and an output corresponds to a particular phase value that lies in $[0, 2\pi]$ range. The accuracy of lookup table can be increased by increasing the sampling rate of output. With the lookup table, the phase of the oscillator can be modified by changing the neuron states of the oscillator accordingly. Also,with the phase value, the neuron states and oscillator's output can be derived from this table.

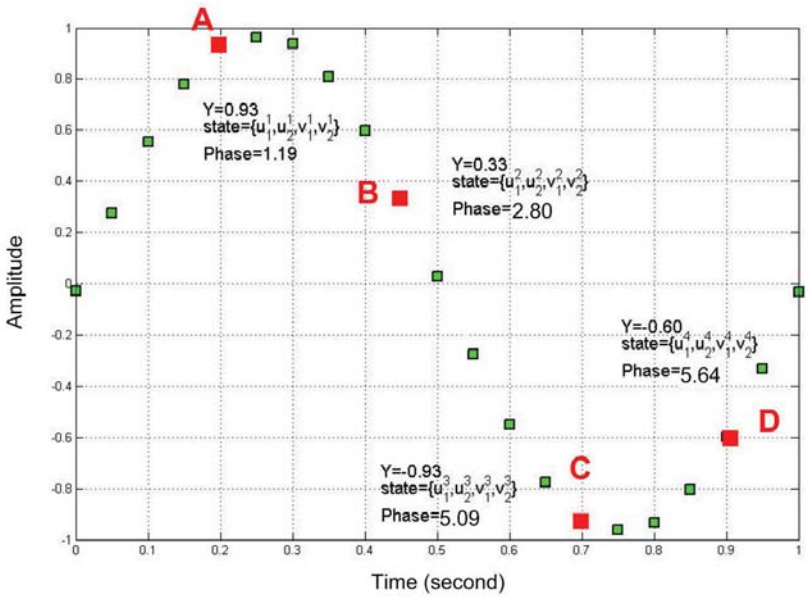


Figure 3.4: A lookup table that mapping the phase of the oscillator to the neuron states and oscillator's output in a single oscillating cycle without any external input. A, B, C, D are four example of lookup table where a phase value is related with an unique neuron states value.

Fig. 3.5 gives two example of phase adjustment by modifying the neuron states in neural oscillator. At time 0.25s, the desired phase value is 0 as shown in the red dot B. In the first demo, the modification of neuron state brings a sudden change in the output which

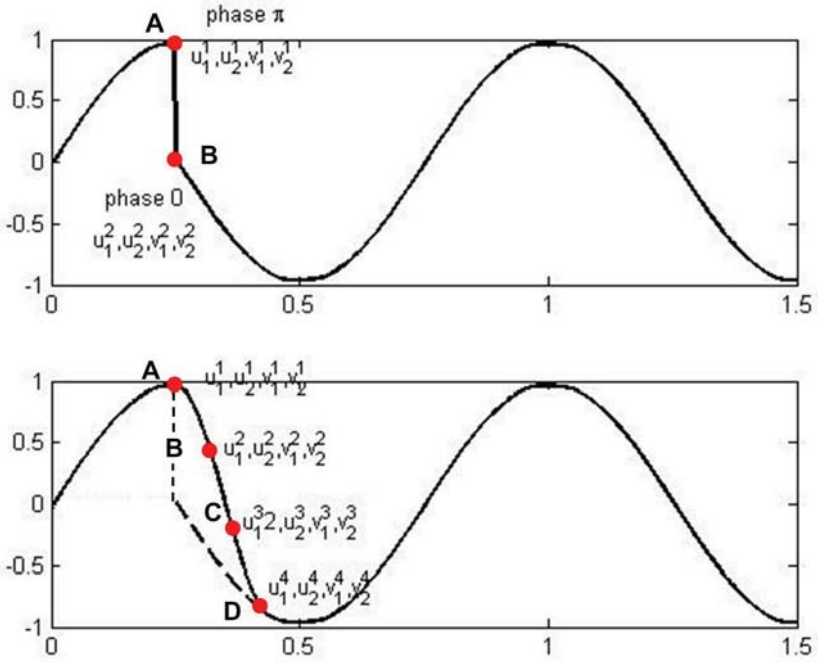


Figure 3.5: Example of phase modification by changing the neuron state's value; upper figure: direct change the neuron state to the desired phase value; bottom figure: a smooth change of neuron state to the desired phase value.

is not recommended. To avoid a sudden change of the output, a smooth modification of the output is used as shown in the second demo. In the transition period, the neuron states are set according to a pre-planned trajectory. During this period, the oscillator cannot response to the external inputs. In this case, if there is any environment change, there will be no response by the oscillator.

3.3.2 Closed Loop Phase Adjustment

Because there is no direct link between neuron states and phase value, phase adjustment on neural oscillator is not as intuitive as on phase oscillator. One solution is to construct a relationship between neuron states and phase value by building a lookup table. However,

it will restrict the performance of neural oscillator as discussed above.

In this study, we propose a new method of phase adjustment between neural oscillators. The idea is inspired by the closed loop feedback control. The closed loop structure is shown in Fig. 3.6. Here g_1 and g_2 are selection functions.

$$g_1 = K\max(e,0), \quad g_2 = K\min(e,0) \tag{3.12}$$

where e is the error between oscillator output and reference input; K is the external input factor described in equation (3.1)-(3.6). The reference input generated by the desired phase value will be input to the closed loop oscillator structure and adjust the phase of the oscillator. The basic idea is that, a strong output (bigger than the reference input) will generate an error signal to inhibit the positive neuron (top neuron in Fig. 3.6) to reduce the oscillator's output; a weak output (smaller than the reference input) will generate an error signal to inhibit the negative neuron (bottom neuron in Fig. 3.6) to increase the oscillator's output. Based on the analysis, the close loop structure enables the oscillator to follow the sensory input. Since the neural oscillator follows the reference input, the phase of the oscillator is adjusted by the reference input. Fig. 3.7 shows an example that oscillator follows the external input. For demonstration, the reference input only adjust the oscillator after 2 s. Here, we set the parameter K non zero value after 2 s; otherwise K equals 0. In the test, the oscillator can follow the reference input smoothly. Therefore, the phase of the oscillator is adjusted by the reference input.

To further explore this result, we have conducted a mathematical study which analyzes

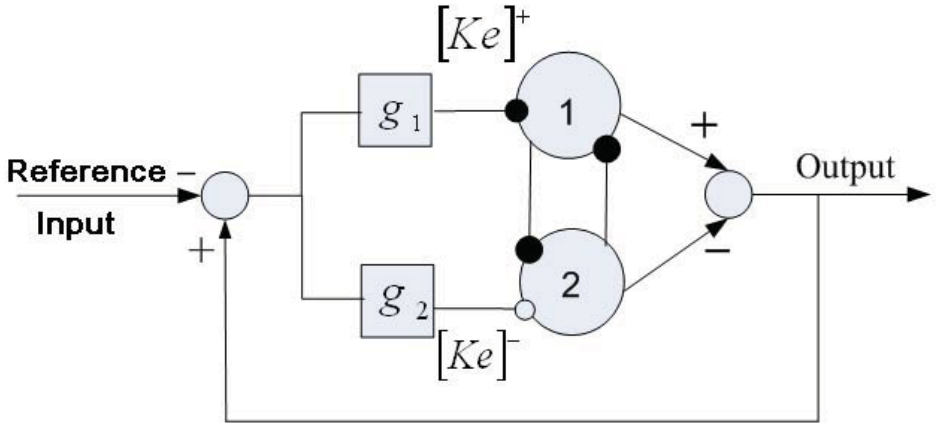


Figure 3.6: Closed loop structure for reference input. Black dot indicates an inhibition connection; hollow dot indicates an excitation connection.

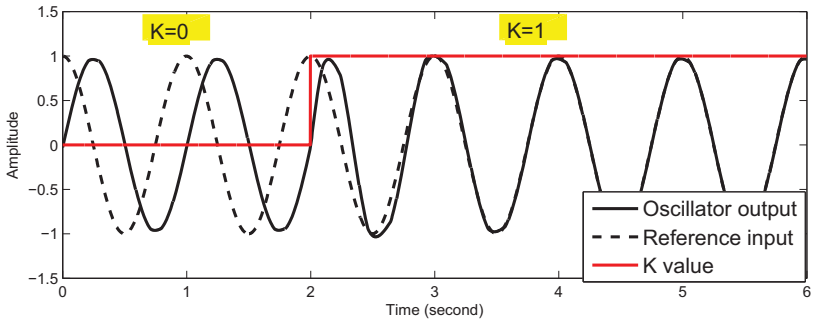


Figure 3.7: Example of oscillator following a reference input where K equals to 0 before 2s and become nonzero after 2 s.

the response of external signal based on this closed loop structure. Based on closed loop model described by Fig. 3.6, we assume a signal R_{in} are input to oscillator through reference input. Here, we assume there is no other external input. The mathematical equation based on Equations (3.1)-(3.4) with this closed loop model is:

$$\tau_1 \dot{u}_1 = c - u_1 - \beta v_1 - a[u_2]^+ - K[Y - R_{in}]^+ \quad (3.13)$$

$$\tau_2 \dot{v}_1 = [u_1]^+ - v_1 \quad (3.14)$$

$$\tau_1 \dot{u}_2 = c - u_2 - \beta v_2 - a[u_1]^+ + K[Y - R_{in}]^- \quad (3.15)$$

$$\tau_2 \dot{v}_2 = [u_2]^+ - v_2 \quad (3.16)$$

To analyze the response of the oscillator, we conduct a piecewise linear analysis on these equations. In this analysis, we separate the state variable u_1, u_2 region into four subset quadrants $\{u_1 \geq 0, u_2 \geq 0\}$, $\{u_1 \geq 0, u_2 < 0\}$, $\{u_1 < 0, u_2 \geq 0\}$, and $\{u_1 < 0, u_2 < 0\}$. The oscillator's response to the reference input is analyzed in each quadrant.

In the first quadrant $\{u_1 \geq 0, u_2 \geq 0\}$, the output of the oscillator is $Y = u_1 - u_2$. We set variable $V = v_1 - v_2$. Combining Equations (3.13)-(3.16), we obtain

$$\tau_1 \dot{Y} = (a - 1 - K)Y - \beta V + KR_{in} \quad (3.17)$$

$$\tau_2 \dot{V} = Y - V \quad (3.18)$$

Substituting V from Equation (3.18)into Equation (3.17), we get

$$\tau_1\tau_2\ddot{Y} + (\tau_1 - \alpha\tau_2)\dot{Y} + (\beta - \alpha)Y = H \tag{3.19}$$

where $\alpha = a - 1 - K$; $H = R_{in} + \tau_2\dot{R_{in}}$. When K is larger than $a - 1 - \frac{\tau_1}{\tau_2}$, the term $\tau_1 - \alpha\tau_2$ is positive. The system has a positive damper which can converge to a stable point. At the stable point, $\dot{Y} = 0$, $\dot{V} = 0$.

$$(a-1-K)Y - \beta V + KR_{in} = 0 \tag{3.20}$$

$$Y - V = 0 \tag{3.21}$$

Therefore, the system will converge to the value $\frac{K}{K+1+\beta-a}R_{in}$. If $K \gg \beta + 1 - a$, the stable output converges to the input R_{in}.

In the second quadrant $\{u_1 > 0, u_2 < 0\}$, the output of the oscillator is $Y = u_1$. The system has a positive damper. In this case, the state values at the stable point can be calculated based on $\dot{u}_1 = 0, \dot{u}_2 = 0, \dot{v}_1 = 0, \dot{v}_2 = 0$. Based on equation (3.13)-(3.16), we

obtain

$$c - u_1 - \beta v_1 - K[u_1 - R_{in}]^+ = 0 \quad (3.22)$$

$$u_1 - v_1 = 0 \quad (3.23)$$

$$c - u_2 - \beta v_2 - a u_1 + K[u_1 - R_{in}]^- = 0 \quad (3.24)$$

$$v_2 = 0 \quad (3.25)$$

when $u_1 - R_{in} >= 0$

$$u_1 = \frac{c + K R_{in}}{1 + \beta + K} \quad (3.26)$$

$$u_2 = \frac{(1 + \beta - a)c + K(c - R_{in})}{1 + \beta + K} \quad (3.27)$$

when $u_1 - R_{in} < 0$

$$u_1 = \frac{c}{1 + \beta} \quad (3.28)$$

$$u_2 = \frac{(1 + \beta - a)c + K(c - (1 + \beta)R_{in})}{1 + \beta} \quad (3.29)$$

If the reference input value is small than the oscillator amplitude which is $R_{in} < Amp \approx \frac{2c}{1+\beta+a}$, and the state value u_2 is always positive. In this case, the stable point will not be

in this quadrant. Therefore, in this quadrant, there is no stable point.

A similar proof can also be given to the quadrant $\{u_1 < 0, u_2 > 0\}$ which shows no stable point in this quadrant.

Based on the analysis, the oscillator will follow the external input when the following conditions are satisfied:

$$K \gg \beta + 1 - a \tag{3.30}$$

$$R_{in} < Amp \approx \frac{2c}{1+\beta+a} \tag{3.31}$$

This conditions are verified by the numerical simulation of constant input. For a periodic input, a small value of K will also enable a following behavior. It only takes a longer time to converge to the input signal. A detail study is still needed for a better understanding of this phenomenon.

3.3.3 Coordination between Oscillators

To achieve a coordination connection, phase adjustment is the key issue. In the previous subsection, we introduce our closed loop structure for the phase adjustment. With this design, the phase of the oscillator can be modified by the reference input. Based on

this result, we develop our method of coordination connection between oscillators. The block diagram is shown in Fig. 3.8.

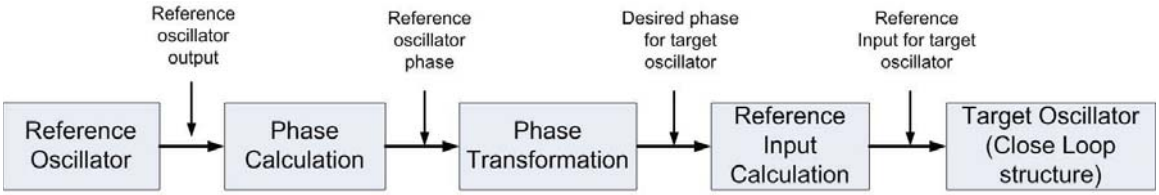


Figure 3.8: A block diagram of coordination process between oscillators.

- **Reference Oscillator**: this oscillator performs like a conductor or principal of instrumental group who generates the reference signal to coordinate other oscillators.

- **Phase Calculation**: this process is to get the phase value of reference oscillator from its output. The formula to get the reference phase value is by Equation (3.11).

- **Phase Transformation**: this process converts the reference oscillator's phase to the target oscillator's phase based on their phase relationship. A detail discussion of phase relationship will be given in the later part.

- **Reference Input Calculation**: this process converts the phase value to the reference input value for the target oscillator based on the lookup table described in Fig. 3.4.

- **Target Oscillator**: this oscillator performs like a member of instrumental group who adjust its phase according to the phase value of reference oscillator. The oscillator has the closed loop structure as shown in Fig. 3.6.

The idea of this block diagram is that the reference input is generated by a reference oscillator instead of other external signals. In order to get the reference phase for the target oscillator, both phase of reference oscillator and the phase relationship between these two oscillators are needed. The reference phase can be calculated by Equation (3.11). Therefore, the key issue is to derive the phase relationship between these two oscillators. Here, the phase relationship is defined as a mathematic equation that can describe the relationship of phase value between these two oscillators as time is increasing.

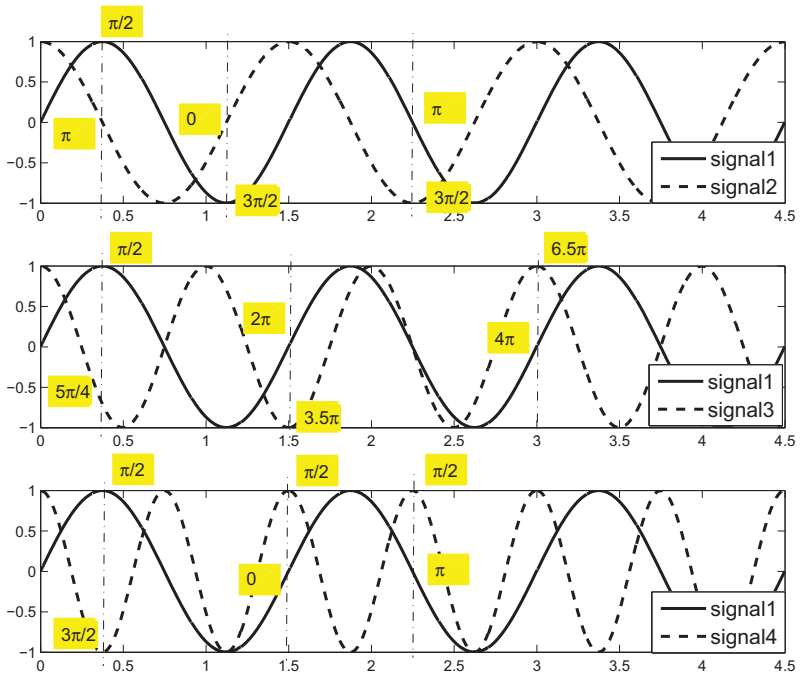


Figure 3.9: Examples of phase relationship between periodic signals. upper figure: sinusoidal wave with same frequency but different initial phase; middle figure: the frequency of signal 3 is 1.5 times higher than signal 1; bottom figure: the frequency of signal 4 is twice of signal 1.

To explore the mathematic phase relationship between two periodic signals, the sinusoidal wave is used as an example for the analysis. Fig. 3.9 shows three examples of

phase relationship between two sinusoidal waves. The phase values of two signals at a particular time are showed in the figure. Upper figure shows two signals with same frequency but different initial phase. The initial phase of signal 1 is 0 while the one of signal 2 is $\frac{\pi}{2}$. The phase difference between two signals is $\frac{\pi}{2}$. The mathematical equation which describes the relationship between the phases of two oscillators is:

$$\phi_2 = mod(\phi_1 + \frac{\pi}{2}, 2\pi) \tag{3.32}$$

Middle figure in Fig. 3.9 shows another example in which two signals with difference frequency and initial phase. In this case, the phase difference between these two signals cannot be interpreted as in the case of same frequency. The period of signal 1 is 1.5 s while the period of signal 3 is 1 s. The initial phase difference between these two signals is pi/2. The phase relationship cannot convert to the $[0, 2\pi]$ range. Because of the non-integral multiple of frequency between these two signals, a phase cycle $[0, 2\pi]$ from signal 3 may map to two phase cycle form signal 1. The relationship between signal 1 and 3 can only be derived according to their cumulated phase value. From the observation, the mathematic equation which describes the relationship between the phases of these oscillators is:

$$\phi_3 = \frac{f_3}{f_1}\phi_1 + \frac{\pi}{2} \tag{3.33}$$

where ϕ_3 and ϕ_1 are the cumulated values as shown in middle figure of Fig. 3.9.

However, in the CPG based control, an oscillator usually has the same frequency as the other ones or twice as high than the other ones. The case that two oscillators in CPG controller have a non-integral multiple frequency relationship is few. In the case that a signal's frequency is integral multiple of the other one's frequency, their phase relationship can be expressed in a $[0, 2\pi]$ range as shown in the bottom figure of Fig. 3.9. The mathematic equation is given as following:

$$\phi_4 = mod(\frac{f_4}{f_1}\phi_1 + \frac{\pi}{2}, 2\pi) \tag{3.34}$$

Inspired by these examples, we develop the mathematical equation to describe the phase relationship between two period signals with different frequency and initial phase. Here, we assume that the frequency of signal 2 is integral multiple of the frequency of signal 1. The equation is given as:

$$\Phi_2 = mod(\frac{F_2}{F_1}\Phi_1 + (\Phi_2^{ini} - \Phi_1^{ini}), 2\pi) \tag{3.35}$$

where Φ_1 and Φ_2 are phase values of periodic signal 1 and 2 respectively; F_1 and F_2 are frequencies of periodic signal 1 and 2 respectively; Φ_1^{ini} and Φ_2^{ini} are initial phase value of periodic signal 1 and 2 respectively.

In the case that requires non-integral multiple frequency relationship between oscillators, Equation (3.34) can be used to calculate the reference phase for the target oscillator.

In this case, the phase value of reference oscillator needs to be cumulated based on the number of periodic cycle counted since the beginning.

Here we give an example of coordination between two oscillators which have different frequencies and initial phase values. The frequency of the target oscillator is twice of the main oscillator. The final output trajectory is the sum of two outputs of the oscillator. When there is an external input given through g_j, the patten of the two oscillator's output changes as shown in Fig. 3.10. After 1 second, the output patten of the two oscillators returns back to the normal patten. However, the final output pattern changes. The reason is that the phase relationship between two oscillators is changed, as indicated by the dotted cycle. The change of phase relationship is because two oscillators response differently to the external input. When CPG is applied to control a walking motion, an external perturbation will result a change of oscillators' output. After the effect of external perturbation, the desired trajectory needs to be recovered. To achieve this, it needs a coordination connection between oscillators.

By virtue of the coordination between oscillators, the desired oscillator output could be recovered. Fig. 3.11 shows the performance of the two oscillators with coordination. When there are external inputs, the oscillators can adjust the phase and recover to the original target output.

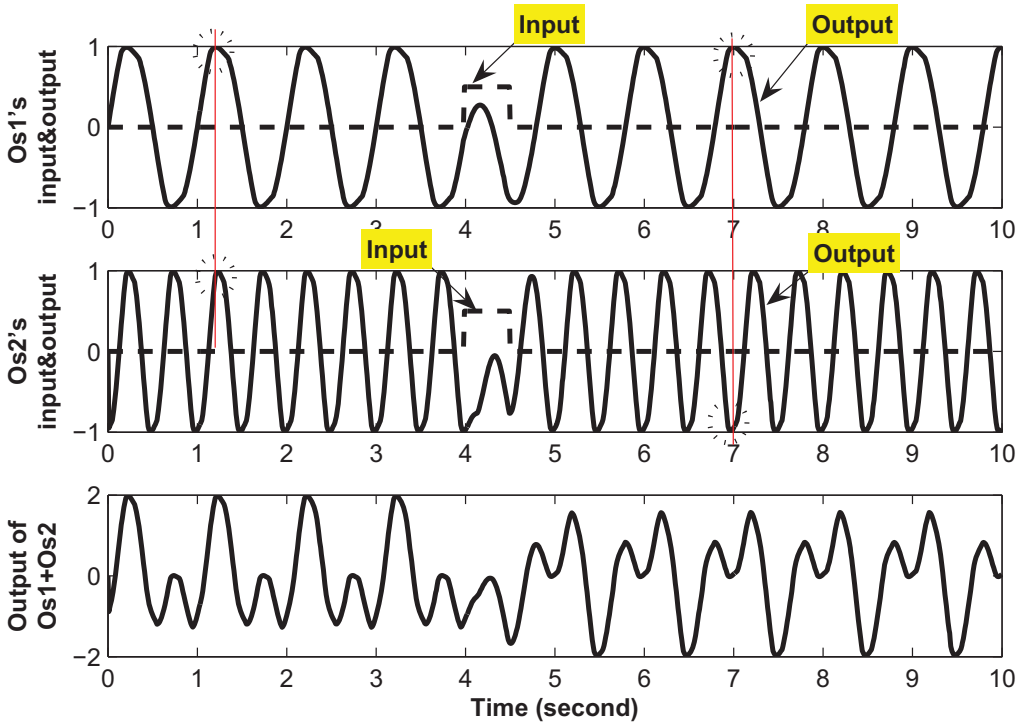


Figure 3.10: Example of two oscillators output without coordination after an external disturbance in the form of nonzero input

3.4 Implementation in 2D Walking Control

In the previous section, we have analyzed the phase adjustment connection between oscillators in CPG controller for a coordination behavior. In this section, we will implement this coordination connection in a 2D bipedal walking control to analyze the effect of this connection.

3.4.1 Control Architecture

A 2D model is developed in a simulation environment, namely Yobotics. It has ankle, knee and hip pitch joints in each leg. The dimension and mass distribution are shown in

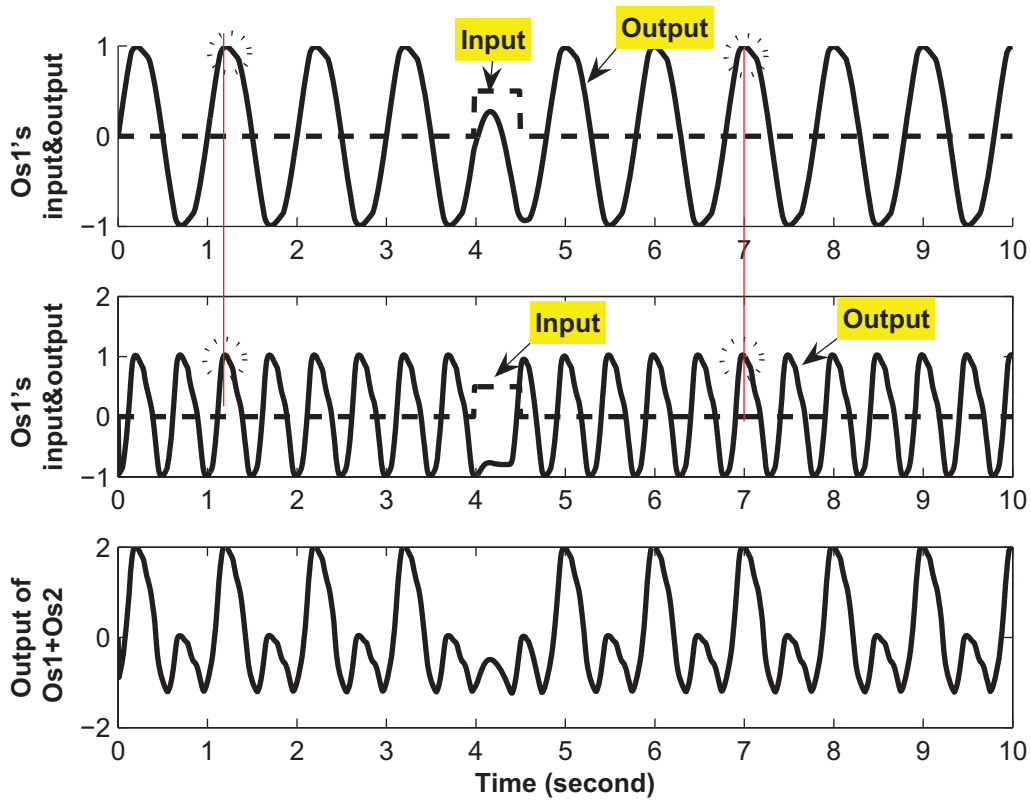


Figure 3.11: Example of two oscillators output with coordination after an external disturbance in the form of nonzero input

Table 3.1: The specifications of simulation model

link	mass(kg)	Ixx(kgm^2)	Iyy(kgm^2)	Izz(kgm^2)	length(m)
Body	25.26	0.6954	0.0842	0.5189	0.30
Thigh	4.25	0.0240	0.0196	0.0096	0.256
Shank	5.10	0.0269	0.0227	0.0100	0.256
Foot	2.52	0.0042	0.0037	0.0035	0.10

Table 3.1.

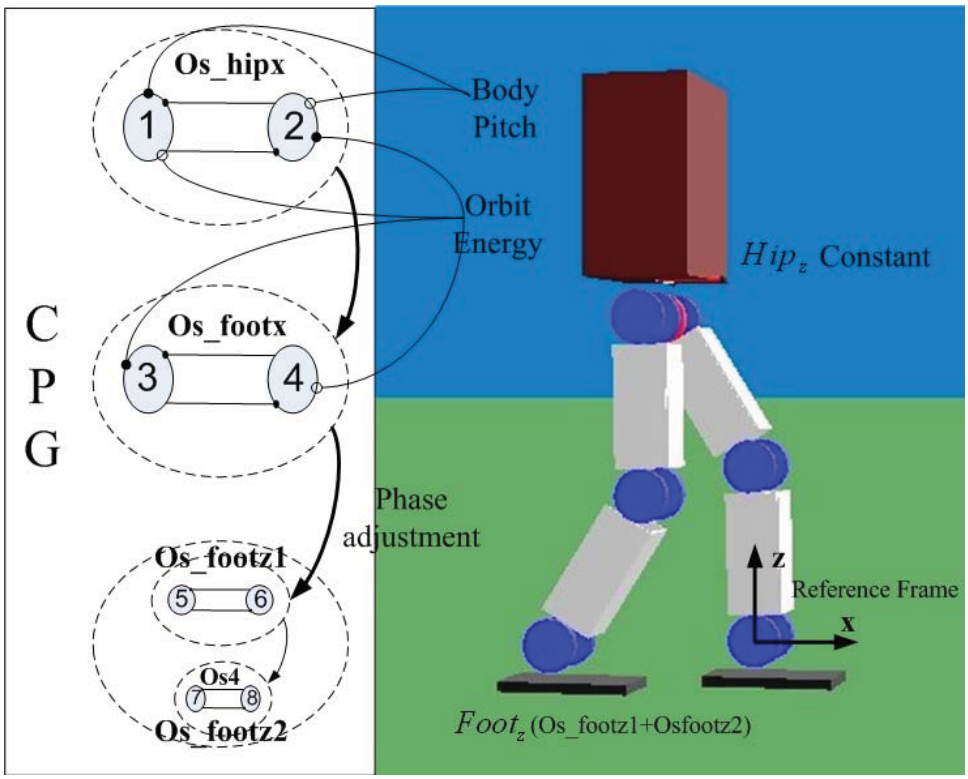


Figure 3.12: Oscillator arrangement and sensory feedback of the CPG structure on the robot

We propose a CPG structure with respect to the position of the leg in the Cartesian coordinate space (see Fig. 3.12). Os_hipx generates the reference hip x direction trajectory(Hip_x). Os_footx and (Os_footz1+Os_footz2) generate the reference swing foot x direction($Foot_x$) and z direction($Foot_z$) trajectories respectively. The reference joint angles are calculated by inverse kinematics. We employed position based control in joint space. Compared

with joint space implementation [65][55], this arrangement significantly reduces the total number of oscillators' parameters and provides a simple way to find effective feedback pathways. Endo et. al [22] successfully implemented a Cartesian based walking control by using two neural oscillators. In their approach, the two oscillators receive sensory inputs but work independently.

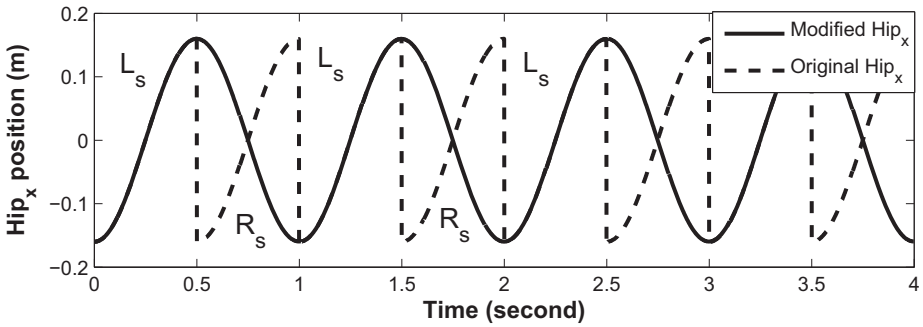


Figure 3.13: Reference Hip_x trajectory. L_s: left support; R_s: right support

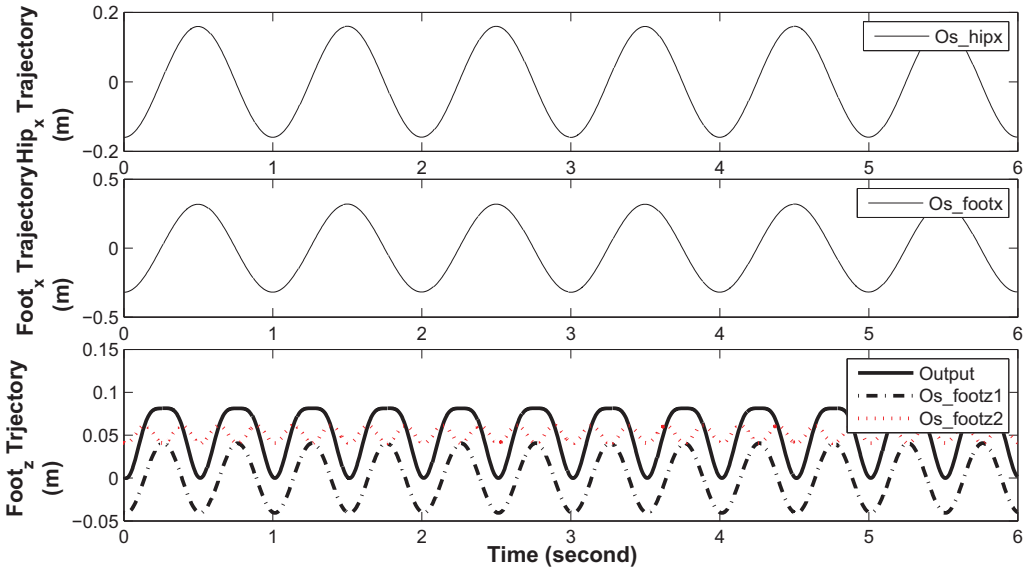


Figure 3.14: Reference trajectory of stance hip and swing foot

In this implementation, the oscillators in CPG are coordinated with each other by phase adjustment as described in subsection "Coordination between Oscillators" Fig. 3.8. Fig.

3.12 shows the reference frame attached to the ankle joint of the stance foot. The hip always moves forward during walking. An example of this trajectory is shown in Fig. 3.13 (dashed line). The Hip_x trajectory has a discrete change when the support leg switches. To obtain a continuous trajectory, we inverse the trajectory every time when a particular support leg touches the ground. The resulting trajectory is shown in Fig. 3.13(solid line) which can be generated by Os_hipx. As the swing foot x direction trajectory($Foot_x$) is similar to the Hip_x trajectory, it can also be generated by an oscillator Os_footx. We assume that the walking height is constant so that Hip_z is unchanged. The foot vertical position ($Foot_z$) is represented by a sum of two oscillators [32](Os_footz1+Os_footz2). Using several oscillators to generate complicated trajectories could also be seen in other research work [55].

Fig. 3.14 shows the reference trajectories of the stance hip and swing foot. Os_hipx gives the Hip_x trajectory; Os_footx gives the $Foot_x$ trajectory; $Foot_z$ is generated by the sum of Os_footz1 and Os_footz2. Phase adjustments are carried out among these oscillators. Os_hipx gives the reference phase to the other oscillators. $Foot_x$ position (Os_footx) adjusts $Foot_z$ (Os_footz1+Os_footz2) such that the foot will not touch down before the swing foot reaches the target position.

Sensory feedbacks are used to adjust the oscillators' outputs. They are input to oscillator through E_{in}. Two kinds of sensory feedbacks are used: body pitch and orbit energy [38], as shown in Fig. 3.12. The body pitch is feedback to Os1 to control the hip x position. The value of orbit energy is used to adjust the outputs of Os_hipx and Os_footx. Feedbacks are used to control the body position and step length in order to stabilize

the walking. Since there exists phase adjustment between (Os_footz1+Os_footz2) and Os_footx, swing foot height could be adjusted according to the swing foot horizontal position.

Swing leg retraction is commonly found in human walking. It is observed that the swing leg moves forward to the maximal forward position and then moves backward just prior to making contact with ground. This backward motion is called the swing leg retraction (Fig. 3.15). In robotics research, many studies have demonstrated the advantage of retraction to achieve stable walking [69, 31]. To achieve retraction, the phase of Os_footx is shifted a little forward to make the swing leg reach the maximal forward extension before it touches down. Therefore, the swing leg can move backward just prior to touching down.

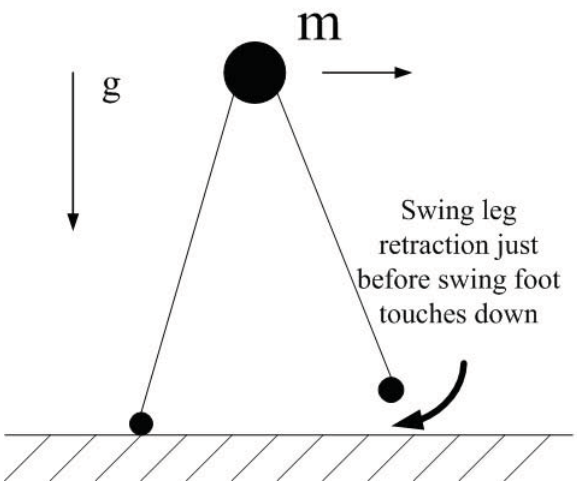


Figure 3.15: Swing leg retraction

3.4.2 Simulation Results

In the simulation, the oscillators' parameters are derived using the following steps [32]:

(1) According to Eq. (3.16), a should be bigger than 1. Here we let $a = 2.5$;

(2) The value of b in Eq. (3.18) is chosen to make $|a-1-b|$ small, typically 0.04-0.1;

(3) According to Eq. (3.18), to simplify the frequency calculation, let $\beta = a$. The frequency formula becomes $F = \frac{1}{2\pi\tau_1}\sqrt{b}$. Then $\tau_1 = \frac{\sqrt{b}}{2\pi F_d}$, where F_d is the desired frequency;

(4) According to Eq. (3.19), $c = \frac{1}{2}(1+\beta+a)A_d$, where A_d is the desired amplitude;

(5) $\tau_2 = \frac{\tau_1}{b}$.

Through these steps, we can roughly obtain all the values of the oscillator parameters based on the step length requirement. The reference trajectories shown in Fig.3.14, which are generated by these parameters, are used to control a 2D walking bipedal robot. Fig. 3.16 shows the snapshots of a level ground walking. The walking step length is 0.35 m and the walking period is 0.5 s. The walking speed is about 0.7 m/s.

In the CPG structure shown in Fig. 3.12, Os_hipx will adjust the output of Os_footx and Os_footx will adjust the outputs of Os_footz1 and Os_footz2. A 50 N external horizontal force is applied forward at the robot hip for 0.5 s. The walking motion recovers after the perturbation. After 4 s, another 50 N external force is applied to the robot. The robot is able to adjust the motion and continue walking. The oscillators' outputs and feedbacks are shown in Fig. 3.17. Because of the pitch and orbit energy feedbacks, the step length increases in x direction to adjust the motion and reduce the effect of the perturbation (shown by dotted circle). The output of Os_footx($Foot_x$) changes more significantly in comparison with Os_hipx(Hip_x). Since there exists coordination between them, the

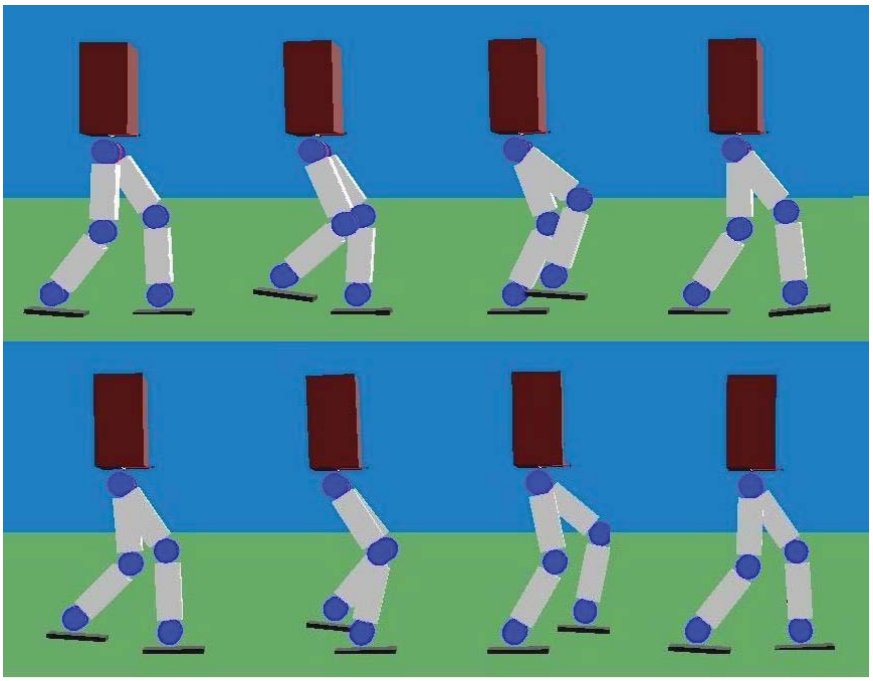


Figure 3.16: Snapshots of forward walking in simulation

oscillators' outputs will be adjusted and their phase relationship will go back to the normal walking mode after the perturbation. The stick diagram of the walking under the influence of external force is shown in Fig. 3.18. When there is no coordination among the oscillators, the phase error cumulates because of the perturbation (shown by dotted circles in Fig. 3.19) and the robot falls after the second push.

Walking with different step lengths and frequencies is tested as well in the simulation. The step length is changed by changing the step length scaling factor c. The walking frequency is changed by changing the time constant τ_1. The ratio between τ_1 and τ_2 is kept constant. As shown in Fig. 3.20, the step length changes from 0. 3m to 0.35 m and then eventually reduces to 0.2 m. The speed varies from 0.46 m/s to 0.8 m/s. Fig. 3.21 shows that the frequency changes during walking. The step length is constant which is 0.3 m. The walking period is 1 s for the first 4 s, 0.8 s for the second 4 s, 1 s for the third

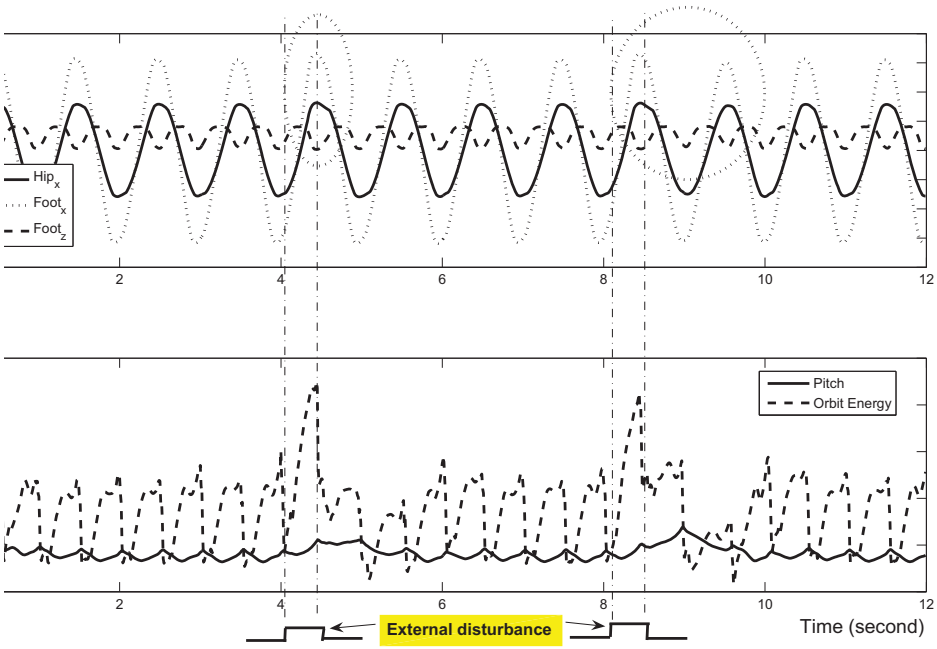


Figure 3.17: Os_hipx,Os_footx and Os_footz1+Os_footz2 outputs and sensory feedbacks in the forward walking with external disturbance

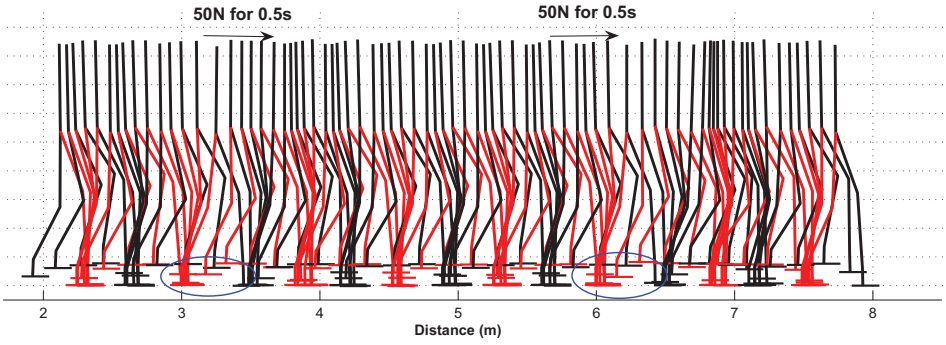


Figure 3.18: Stick diagram of walking with external disturbance

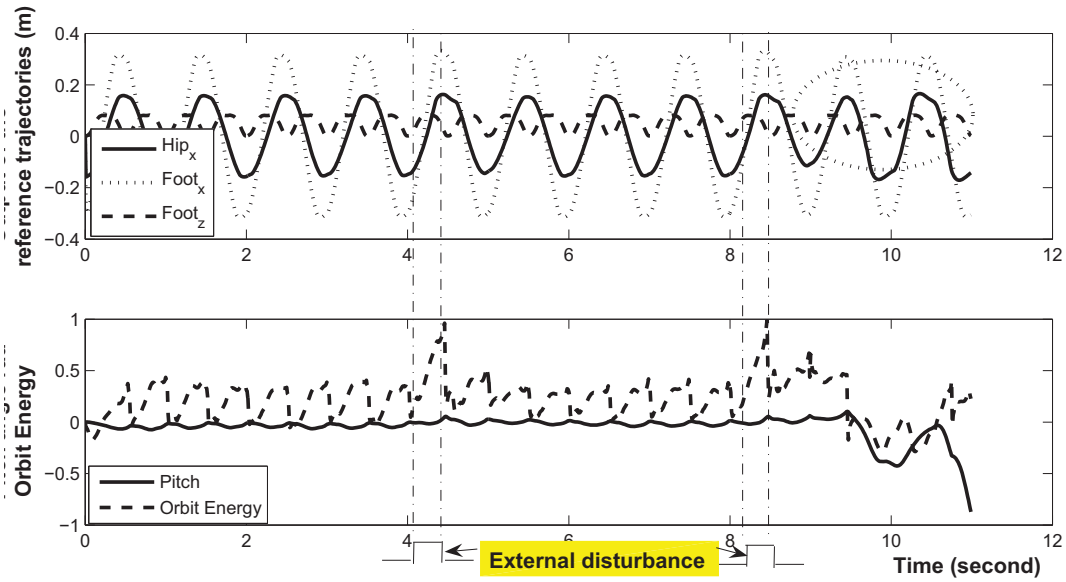


Figure 3.19: Os_hipx,Os_footx and Os_footz1+Os_footz2 outputs and sensory feedbacks in the forward walking without coordination

4 s, and 1.2 s for the rest. The simulation shows that the oscillators can coordinate their outputs when the walking step length and frequency are changed.

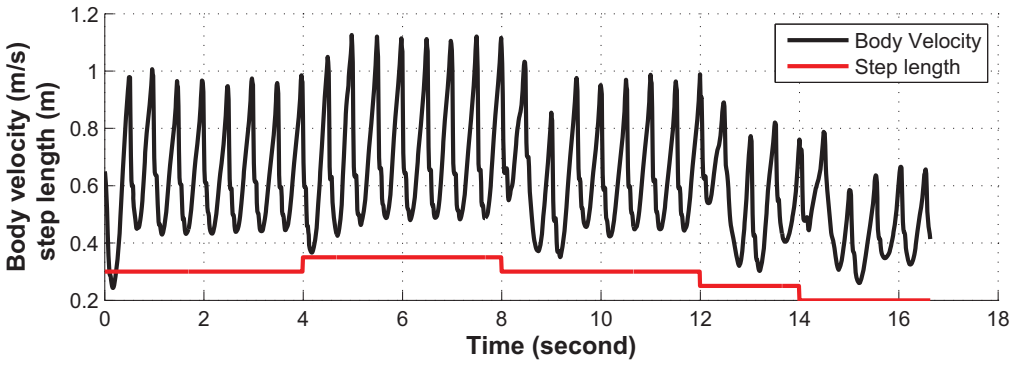


Figure 3.20: Body velocity plot when walking step length is changed

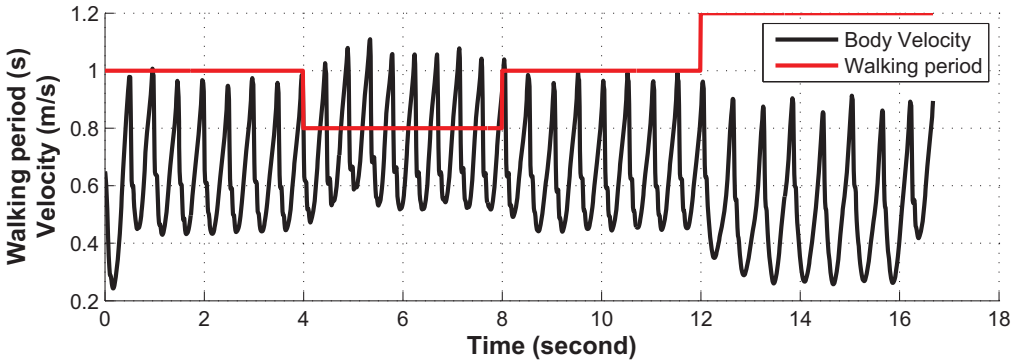


Figure 3.21: Body velocity plot when walking frequency (walking period) is changed

3.5 Summary

This chapter presents a coordination method between oscillators in CPG. In this method, the output of a main oscillator is used to adjust the output of other oscillators to maintain the phase relationship between them. Because the method is not based on any external input outside the CPG, the whole CPG system can be coordinated without receiving any external command. The CPG controller has been implemented on a 2D robot to control it to walk. Several external forces are applied on the robot during walking to verify the effectiveness of this coordination method. The neural oscillators in CPG adjust their outputs according to the external input and result in a robust walking behavior.

In the simulation, the body velocity in x direction is used as a feedback to the oscillators. When the robot is pushed by external force, the body velocity changes which feedbacks to the CPG controller. An interesting result is that, with the velocity feedback to the oscillator, the oscillators change their output to balance the body. In the simulation, the CPG controller increases the step length to overcome the effect of push. The simulation results indicate the important of sensory feedback in the CPG based walking control.

A good CPG structure without a proper sensory feedback design cannot achieve an adaptive walking behavior. In this chapter, only one type of sensory feedback is used in the control. In the next chapter, we will further explore the effect of sensory feedback to the CPG controller by analyzing more types of the sensory feedback in the controller.

Chapter 4

Coordination between CPG and Sensory Feedback

4.1 Introduction

In the previous chapter, we propose the phase adjustment method between oscillators to achieve the coordination behavior. On the other hand, to have a rhythmic and adaptive walking, CPG controller also needs to interact with the environment. In this case, sensory feedback provides the environment information to the oscillators in CPG. In the previous 2D walking simulation, sensory feedback was input to the oscillator through external input E_{in} (Equation (3.1)-(3.4)). According to the feedback, oscillators modified their outputs to overcome the external perturbations. The feedback information helped the robot achieve a robust walking in the 2D dynamic simulation.

In the 3D walking, double supporting phase is an important procedure to shift the center of mass (CoM) towards the supporting polygon for a comfort foot lifting. This procedure is omitted in the 2D walking simulation. In the double supporting phase, both feet remain on the ground. The foot trajectories in Cartesian space are kept constant. To generate the desired trajectories, the corresponding oscillators' outputs need to be kept constant too. However, suspending the activity of neural oscillator to a desired value is challenging by external input. To enable the oscillator to have a wider range of applications, more ways of modifying the output of the oscillator need to be developed.

In this chapter, we further explore the effect of sensory feedback on the output of the oscillator. Two new ways of modifying the output of the oscillator are introduced. Based on the way of modifying the output of the oscillator, we classify the sensory feedback into three types: inhibition input, triggering input and parameter modification input. Inhibition input is the input given through the external input E_{in}. Triggering input is a new type of input which is designed to activate or suspend the activity of the oscillator. Parameter modification input is another new type of input which is designed to modify the frequency or amplitude of the oscillator. A more detailed information of these three inputs is given in next section. The purpose of this classification is to enable a better way of feedback design based on the required response pattern of the oscillator.

With the proposed sensory feedback design, we implement the CPG controller to control a 3D dynamic stepping motion. A robust stepping motion is achieved in the simulation. The sensory feedback adjusts the trajectories of the CPG controller when perturbations are applied. This design of sensory feedback helps the CPG controller to successfully

control the bipedal walking in 3D dynamic environment.

4.2 Sensory Inputs to the Oscillator

In the CPG control, sensory feedback is used to modify the output of the oscillators to adapt to the environment. In Equation (3.1)-(3.4), E_{in} is a common way of adding sensory feedbacks. It performs as an inhibition term to the differential equations, namely inhibition input. The inhibition input is the most widely used in the CPG research as a way to receive the sensory feedback [64][22][55][67]. Many 2D dynamic simulation results showed that the CPG controller generated robust bipedal walking pattern with appropriate inhibition input parameters [64][50][32].

Sensory feedback can also be used to activate the activity of the oscillator. Sherwood's neuron study [59] indicated that the motor neuron releases a flood of neurotransmitters that bind to postsynaptic receptors and triggers a response in the muscle fiber based on receiving adequate stimulation. Grillner's experiments [27] also showed that a cat could spontaneously initiate locomotor episodes by nociceptive stimuli or electrical stimulation of the subthalamic nucleus. These studies indicate that the state of the neuron is activated or suspended by the external stimulation. Inspired from these experiments, we propose a new type of input, namely triggering input. This type of input can be used to activate the certain state of the neuron such as activating or suspending the activity of the neuron. An activating signal is generated based on several sensors' readings. For example, a triggering input, which activates the oscillator for foot trajectory in vertical

direction, is generated by considering the load sensors and position sensors on both feet. A suspending signal for this oscillator is generated based on the ground contact information and swing period. This signal can be considered as a signal from top layer (such as brain) which analyzes the sensory feedback and gives the command accordingly to the oscillators. This property can be greatly useful in designing double supporting phase trajectories. Also, an under-actuated oscillators can be designed to offer an additional motion for the CPG controller when the environment changes. For example, a sudden stepping forward can be activated there is strong push from back.

The phase resetting signal which is used to rest the phase value of the oscillator can be found in many CPG research works. In general, a phase resetting signal is applied to all the oscillators when a single walking cycle is completed. This walking cycle includes two single supporting phase of walking. The state value of phase will be reset when the oscillator receives the phase resetting signal. Yamasaki et al. [70] investigated the role of phase resetting in biped locomotion. Their studies reveal possible contribution of phase resetting during walking to gait stability against external perturbations. Nakanishi et al. [51] also showed the importance of phase resetting in the bipedal walking control by the oscillators. Because the phase resetting signal is used to trigger the event of phase resetting, we classify this type of signal as triggering input.

Changing the inner parameters of the oscillator, such as τ_1, τ_2, c, a, β, is another way to modify the oscillating pattern, namely parameter modification input. A robot may need to change the walking pattern when environment changes. For example, a change from flat terrain walking to slope walking may result a change of oscillator's amplitude

in CPG; a change from slow walking to fast walking due to the big external push may result a change of oscillator's frequency. In this case, changing the oscillator's parameters is a simple and straightforward way to change the oscillating pattern. Oscillator's frequency is determined by the time constant of the oscillator. Since the change of environment is not considered in this thesis, this type of input is not detail studied. However, by enabling the modification of parameters according to the sensory inputs, it helps the oscillator to adapt to a different environment. In this case, the modification of the oscillator's parameters should satisfy with the oscillating requirement of the oscillator.

In summary, we classify the sensory inputs to a oscillator into three types according to the way they change the output of the oscillator.

(1) Inhibition inputs: they are inputs which are added as a term in the differential equations.

(2) Triggering inputs: they are inputs which are used to modify the state values of the oscillator. It can be used to trigger the activity of an oscillator which is newly introduced in this thesis, or reset the phase value of an oscillator.

(3) Parameter modification inputs: a new type of input introduced in this thesis which consists of inputs to modify the parameters of the oscillators to change the oscillating pattern.

The purpose of the classification is to make the sensory feedback design simple and straightforward. The diagram in Fig. 4.1 shows how sensory input affects the output of the oscillator. Among these three types of inputs, triggering inputs and parameter

modification inputs have not been widely explored by researchers. In this thesis, besides using the inhibition inputs, the triggering input will be explored in designing the CPG controller. These sensory-feedback design will be used to control a 3D stepping motion. Our primary goal is to design a CPG controller which can response correctly to the sensory feedback signals and achieve an adaptive behavior.

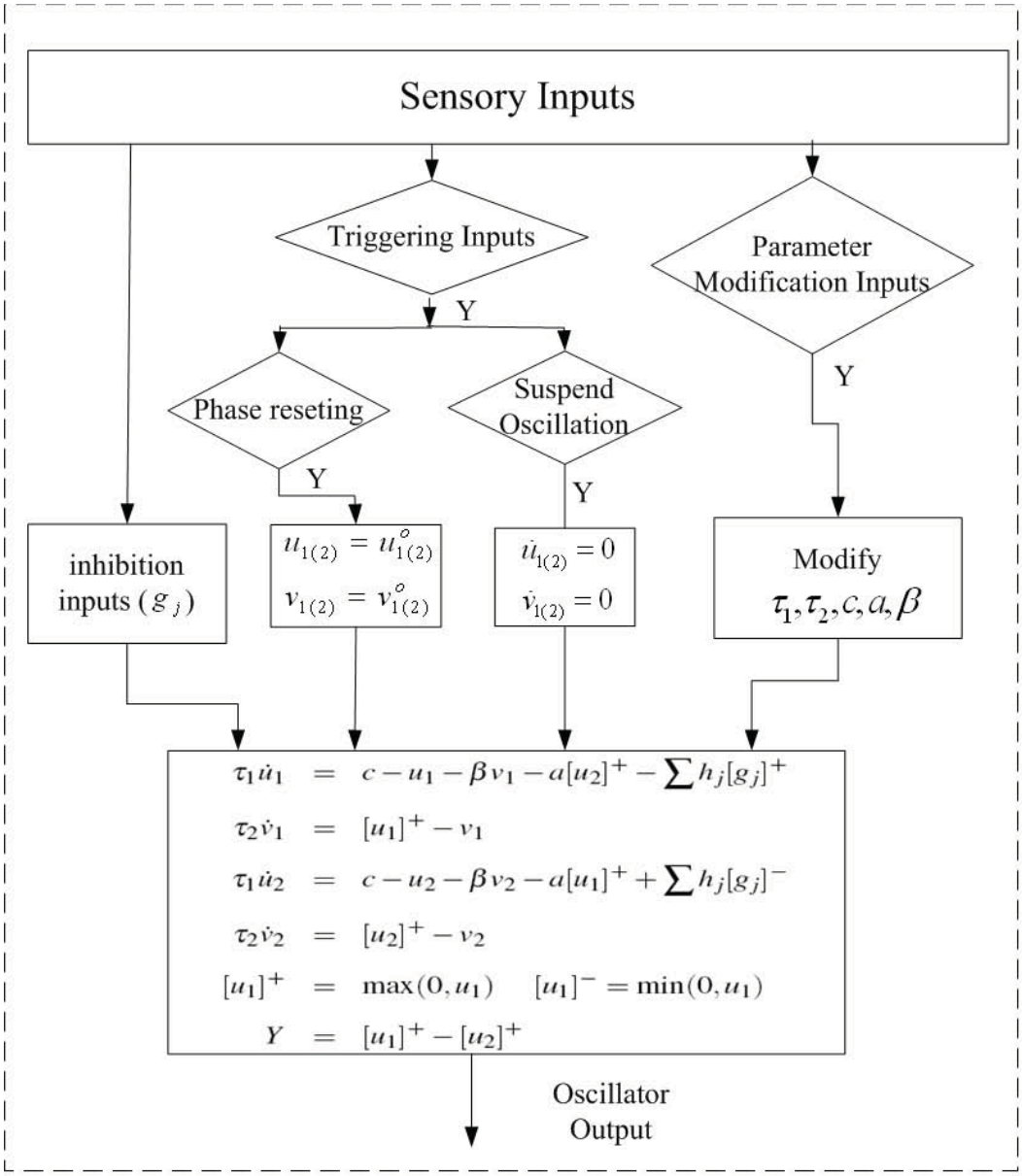


Figure 4.1: The diagram of how sensory input affects the output of the oscillator, $u^o_{1(2)}$ and $v^o_{1(2)}$ are state value for the desired phase

4.3 Stepping Motion Controlled by CPG

In the previous section, we have introduced three types of sensory inputs to the oscillator. These sensory inputs are useful when implement the CPG controller in a 3D walking control. In this section, we will present the implementation of the CPG controller with senosry feedback in the control of a stepping motion in 3D environment.

4.3.1 Proposed Stepping Motion Description

Stepping motion is one of the basic locomotion behaviors. A walking motion includes the motions in both frontal and sagittal plane. 2D walking in sagittal plane has been widely studied in CPG research due to its simpler dynamics compared with 3D walking. However, because of the difference between 2D and 3D dynamics, it is difficult to apply the 2D walking algorithms in the 3D walking control directly. On the other hand, stepping motion can be considered as a 3D dynamic walking with 0 forward speed. It is simpler than walking forward as there is no sagittal movement. However, a forward walking can be easily achieved from a robust stepping motion by moving the body and swing foot forward.

Generally, a stepping motion can be divided into four sub-phases:

(1) DS_l: double supporting phase while the body is moving to the left.

(2) SS_l: single supporting phase with left support.

(3) DS_r: double supporting phase while the body is moving to the right.

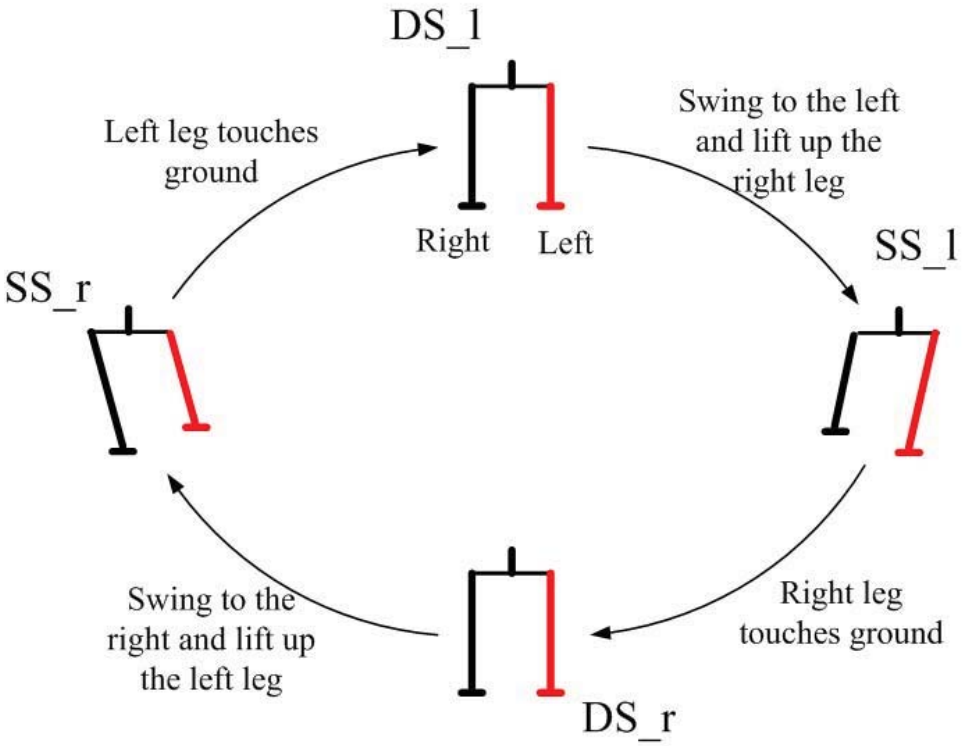


Figure 4.2: The cycle of marching on the spot

(4) SS_r: single supporting phase with right support

A successful switch from one phase to the next phase enables a continuous stepping motion. To walk continuously, the supporting leg must be alternated. Bipedal walking is achieved by a smooth alternation of the two supporting legs. In the stepping motion, the load shifts from one foot to the other foot when the supporting leg exchanges. When the load on a foot reduces to a certain value, the foot can be lifted up to let the other foot support the robot body without losing system stability. The load on both feet can be used as a feedback to indicate when to lift up the foot. In the simulation, four load sensors are place at each corner of the foot. By calculating the force values on eight load sensors on both feet, we can calculate the Center of Pressure (CoP) [35]. The CoP information is used to indicate when to lift up the foot. Two conditions are required here

for transition from double support to single support. The transition will only happen when both conditions are satisfied.

(1) When the CoP position reaches a preset value.

(2) When the double supporting phase period exceeds a preset value.

The purpose of setting the timing requirement (second condition) is to avoid a sudden transition.

Two conditions are adopted for transition from single support to double support. The transition will only happen when both conditions are satisfied.

(1) When the swing foot touches the ground.

(2) When the single supporting phase period exceeds a preset value.

Fig. 4.3 shows the state transition diagram between double supporting phase and single supporting phase.

4.3.2 Arrangement of Oscillator and Sensory Feedback

In the stepping motion analysis, a stepping motion can be divided into three parts: the moving of body hip in frontal plane; lifting up right leg with left leg support and lifting up left leg with right leg support. In this simulation, the CPG controller is designed to control the position of the leg in the Cartesian coordinate space.

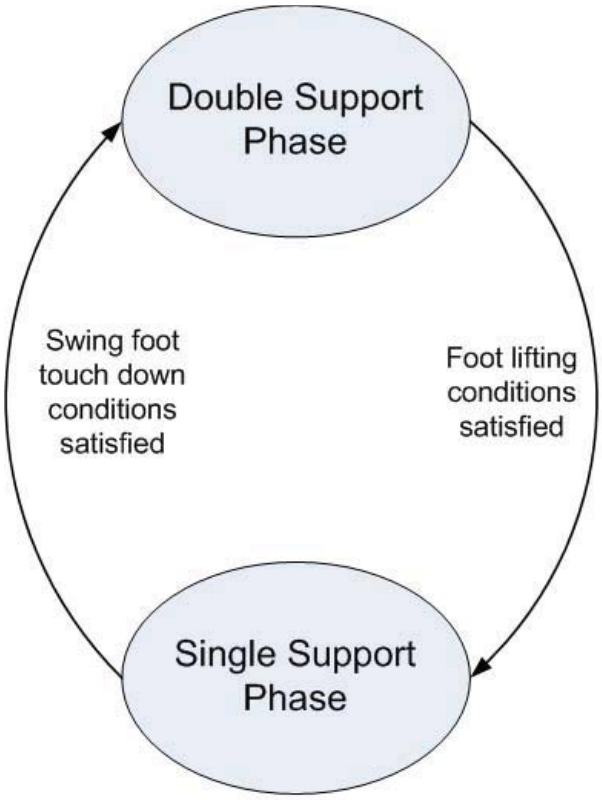


Figure 4.3: The state transition diagram between double supporting phase and single supporting phase

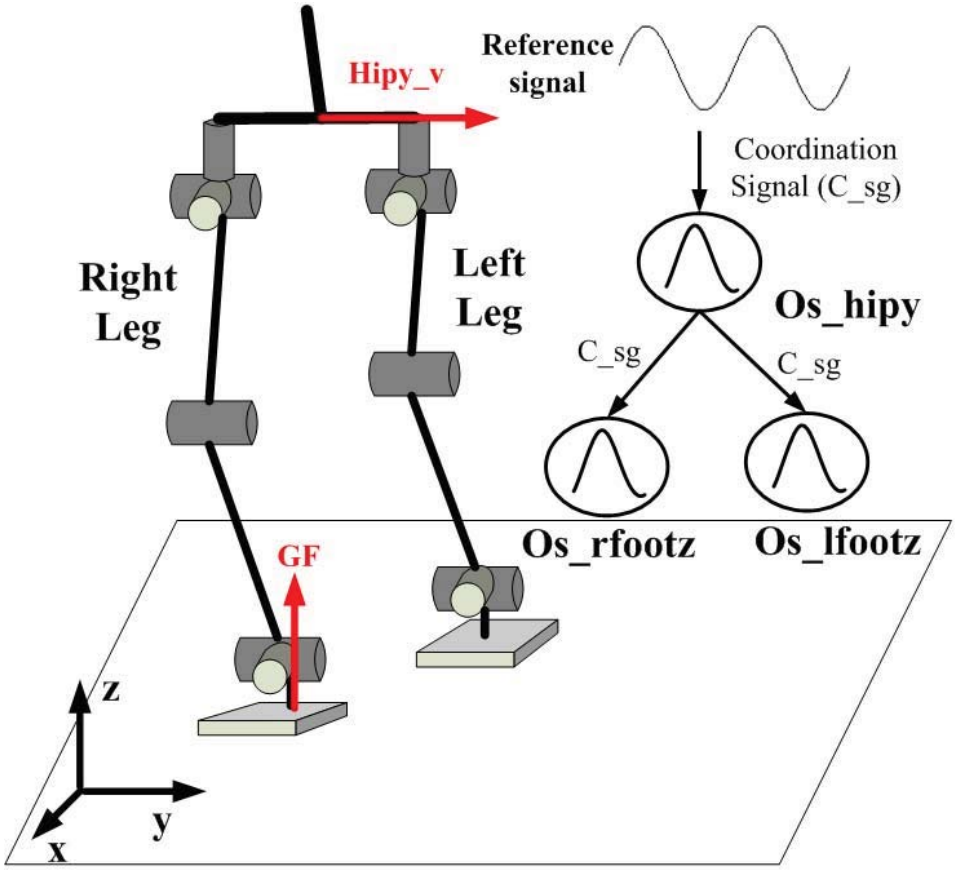


Figure 4.4: The arrangement of the oscillators for stepping motion; only coordination connection between oscillators (C_sg) are shown in the figure

As shown in Fig. 4.4, oscillator *Os_hipy* generates the reference trajectory for body hip in the y direction. Oscillators Os_lfootz and Os_rfootz generate the reference trajectories for both feet in z direction, respectively. Since the robot is not moving forward, the positions of hip and both feet in x direction are kept constant. Fig. 4.4 also indicates the coordination connection between oscillators. The oscillator *Os_hipy* is adjusted by a reference signal. This reference signal is a sinusoidal wave changed according to the system time. It provides the reference phase for the oscillator *Os_hipy* when there are external perturbations. In the stepping motion, the swing foot lifts when the CoM falls in the desired range. The activation of foot trajectories is related with the hip body motion. We set oscillator *Os_hipy* to generate coordination signals for both Os_lfootz and Os_rfootz.

Two types of sensory feedback are used here: ground reaction force on each foot and body velocity in y direction. The sensory input to the oscillators is shown in Fig. 4.5.

In the stepping motion, the body swings in the frontal plane. Here, we use the body velocity in y direction $Hipy\_v$ as an inhibition input to *Os_hipy* to perform as a damping effect on the swing motion. It can help to prevent the overshot of the swing motion.

In the stepping motion, the robot needs to know when to lift up the foot where a triggering input is required. In this design, the CoP value is used as a trigger to indicate when to lift up the foot. The CoP value can be calculated from the ground reaction forces on both feet. When the CoP value falls in a preset range, either Os_lfootz or Os_rfootz will be activated depending on which leg is the swing leg. After the swing foot touches the ground, the output of the oscillator can be suspended by the signal of touching down.

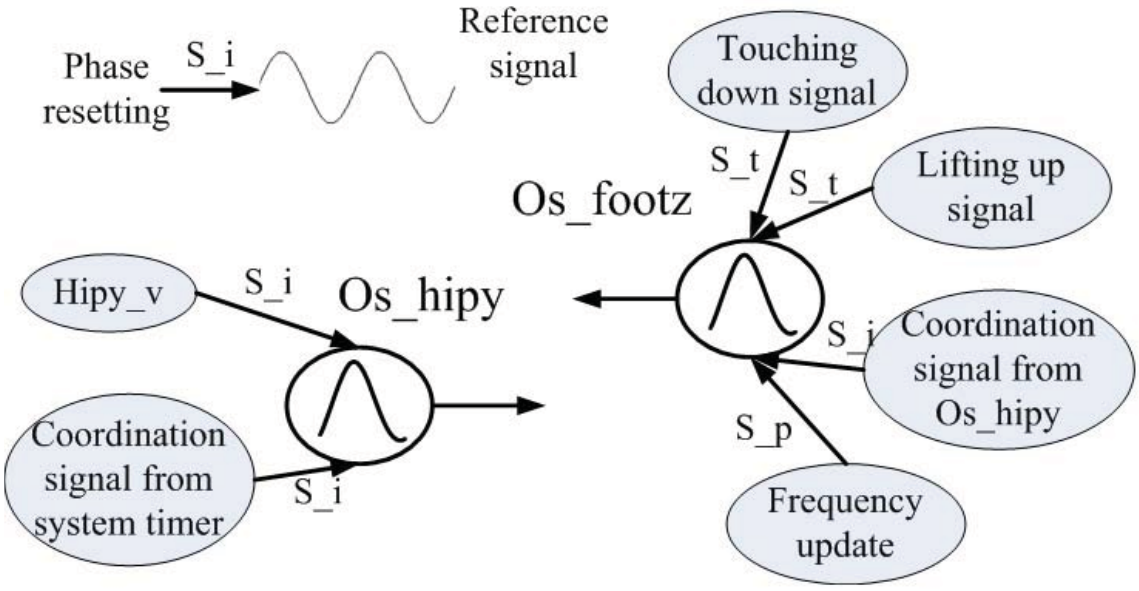


Figure 4.5: The sensory input to the oscillator; S_i, S_t and S_p are the inhibition input, triggering input and parameter modification input which show how the the sensory feedback input to the oscillators

After both feet have touched the ground in turn, phase resetting signal will reset the phase of all the oscillators.

Since the timing of foot lifting may not be the desired one, the frequency of the foot oscillator needs to be modified to synchronize with the body motion. The parameter modification input is used to change the time constant of the oscillator. In the simulation, the timing of foot lifting will generate the frequency update signal to modify the time constant of the oscillator.

In the stepping motion, the phase resetting signal is activated after both feet have touched down sequently. In this simulation, the robot swings to the left side at first. Therefore, the right foot lifts up first and left foot lifts up later. The phase resetting signal is designed to be activated when the left foot has touched ground. In the implementation, to avoid the discontinuity of the oscillator's output, the phase resetting signal is not directly

introduced to all the oscillators. Instead, in our setting, we first introduce the phase resetting to the reference signal. Then the reference signal will generate the coordination signals for the rest of oscillators to achieve the desired relative phase.

4.3.3 Discrete Time Oscillator Model

With the arrangement of the oscillators and sensory feedback, the proposed CPG controller can generate the reference trajectories for the stepping motion. To implement the controller in the simulation, we convert the differential equations of the neural oscillator model described by (3.1)-(3.6) into discrete time equations. Fourth order Runge-Kutta ($4R\_K$) approach is used to do the conversion. The discrete time equations are shown as:

$$[U_{out}^{1(2)}(i), V_{out}^{1(2)}(i)] = 4R\_K(U_{in}^{1(2)}(i), V_{in}^{1(2)}(i), a, s) \tag{4.1}$$

$$Y = max(0, U_{out}^{1}) - max(0, U_{out}^{2}) \tag{4.2}$$

where U and V are the states of the oscillator; s is the input of sensory feedback; a is the coordinating input from other oscillators and Y is the oscillator output.

In the discrete time domain, to activate the oscillator output, the previous step's output is adopted as the new input to the oscillator.

$$[U_{in}^{1(2)}(i+1), V_{in}^{1(2)}(i+1)] = [U_{out}^{1(2)}(i), V_{out}^{1(2)}(i)] \tag{4.3}$$

To suspend the neural oscillator output, the next input to the oscillator is taken to be the same as previous step's input such that the oscillator output is not changed.

$$[U_{In}^{1(2)}(i+1), V_{In}^{1(2)}(i+1)] = [U_{in}^{1(2)}(i), V_{in}^{1(2)}(i)] \tag{4.4}$$

With this conversion, the CPG controller can be used in the simulation to generate the reference stepping trajectories.

4.3.4 Simulation Experiments

Simulation Environment

The simulation software used here is Webots, which allows users to conduct realistic dynamical simulation of robots in the 3D virtual environment. The simulated robot has a height of 175cm and a weight of 75 kg. It has 42 DoFs (7 on each leg, 6 on each arm, 6 on each hand, 2 on the waist and 2 on the neck). The schematic diagram of the robot is shown in Fig.4.6.

In the oscillator arrangement, three neural oscillators generate the reference trajectories for body hip position in y direction and feet position in z direction. In the stepping motion, we assume that the position of hip in x and z direction are kept constant. The position of both stance and swing feet in x and y direction are also kept constant. Through the inverse kinematics, the joint angles of hip yaw, hip roll, hip pitch, knee pitch, ankle

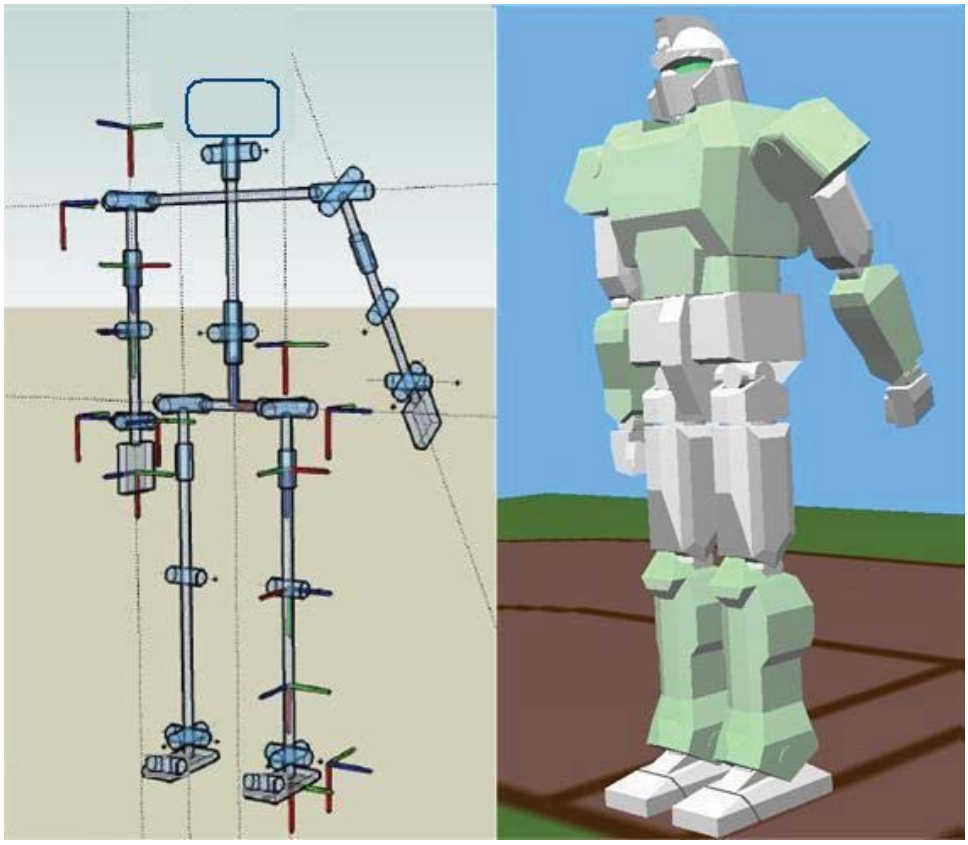


Figure 4.6: Schematic diagram of the simulated robot

roll and ankle pitch on both left and right leg can be calculated.

$$[Hip^{l(r)}_{y(r,p)}, Knee^{l(r)}_{p}, Ankle^{l(r)}_{r(p)}] = IK(hip_{x(y,z)}, lfoot_{x(y,z)}, rfoot_{x(y,z)}) \tag{4.5}$$

Here, IK indicates inverse kinematics. Based on inverse kinematics, the joint angles on both legs can be calculated by the position values of body hip, left foot and right foot. $Hip^{l(r)}_{y(r,p)}$ is the angle value of left(right) hip yaw(roll or pitch); $Knee^{l(r)}_{p}$ is the angle value of left(right) knee pitch; $Ankle^{l(r)}_{r(p)}$ is the angle value of left(right) ankle roll(pitch); $hip_{x(y,z)}$ is center of body hip position in x(y or z) direction; $lfoot_{x(y,z)}$ is the left foot position in x(y or z) direction; $rfoot_{x(y,z)}$ is the right foot position in x(y or z) direction.

The reference joint angle is calculated based on the inverse kinematics according to the desired trajectories in Cartesian space. Then a PID control law is used in the joint control which is integrated in the dynamic simulation software "Webots".

Oscillator Output Adjustment

To maintain the system stability and reduce the energy consumption during foot lifting, the foot is lifted up when the CoP position falls into a preset range. This range will be analyzed in the later simulation. In order to achieve a natural lifting-up motion, a triggering input $lift\_s$ is used to adjust the output of the $footz$ oscillators. When the CoP position falls into the preset range, $lift\_s = 1$. It activates the neural oscillator to generate reference trajectory to lift up the foot. When the foot touches down, $lift\_s =$

-1. It suspends this neural oscillator's output.

Since the timing of lifting up the foot is dependent on the CoP location of the robot, it is not fixed. When the timing of lifting up is earlier than desired, the period of the oscillator should be longer. When the lifting of the foot is delayed, the period of the oscillator should be shorter. Fig.4.7 shows an example of the swing foot oscillator output during half walking cycle, where T is the walking cycle period; T_{s1} and T_{s2} are the two double supporting periods before and after the foot lifting period respectively; and T_l is the oscillator period. If we set $T_{s2} = 0.05T$, we obtain

$$T_l = 0.45T - T_{s1} \tag{4.6}$$

T_l generates the frequency update signal to modify the time constant τ_1 of the oscillator as shown in Fig. 4.5.

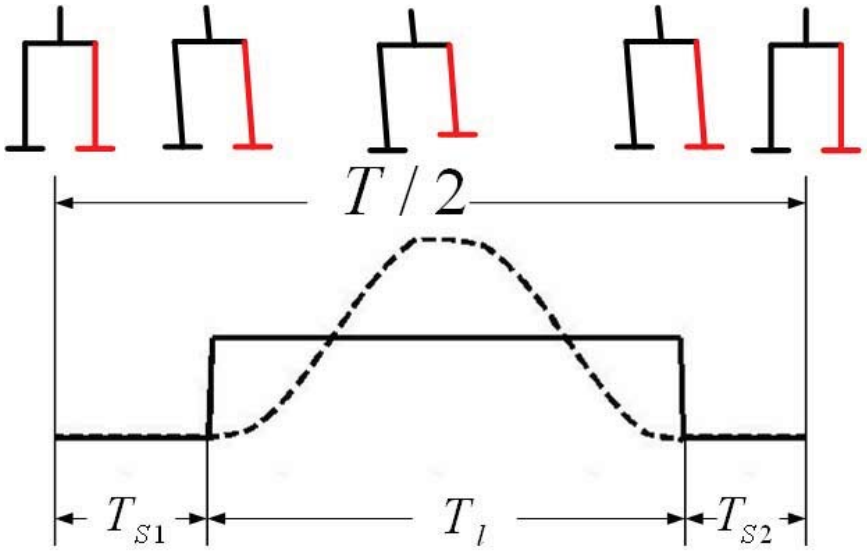


Figure 4.7: The output of swing foot oscillator in half walking cycle

When the lifting foot touches down, $lift\_s = -1$ which will suspend the oscillator's

output. Touching down can be divided into two cases. Case 1 is that the lifting foot touches the ground later than the reference trajectory. Case 2 is that the lifting foot touches the ground earlier than the reference trajectory. In case 1, the reference oscillator $foot\_z$ trajectory will continue to decrease to ensure that the swing foot touches on the ground. On the other hand, in case 2, when the swing foot touches the ground before the reference command, the oscillator output $foot\_z$ will be suspended and reset. In both early and late touching down, the oscillator's trajectory is discontinuous. An additional smooth trajectory will be added to ensure the overall reference trajectory returns to 0 which corresponds to the foot touching the ground. Fig.4.8 shows an example of the $foot\_z$ trajectory adjusted by touching down signal.

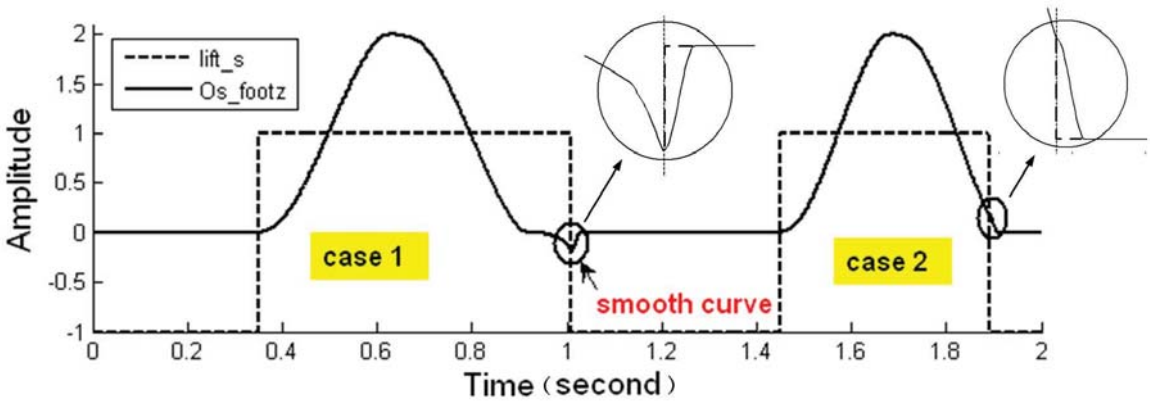


Figure 4.8: An example of *Foot_z* output adjusted by touching down signal; the trajectories shown in circle is the smooth trajectory which make the reference trajectories return back to 0

Simulation Results

In the first experiment, the CPG structure is used to control a humanoid robot walking on the spot. The period of stepping cycle is 1.2s. The trigger to activate the foot oscillator is the CoP location in y direction. Here, we set the origin at the center of the two feet.

If the CoP location in y direction is less than -0.01m, the right foot will lift up. If the CoP location in y direction is bigger than 0.01m, the left foot will lift up. Based on the foot size, the supporting polygon of the left support and the right support in y direction are [0.043 0.187]m and [-0.187 -0.043]m respectively. The robot lifts up the swing foot before the CoP moves into the supporting polygon.

The resultant trajectories generated by the oscillators are shown in Fig.4.9. GF denotes the ground reaction force in Z(vertical) direction. The trajectories of both feet in Z direction are activated by the triggering inputs. As shown in the swing period subplot in Fig.4.9, the swing period approaches a stable patten every two steps. The left swing period is a little longer than the right one. The relation between hip position and velocity in y direction is shown in the Fig.4.10. The trajectory approaches a limit cycle behavior. However, the actual robot swings further to the left side than right side. This results a difference of swing time between both legs. In this experiment, the right foot is lifted up first. We also tested the case of lifting up the left foot first. In this case the swing period of left support is shorter than the right support. These results show that the sequence of lifting up the foot may result in a asymmetry between both legs.

A further exploration founds that the difference in swing period between left and right supporting phase is because of the phase resetting signal. In our experiment, the phase resetting is given based on a whole walking cycle which is only after the right supporting phase. There is no phase resetting after the left supporting phase. To balance both left and right supporting phase, we add another phase resetting after the left supporting phase as shown in Fig. 4.11. The resultant trajectories generated by the oscillators are shown

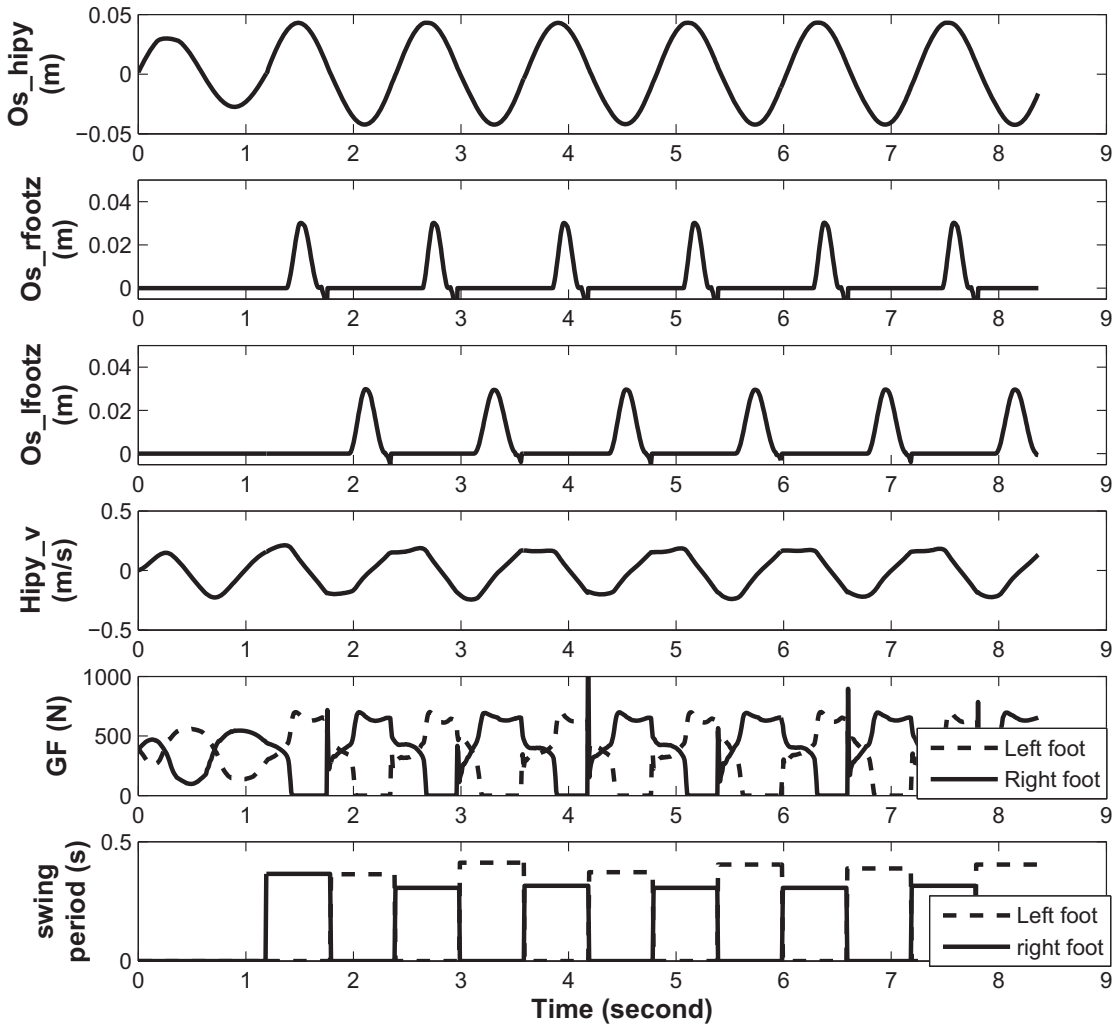


Figure 4.9: Simulation data: the reference trajectories generated by oscillators, the body velocity, the ground reaction force on both feet and the swing time in each step (from top to bottom); the foot trajectories are activated by the triggering inputs

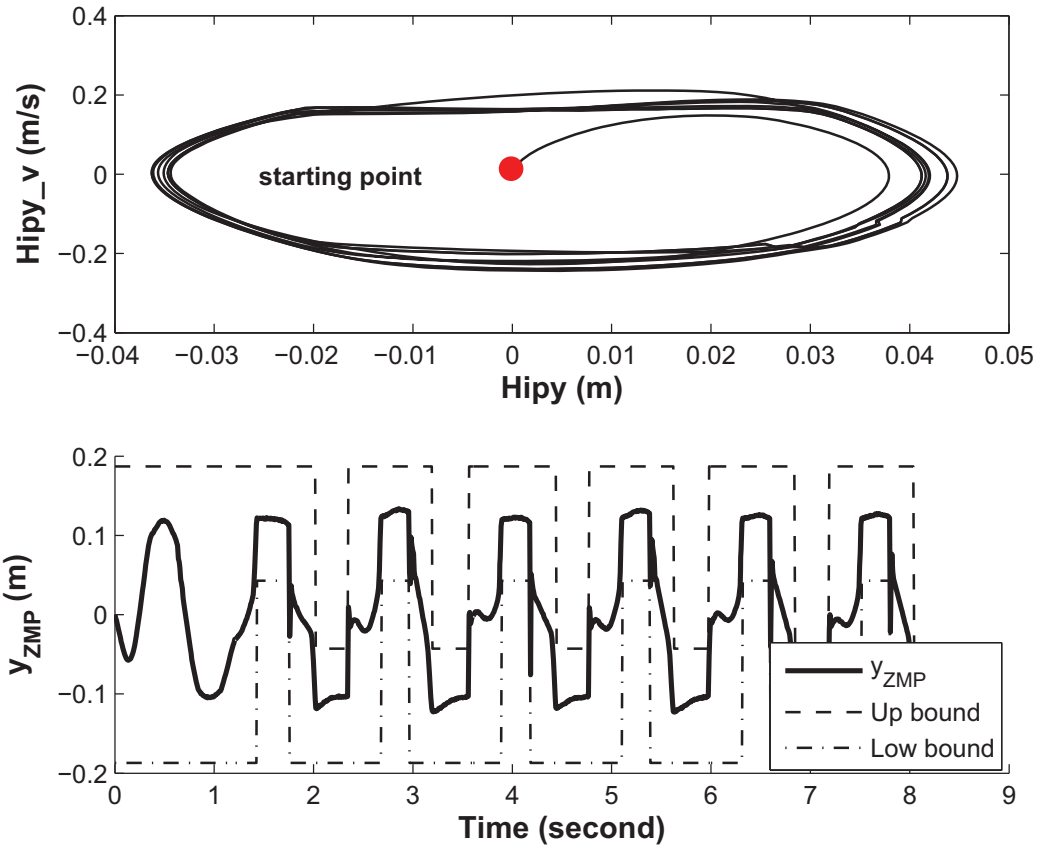


Figure 4.10: Simulation data: the limit cycle behavior of body hip position and velocity in y direction (top); a plot of ZMP trajectory (bottom)

in Fig. 4.12. In this case, the swing periods on both supporting phases converge to a similar value. The hip swing trajectory is also more symmetric with two phase resetting signals (as shown in Fig.4.13.

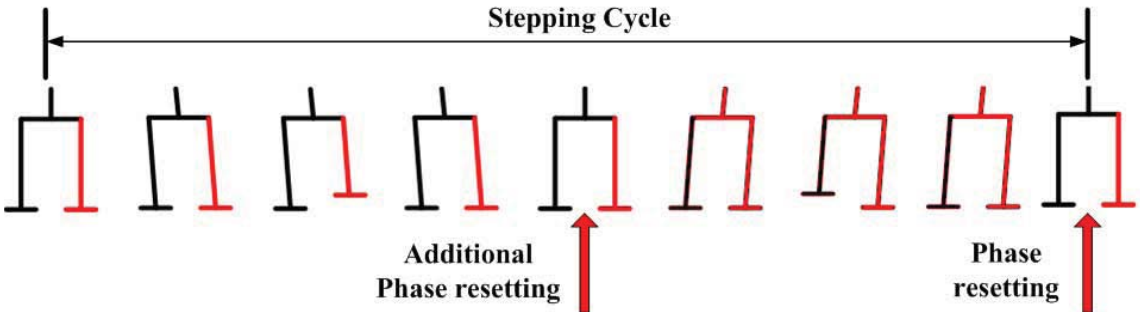


Figure 4.11: Additional phase resetting signal during the stepping motion cycle.

To generate the reference trajectories for 3D walking control, having a double supporting phase is quite important. We tried to test the control of 3D walking without the double supporting phase. In the experiment, the motion is very unstable and unnatural. Fig. 4.14 shows the response of the CPG controller without the double supporting phase. In each cycle, the foot touches the ground much earlier than desired. As shown in Fig. 4.15, the stepping motion does not approach a limit cycle and the CoP location in single support is always on the boundary. This shows that the motion without the double supporting phase is not stable.

In this study, the length of double supporting period is chosen based on the experiment tests. In the previous experiment, if the CoP is located at 0.01m or further, the left supporting phase will be activated. If the CoP is located at -0.01m or further, the right supporting phase will be activated. Here, the value of 0.01m is chosen based on several tests. Fig. 4.16 shows the swing period of stepping motion with different CoP triggering value. Based on the comparison, the swing period of CoP value at 0.01 is more stable

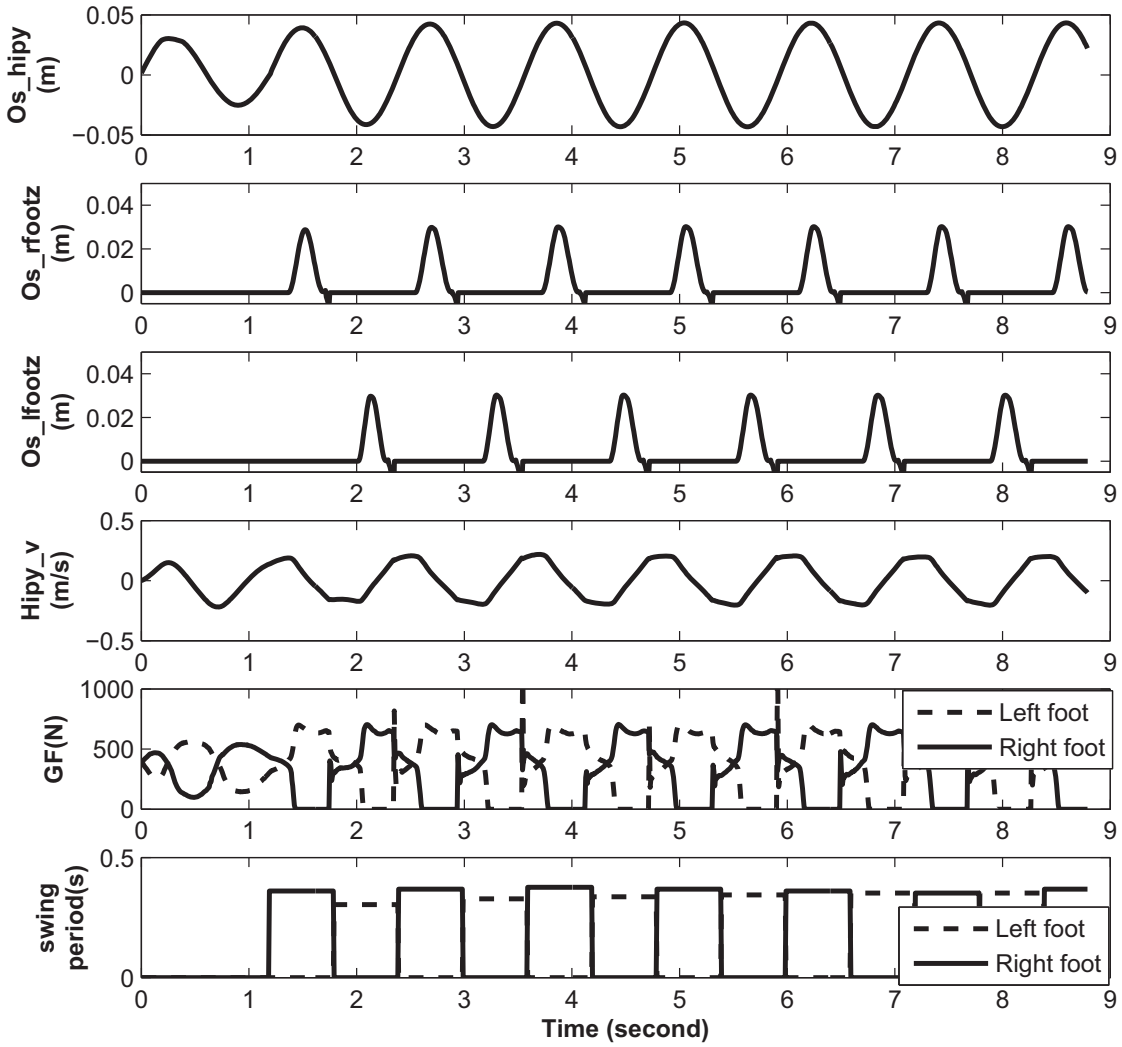


Figure 4.12: Simulation data: the reference trajectories generated by oscillators, the body velocity, the ground reaction force on both feet and the swing time in each step (from top to bottom); the controller is adjusted by two phase resetting signal.

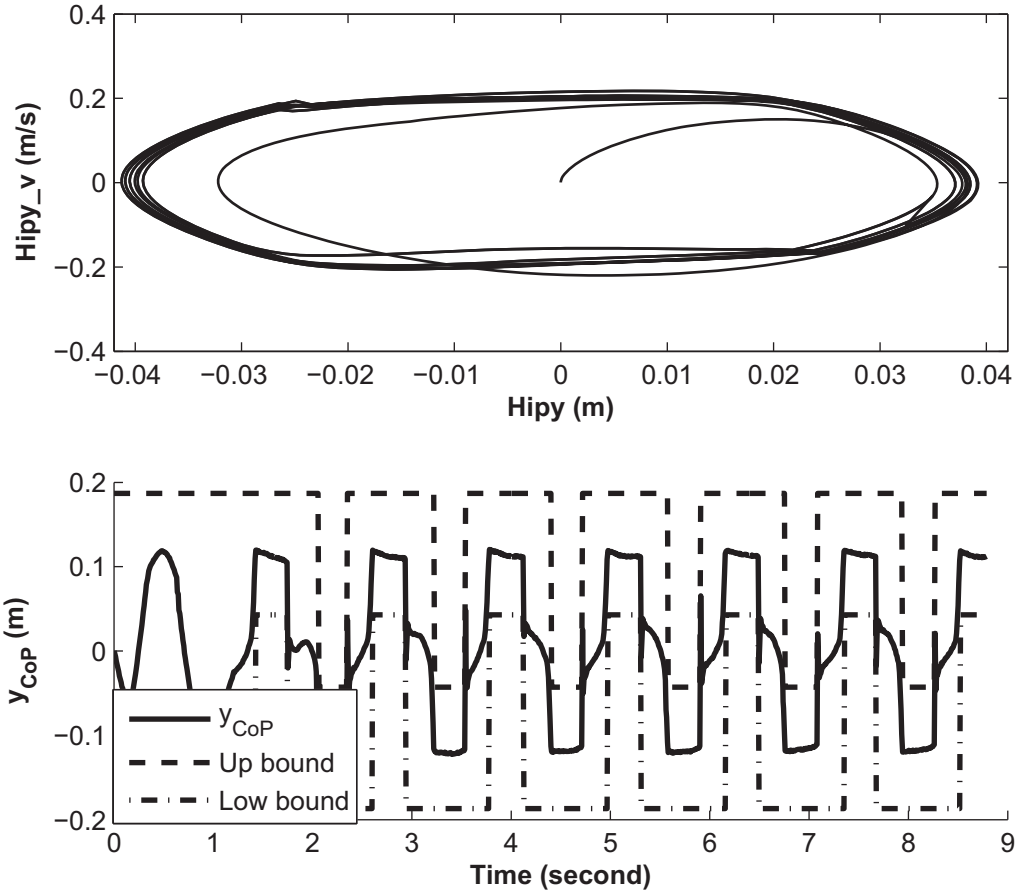


Figure 4.13: Simulation data: the limit cycle behavior of body hip position and velocity in y direction (top); a plot of CoP trajectory (bottom)

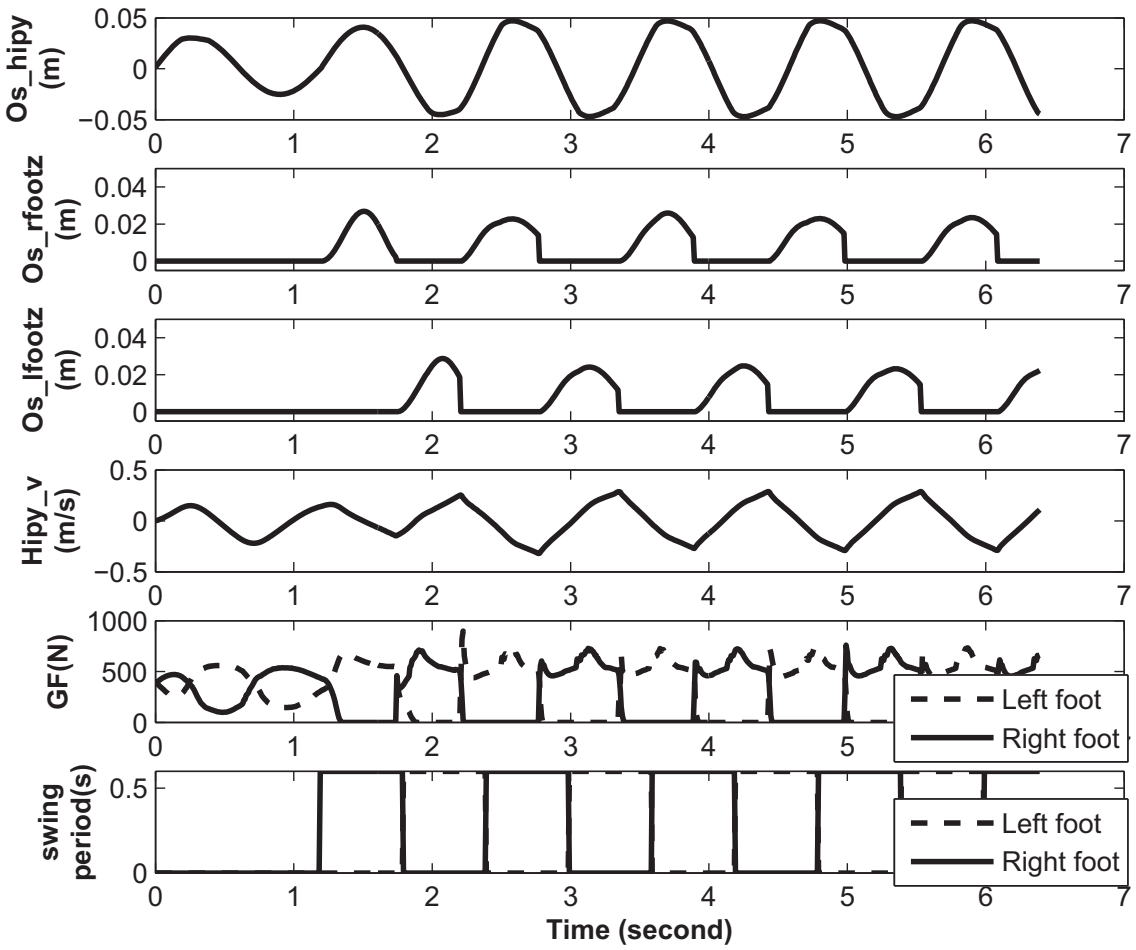


Figure 4.14: Simulation data: Without double supporting phase, the foot touches much earlier than the desired time (top 2 & 3).

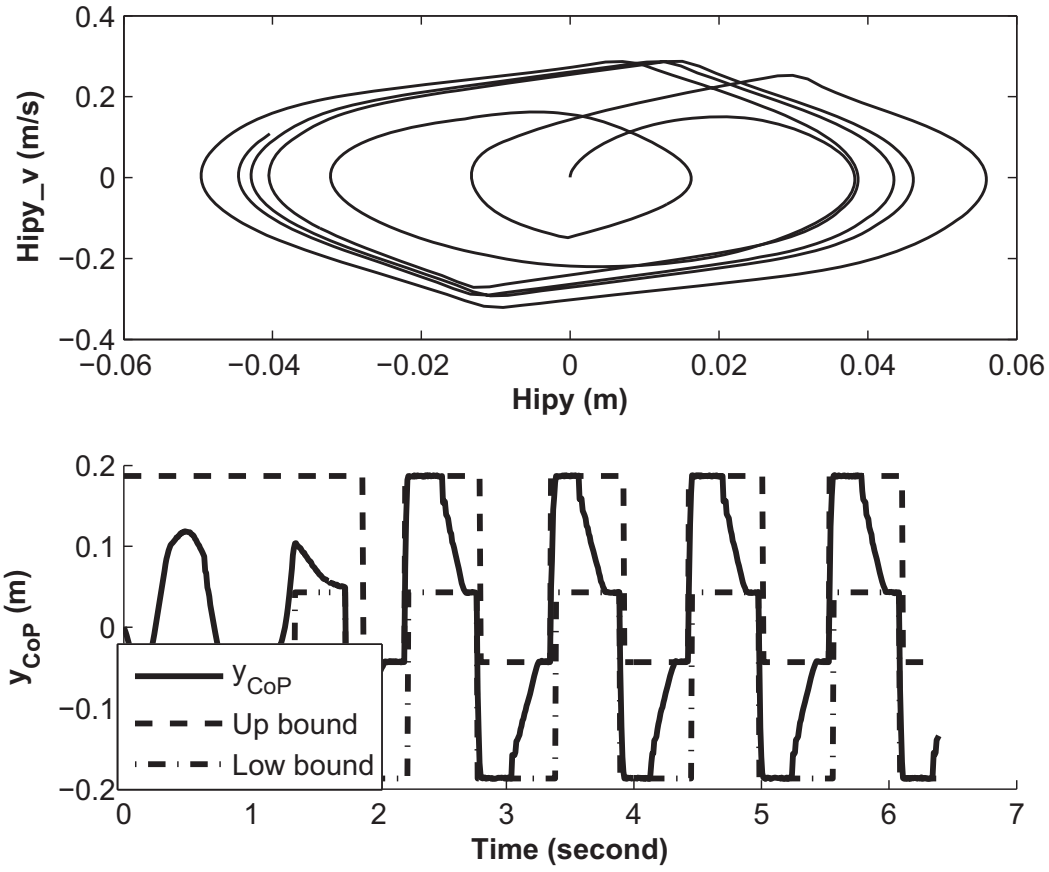


Figure 4.15: Simulation data: the stepping motion does not approach to a limit cycle (top); the CoP location in single support is always on the boundary (bottom)

than other values for this particular frequency.

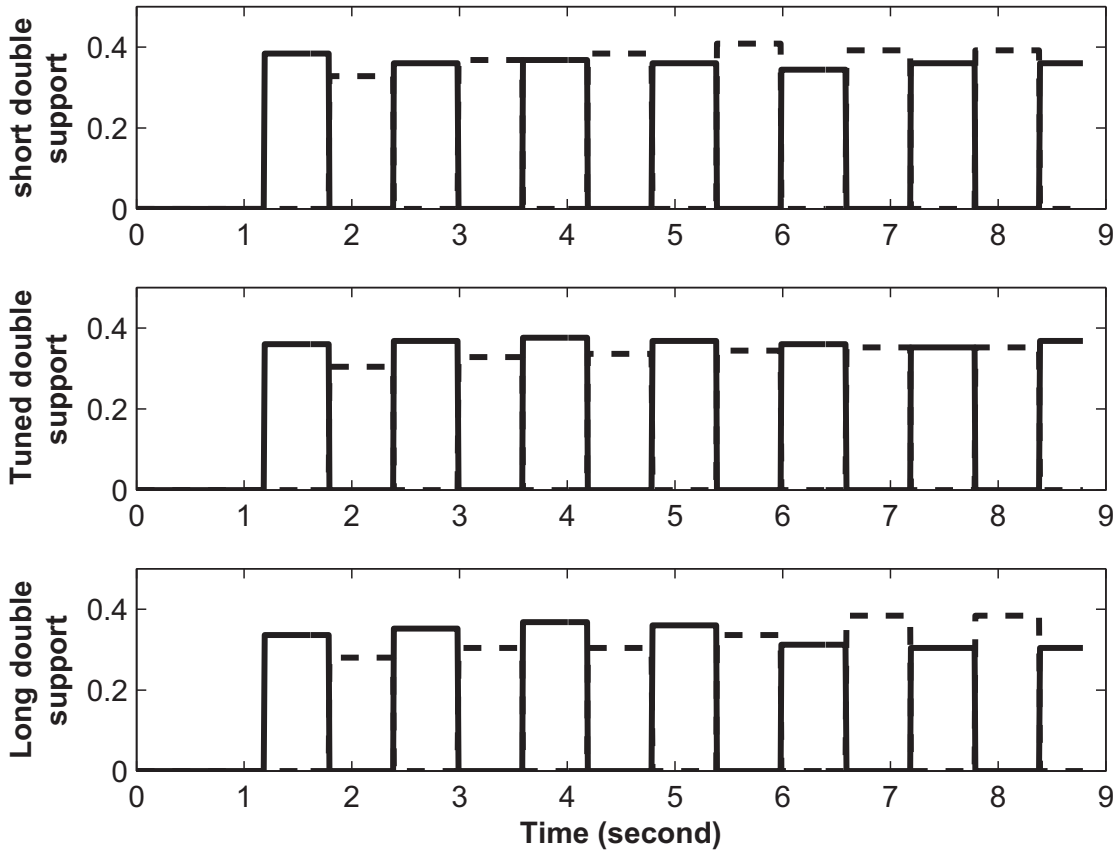


Figure 4.16: Simulation data: a compare of swing period with different CoP triggering value: CoP=0 (top), CoP=0.01(middle), CoP=0.02 (bottom)

4.3.5 Perturbation Test I

To verify the robustness of the controller, we apply an external force on the robot during the stepping motion. As shown in Fig. 4.17, a moving object in the form of a ball hits the side of the robot during its stepping motion. The impact results in a force of 980N applying for an duration of 0.008s on the robot's torso. That is, the resultant impulse added to the system is 7.84Ns. Fig. 4.18 shows that the velocity and the CoP location

of the robot are shifted because of the push (shown by the circle). The change in the body velocity causes a change in the reaction forces acting on both feet. According to the force information, the timing of lifting up the foot is adjusted by the triggering inputs. The period of the corresponding oscillator is also adjusted by the parameter modification inputs as shown in Fig. 4.18 (swing period subplot, dashed circle). These adjustment help the robot balance the motion when the push is applied. After three steps, the stepping motion returns to the normal pattern. The swing period is also converge back to the normal walking period.

We increase the impact force to 1500N on the robot with the same duration of 0.008s. The influence on the robot is more obvious as shown in Fig. 4.19. After the perturbation, the stepping motion can converge back to the normal stepping motion cycle. This CPG controller shows a strong robustness against the external perturbation in the control of stepping motion.

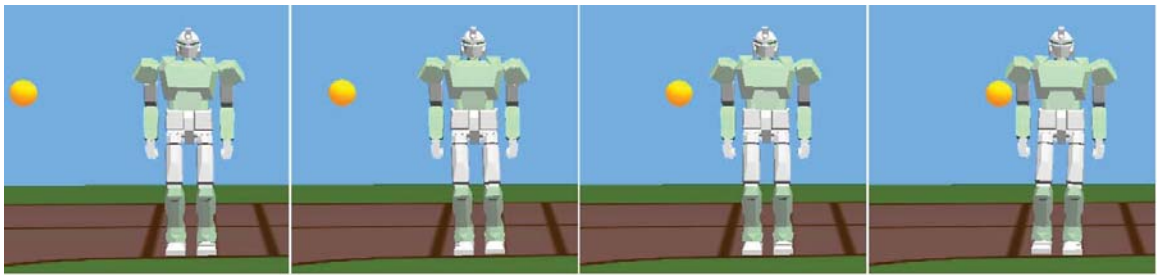


Figure 4.17: An example of push applied on the robot.

4.3.6 Perturbation Test II

In the perturbation test I, based on ground reaction force, the triggering inputs guide the robot to lift up the foot. This results in a robust stepping motion when a push is applied.

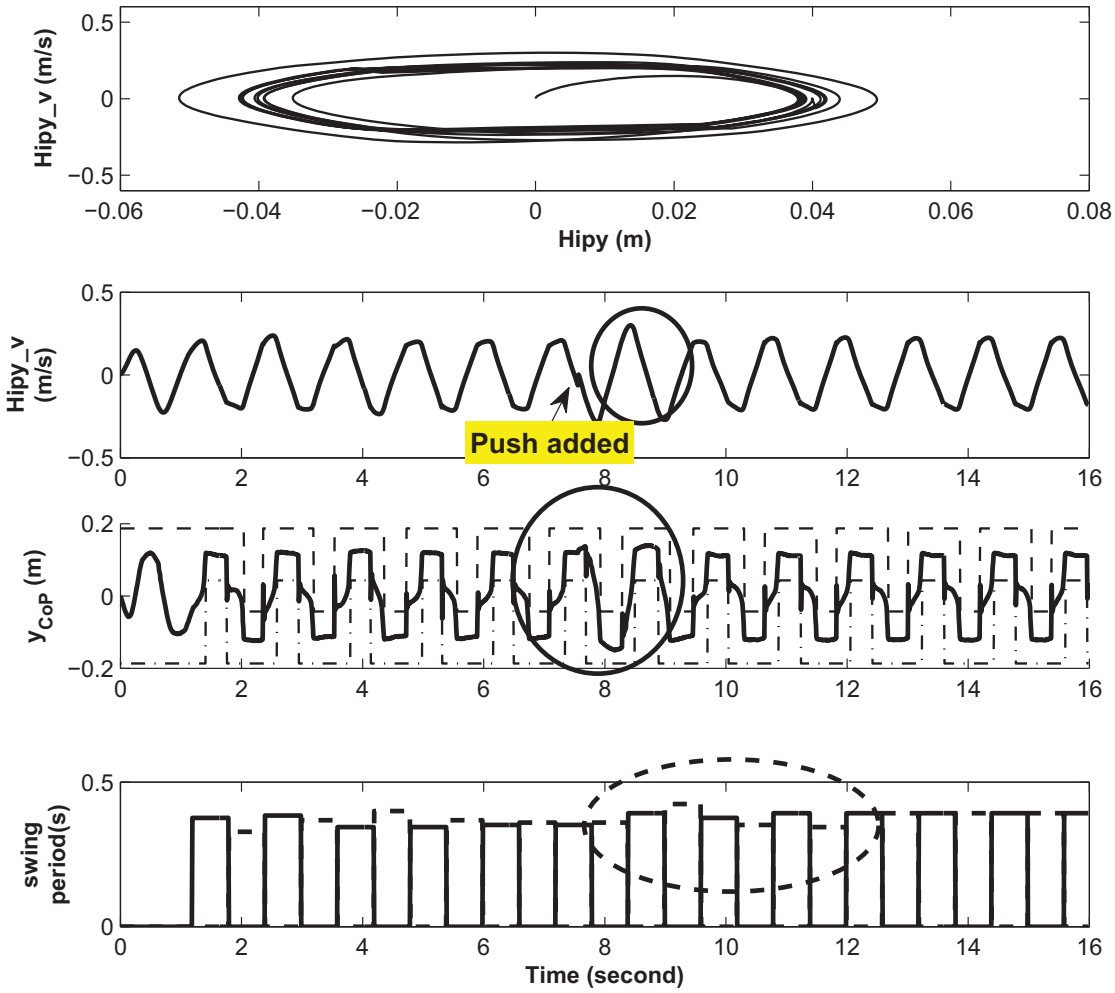


Figure 4.18: Simulation data: the limit cycle behavior of body motion, the body velocity in y direction, CoP trajectory and swing period of both leg (from top to bottom); The circle in the subplot of body velocity shows the velocity changes because of the push; the dashed circle in the subplot of swing period shows the adjustment of swing period during the push

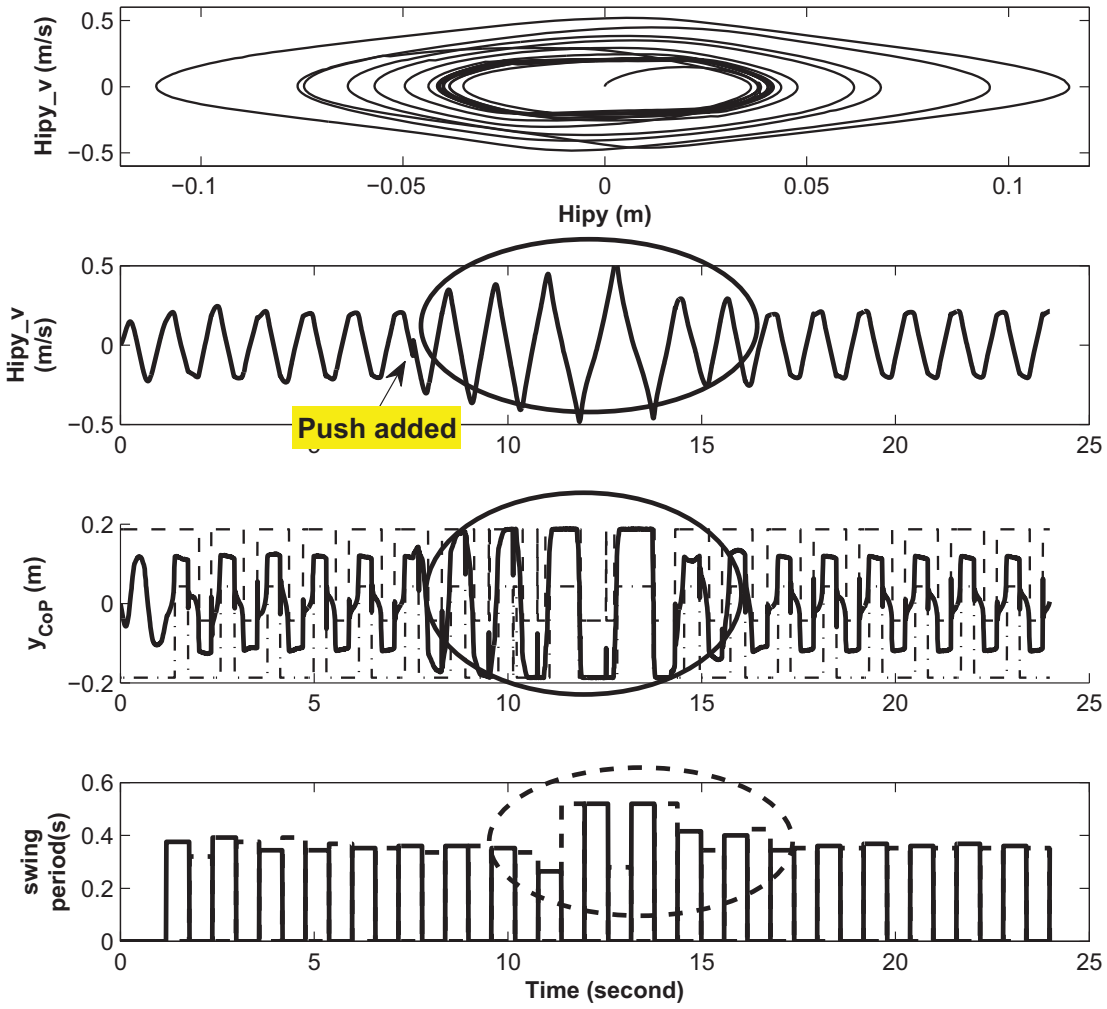


Figure 4.19: Simulation data: a bigger force applied on the robot; the CoP location is almost at boundary of the foot because of the force (circled); the swing period converge back to the normal walking pattern after the perturbation (bottom)

However, when a larger force is applied on the robot, the stability margin of stepping motion is not enough to maintain the stability. In this case, additional motions of the body and the feet are necessary to balance the robot. Typically, when a push is applied from sideway, the robot needs to take a sideways step. Here, we adopt a big change of velocity as an indication of a push. Four strategies for balancing the robot after a push are adopted here:

(1) If the velocity change is bigger than a preset value, a push is detected.

(2) An additional motion is added to the swing foot to balance the robot. The direction of the motion is according to the applied force. The amount of the added motion is dependent on the velocity change.

(3) To avoid premature landing on ground, the additional motion of the swing foot must be completed before the foot touches down.

(4) An additional hip motion with the same amount and direction as the foot motion is added to shift the body following the swing foot.

Because of the triggering input, we can use it to trigger the additional movement during stepping motion. In this thesis, we propose a new oscillator arrangement for CPG design. The main idea is that the normal motion of the walking is controlled by several main oscillators. In the case that additional adjustment is needed, a supplemental oscillator will be activated to give this additional motion. In this arrangement, the design of CPG structure is very straightforward. Each oscillator can be designed to be responsible for a specified motion trajectory.

Three supplemental oscillators are added to provide the addition motions. There are Os_hipyv_ad, $Os\_lfootxv\_ad$ and $Os\_lfootyv\_ad$, which generate the reference velocity trajectories for additional motion of body hip and both feet in y direction, respectively. A push applied on the robot torso will cause a sudden change of body velocity. When this velocity change is bigger than a threshold, it will trigger the activity of these oscillators. Coordination signals are connected between them as shown in Fig. 4.20.

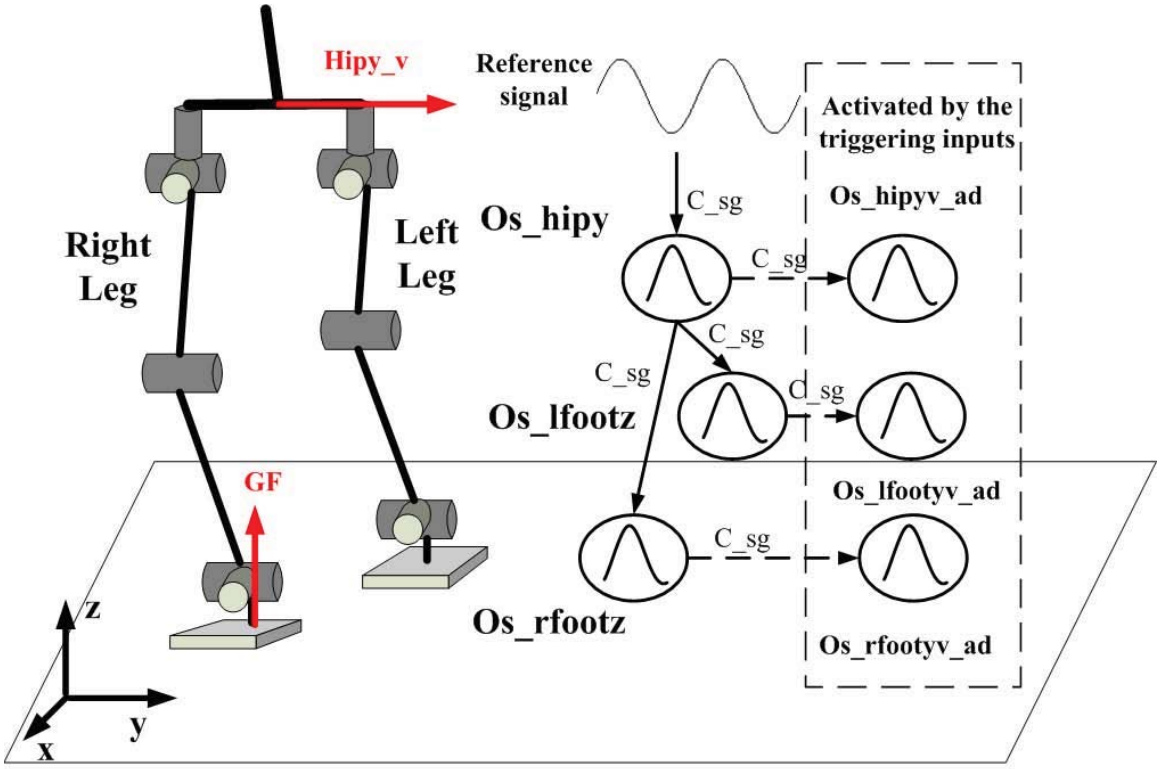


Figure 4.20: The arrangement of oscillators with additional motions when a push is applied; the oscillators in side the dashed line are only activated by the triggering input, the coordination adjustment is also connected when these oscillators are activated

When the push is applied on the side of the robot, we only consider the case that the ball hits the supporting side of the body as shown by Fig. 4.21. When the push is applied on the side of the swing leg, it will cause the swing leg to move towards the supporting leg. Because of the joint configuration design, the robot cannot have a cross leg motion as

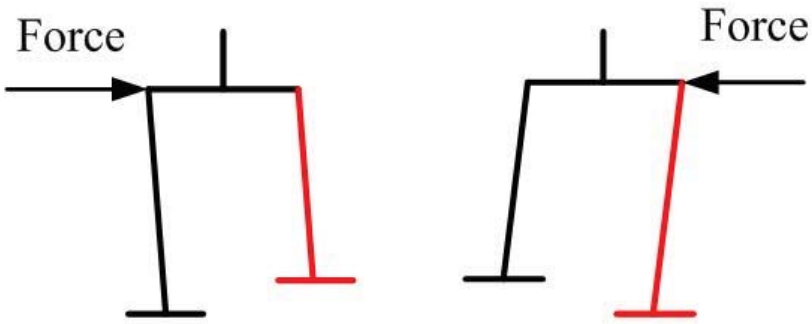


Figure 4.21: The push is applied on the supporting side.

human can. Therefore, we do not give the force on the side of swing leg. Inspired from the idea of capture point [52], we adopt a simple rule to control the amount of additional stepping motion caused by the push:

$$stepy\_ad = 1.5\sqrt{\frac{H_{CoM}}{g}}\delta v_y \tag{4.7}$$

where, $stepy\_ad$ is the amount of additional foot movement which is proportional to the velocity change δv_y in y direction; H_{CoM} is the height of CoM; g is the gravity constant. From the amount of movement, the amplitude of the oscillators which gives the additional reference velocity trajectories can be calculated.

In the simulation, we increase the impact force to 1953N with duration 0.008s on the side of the robot. The impulse is 15.624Ns. Fig. 4.22 shows the oscillators' outputs and the corresponding body velocity in y direction. The velocity of body hip in y direction reduces significantly as shown by cycle line in Fig. 4.22. Since the force is applied when the right foot is swinging, oscillator $Os\_rfootyv\_ad$ is activated. The right foot takes a sideways step on the right. Oscillator Os_hipyv_ad is also activated to move the hip and follow the right foot motion. After right foot touches down, the left foot take one

step right to return back normal stepping motion. The sequence of activation for these three oscillators is shown in Fig. 4.22. Fig. 4.23 shows the limit cycle behavior of this motion. Because of the push, the limit cycle shift to the right side. From the lifting up period, it can be seen that the robot takes one step sideways to reduce the effect of push. After that, another 10 steps are used to return to the normal stepping motion because of the applied push.

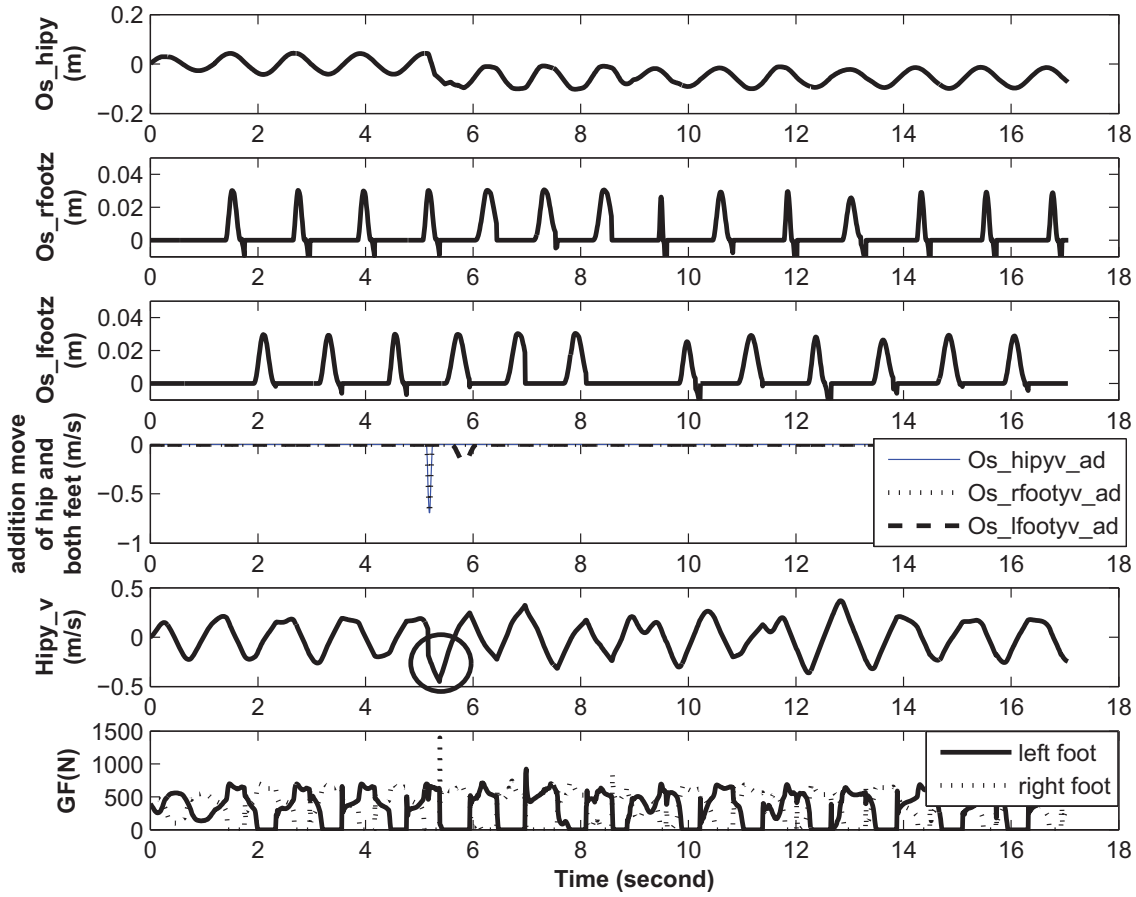


Figure 4.22: Simulation data: the reference output of the oscillators' trajectories, body velocity in y direction and ground reaction force on both feet (from top to bottom); the body velocity changes when the push is applied as shown in the cycle of body velocity subplot; the velocity change triggers the activity of side step motion as shown in subplot of additional move of hip and both feet

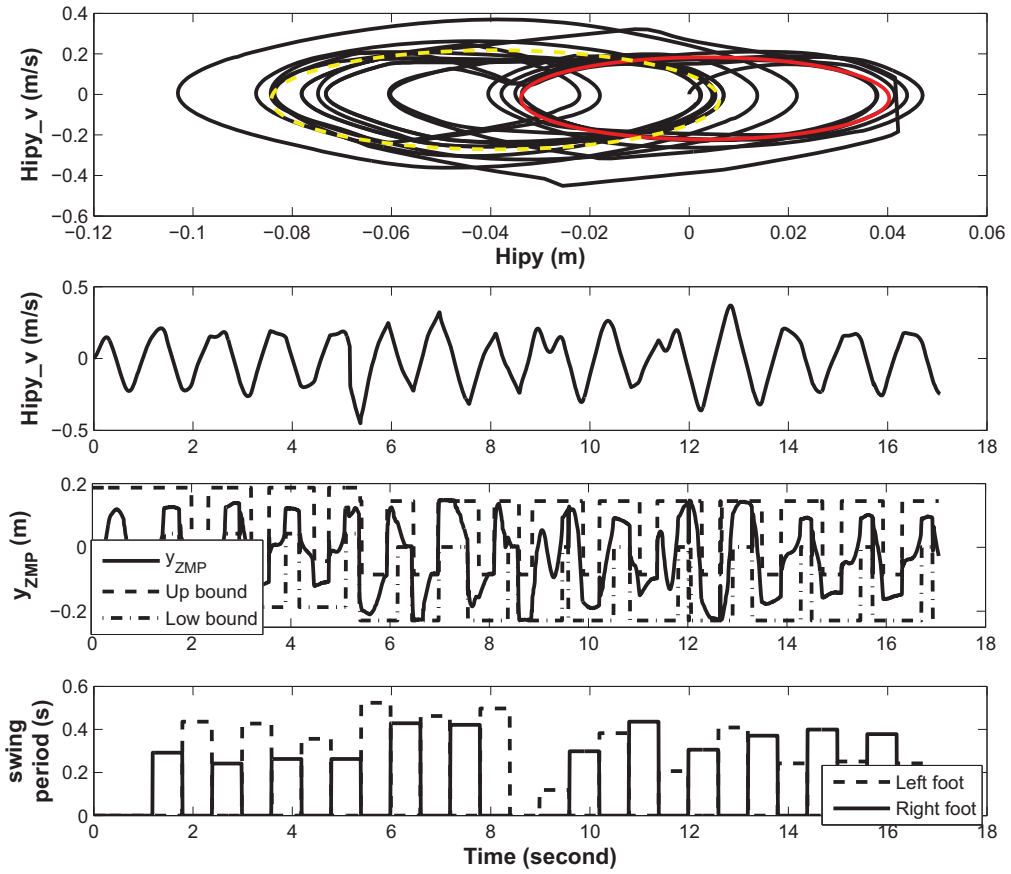


Figure 4.23: Simulation data: the limit cycle behavior of body motion, the body velocity in y direction, ZMP trajectory and swing period of both leg (from top to bottom); the two circle in the subplot of limit cycle trajectory shows two stable motion before and after push

4.3.7 Forward Walking

In the previous section, the robot achieves a stable stepping motion with the sensory feedback design. Here, we further implement the controller for forward walking. To achieve forward walking, we add three more oscillators to control the forward motion of the body and two feet. The arrangement of the oscillators for the forward walking control is shown in Fig. 4.24. Oscillators Os_hipy, Os_lfootz and Os_rfootz generate the same reference trajectory as in the stepping motion. In the simulation, we assign the origin to the midpoint of the two feet in the beginning standing phase. After the robot moves forward, the x direction trajectories of hip and both feet keep on increasing. However, the velocities of these trajectories are almost periodical for each step. In this case, three oscillators: Oscillators Os_hipx_v, Os_lfootx_v and Os_rfootx_v generate the reference velocity trajectories of the body hip and feet in x direction, while the position trajectories of hip and feet in x direction will be the integration of the corresponding oscillators' output. In this implementation, the walking scenario is a level ground walking. The desired motions of the hip in z direction and the feet in y direction are set to be constant. The reference joint angles are then calculated by inverse kinematics.

Fig. 4.25 shows both reference trajectories generated by the oscillators and the actual trajectories of the robot. In the initial standing posture, both feet are parallel to each other. The period of walking cycle is 1.2s. During the initial 1.2s, the robot starts to swap but does not lift up the foot. During 1.2-1.8s, the robot starts the first step. The first step is different from the normal walking cycle, as the initial body hip speed in x direction is 0m/s. In the first step, the maximum speed of the swing leg in x direction

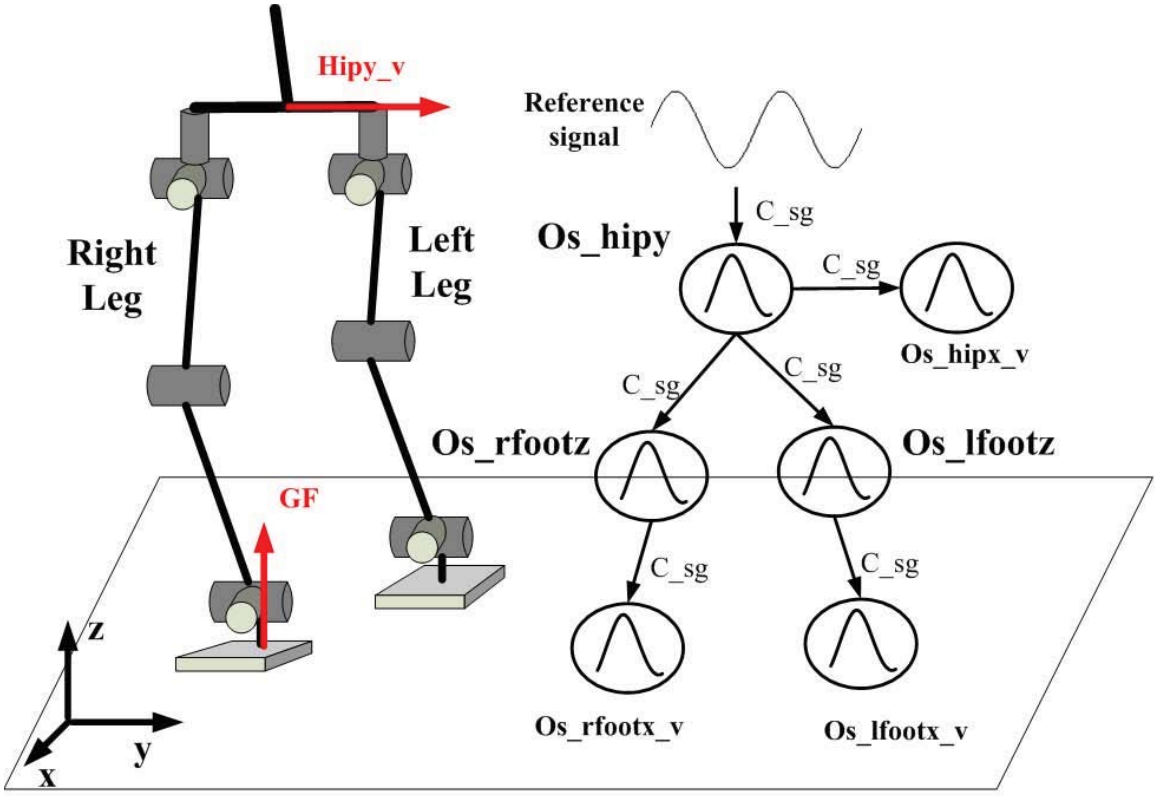


Figure 4.24: The arrangement of oscillators for forward walking; oscillators *Os_hipx_v*, *Os_lfootx_v* and *Os_rfootx_v* provide the trajectories of forward motion

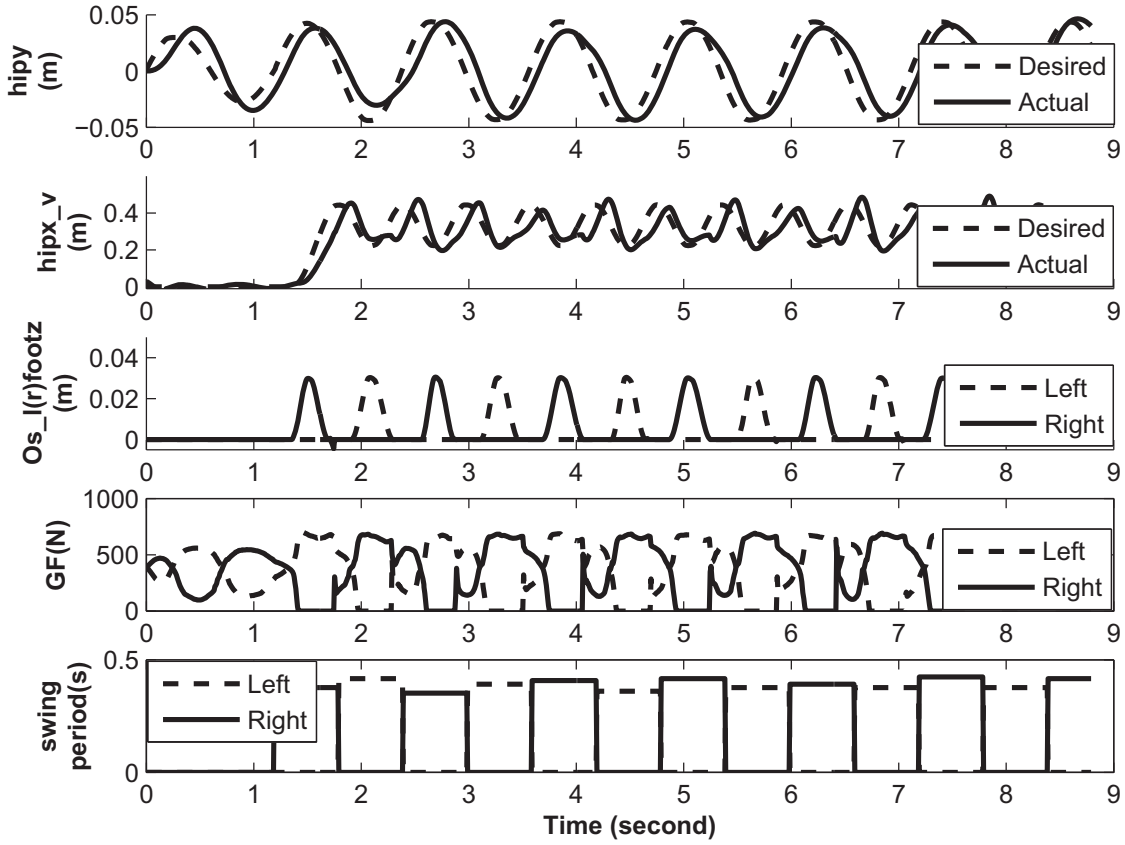


Figure 4.25: Simulation data: the reference trajectories generated by oscillators and the corresponding ground reaction force on both leg; the body hip trajectory is calculated by the accumulation of the body hip velocity in x direction; the swing period converge to a similar value in the forward walking (bottom)

is designed to ensure that the body hip is in the middle of both feet when the swing leg touches the ground. After 1.8s, the robot starts the normal walking cycle. The desired step length is 0.22m. The average walking speed is 0.35m/s. As shown in the Fig. 4.25 the actual trajectory of body hip velocity in X directory is a little different from the desired one. This difference may accumulate and cause the robot to fail in a long distance walk. To remove the accumulated error, a reset of body hip position in X direction is needed after a certain walking distance. The stick diagram of forward walking in sagittal plane in shown in Fig 4.26.

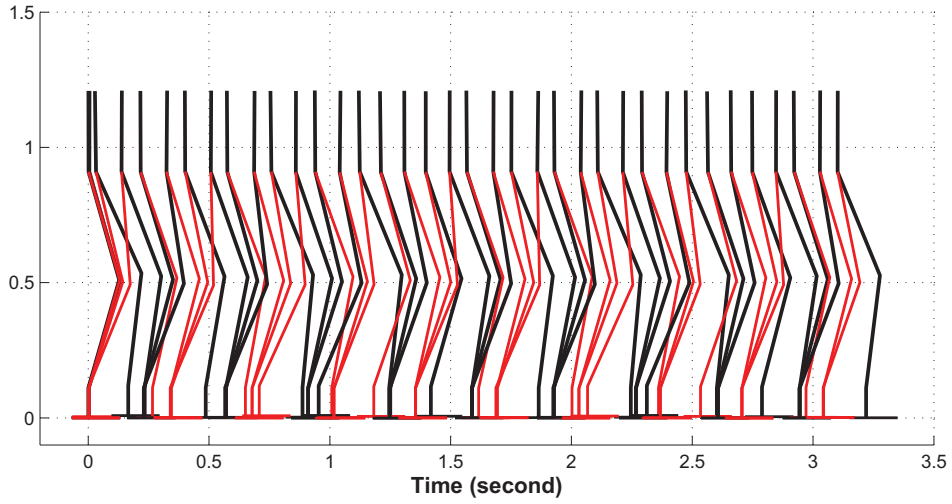


Figure 4.26: A stick diagram (in sagittal plane) of forward walking

4.4 Summary

In this Chapter, the CPG controller was implemented to control a stepping motion of a humanoid robot in 3D dynamic environment. Two new types of sensory input are introduced to help the controller generated the desired trajectories for the 3D walking.

Based on the way of modifying the output of the oscillator, we classify the sensory inputs into three types: inhibition input, triggering input and modification input. The purpose of this classification is to make the feedback design easier. We have explored all these three types of sensory feedback in the stepping motion control. In the simulation, the sensory feedbacks modify the output of the oscillators to balance the robot motion when pushes are applied. The proposed sensory feedback design works well in the control of stepping motion in a 3D environment.

Sensory feedback can modify the output trajectories of the oscillators to balance the robot after push. If the external push is strong, the modification of oscillator's output by sensory feedback may not be enough to overcome the push. Therefore, we propose to use an additional oscillator to provide an additional motion. This oscillator is activated by a triggering input. Also, we propose a new oscillator arrangement method. The main idea is that the normal motion of the walking is controlled by several main oscillators. In the case that an additional adjustment is needed, a supplemental oscillator will be activated to give this additional motion. This arrangement makes the design of CPG structure simple and straightforward.

In both Chapter 3 and this Chapter, we have analyzed the coordination between oscillators and the coordination between CPG and sensory feedback. Coordination between oscillators enables oscillator to respond to the change of other oscillator's output. Coordination between CPG and sensory feedback enables the oscillator response to the change of environment. Stable stepping and walking motions have been achieved in the simulation. In the next Chapter, we will implement the CPG controller on a physical

humanoid robot NUSBIP-III ASLAN. A detail implementation of CPG based controller on this robot will be presented.

Chapter 5

Real Implementation on ASLAN

In Chapter 3, we proposed a coordination connection between oscillators in CPG. In Chapter 4, we implemented this CPG controller in controlling a robot walking in a 3D dynamic simulation environment. Sensory feedbacks were analyzed to enable an adaptive walking behavior to the external pushes. The robot achieved both stable stepping and walking motion in the simulation. In this Chapter, we will describe the implementation of the CPG based walking controller on the physical humanoid robot NUSBIP-III ASLAN. In the first section, we will introduce the hardware platform of our robot. Then, we will present the oscillator arrangement for the control of ASLAN. After that, the CPG based controller used in the simulation is implemented on ASLAN. Two walking scenarios are tested: stepping motion and walking forward.

5.1 Hardware Platform

5.1.1 ASLAN Overview

The robot NUSBIP-III ASLAN has approximate height of 120cm and approximate weight of 60Kg as shown in Fig. 5.1. It consists of 23 actuated rotational joints, 2 cameras and onboard computing devices. Thirteen of the joints are the most relevant for walking: six in each leg and one in the waist for the yaw motion. The neck which holds the articulated camera has 2 DoFs. Fig. 5.2 shows the kinematic configuration of the robot. The actuator information of each joint is shown in Table 5.1.

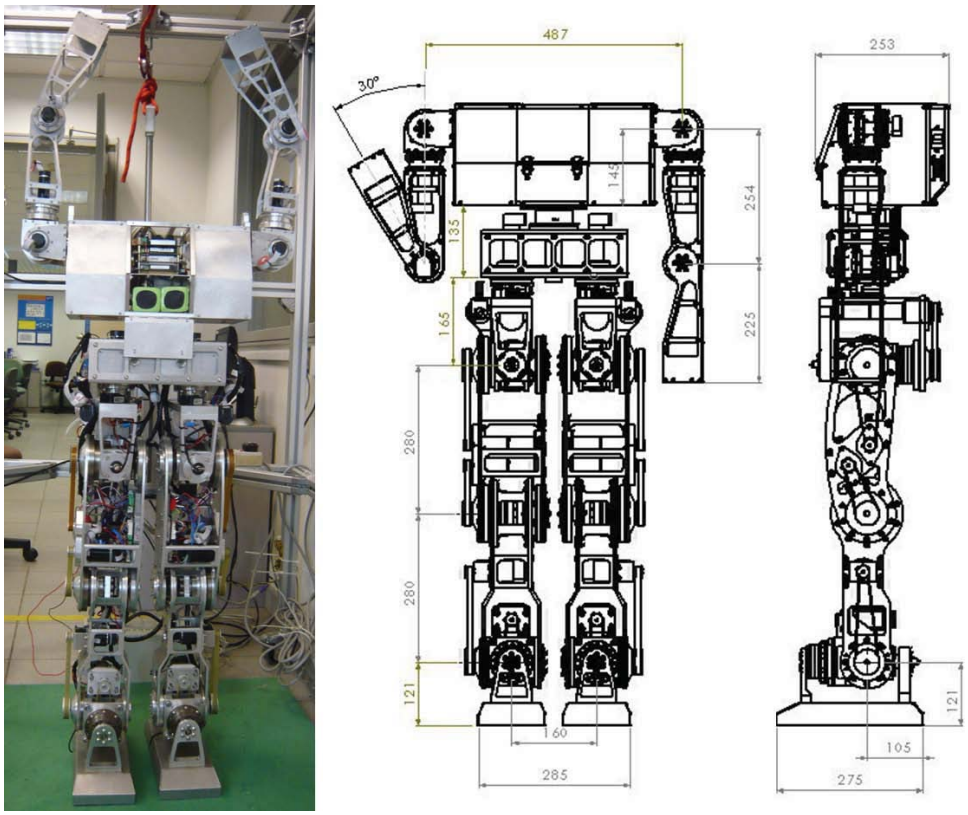
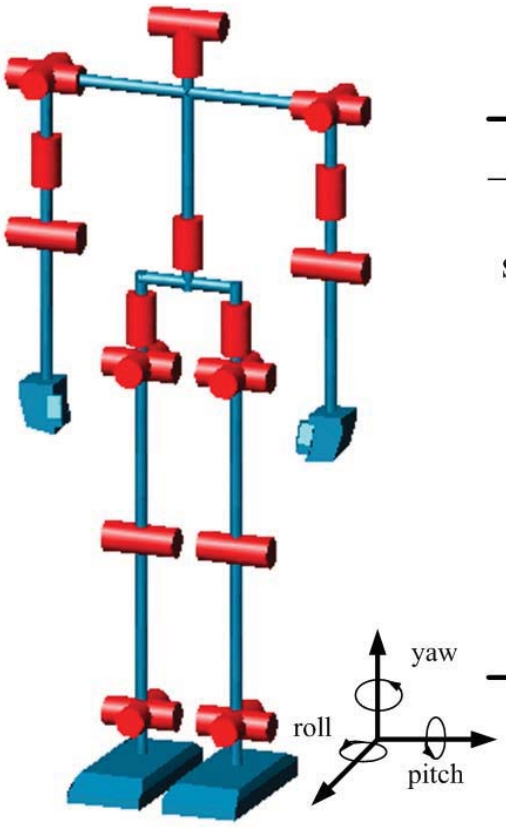


Figure 5.1: Photograph and technical layout of the 23-DoF humanoid walking robot ASLAN



Joint	DoF
Head	2
Shoulder	3
Elbow	1
Pelvis	1
Hip	3
Knee	1
Ankle	2
Total	23

Figure 5.2: Kinematic structure of ASLAN

Table 5.1: Gear sizes, reduction ratios and joint working ranges HD: harmonic drive; PB: pulley belt; GH: gear head; BM: brush motor; BLM: brushless motor

Joint		Reduction Gear	Input Gear	Motor Power (W)	Workspace (deg)
Hip	Pitch	HD 160:1	PB 3:1	200 BLM	$-135 \sim 45$
	Roll	HD 160:1	PB+GH 20:1	200 BLM	$-20 \sim 45$
	Yaw	HD 120:1	PB 1:1	60 BM	$-45 \sim 45$
Knee	Pitch	HD 160:1	PB 2.3:1	200 BM	$-10 \sim 140$
Ankle	Pitch	HD 120:1	PB 2.3:1	200 BM	$-20 \sim 45$
	Roll	HD 120:1	PB 2:1	200 BM	$-60 \sim 42$
Pelvis	Yaw	HD 120:1	PB 1:1	60 BM	$-60 \sim 42$
Shoulder	Pitch	HD 120:1	PB 1:1	20 BM	$-180 \sim 180$
	Roll	HD 120:1	PB 1:1	20 BM	$-15 \sim 195$
	Yaw	HD 120:1	PB 1:1	20 BM	$-180 \sim 180$
Elbow	Pitch	HD 120:1	PB 1:1	20 BM	$0 \sim 150$

5.1.2 Control System

The main processor of ASLAN is a PC/104. All the control algorithms are run on this board. The amplifier used for ASLAN is ELMO which supports CAN-bus communication. All the ELMO amplifiers are connected to a CAN-bus card through CAN BUS. The CAN-bus card stacks on the PC/104 card through PC/104 BUS. In this case, the motion command is sent through CAN-bus. To get the sensory feedbacks, a DAQ card is used to read sensory signals. The absolute encoders, accelerometer and force/torque sensors are connected to the DAQ card. The gyro sensors are directly connected to PC/104 through RS232. The control structure is shown in Fig. 5.3.

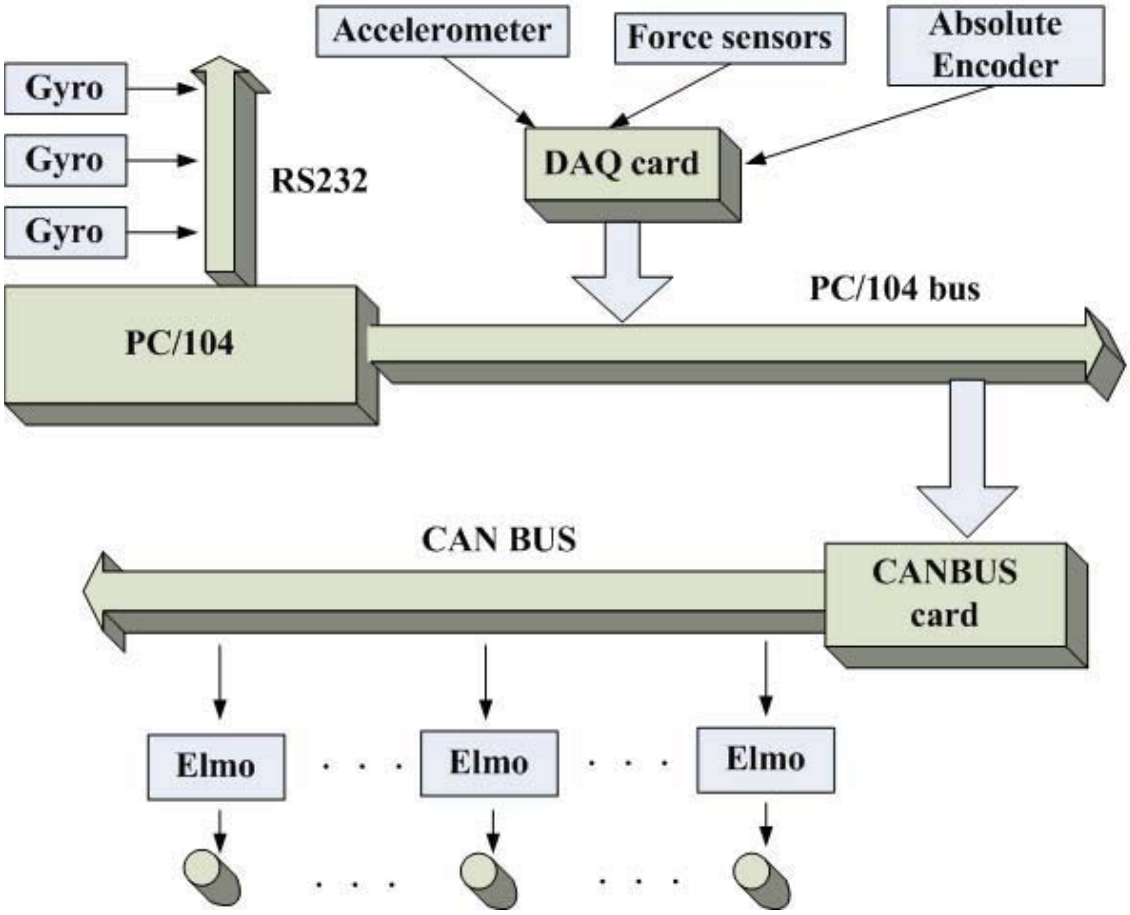


Figure 5.3: Control structure of ASLAN

5.2 Oscillator Arrangement

A similar CPG structure as used in the simulation is used to control the walking of ASLAN. The oscillator arrangement is shown in Fig. 5.4. The robot is controlled by oscillators: Os_hipy, Os_lfootz , Os_rfootz, Os_hipx_v, Os_lfootx_v and Os_rfootx_v. This arrangement is the same as the oscillator arrangement in the forward walking simulation.

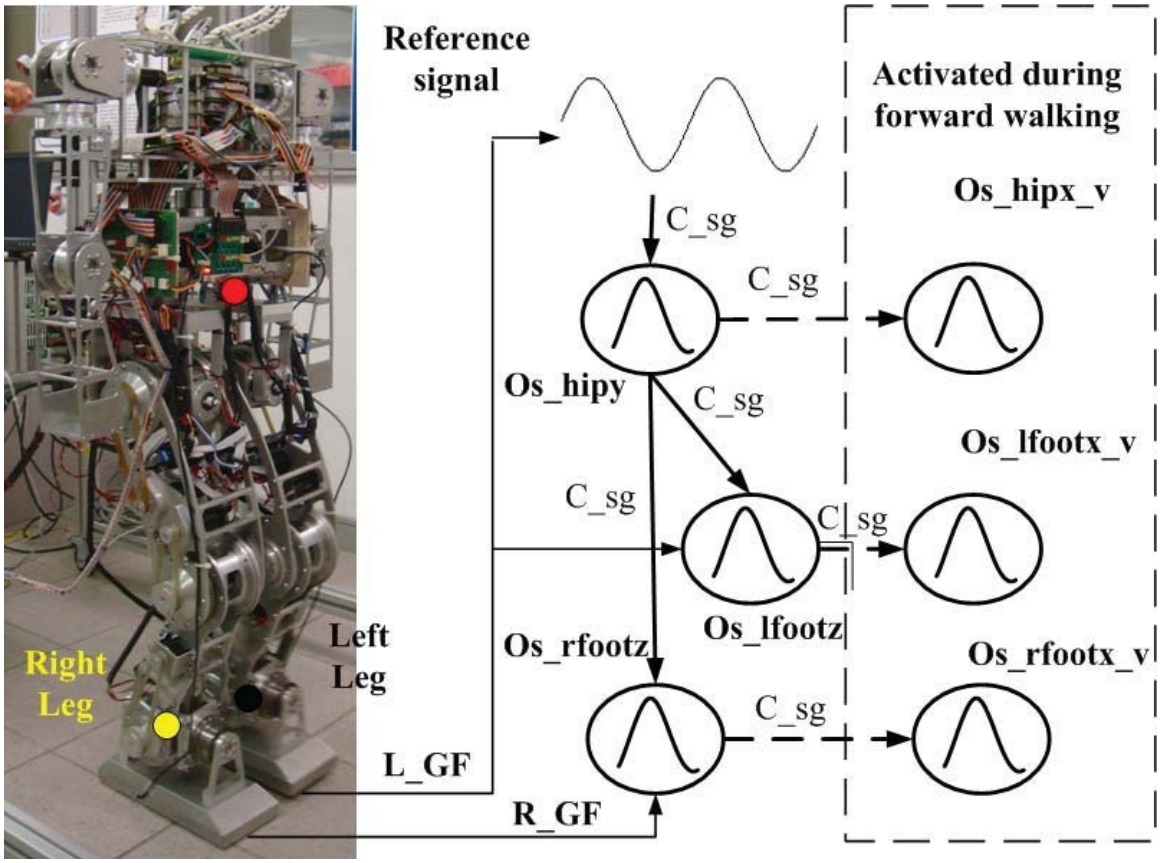


Figure 5.4: The arrangement of oscillators to control the walking of ASLAN

One difference in CPG structure between simulation and real implementation is the sensory feedback. Currently, the accelerometer and gyros have not been mounted on the robot yet. The available sensory information is the ground reaction force from the

force/torque sensors on both feet. The force feedback from the right foot is used to activate or suspend oscillators Os_rfootz and Os_rfootx_v. The force feedback from the left foot is used to control oscillators Os_lfootz and Os_lfootx_v. In this walking control, the trajectories are chosen to lift the right foot first and then the left foot. In this case, the touching down timing of the left foot is used to indicate the end of one walking cycle. When the left foot touches down, the timing of the reference signal will be updated.

This CPG structure is designed for walking on level ground. When the robot performs a stepping motion, the forward motion will be canceled. As shown in Fig. 5.4 the oscillators inside the red dashed line will not be activated.

Table 5.2 shows all the parameter values of the oscillators in CPG controller. In the walking control, the swing period of the foot trajectory is related with the timing of foot lifting. This has been analyzed in the subsection of Chapter 4 entitled "Arrangement of Oscillator and Sensory Feedback". The equations for the time constant τ_2 are given by:

$$\begin{cases} \tau_{2\_rfz} = \tau_{2\_rfx} = \frac{T_l}{2.96\pi} & \text{left support} \\ \tau_{2\_lfz} = \tau_{2\_lfx} = \frac{T_l}{2.96\pi} & \text{right support} \end{cases} \tag{5.1}$$

The equations for the constant stimulus c of oscillators Os_lfootx_v and Os_rfootx_v

Table 5.2: The values of oscillators'parameters in CPG controller

	a	β	c	τ_1	τ_2	K
Os_hipy	2.5	2.5	0.12	0.232	0.157	0.3
Os_lfootz	2.5	2.5	0.06	$1.48\tau_{2\_lfz}$	τ_{2_lfz}	0.1
Os_rfootz	2.5	2.5	0.06	$1.48\tau_{2\_rfz}$	τ_{2_rfz}	0.1
Os_hipx_v	2.5	2.5	0.12	0.116	0.0785	0.05
Os_lfootx_v	2.5	2.5	c_{lfx}	$1.48\tau_{2\_lfx}$	τ_{2_lfx}	0.3
Os_rfootx_v	2.5	2.5	c_{rfx}	$1.48\tau_{2\_rfx}$	τ_{2_rfx}	0.3

are given by:

$$\begin{cases} c_{rfx} = \frac{T_l c_{ref}}{0.25T} & \text{left support} \\ c_{lfx} = \frac{T_l c_{ref}}{0.25T} & \text{right support} \end{cases} \tag{5.2}$$

T_l is the period of the oscillator given by Equation 4.6. c_{ref} is the desired stimulus when the oscillator period is one quarter of the whole walking cycle period. The purpose is to make the step length a constant.

K is the scaling factor to modify the speed of coordination adjustment as shown in Fig. 3.2.

5.3 CPG Based Stepping Motion

In Chapter 4, we used a CPG controller to control the stepping motion of a simulated robot. Here, we implement this controller to control the stepping motion of ASLAN.

Fig. 5.5 shows the system architecture of the CPG controller. Before sending reference trajectories to the drive units, the trajectories are modified by a compensator. The pur-

pose of adding the compensator is to overcome the difference between simulation and real hardware. Due to the limit power of the DC actuator, the hip roll joint had problem to lift up the swing leg by following the reference joint trajectories. To compensate for this limitation, a sinusoidal wave is added on the hip roll trajectory of the support leg for every step. This compensator is only applied on the hip roll joints. Other joints will not be affected. This is to help the robot lift up the foot. A local PID control loop before the drive unit enables the joints to follow the reference trajectories. The ground reaction force value is detected by the force/torque sensors and input to the oscillators.

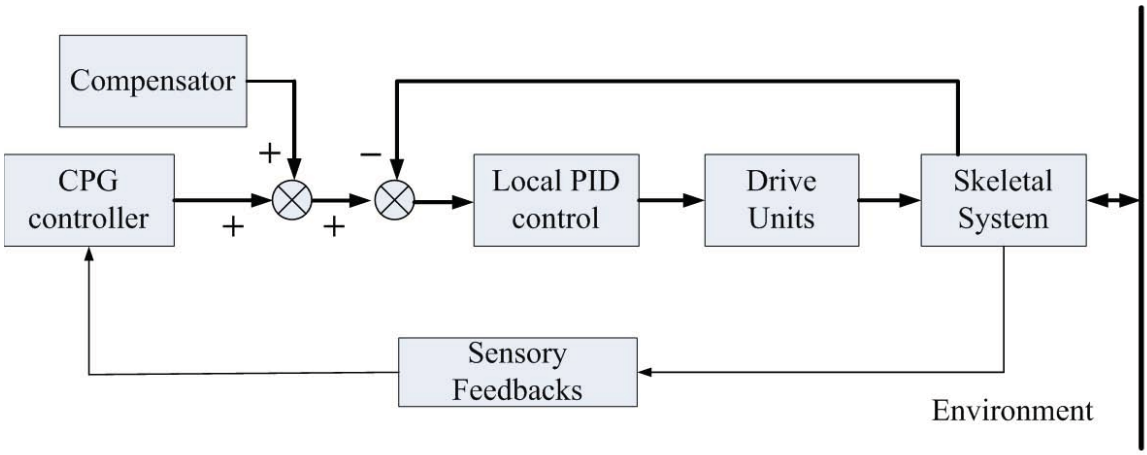


Figure 5.5: System architecture of CPG based walking control of ASLAN

In the experiment, the period of stepping motion is 1.2s; the height of the CoM is designed to be constant at 0.63m in height; the position of CoM and both feet in sagittal plan is designed to 0. The CPG controller generates the reference trajectories for CoM in frontal plan and both feet in height. The experimental data are shown in Fig. 5.6. In the first cycle, the robot starts to swap but does not lift the foot. In the second cycle, it starts stepping. As indicated by the output of Os_rfootz in Fig. 5.6, the right foot touches down a little late in the first two step. Because of this delay, the swing period

of the left foot is reduced to follow the hip motion. After two steps, the swing period of both feet become stable. The reference joint's trajectories are calculated according to the Cartesian space trajectory by inverse kinematics. Fig. 5.6 shows the reference joint trajectories of left leg during the stepping motion. The hip roll joint angle is modified by the compensator as indicated in the red circle. This modification helps the robot to lift up the heavy foot.

In this experiment, the final swing period of both feet is still not the same. Since the motion becomes stable, the adjustment of swing time is stopped. Fig. 5.8 shows a sequence of snapshots of the stepping motion by ASLAN.

5.4 CPG Based Level Ground Walking

After achieving a stable stepping motion, we test forward walking by including the forward motion. The oscillators inside the dashed line in Fig. 5.4 will be activated.

Fig. 5.9 shows the reference trajectories generated by the oscillators. In the initial standing posture, both feet are parallel to each other. The period of the walking cycle is 1.5s. During 0-1.5s the robot starts to swap but does not lift the trailing leg. The robot starts the first step after 1.5s. The first step is designed for the same purpose as in the forward walking simulation. In the first step, the maximum speed of the swing leg in the x direction is designed to make the body hip stay in the middle of the two feet when the swing leg touches the ground. According to the ground reaction force feedback, the robot adjusts the timing of foot lifting. After several steps, the swing time of both

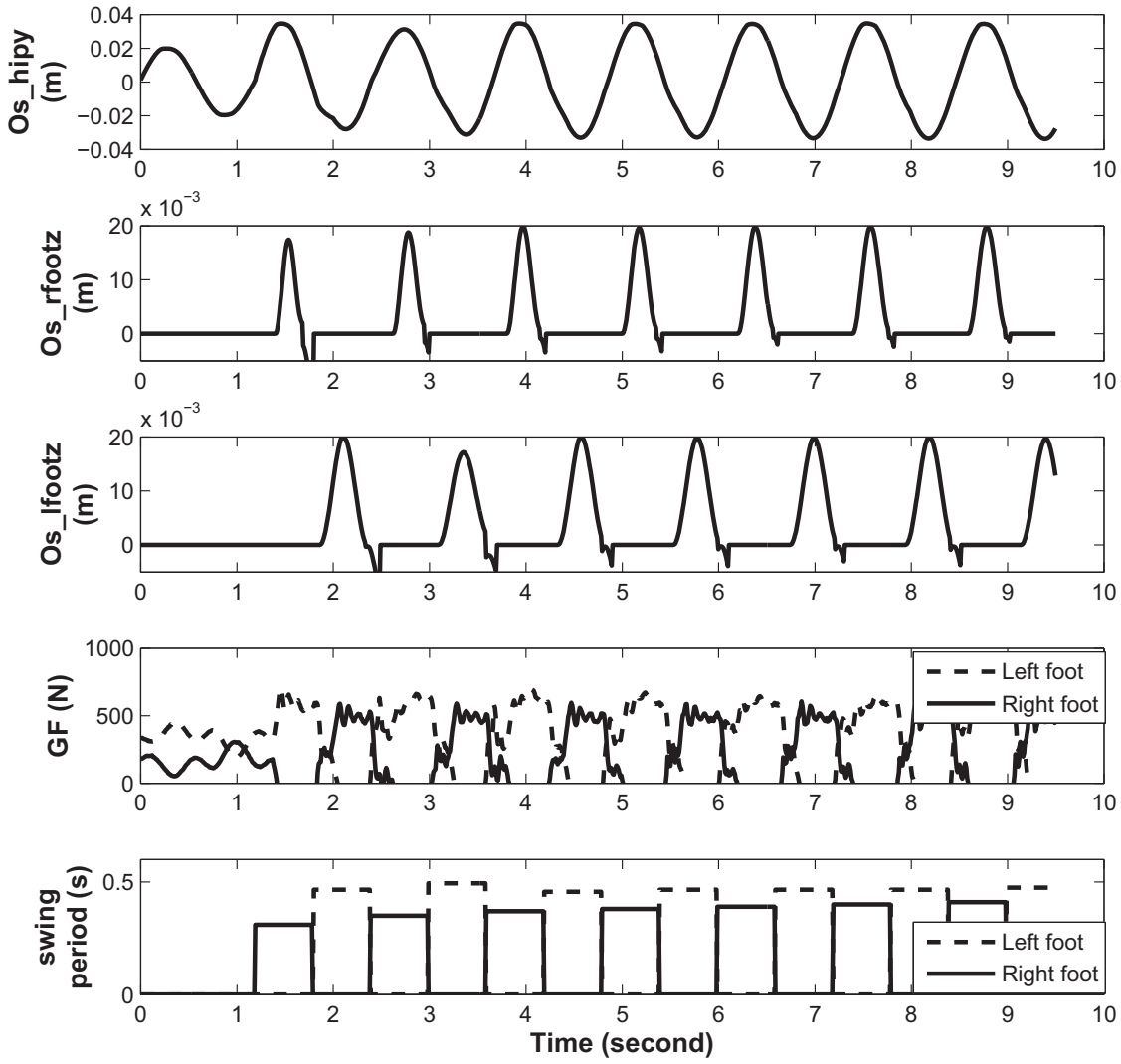


Figure 5.6: Experimental data: the reference trajectories generated by the oscillators, the ground reaction force on both feet and the swing time in each step

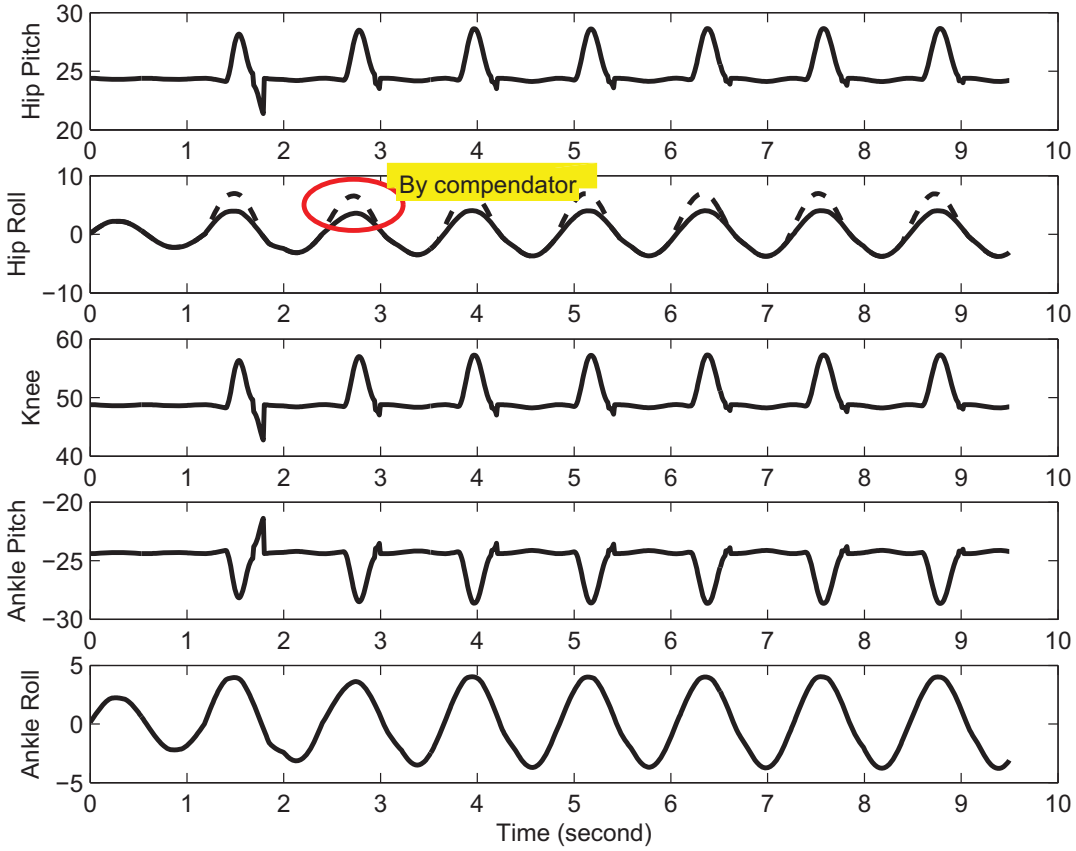


Figure 5.7: Experimental data: the reference joint trajectories calculated by inverse kinematics, the red circle indicate the modification of hip roll joint by the compensator.

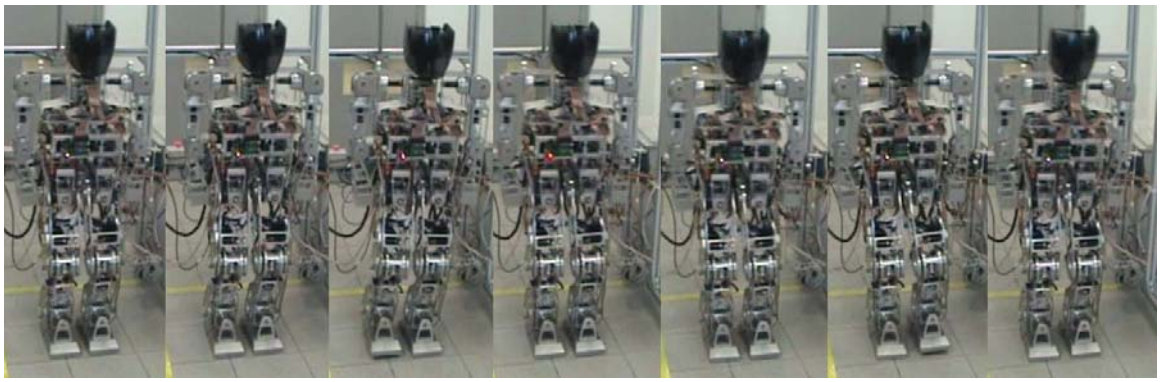


Figure 5.8: Snapshots of the stepping motion by ASLAN

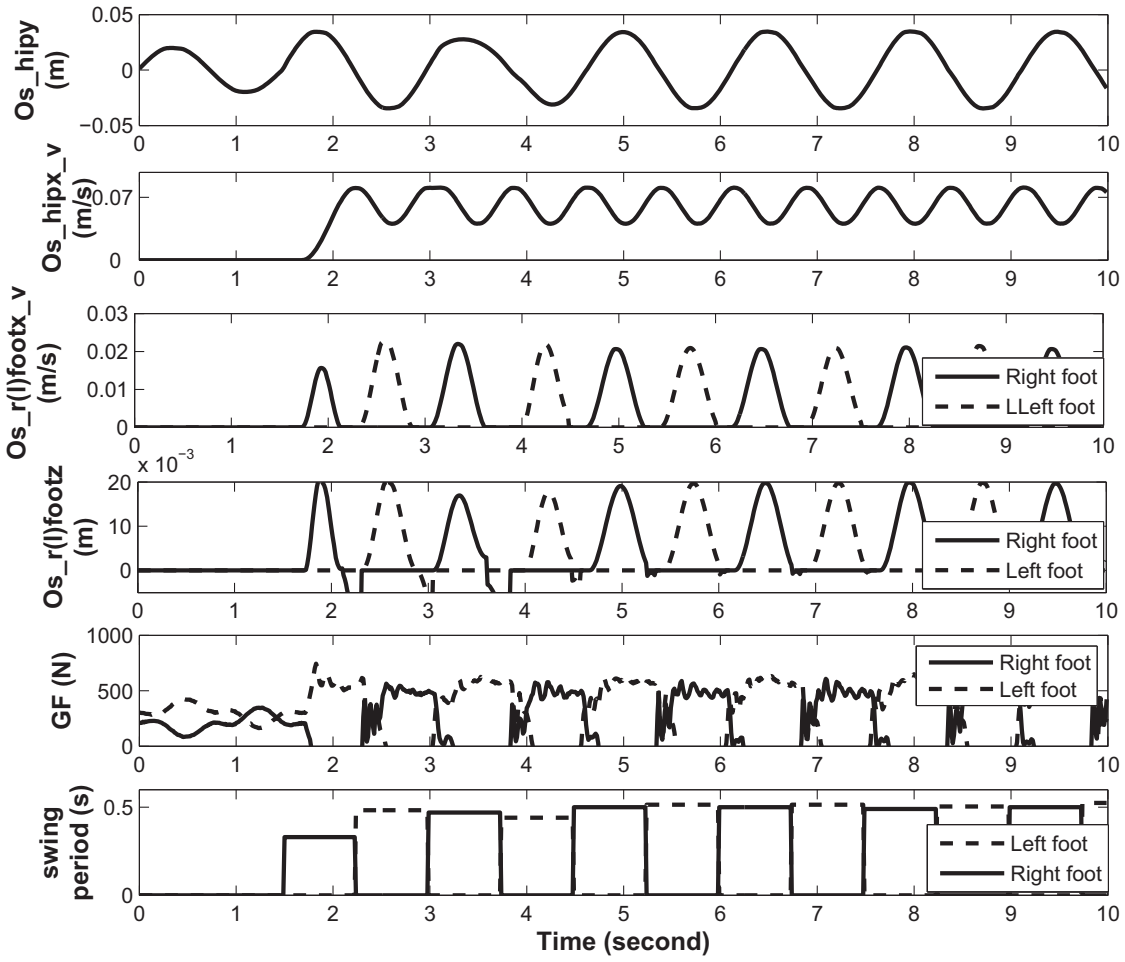


Figure 5.9: Experimental data: the reference trajectories generated by the oscillators, the ground reaction force on both feet and the swing time in each step

feet reaches a stable state. Fig. 5.9 shows the swing period during walking. Fig. 5.10 displays the snapshots of one cycle of level ground walking by ASLAN. The step length is 0.06 m. In this experiment, the distance of the walking is about 1.2m.

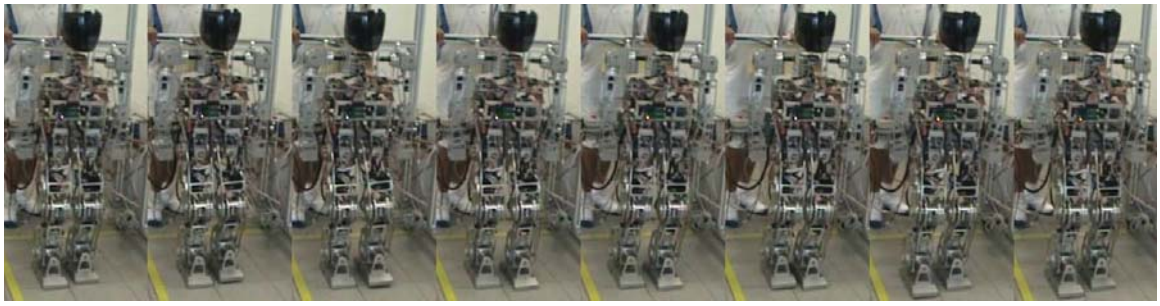


Figure 5.10: Snapshots of one cycle of level ground walking

5.5 Summary

We have implemented the CPG based controller on ASLAN. Two basic motions which have been successfully achieved in simulation are tested. The robot achieves stable stepping motion and forward walking. The proposed CPG based controller is successfully implemented on the physical humanoid robot.

A real implementation usually faces many problems such as lack of torque output, friction with ground, lack of sensory feedbacks. Because of these problems, implementing on real robot is much more challenging than doing in the simulation. As for ASLAN, its foot is very heavy. We have to add a hip roll compensator to help the robot lift the trailing leg. Also, because of the heavy foot, it is very difficult to achieve a large step length walking. In this walking control, we have only tested two simple motions. To better verify the controller and further improve it, more motions with different environ-

ments need to be tested. This requires more sensory feedback to know the environment.

More sensors such as accelerometers and gyros are needed to be mounted on the robot.

Chapter 6

Conclusion

The main objective of this thesis is to analyze the coordination between oscillators in the CPG based walking control. In particular, we analyze the oscillator model proposed by Matsuoka and investigate how the coordination between oscillators contribute to a robust walking behavior. In the Bio-inspired CPG approach, a controller is usually composed of many oscillators to generate reference trajectories. In the current CPG research, much effort has been put in the design and optimization of the oscillator model. However, few research work investigates the connection between oscillators in the control. In the CPG based control, oscillators work together to generate the reference trajectories for the robot. Therefore, the coordination between oscillators is very critical, especially in the case of environment change.

The method of coordination in this thesis is based on the entrainment property of the neural oscillator. In detail, the difference between desired output and real output is used

to adjust the oscillator's output. The desired output is generated by a main oscillator. With this connection, the main oscillator can adjust the output of other oscillators and maintain a certain phase relationship between them. The proposed method can also handle the case that the main oscillator and the target oscillator have different frequencies and phase relationship.

To achieve an adaptive behavior, the robot needs to interact with the environment. Sensory feedbacks provide the environment information to the robot. In this thesis, the sensory feedback are analyzed to help the CPG controller to be implemented in 3D environment. Two new types of sensory inputs are introduced to modify the output of the oscillator for the 3D dynamic walking implementation. There are triggering input and parameter modification input. Together with inhibition input, these three types of sensory input are used for the feedback design. With the proposed sensory feedback design, a stable stepping motion was achieved in the simulation. External perturbations are applied on the robot to verify the robustness of the controller. The simulation results show that the coordination connection and sensory feedback in CPG controller greatly help the robot balance itself when external pushes are applied.

The proposed CPG controller was also implemented on the humanoid robot NUSBIP-III ASLAN. ASLAN was developed as a platform to test various walking algorithms. With the coordination between oscillators, the CPG controller achieves a stable stepping and walking motion for ASLAN.

6.1 Summary of Research Contribution

The main research contributions reported in this thesis are summarized as below.

Coordination between Neural Oscillators

This thesis analyzes the importance of coordination between oscillators in the CPG based walking control. A connection between oscillators is developed to explore the inherent coordination between oscillators in CPG. This connection enables the oscillators to coordinate with each other. Dynamic simulation results show that the CPG controller with coordination between oscillators is more robust than the one without coordination in the control of robot walking. This can be attributed to the fact that oscillators synchronize their outputs with other oscillators when external pushes are applied on the robot. This connection can also guide the robot to return back to normal walking pattern after the pushes.

Sensory feedback design

Sensory feedbacks play an important role in achieving an adaptable walking with external perturbation. In this thesis, two new types of sensory inputs are introduced to implement the CPG controller in 3D walking. There are triggering input and parameter modification input. In the 3D walking, a double support phase is needed during the walking. The triggering input enables the oscillator to suspend its output trajectory during the double support phase. The activity of this oscillator will be activated again when the robot shifts to single support phase. Parameter modification input provides the

controller a simple way to modify the oscillation pattern like frequency or amplitude.

To design the triggering input, we analyze the timing issue of foot lifting during the walking. A correct timing can avoid a sudden change of ground reaction force on supporting leg. It also helps to reduce the torque output to lift up the trailing leg. In our study, the ground reaction force values on both feet are used to indicate when to lift up the foot. When the force value on the supporting foot is larger than a preset value, a triggering input will be sent to activate the oscillator to generate foot trajectory. With the proposed sensory feedback design, the CPG controller achieves a stable stepping motion control in the simulation.

Oscillator arrangement in CPG

In this thesis, we propose a way of oscillator arrangement for CPG structure design. In the design, the standard motion's trajectories are generated by several main oscillators. If an additional adjustment is needed, a supplemental oscillator will be activated to give this additional motion. This supplemental oscillator is activated by triggering inputs. In this arrangement, the design of CPG structure is very straightforward. Each oscillator is responsible for a specified motion trajectory.

Implementation of the CPG Controller on a 3D Physical Robot

This thesis also introduces the description of a humanoid robot NUSBIP-III ASLAN. ASLAN is a humanoid robot developed in National University of Singapore which serves as a platform to test various walking algorithms. Compared with previous design NUSBIP-II, the new robot has been improved in many areas such as control system,

actuators and sensor feedbacks. The target is to make it a fully autonomous humanoid robot that can walk in human environment. The proposed CPG controller has been successfully implemented on ASLAN. Stable stepping and forward walking motion have been achieved with the proposed controller.

6.2 Directions for Future Work

Several directions are available for the future research based on the work in this thesis.

Coordination Structure Design

In this thesis, the proposed coordination method can maintain a desired phase relationship between oscillators in the control. It can make the robot return back to a desired walking pattern after the external perturbation. In other cases, for example, the robot needs to change walking pattern such as from level ground walking to staircase walking. In this case, the phase relationship between oscillators in level ground walking will be different from the one in the staircase walking. Our proposed method cannot be applied in this case. The phase relationship cannot be changed for a new environment. In this case, a smooth transition from a pattern to another in CPG control is an interesting topic for further study.

Model Design

In this thesis, the proposed coordination connection is based on the oscillator model proposed by Matsuoka. It is understood that many other oscillator models have also

been widely used in the CPG control. Each of them has advantages and disadvantages in the walking control. Exploring a better oscillator model is still an interesting topic for future study. Although the proposed connection method is based on Matsuoka's oscillator model in this study, the coordination idea brought by our study can be widely applied to any oscillator model of CPG approach.

Sensory Feedback design

In this thesis, the sensory feedback design is based on an intuition approach. This method is highly depended on the human experience. In the final goal, the CPG controller should know how to response to a perturbation even without the knowledge from human experience. In this case, a learning system or other intelligent method can be used. Exploring this area is also very interesting for future research.

Hardware Improvement

In the implementation of CPG based controller, only the ground reaction force values are fed back to the oscillators. To detect an external push applied on the robot, the inertial sensors such accelemeters and gyros should be mounted on the robot. With more sensory feedback information, more walking motions can be tested.

Bibliography

[1] *Actroid*. Osaka University, June 2010. [online] http://news.cnet.com/2300-11394_3-6132975-4.html.

[2] *Artifical leg*, June 2010. [online] http://blogs.franklinnow.com.

[3] *ASIMO*. HONDA, June 2010. [online] http://world.honda.com/ASIMO.

[4] *HRP robots*. AIST, June 2010. [online] http://www.is.aist.go.jp/humanoid/m_movies/index.html.

[5] *Murata boy*. Murata, June 2010. [online] http://www.murataboy.com/en/index.html/.

[6] *PETMAN*. Boston Dynamics, June 2010. [online] http://www.bostondynamics.com/robot_petman.html.

[7] *QRIO*. SONY, June 2010. [online] http://www.sonyaibo.net/aboutqrio.htm/.

[8] *Toyota robot*. Toyota, June 2010. [online] http://www.toyota.co.jp/en/special/robot/.

[9] M. Amemiya and T. Yamaguchi. Fictive locomotion of the forelimb evoked by stimulation of the mesencephalic locomotor region in the decerebrate cat. *Neurosci Lett*, 50:91–96, 1984.

[10] S. Aoi and K. Tsuchiya. Locomotion control of a biped robot using nonlinear oscillators. *Autonomous Robots*, 19:219–232, 2005.

[11] P. Arena. A mechatronic lamprey controlled by analog circuits. In *In Proceedings of the 9th IEEE mediterannean conference on control and automation*, 2001.

[12] P. Arena, L. Fortuna, M. Frasca, and G. Sicurella. An adaptive, self-organizing dynamical system for hierarchical control of bio-inspired locomotion. *IEEE Transactions on Systems, Man and Cybernetics*, 34(4):1823–1837, 2004.

[13] J. S. Bay and H. Hemami. Modeling of a neural pattern generator with coupled nonlinear oscillators. *IEEE Transactions on Biomedical Engineering*, 34(4):297–306, 1987.

[14] R. Beer and J. Gallagher. Evolving dynamical neural networks for adaptive behavior. *Autonomous Robots*, 1(1):91–122, 1992.

[15] J. Cazalets, Y. Sqalli-Houssaini, and F. Clarac. Activation of the central pattern generators for locomotion by serotonin and excitatory amino acids in neonatal rat. *J. Physiol*, 455:187–204, 1992.

[16] C. M. Chew. *Dynamic Bipedal Walking Assisted by Learning*. PhD thesis, Massachusetts Inustitute of Technology, 2000.

[17] C. M. Chew and G. Pratt. Dynamic bipedal walking assisted by learning. *Robotica*, 20(5):477–491, 2002.

[18] A. H. Cohen, P. J. Holmes, and R. Rand. The nature of coupling between segmented oscillations and the lamprey spinal generator for locomotion: A mathematical model. *Journal of Mathematical Biology*, 13:345–369, 1982.

[19] A. Crespi and A. Ijspeert. Amphibot ii: An amphibious snake robot that crawls and swims using a central pattern generator. In *In Proceedings of the 9th international conference on climbing and walking robots*, pages 19–27, 2006.

[20] H. Cruse, D. Brunn, C. Bartling, J. Dean, M. Dreifert, and T. Kindermann. Walking: A complex behavior controlled by simple networks. *Adaptive Behavior*, 3(4):385–418, 1995.

[21] G. Endo, J. Morimoto, T. Matsubara, J. Nakanishi, and G. Cheng. Learning cpg-based biped locomotion with a policy gradient method: Application to a humanoid robot. *International Journal of Robotics Research*, 27:213–228, 2008.

[22] G. Endo, J. Morimoto, J. Nakanishi, and G. Cheng. Experimental studies of a neural oscillator for biped locomotion with qrio. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 596–602, 2005.

[23] Y. Fukuoka, H. Kimura, and A. H. Cohen. Adaptive dynamic walking of a quadruped robot on irregular terrain based on biological concept. *International Journal of Robotics Research*, 22:187–202, 2007.

[24] D. Garis. Analysis of neural oscillator for bio-inspired robot control. In *IEEE Conference on Robotics, Automation and Mechatronics*, pages 204–206, 2006.

[25] M. Golubitsky and I. Stewart. *The symmetry perspective: From equilibrium to chaos in phase space and physical space*. Birkhauser, 2002.

[26] A. Goswami, B. Espiau, and A. Keramane. Limit cycles and their stability in a passive bipedal gait. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 246–251, 1996.

[27] S. Grillner. *Control of locomotion in bipeds, tetrapods and fish*. American Physiological Society, Bethesda, 1981.

[28] S. Grillner. Neurobiological bases of rhythmic motor acts in vertebrates. *Science*, 228:143–149, 1985.

[29] R. Heliot and B. Espiau. Multisensor input for cpg-based sensoryłmotor coordination. *IEEE Transactions on Robotics*, 24(1):191–195, 2008.

[30] M. Hirose and K. Ogawa. Honda humanoid robots development. *Philosophical Transactions of the Royal Society London, Series A (Mathematical, Physical and Engineering Sciences)*, 365(1850):11–19, 2007.

[31] D. G. Hobbeien and M. Wisse. Swing-leg retraction for limit cycle walkers improves disturbance rejection. *IEEE Transactions on Robotics*, 24:377–389, 2008.

[32] W. Huang, C. M. Chew, and G. S. Hong. Coordination between oscillators: An important feature for robust bipedal walking. In *Proceedings of the IEEE Interna-*

tional Conference on Robotics and Automation, pages 3206–3212, Pasadena, CA, May 2008.

[33] K. Ichiro, O. Sadamu, S. Katsuhiko, N. Seinosuke, S. Shigeki, M. Toshiaki, K. Tetsunori, and F. Eizo. Robot musician 'wabot-2' (waseda robot-2). *Robotics Amsterdam*, 3(2):143–155, 1987.

[34] A. J. Ijspeert. Central pattern generators for locomotion control in animals and robots: A review. *Neural Networks*, 21:642–653, 2008.

[35] T. Ishide and Y. Kuroki. Development of sensor system of a small biped entertainment robot. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 648–653, New Orleans, 2004.

[36] S. Ito, H. Yuasa, Z. Luo, M. Ito, and D. Yanagihara. A mathematical model of adaptive behavior in quadruped locomotion. *Biological Cybernetics*, 78(5):337–347, 1998.

[37] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa. Biped walking pattern generation by using preview control of zero-moment point. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1620–1626, Taipei, Taiwan, 2003.

[38] S. Kajita, O. Matsumoto, and M. Saigo. Real-time 3d walking pattern generation for a biped robot with telescopic legs. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2299–2306, Seoul, Korea, May 2001.

[39] S. Kajita and K. Tani. Experimental study of biped dynamic walking in the linear inverted pendulum mode. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2885–2891, Nagoya, Japan, May 1995.

[40] K. Kaneko, F. Kanehiro, S. Kajita, H. Hirukawa, T. Kawasaki, M. Hirata, K. Akachi, and T. Isozumi. Humanoid robot hrp-2. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1083–1090, 2004.

[41] H. Kimura, S. Akiyama, and K. Sakurama. Realization of dynamic walking and running of the quadruped using neural oscillators. *Autonomous Robots*, 7(3):247–258, 1999.

[42] B. Klaassen, R. Linnemann, D. Spenneberg, and F. Kirchner. Biomimetic walking robot scorpion: Control and modeling. *Autonomous Robots*, 41:69–76, 2002.

[43] E. Klavins and D. E. Koditschek. Phase regulation of decentralized cyclic robotic systems. *International Journal of Robotics Research*, 21:257–277, 2002.

[44] M. Lewis and A. Fagg. Genetic programming approach to the construction of a neural network for control of a walking robot. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2618–2623, 1992.

[45] H. Masato and T. Tooru. Development of humanoid robot asimo. *Honda R and D Tech Rev*, 13(3):1–6, 2001.

[46] K. Matsuoka. The dynamic model of binocular rivalry. *Biological Cybernetics*, 49:201–208, 1984.

[47] K. Matsuoka. Sustained oscillations generated by mutually inhibiting neurons with adaptation. *Biological Cybernetics*, 52:367–376, 1985.

[48] T. McGeer. Passive dynamic walking. *Int. J. Robot*, 9(2):62–82, 1990.

[49] W. Miller, P. Latham, and S. Scalera. Bipedal gait adaptation for walking with dynamic balance. In *Proceedings of American Control*, pages 1603–1608, 1991.

[50] S. Miyakoshi, G. Taga, Y. Kuniyoshi, and A. Nagakubo. Three dimensional bipedal steping motion using neural oscillators-towards humanoid motion in the real world. In *Proceedings of the IEEE International Conference on Intelligent Robot and Systems*, pages 84–89, 1998.

[51] J. Nakanishi, J. Morimoto, G. Endo, G. Cheng, S. Schaal, and M. Kawato. A framework for learning biped locomotion with dynamical movement primitives. In *Proceedings of the IEEE-RAS International Conference on Humanoid Robots*, pages 925–940, 2004.

[52] J. Pratt, J. Carff, S. Drakunov, and A. Goswami. Capture point: A step toward humanoid push recovery. In *Proceedings of the IEEE-RAS International Conference on Humanoid Robots*, pages 200–207, 2006.

[53] M. Raibert. *Legged Robots That Balance*. MIT Press, Cambridge, 1986.

[54] M. Raibert and J. Hodgins. Animation of dynamic legged locomotion. *Computer Graphics*, 25(4):349–358, 1991.

[55] L. Righetti and A. J. Ijspeert. Programmable central pattern generators: an ap-

plication to biped locomotion control. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1585–1590, May 2006.

[56] L. Righetti and A. J. Ijspeert. Pattern generators with sensory feedback for the control of quadruped locomotion. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 819–824, Pasadena, US, May 2008.

[57] S.Grillner and P. Zangger. How detail is the central pattern generation for locomotion. *Brain Research*, 88:367–371, 1975.

[58] J. Shan and F. Nagashima. Neural locomotion controller design and implementation for humanoid robot hoap-1. In *Proceedings of The 20th Annual Conference of the Robotics Society of Japan*, 2002.

[59] L. Sherwood. *From Cells to Systems*. California: Brooks/Cole, 2001.

[60] M. Shik, F. Severin, and G. Orlovskii. Control of walking and running by means of electrical stimulation of the mid-brain. *Biophysics*, 11:756–765, 1966.

[61] J. Smith, J. Feldman, and B. J. Schmidt. Neural mechanisms generating locomotion studied in mammalian brain stem-spinal cord in vitro. *FASEB*, 2:2283–2288, 1988.

[62] S. Strogatz. *Nonlinear Dynamics and Chaos with application to physics Biology, Chemistry, and Engineering*. Addison-Wesley Publishing Company, 2000.

[63] S. P. Swinne. Intermanual coordination: from behavioural principles to neural-network interactions. *nature*, 3(5):348–359, 2002.

[64] G. Taga. A model of the neuro-musculo-skeletal system for human locomotion i. emergence of basic gait. *Biological Cybernetics*, 73:97–111, 1995.

[65] G. Taga, Y. Yamaguchi, and H. Shimizu. Self-organized control of bipedal locomotion by neural oscillators in unpredicatable environment. *Biological Cybernetics*, 65:147–159, 1991.

[66] M. Vukobratovic, A. A. Frank, and D. Juricic. On the stability of biped locomotion. *Biomedical Engineering, IEEE Transactions on*, BME-17(1):25–36, 1970.

[67] M. M. Williamson. Neural control of rhythmic arm movements. *Neural Networks*, pages 1379–1394, 1998.

[68] T. Wimbo, D. Nenchev, A. Albu-Scha, and G. Hirzinger. Experimental study on dynamic reactionless motions with dlr's humanoid robot justin. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5481–5486, 2009.

[69] M. Wisse, C. Atkeson, and D. Kloimwieder. Swing leg retraction helps biped walking stability. In *International Conference on Humanoid Robots*, pages 295–300, 2005.

[70] T. Yamasaki, T. Nomura, and S. Sato. Possible functional roles of phase resetting during walking. *Biological Cybernetics*, 88:468–496, 2003.

[71] R. Zaier and F. Nagashima. Motion generation of humanoid robot based on polynomials generated by recurrent neural network. In *Proceedings of the First Asia International Symposium on Mechatronics*, 2004.

[72] D. Zhang, K. Zhu, and L. Lan. Genetic programming. evolution of a time dependent neural network module which teaches a pair of stick legs to walk. In *Proceedings of the IEEE International Conference on Robotics, Automation and Mechatronics*, pages 1–6, 2006.

[73] T. Zieliinska. Coupled oscillators utilised as gait rhythm generators of a two-leggeeed walking maachine. *Biological Cybernetics*, 74:263–273, 1996.

Appendix I: ASLAN Description

A.1 History

In humanoid research, designing a physical humanoid robot is very important. It can be used as a platform to verify different walking algorithms. In general, a humanoid robot has head, trunk, two legs, two arms and two feet. Each part has several DoFs to enable execution of desired motion.

In National University of Singapore, the research of designing a fully autonomous humanoid robot has been carried out for many years. In 2003, the first version NUSBIP-I was developed. NUSBIP-I is a bipedal robot with 6 DoFs on each leg. As a prototype platform, NUSBIP-I fails to achieve walking motion. However, its design and joint configuration provide a reference for later designs. As for the second version, NUSBIP-II followed the joint configuration of NUSBIP-I, but with an improved mechanical design. With the new system, the robot achieved stepping motion. The problem of NUSBIP-II is that many joints have backslash and weak torque output. Because of this, it also fails to achieve walking motion.

To further improve NUSBIP-II, NUSBIP-III project was started in 2007. The new robot is mounted with more powerful motors and harmonic drives. This overcomes the problem of backslash and weak torque output. With the new design, the robot can perform level ground walking, slope walking and staircase walking.

Starting from 2009, a new improvement of NUSBIP-III is carried out. The purpose is to make the robot to be a fully autonomous humanoid robot. Head, arms are added on the robot. A yaw joint is added on the trunk. Battery is used to replace the external power and make the robot autonomous. The new robot is named NUSBIP-III ASLAN.

A.2 Mechanical Design

Fig. A.1 shows the design of ASLAN in SolidWorks. Generally, the robot structure can be divided into five parts: trunk (including head), arm, waist, leg and foot. Each of them will be introduced in the following subsections.

A.2.1 Head and Trunk Design

The head is designed to hold the vision system of the robot. In this design, two cameras are mounted in the head. The neck has 2 DoFs which are pitch and yaw joints. The motors used are DC servo motors. To improve of the outlook of the ASLAN, a plastic cover is added on the head as shown in Fig. A.2.

The purpose of the body trunk is mainly to hold the electrical devices, sensors and

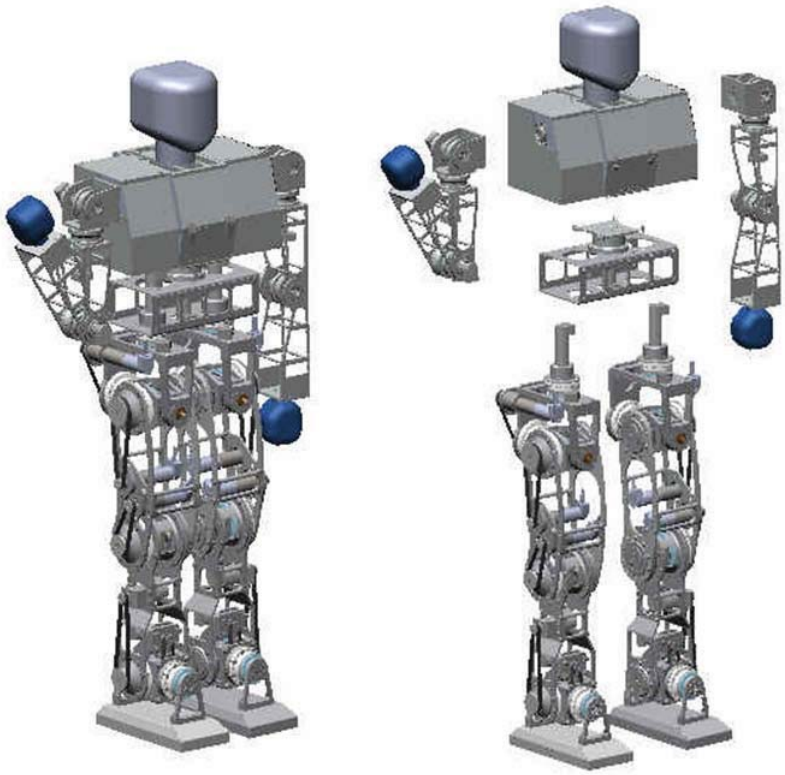


Figure A.1: The design of ASLAN in SolidWorks.

Figure A.2: The head design of ASLAN.

batteries. The trunk can be divided into several layers as shown in Fig. A.3. The main processor PC/104 unit with DAQ card and CANBUS card are mounted at the center of the trunk. The battery is placed under PC/104 to lower the CoM. In this figure, the accelerometer and gyro sensors have not been mounted on the body yet. The mounting location is indicated in the SolidWorks design on the left side of the Fig. A.3.

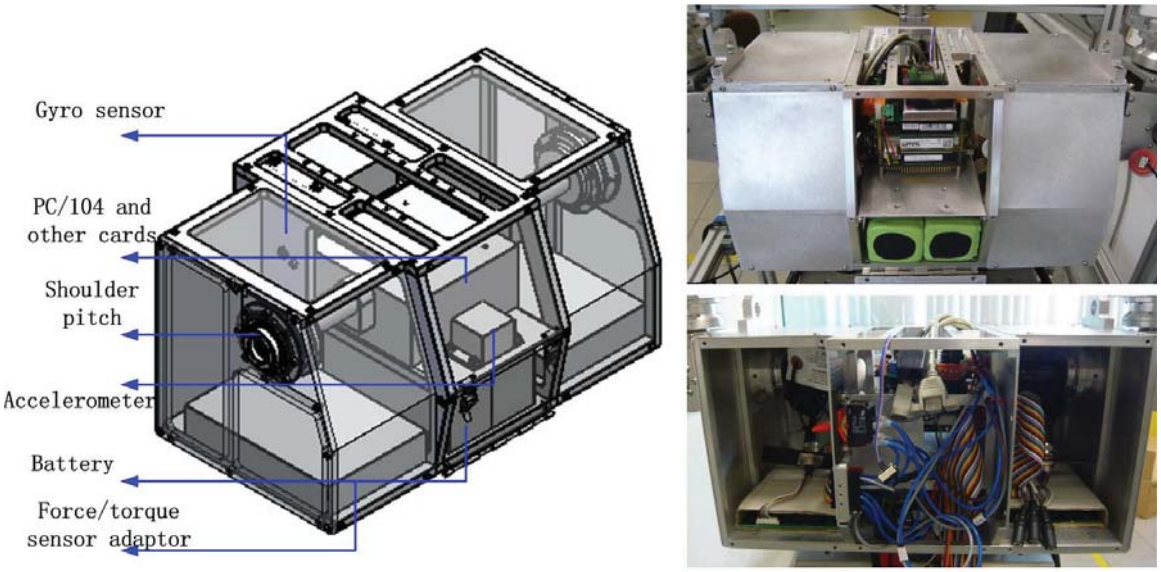


Figure A.3: The trunk design of ASLAN.

A.2.2 Arm Design

The arms are designed to be simple and light-weight. Each arm consists of 4 DOFs: 3 DOFs at the shoulder and 1 DOF at the elbow. Shoulder has pitch, roll and yaw joints which intersect at the same point with the pitch-roll-yaw configuration. The elbow has a pitch joint perpendicular to the yaw axis of the shoulder. Due to this arrangement the elbow pitch axis is not always parallel to the shoulder pitch axis. All the joints are connected with the harmonic gears and motors. Fig. A.4 shows the arm design of

ASLAN.

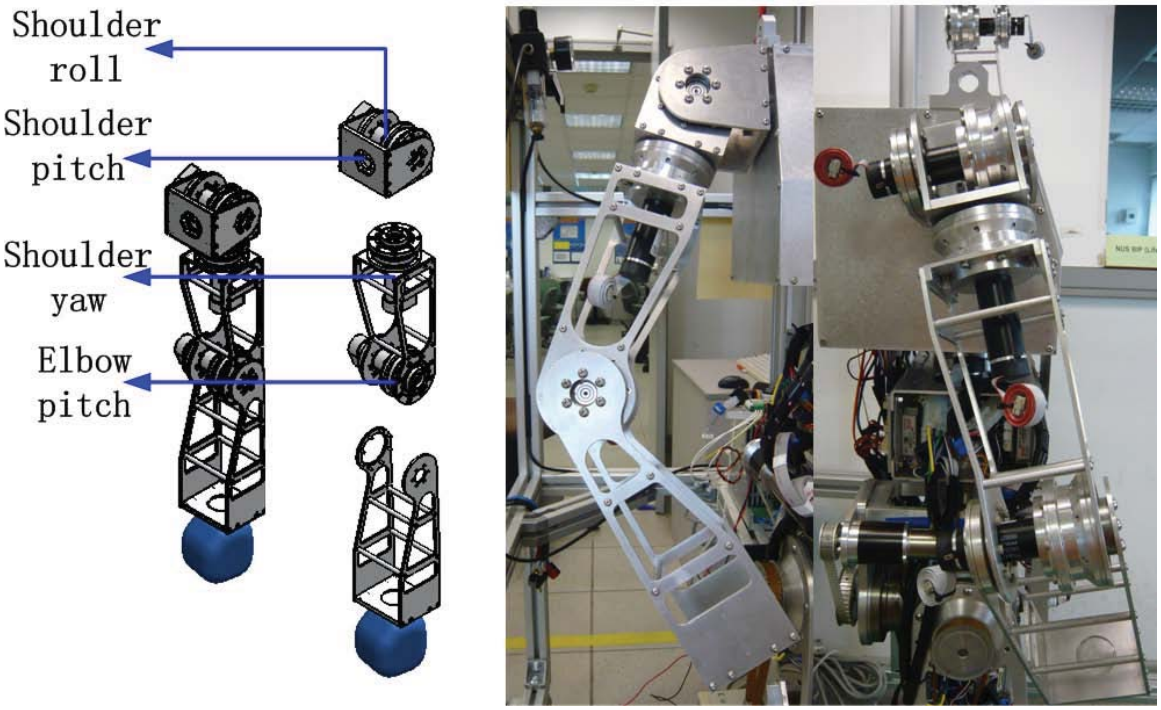


Figure A.4: The arm design of ASLAN.

A.2.3 Waist Design

In the bottom part of trunk, a yaw joint is added between body trunk and legs to enable a yaw motion of trunk as shown in Fig. A.5. The robot can swing the body during walking to balance the motion and improve the stability. The available space in the waist is used to hold the amplifiers which control the hip and knee motors. This is can also help to shift the CoM position closer to the hip.

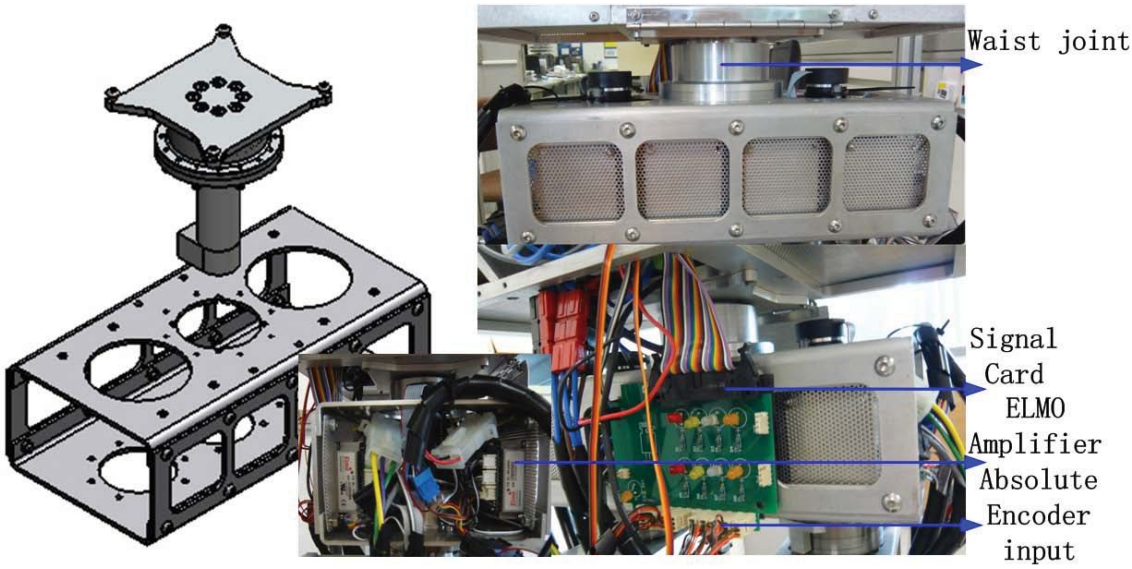


Figure A.5: The waist design of ASLAN.

A.2.4 Leg Design

The leg design follows the NUSBIP-II joint configuration. Each leg has 6 joints in total. To overcome the backslash problem, harmonic gears are used to replace the normal gear heads. Also, more powerful motors are used to replace old ones used in NUSBIP-II. The joint is not directly connected to the motor. A belt driven system is adopted to achieve better mass distribution and larger torque output. Fig. A.6 shows the leg design. As shown in the figure, the thigh links have a curve shape. This is to allow a larger knee rotation range.

A.2.5 Foot Design

The foot contacts with the ground with an aluminium plate. As can be seen in Fig. A.7, a force/torque sensor is placed between foot plate and the ankle joint to sense the force

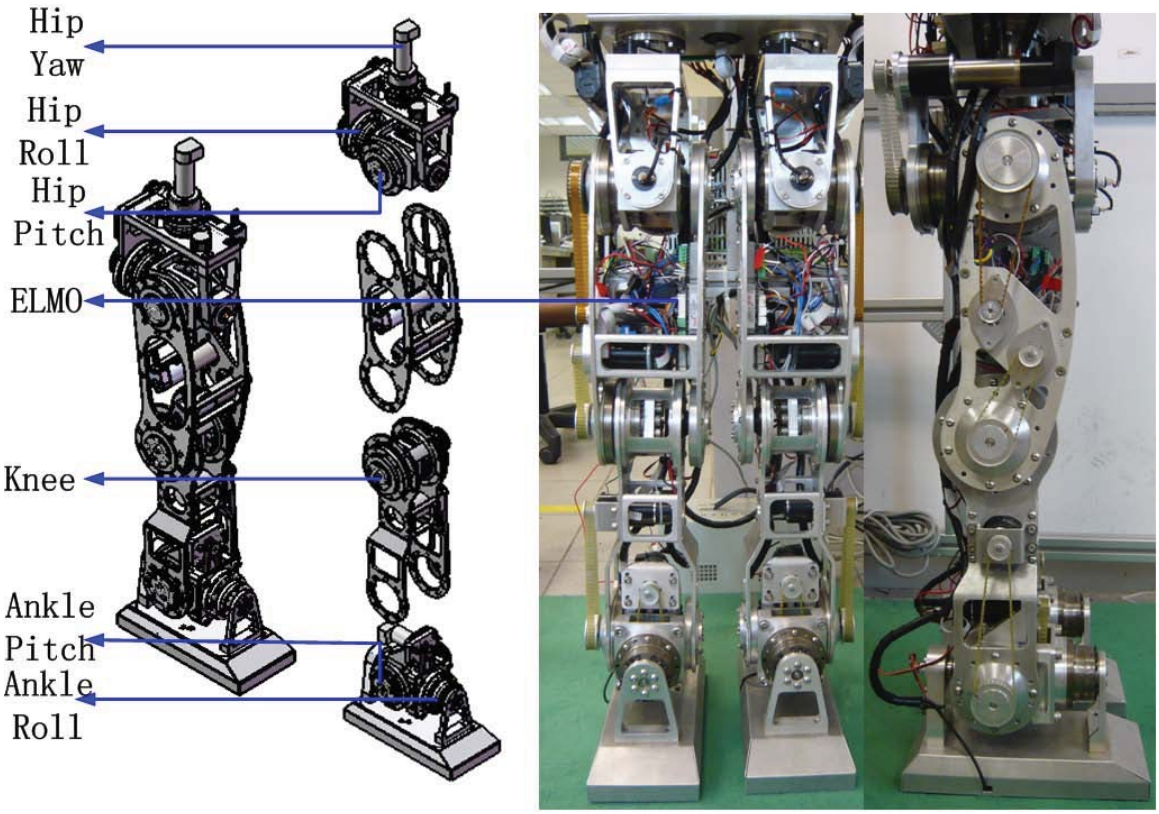


Figure A.6: The leg design of ASLAN.

and torque value. To protect the sensor and make the feet look nicely, additional covers are added on both feet.

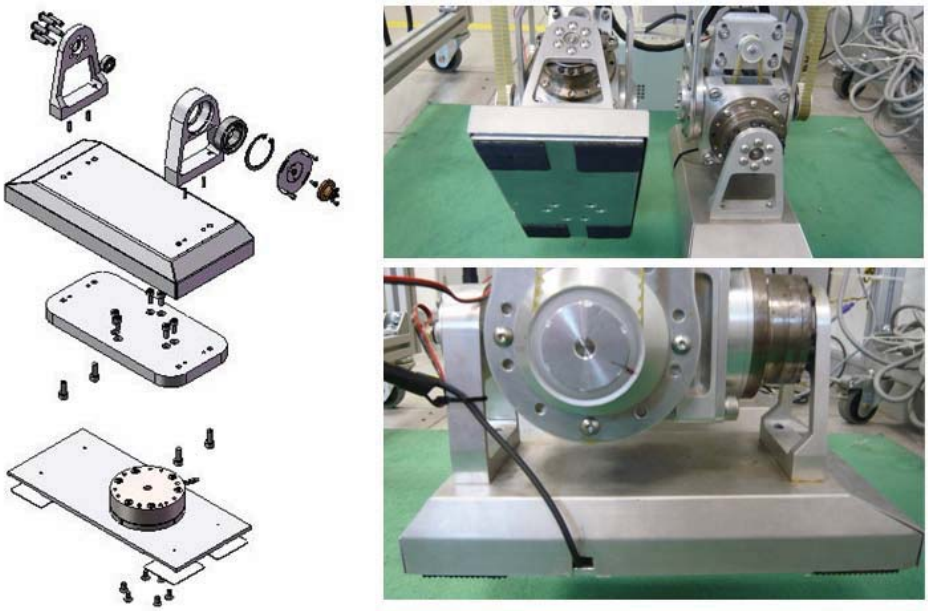


Figure A.7: The foot design of ASLAN.

In the foot design, we add four small rubbers on each edge of the foot to increase the friction and absorb the impact. The design is shown in Fig. A.7. During walking, the rubber may be damaged easily because of the huge impact between foot and ground. Therefore, instead of directly attaching the rubbers on the foot plate, we attach the rubber on a thin aluminium plate which is then screwed into the foot plate. If one rubber is damaged, we can easily replace an aluminium plate instead of a whole foot plate,

A.3 Control System

A.3.1 Sensors

Fig. A.8 shows the pictures of two types of absolute encoder used on the robot. The absolute encoder sensors are used to get the absolute value of joints. They are used to initialize the joints of the robot before starting the motion. Althrough the encoders mounted on the motors can provide more precise joint angle value, it can not provide the absolute angle value. They are used in the control after inlitialization. MAE3 is an absolute magnetic kit encoder that provides shaft position information with no stops or gaps. It can be easily mounted to an existing shaft. The output of the MAE3 encoder is PWM signal. All joints on the leg except hip yaw are mounted with this sensors. On the hip yaw joint, a wire sensor is used because it is very difficult to mount the MAE3 on this joint. The output of this sensor is analog signal which is read by the DAQ card. Fig. A.9 shows other sensors used on ASLAN. Accelerometer and gyros are used to provide inertial information of the body. Force/torque sensor provides the ground reaction force information on the foot.

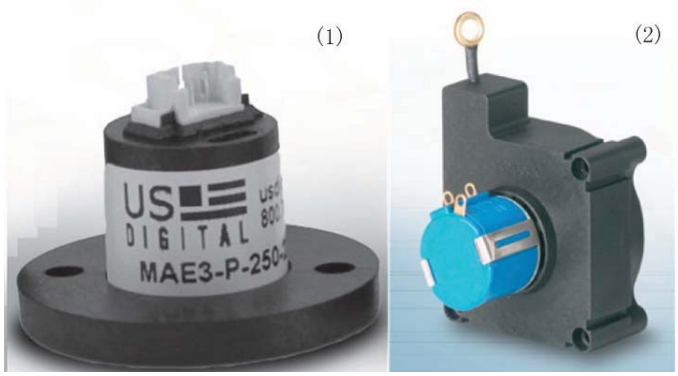


Figure A.8: The absolute encoder of ASLAN (1):MAE3, (2):wire sensor

Figure A.9: Other sensors of ASLAN (1): accelerometer, (2) gyro, (3) force/torque sensor

A.3.2 Drive Unit

The drive unit for controling a joint usually consists of an amplifier, gear reduction, motor and tech encoder. The desired position is sent from PC/104 to ELMO amplifier through CAN BUS. ELMO controls the motor to reach the desired position by the local feedback loop. In this local control loop, the PID parameters are automatically tuned by the amplifier itself. Fig. A.10 shows a result of position tracking and velocity tracking by the ELMO after auto-tuning. For ASLAN, the gear reduction is a combination of harmonic drive and pulley belt system.

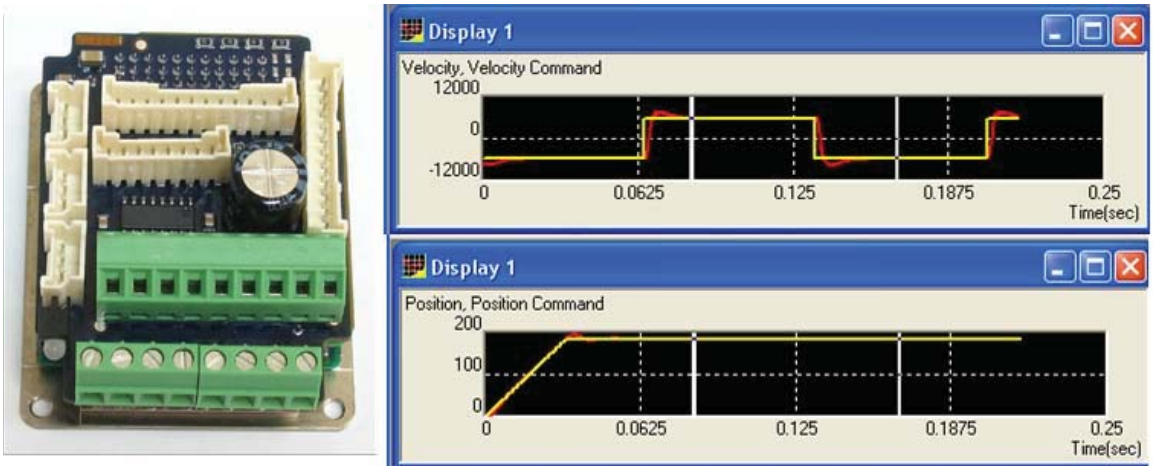


Figure A.10: The amplifier of the ASLAN and an example of position and velocity tracking

A.3.3 Programming Environment

The original programming environment of NUSBIP-III is RT-Linux. RT-Linux is designed to achieve real time control. A high real time sample rate can be achieved in RT-Linux. One disadvantage of RT-Linux is that the interface is not user friendly. To overcome this problem, we migrate the system to Windows in current version. Windows has a friendly user interface and programming environment. To make the controller run in real time, we install RTX in the Windows. RTX is a software package to achieve real time control in Windows. Our controller is run under RTX environment. The programming loop basically includes reading sensory feedback, calculating the desired trajectories and sending the motion command. The motion command is executed in every 10ms.

The ELMO amplifier supports CANBUS communication. Thus, the motion command is sent through CANBUS. One problem of using ELMO amplifier is that its driver cannot be directly added into RTX program, because it includes some MFC applications which cannot be used in RTX. The version which can be used in RTX is still under development. To solve this problem, a share memory between RTX and ELMO driver program is created. The reference trajectories are sent from main loop in RTX to share memory every 10ms. The ELMO driver program will keep on checking the share memory and execute the new trajectories when they are updated. This can ensure the motion command is executed in real time. The communication diagram is shown in Fig. A.11.

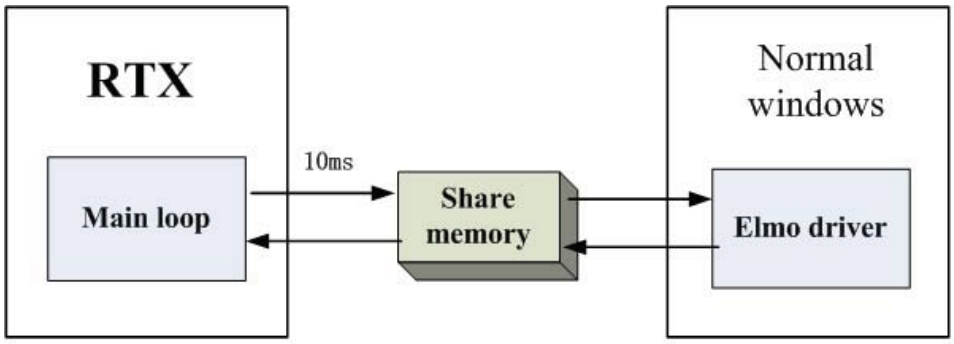


Figure A.11: The diagram of communication between main program and ELMO driver program

Appendix II: Conditions for Limit Cycle Behavior

In this part, we will analyze the Mastuoka's neural oscillator model and derive the conditions to have a limit cycle behavior.

According to the Poincaré-Bendixson Theorem[62], a trajectory inside region $\Re$ has limit cycle behavior if $\Re$ is bounded and the trajectory has no stable equilibrium point. For the neural oscillator model, Matsuka[47] has proved that the output of neural oscillator is bounded. In this thesis, we focus on analyzing the stable equilibrium point of the neural oscillator model described by the equations (3.1)-(3.6).

To check if there is a stable equilibrium point, we separate the state variable u_1, u_2 region into four subset quadrants $\{u_1 \geq 0, u_2 \geq 0\}$, $\{u_1 \geq 0, u_2 < 0\}$, $\{u_1 < 0, u_2 \geq 0\}$, and $\{u_1 < 0, u_2 < 0\}$. We will analyze the stable equilibrium point in each quadrant. Here we assume there is no external input; that is $g_j = 0$.

In the first quadrant $\{u_1 \geq 0, u_2 \geq 0\}$, the output of the oscillator is $Y = u_1 - u_2$. We set

variable $V = v_1 - v_2$. Combining equation (3.1)-(3.4), we obtain

$$\tau_1 \dot{Y} = -Y - \beta V + aY \tag{B.1}$$

$$\tau_2 \dot{V} = Y - V \tag{B.2}$$

Substituting V into equation (3.7) and equation (3.8), we get

$$\tau_1 \tau_2 \ddot{Y} + (\tau_1 + \tau_2 - a\tau_2)\dot{Y} + (1 + \beta - a)Y = 0 \tag{B.3}$$

In the second quadrant $\{u_1 \geq 0, u_2 < 0\}$, $Y = u_1$. We set variable $V = v_1$. Equations (3.1) and (3.2) become

$$\tau_1 \dot{Y} = c - Y - \beta V \tag{B.4}$$

$$\tau_2 \dot{V} = Y - V \tag{B.5}$$

Substituting V into equation (3.10) and equation (3.11), we get

$$\tau_1 \tau_2 \ddot{Y} + (\tau_1 + \tau_2)\dot{Y} + (1 + \beta)Y - c = 0 \tag{B.6}$$

In the third quadrant $\{u_1 < 0, u_2 \geq 0\}$, $Y = -u_2$. We set variable $V = v_2$. Equations

(3.3) and (3.4) become

$$\tau_1 \dot{Y} = -c - Y + \beta V \tag{B.7}$$

$$\tau_2 \dot{V} = -Y - V \tag{B.8}$$

Substituting V into equation (3.13) and equation (3.14), we get

$$\tau_1 \tau_2 \ddot{Y} + (\tau_1 + \tau_2)\dot{Y} + (1+\beta)Y + c = 0 \tag{B.9}$$

In the first quadrant, equation (3.9) has a equilibrium point at state $\{Y^*, \dot{Y}^*\} = \{0,0\}$. When $1 + \frac{\tau_1}{\tau_2} - a < 0$, equation (3.9) has a negative damping and the equilibrium point is an unstable equilibrium point. With this constraint, if the trajectory does not start at point $\{Y^*, \dot{Y}^*\} = \{0,0\}$, which is $\{u_1^* = u_2^*, v_1^* = v_2^*\}$, we can say that there is no equilibrium point in this quadrant.

In the second quadrant, equation (3.12) has a fixed point at $\{Y^*, \dot{Y}^*\} = \{\frac{c}{1+\beta}, 0\}$. In this case, $\{u_1, u_2\} = \{\frac{c}{1+\beta}, c(1 - \frac{a}{1+\beta})\}$. If $1 - \frac{a}{1+\beta} > 0$, which is $a < 1 + \beta$, this point is outside the second quadrant. Therefore, with this constraint, there is no equilibrium point in this quadrant.

In the third quadrant, equation (3.15) has a fixed point at $\{Y^*, \dot{Y}^*\} = \{\frac{-c}{1+\beta}, 0\}$. A similar proof as in the second quadrant can be applied in this quadrant. When $a < 1 + \beta$, there is no equilibrium point in this quadrant.

Because of the inter-inhibition property, two neuron states will not be negative at same time. The oscillator will not go to the fourth quadrant.

In summary, based on the Poincaré-Bendixson theorem, the neural oscillator model described by equations (3.1)-(3.6) has a unique limit cycle behavior if

$$a - 1 - \frac{\tau_1}{\tau_2} > 0 \tag{B.10}$$

$$a - 1 - \beta < 0 \tag{B.11}$$

Appendix III: Amplitude of Neural Oscillator

As written in the Chapter 3 section 2, we have done a piecewise linear analysis to this oscillator model. We separate the region into four subset quadrants:

$$\{\{u_1 \geq 0, u_2 \geq 0\}, \{u_1 \geq 0, u_2 < 0\}, \{u_1 < 0, u_2 \geq 0\}, \{u_1 < 0, u_2 < 0\}\}$$

When $|a - 1 - \frac{\tau_1}{\tau_2}|$ is small, two neurons are always in the active state which is in the $\{u_1 \geq 0, u_2 \geq 0\}$ quadrant. This can be seen by plot the states of the neuron during the oscillation. In this case, the oscillator equations are:

$$\tau_1 \dot{u}_1 = c - u_1 - \beta v_1 - a u_2 \quad \text{(C.1)}$$

$$\tau_2 \dot{v}_1 = u_1 - v_1 \quad \text{(C.2)}$$

$$\tau_1 \dot{u}_2 = c - u_2 - \beta v_2 - a u_1 \quad \text{(C.3)}$$

$$\tau_2 \dot{v}_2 = u_2 - v_2 \quad \text{(C.4)}$$

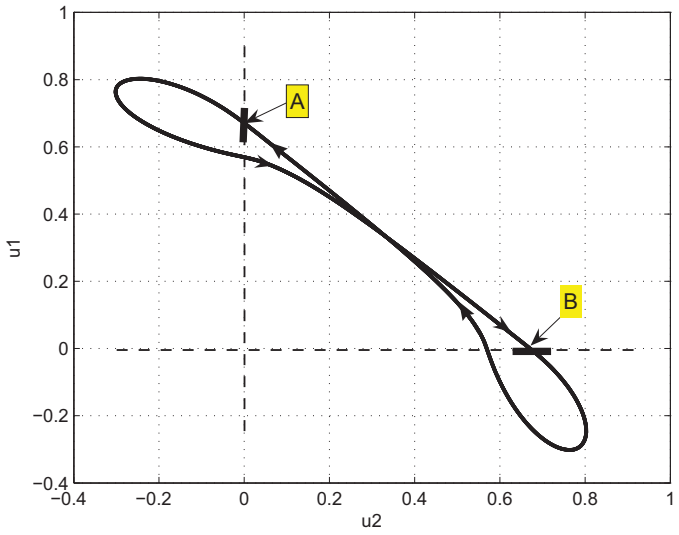


Figure C.1: The output of the neuron in u1-u2 plane.

In this quadrant, the two neurons of the oscillator are symmetrical. In the steady state, as shown in Fig. C.1, the value of $\dot{u}_2$ at point A is the same as the value of $\dot{u}_1$ at point B. Because of the symmetry, the value of $\dot{u}_2$ at point A is the negative value of $\dot{u}_2$ at point B. Therefore, in the steady state and in this quadrant: $\dot{u}_2 = -\dot{u}_1, \dot{v}_2 = -\dot{v}_1$. In this case, we get:

$$c - u_1 - \beta v_1 - a u_2 = -(c - u_2 - \beta v_2 - a u_1) \tag{C.5}$$

$$u_1 - v_1 = -(u_2 - v_2) \tag{C.6}$$

Then we get $(1+\beta+a)(u_1+u_2)=2c$. Because $u_2=0$ at point A, $u_{1A}=u_{2B}=\frac{2c}{1+\beta+a}$.

When $|a-1-\frac{\tau_1}{\tau_2}|$ is small, we use this value to approximate value of oscillator amplitude.

Note: when $|a-1-\frac{\tau_1}{\tau_2}|$ is big, the neuron output will stay longer in quadrants $\{u_1 \geq 0, u_2 < 0\}, \{u_1 < 0, u_2 \geq 0\}$. This will increase the error between our approximated value and the actual value.

Author's Publications

Journal Papers

W. Huang; C. M. Chew; Y. Zheng; G. S. Hong; Bio-inspired locomotion control with coordination between neural oscillators; International Journal of Humanoid Robotics, v 6, n 4, p 585-608, Dec. 2009

Conference Papers

W. Huang; C. M. Chew; G. S. Hong; Coordination between oscillators: An important feature for robust bipedal walking; IEEE International Conference on Robotics and Automation, p 3206-3212, 2008

W. Huang; C. M. Chew; Y. Zheng; G. S. Hong; Pattern generation for bipedal walking on slopes and stairs; IEEE-RAS International Conference on Humanoid Robots (Humanoids 2008), p 205-10, 2008

W. Huang; C. M. Chew; G. S. Hong; Coordination in CPG and its application on bipedal walking; IEEE International Conference on Robotics, Automation and Mechatronics, RAM 2008, p 450-455

W. Huang; C. M. Chew; G. S. Hong; N. Gnanassegarane; Trajectory Generator for Rhythmic Motion Control of Robot using Neural Oscillators ADVANCES IN CLIMBING AND WALKING ROBOTS (pp 383-392)

A. H. Adiwahono; C. M. Chew; W. Huang; Y. Zheng; Push recovery controller for bipedal robot walking; IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), p 162-7, 2009

A. H. Adiwahono; C. M. Chew; W. Huang; V. H. Dau; Humanoid robot push recovery through walking phase modification; IEEE Conference on Robotics, Automation and Mechatronics (RAM 2010), p 569-74, 2010