QUADRUPEDAL WALKING GAIT GENERATION USING GENETIC ALGORITHM

YONG SEE WEI

NATIONAL UNIVERSITY OF SINGAPORE

2014

QUADRUPEDAL WALKING GAIT GENERATION USING GENETIC ALGORITHM

YONG SEE WEI

B.Eng.(Hons.), NUS

A THESIS SUBMITTED

ENGINEEERING

DEPARTMENT OF

ELECTRICAL & COMPUTER ENGINEERING

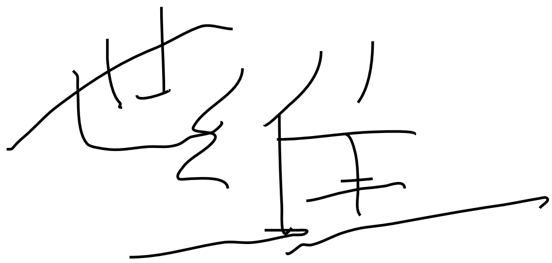
NATIONAL UNIVERSITY OF SINGAPORE

2014

DECLARATION

I hereby declare that the thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.



Yong See Wei

23 May 2014

Acknowledgements

With immense pleasure I express my deep gratitude and thanks to my supervisor, Assoc Prof. Prahlad Vadakkepat for giving me this opportunity to work under him and also his continuous supports and guidance in all the stage of my study including giving me the freedom that nurtured my growth in independence thinking. Not to mention his encouragement and confidence on me in making this study possible. I have greatly benefit from his invaluable technical insight, knowledge, and experience shared.

I would also like to thank Prof. Tong Heng Lee, Assoc Prof. Abdullah Al Mamun, Assoc Prof. Gerard Leng, Assoc Prof. Arthur Tay and Dr. Tang for their kind help and suggestions. I am also grateful to my fellow project mate Mr. Tan Shin Jiuh, who has given me many useful advice. My gratitude also goes to my senior Dr. Dip Goswami and Dr. Pramod Kumar together with my colleagues Mr. Daniel Hong, Mr. Ng Buck Sin and Mr. Christopher Moy Shin Lee Lan Chong. Finally, I show my appreciation to lab officer Mr. Tan Chee Siong for his supports and giving timely reply for my requests.

I am also grateful to have the chance to pursuit my doctoral degree in National University of Singapore. I would also like to thank all the member of the Electrical and Computer Engineering for their support provided during my time in graduate school.

Lastly, I would like to give the biggest appreciation to my family for their support, patience and confidence to me during this period. Without them, I would not possible achieved what I have.

Table of Contents

Chapter 6

Towards the Quadruped System: The Quadruped Configuration

Chapter 7

Conclusions and Future Directions

Summary

Quadrupedal locomotion is a heavily researched topic for the need of generating stable gaits for high speed locomotion on tough terrains. The challenge in quadrupedal locomotion is in generating locomotion gaits while maintaining stability. In this dissertation, quadrupedal locomotion gait generation is carried out in a virtual world obeying the laws of physics. Genetic algorithm (GA) is used to evolve the quadrupedal gaits. The evolution of the quadrupedal configuration is carried out in stages. The frontal single limb getting up gait is generated and optimized utilizing GA. The concept is then extended to double limbs. The gaits are generated finally for a quadrupedal system. The dynamics of the configuration is taken care of by the Chron::Engine and no mathematical modelling is carried out.

The mammal and reptile type quadrupeds have different limb configurations each having their own advantages for locomotion. The limb configuration considered in this work combines certain advantages from the mammal (dog) and reptile limb configurations. The simulation is carried out in three phases: (1) single limb configuration, (2) double frontal limb configuration with a support structure at the rear, and, (3) quadrupedal configuration.

The objective of the single limb configuration is to get up and sustain the posture for a period of time without collapsing. GA is utilized to optimize the generated gait. The single limb configuration is used as the basic building block for the quadruped model.

The double frontal limbs are simulated with a support wheel structure at the rear. The frontal limbs are formed by two single limb structures. The rear wheels are able to toggle between passive and active modes. When the rear wheels are passive, they act as a support and study is carried out on whether the frontal limbs are able to achieve stable walking gaits. When the wheels are active, they simulate a force generated from the hind limbs in a

quadruped. This is to study whether the frontal limbs are able to maintain stable gaits when disturbance acts from rear like in quadrupedal locomotion.

The quadrupedal configuration utilizes a homogeneous configuration having the same configuration in each limb. The front limbs provide steering while the hind limbs provide the driving force. In order to achieve stability in walking, the hind limbs' driving force need to be optimized.

GA is utilized to optimize the gaits in the three phases of simulation. There is no mathematical model derived for the quadruped. The application of joint functions make the quadruped to move. Relevant cost functions for different stages, single limb, double limbs and quadrupedal limbs, are utilized as cost functions for the GA.

The optimized limb configurations lead to stable walking gaits and postures for a period after getting up. The double limbs configuration is able to walk when a pushing force is generated from the rear. The optimized walking gait for the quadrupedal configuration a typical trot gait.

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Background

Ever since the start of human civilization, transportation had played an important role. By utilizing different types of transportation, humans could shorten travel time and explore different territories. Human could reduce the effort in transportation by using animals to pull carts or sailing with the wind. With the advances in technology, we are able to travel around the world with various transportations which include land vehicles, airplanes and ships.

Autonomous vehicles capable of moving on their own without human control have attracted lot of attention lately. The introduction of autonomous vehicles into our lives can ease the effort of traveling. Autonomous vehicles are especially useful in exploring areas which are difficult or hazardous to reach. Some environments are too hazardous to send humans over before knowing the prevailing conditions, thus autonomous vehicles with surveillance capabilities are a natural choice. By sending in surveillance autonomous vehicles, information can be collected from the field. Some environments are too difficult to be accessed by humans and, sending in surveillance autonomous vehicles is desirable.

Various types of autonomous vehicles exist in the world today, including land, water and air vehicles. Being the most popular among them, autonomous land vehicles are the most researched on.

Autonomous land vehicles can have different locomotion mechanisms including wheeled and legged locomotion. Wheeled autonomous vehicle is most common today as it is easier to design and control as compared to legged locomotion. However, wheeled locomotion is limited for travel on even terrains. In real world explorations, the terrain is uneven. Legged locomotion is a natural choice to deal with uneven terrains.

There are several types of legged locomotion such as bipedal (two legged), quadrupedal (four legged) and multipedal. This thesis focuses on quadrupedal locomotion. The challenge in quadruped locomotion lies on obtaining a stable gait. The strategy to generate certain joint motions is dependent on the platform size and, the mechanical and electrical designs. Uneven terrains greatly increase the difficulty in achieving stable gaits. Most of the quadruped designs are inspired by the biological counterparts in nature world.

1.2 Quadruped systems in natural world

Animals that move with four limbs on the ground are known as quadruped. The most commonly seen quadrupeds are mammals, reptiles and amphibians. The strengths of quadrupedal locomotion are due to their speed, power and stability in the course of moving. For example, cheetah is categorized as the fastest moving animal which can move at a speed of 70 mph (114km/h) for 250 meters with acceleration comparable to a modern sports car [1]. Horses have been utilized for transportation by men since ancient times as horses can travel with a speed between 10–16 m/s for a long duration [2]. Pronghorns, which are the fastest animal in America, are built for both speed and endurance [3].

The leg configuration of a reptile quadruped is different than a mammal quadruped. Reptile quadrupeds have relatively larger triangular support polygon in feet positioning yielding higher static stability. However, this advantage comes at the cost of higher joint torques on legs to support the body [4]. In the case for mammal quadrupeds the legs require less torque to support the body at the cost of lower static stability.

Motion with four legs is more stable than with two legs as it is easier to shift the center of gravity (COG) of the body. This is also the reason why a baby learns to crawl with four limbs before learning to stand and walk with two legs. Stability is important for wild animals as they often need to move on uneven terrains. A good example for quadruped to handle uneven terrains efficiently is the climbing ability of mountain goats [5]. Mountain goats are able to climb steep cliffs in a very nimble manner. Unlike other quadrupeds like cats and squirrels that climb with the ability to grasp on climbing surfaces, mountain goats which is not really a climber have the ability to run at high-speeds over broken and uneven grounds.

1.2.1 Efficiency in locomotion and survivability

Stable locomotion and survivability are crucial for quadrupeds. Predators hunt for food while preys run away from predators to stay alive. Locomotion speed is important in the movement of quadrupeds as speed determines how fast a predator can catch a prey or vice versa. However, not every animal relies on speed for survival. For example, tortoise moves very slowly and its shell protects it well [6].

Although speed is an important factor for survival, predators are able to catch a prey running at a higher speed which is possible due to their agility. If a predator is able to accelerate to a higher speed than its prey in a shorter time, it can reach the prey before the prey manages to escape. A prey can outrun a predator by swerving in the right timing. An animal can have very high radial acceleration even with slower speed if the turning angle is large enough. The endurance of an animal to move in high speed also affects its survivability. It is inevitable that quadrupeds consume energy while looking for food.

1.2.2 Quadruped locomotion gaits

The locomotion gaits of quadruped can be categorized into statically stable moving gaits and dynamically stable moving gaits. In order for a quadruped to achieve statically stable locomotion gaits, three limbs must be in contact with ground at any instance during walking [7] which gives the gait a duty factor of greater than 0.75. The gait duty factor is the fraction of time during which a limb comes to contact with the ground in a walking cycle. The duty factor is in a range of 0 to 1, and a higher value indicates that the gait yields slower speed.

Quadruped can achieve dynamically stable moving gaits with two limbs or less contacting with the ground at any instance during motion. Some common dynamically stable moving gaits are trot and gallop. The dynamically stable moving gaits can be further categorized to running and walking gaits. They can be differentiated by the duty factor where walking has a duty factor greater than 0.5 while running has a duty factor lower than 0.5. Walking gait is the most common gait where at least one foot is on the ground at any instance during the course of walking. However, running and leaping gaits can have every limb off the ground at some instance during motion. Running and leaping gaits are faster as compared to walking but come with the price of higher energy consumption.

A quadruped is walking in a stable manner when its COG lies within the support polygon formed by the legs contacting the ground. The main objective of statically stable moving gaits is to ensure that the COG is always within the support polygon. For stable dynamic walking gaits, the support polygon may not exist as only one or two legs be in contact with the ground. Therefore, stability cannot be verified in the same way as for stable static walking gaits.

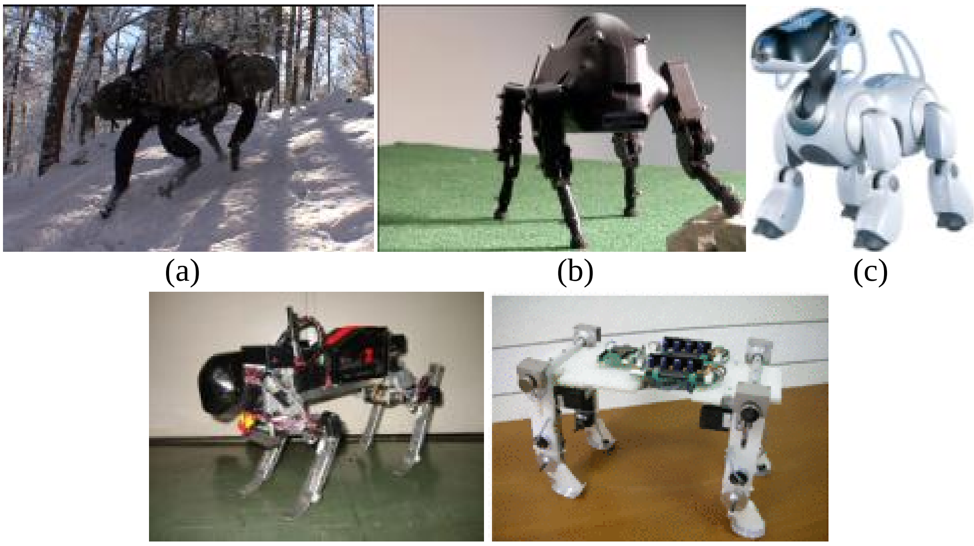
1.3 Quadruped robotic system

There are various quadruped platforms created for the study of quadruped behavior. BigDog by Boston Dynamics [8 - 11] is a quadruped robotic platform which has achieved stable locomotion through rough and uneven outdoor terrains. Big Dog is a self-contained quadruped which can move with a speed comparable to human walking speed and has the ability to recover from disturbance. Big Dog has hydraulically actuated joints. A smaller version of Big Dog named as Little Dog is developed to study the fundamental relationships among motor learning, dynamic control, perception of the environment, and rough-terrain locomotion [12, 13].

Tekken, a quadruped developed in Japan is designed based on the biological concept [14 - 17]. Tekken is a quadruped with 4 degree of freedoms (DOF) in each leg and is controlled by a neural system model consists of a central pattern generator (CPG), reflexes and responses. This is based on the concept of periodical joint movement in a walking animal. Tekken is capable of walking on an uneven terrain and also running on a flat terrain. Upgraded version namely Tekken 3 and 4 are also developed which are capable of stable walking for a longer duration [18].

AIBO, an electronic pet toy designed and manufactured by Sony has been used as a platform to test out various theories not only in walking gaits, but also in human-robot interaction and pattern recognition [19 - 22]. Each of AIBO legs has 3 DOF which makes up of a total of 12 DOF for the robot. AIBO is a platform which is quite widely used in research due to the easiness to obtain the platform even with some limitations such as the short leg length and fixed DOF.

CATIA has developed a quadruped robot which implements a slider crank mechanism on the leg [23]. This design ensures that the actuator of the hip joint can operate under a constant angular velocity without the need of reversing direction. Each leg of the robot is designed with two active actuators. With the hip joints operating under a constant velocity, the calculation for ankle joint movement is simplified. Fig. 1.1 shows examples of the current quadruped robotic platforms.



(a) (b) (c)

(d) (e)

Figure 1.1: Quadruped robotic platform. (a) Boston Dynamic Big Dog (b) Boston Dynamic Little Dog (c) Sony AIBO (d) Tekken 2 (e) CATIA's quadruped robot

The development of a quadruped platform involves several aspects including the mechanical design, control strategy, sensors fusion, power management and gait generation. In this dissertation, the focus is on the limbs configuration and gait generation.

1.3.1 Design of quadruped limbs

The design of a quadruped robot is important as the design will have an effect on the stability and affect the torque required to support the body. A widely used approach is to develop the quadruped robot based on the shape of an animal which is usually based on the skeletal approximation of an actual animal. Fig. 1.2 shows the skeleton configuration of a dog limb [23]. The skeleton configuration of a dog limb has a total of 4 parts, namely the scapula, humerus, anterbrachium and carpal. Based on this configuration, a robot limb will have four links and four joints. A quadruped robot that utilizes such a limb configuration will have 20 links and 20 joints for limbs. Therefore, this configuration is high in computational cost and difficult to control. In order to reduce the complexity of the configuration, approximations can be made by combining the two links into one. P.T. Doan et al. developed a quadruped robot based on the skeletal approximation of a dog [24, 25]. This quadruped robot has two links in each limb which is equivalent to three segments structure of an animal's leg. This design reduces the complexity of the leg configuration. The quadruped leg in discussion is shown in Fig. 1.3 [24].

Alexander Sprowitz et al. proposed a compliant quadruped robot based on a cheetah [26]. This robot utilizes a spring-loaded limb, which can modify limb stiffness by using a spring with variable stiffness. At higher-speed, the momentum-triggered limb compresses leading to smooth hip trajectory patterns. However, there is a tradeoff between the robot speed and limb stiffness, as a high stiffness limb works well in high speed locomotion but not in low speed. At low speed, the high stiffness of the limb causes the leg not to compress enough which can destabilize the robot. While at high speed, a low stiffness limb configuration can cause the robot to collapse. This implies that depending on the travel speed, limb stiffness can play a role in leg design.

Xin Li et al. summarized that quadruped leg design can be divided into four different types namely 1) total knee type, 2) wholly elbow type, 3) outside knee, elbow type and, 4) inside knee, elbow type (Fig. 1.4) [27]. The different choices in leg design eventually leads to a different standing up, balancing and locomotion strategies. By utilizing a homogenous design for forelimbs and hind limbs, the complexity in controlling the robot is reduced.

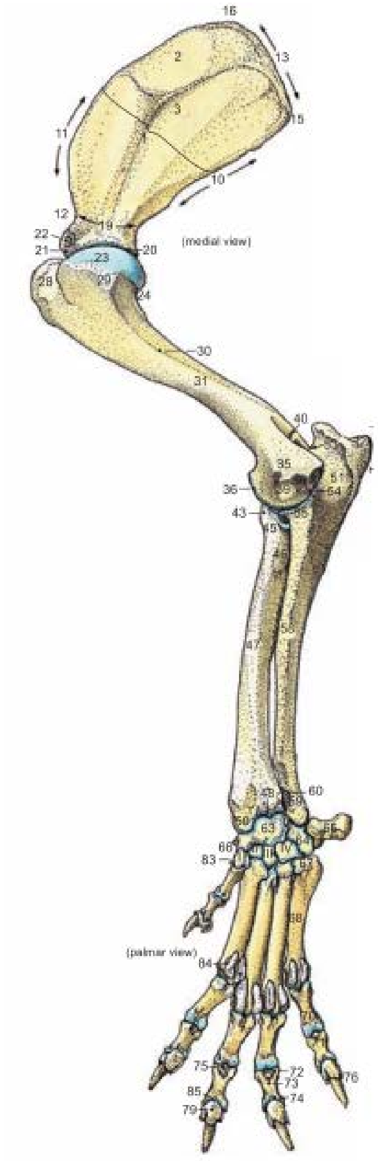


Figure 1.2: Skeleton configuration of dog limb

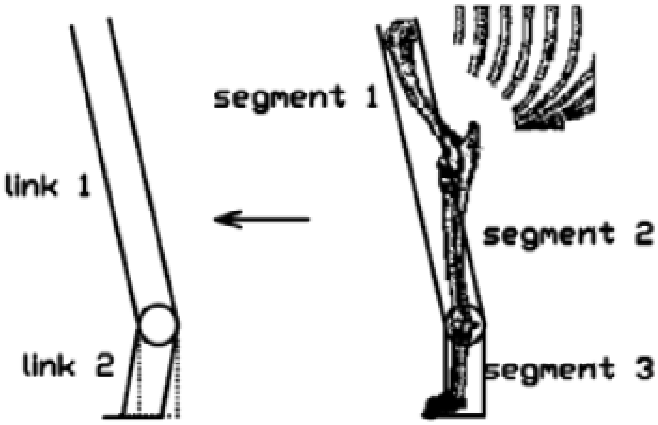


Figure 1.3: The quadruped leg

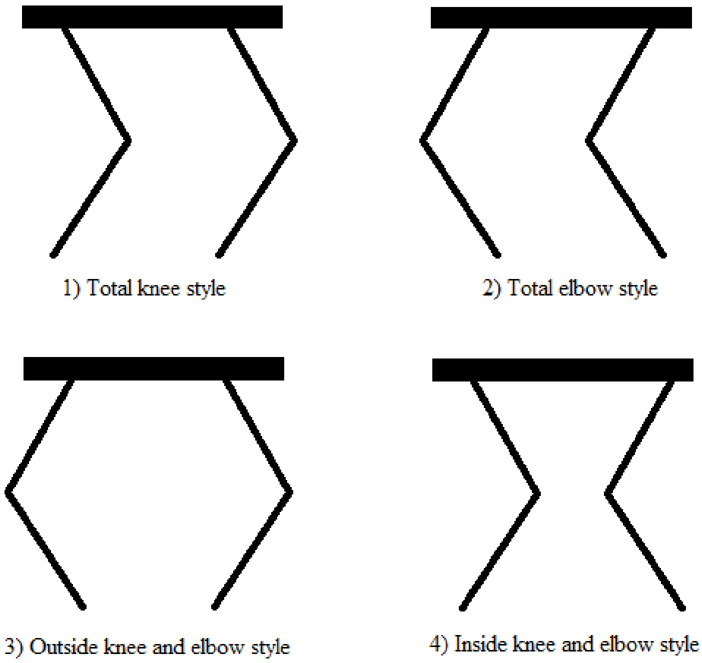


Figure 1.4: The design style of a quadruped leg

There are reported works on quadruped designs which are not based on biological models. Quadruped robot can be designed such that the leg configuration can overcome certain torque constraints to perform various tasks such as wall climbing [28]. Various quadruped platforms shown in [28] are designed to successfully achieve various tasks such as roller skating and stair climbing. The designs are based on some walking features such as: 1) robot can move over rugged terrains and not damaged while stepping over objects, 2) robot can realize a holonomic omni directional motion and 3) robot is a stable and active platform even on a rugged terrain.

1.3.2 Strategy for stable quadruped gait control

Although there are many quadruped robotic platforms developed, the challenge lies in achieving stable gait control. The quadruped locomotion is categorized into statically stable moving gaits and dynamically stable moving gaits. One of the main difference between these two categories is the number of legs contacting the ground. Due to the need for higher effort to achieve stable dynamically moving gaits, most of the quadruped robots are designed with statically stable moving gaits [29].

One of the most popular strategies in achieving stable gait control is to maintain the COG of the quadruped robot within the support polygon. The position of the COG is calculated in order to ensure that the COG of the quadruped robot always fall within the polygon that is formed by the legs contacting the ground [24, 30].

There is a research focusing on acquiring a mathematical model of the quadruped model and calculating the joint trajectories. Research in this area tends to obtain a kinematic model [31] or dynamic model [32,33] of the quadruped model and proceed to gait control using the model derived. The computational cost of the mathematical based control increases with the DOF.

There are reported works focusing on Central Pattern Generators (CPG) which utilizes neural like control laws capable of generating periodic signals for gait control [34]. In the real world, animals often move with periodic motion and by utilizing such an approach in quadruped robotic platforms can simplify the gait generation process. There are different strategies in CPG based gait control including CPGs generating toe trajectories and CPGs generating the different joint control signals [35].

As for dynamic walking gaits, a prominent approach is to design the robot platform such that it is able to achieve hopping behavior. Ioannis Poulakakis et al. experimented two variations of bounding gaits on Scout II robot (Fig. 1.5) [36]. The Scout II robot is designed such that each leg has the ability to generate a force which opposes the gravity force. The cycle of a bounding gait is shown in Fig. 1.6. Fumiya Iida et al. designed a quadruped robot based on the concept of spring loaded inverted pendulum (Fig. 1.7) [37]. With the aid of springs on the legs, the robot achieved bounding gaits without using any sensor information.

Figure 1.5: Scout II robot

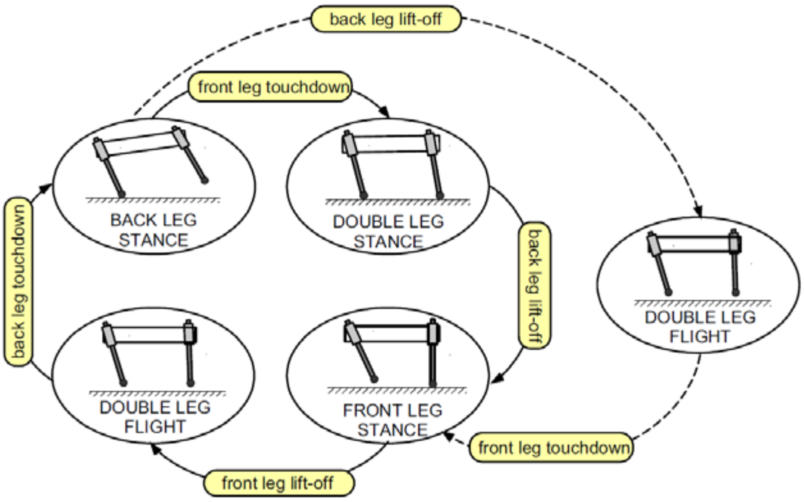


Figure 1.6: Bounding Gait

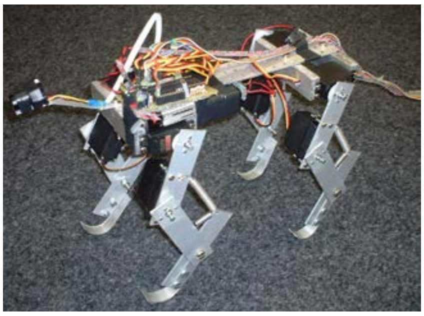


Figure 1.7: Quadruped robot based on spring loaded inverted pendulum

1.4 Multibody simulation and physics engine

Multibody system is used to model the dynamic behavior of interconnected bodies that are rigid or flexible and which may perform translational or rotational motion. Examples of actual system that can be modeled as a multibody system include automobiles, space structures and robots [38 - 40]. The history of multi body model simulation goes back to sixties of twentieth century for the first launch of artificial satellite "Sputnik" by the Soviet Union [41]. It is important to study the theoretical part of an engineering system before launching an actual platform to reduce unwanted accidents and ensuring smoother operation. Simulation give a wide range of option for solving many problems by investigating, designing, visualizing and testing a system before it even exists [42].

Robotic system is suitable to be modeled as a multibody system. Steven Dubowsky et al. presented a method for planning and controlling the motion of space robotic systems which are formulated as a multibody system [43]. The understanding of the dynamic behavior of the system is the key to solve problem related to the multibody system. Delia Paulina Aguirre et al. presented a work on multibody modelization of a biped robot in which the multibody model is assumed to be a tree-like model [44]. The dynamic behavior of the robot is taken care by utilizing a software that will generate dynamic equations for the model. Zoltan Nagy et al. describe the motion of a microbot by applying of nonsmooth multibody dynamics [45]. The proposed method for modeling is general and is able to be applied to large range of different robots.

For quadruped robot, work has been done on constructing a dynamic model to analyse the stability of a quadruped running controller [46 - 48]. The quadruped robot, Rush is constructed to study efficient running on flat and rough terrain. Through the simulation results, the quadruped robot is able to achieve self stabilization and energy efficient running gait. It is important to understand the dynamic behavior of the system before building the actual quadruped robot [64]. The general dynamic equation of the system can be expressed as below :

$$\tau = M(\theta)\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} + N(\theta, \dot{\theta}) \tag{1.1}$$

where τ is the vector of joint torques, θ is the vector of joint angles, $M(\theta)$ is the inertia matrix, $C(\theta, \dot{\theta})$ is the coriolis matrix and $N(\theta, \dot{\theta})$ includes gravity terms and other forces act on the joints [50].

Simulation can be done with the aid of numerical computing software such as MATLAB or some dedicated simulators. The dynamic of the system is taken care by the simulator. The Unified System for Automation and Robot Simulation (USARSim) is a robot and environment simulation tool which is based on a gaming engine [51, 52]. The robot models simulated with USARSim are shown in Fig. 1.8. USARSim is developed as a research and educational tool. Another notable simulation tool available is the Microsoft robotics studio (MSRS) [53, 54]. MSRS provides various built-in models for use. Various types of sensors are also available. The simulation environment of MSRS is shown in Fig. 1.9.

Figure 1.8: USARSim: Robot models

Figure 1.9: MSRS: Robot and environment

The dynamics of a robotic system can also be taken care by utilizing physics engine in the simulation. Physics engine incorporates laws from the physical world in a virtual environment. Properties such as rigid body dynamics including collision detection, soft body dynamics and fluid dynamics can be incorporated. The advantage of a physics engine is that it takes care of the model kinematics and dynamics. Physics engines are usually used in gaming and scientific simulation. Gaming engine requires real time simulation while scientific engine requires high precision simulation. The accuracy of simulation in gaming engine is not important as compared to precision simulation, hence it is important to choose an appropriate physics engine to ensure sufficiently accurate simulation results. In the field of research, high precision simulation is more important.

1.5 Quadrupedal gait generation in virtual world

The design of a quadruped robotic platform is inspired from the quadruped in natural world. The advantages of the quadruped in natural world including stability and the ability to walk on uneven terrains, is desirable in a quadruped robot platform. The mathematical model of a quadruped is complex and the complexity is directly proportional to the degrees of freedom.

Besides, the leg configurations are not considered within a mathematical model. This thesis focuses on achieving stable quadruped walking gait without deriving a mathematical model. The thesis also investigates the effect of the leg configuration in quadrupedal locomotion.

In order to achieve this objective, a simulated quadrupedal configuration is built. The quadrupedal configuration is built in a progressive manner. Single limb configuration is first built, followed by double limb configuration and finally the quadruped configuration. The optimal leg configuration design obtained is used in the two legged configuration and quadruped model. Genetic Algorithm (GA) is used to search for the optimal leg configuration design and stable walking gaits. The challenge narrows down to discover the proper cost functions for the fitness calculation.

1.6 Major contributions of the Thesis

The major contribution of the dissertation is the generation of a quadrupedal walking gait using GA. The gait generation of the quadrupedal configuration is simulated and the dynamics of the configuration is taken care of by the physics engine ‘Chrono::Engine’.

The simulation is carried out in three phases: (1) single limb configuration, (2) double frontal limb configuration with a support structure at the rear, and, (3) the quadrupedal configuration. The progressive manner in simulation increases the success rate of the final result. As the gait generation utilizes GA to achieve optimal results, identifying the cost functions for different simulation stages is important. The working cost function for different stages are identified.

In a single limb configuration, the joint functions which enable the single frontal limb configuration to get up from resting posture is optimized utilizing GA. The single frontal limb configuration is able to maintain the standing posture for a period of time. This finding leads to the ability of a single frontal limb configuration to self sustain and reduce the torque required while maintaining the standing posture.

The single limb configuration is extended to a double frontal limb configuration. The double frontal limbs are simulated with a support wheel structure at the rear. The optimized joint functions are obtained and applied on to the double frontal limbs. The double frontal limbs configuration is capable of achieving walking gaits even when there is a disturbance force applied at the rear. This finding shows that the double frontal limbs configuration is capable of working well with a support wheel structure in the absence of a pair of rear limbs.

The quadrupedal configuration utilizes a homogeneous configuration with the same configuration in each limb. The front limbs provide steering while the hind limbs provide the driving force. The hind limbs act as a driving force during both the standing up and walking phase of the quadruped. In order to achieve stability in walking, the hind limbs' driving forces need to be optimized.

1.7 Thesis Organization

The thesis is organized as follows. Chapter 2 provides the multi body model simulation and GA optimization. Simulation of single quadruped leg configuration is provided in Chapter 3. Chapter 4 describes the double frontal limb configuration and gait generation for standing up. Gait generation for walking with double frontal limb configuration is discussed in Chapter 5. Chapter 6 expands the approach to a quadruped configuration. The final chapter summarizes the work and possible future research directions are highlighted.

Chapter 2

Multibody simulation & Genetic Algorithm

2.1 Introduction

In this chapter, the utilization of physics engine with GA to solve a robotic related problem is discussed. A physics engine named Chrono::Engine is used to provide the simulation of rigid body dynamics needed for the experiment. The initialization of GA in the simulation environment and the process of evaluation are stated in this chapter. Lastly, a double joint pendulum problem is defined in order to test the feasibility of the Chrono::Engine with GA.

2.2 Multi Body simulation based on biological concepts

The quadrupeds in natural world have large number of joints and muscles. It is not possible to completely simulate quadrupeds which directly follow the configuration of an animal. An approximate model can be simulated where certain parts of the animal are ignored. In the nature world, animals move their joints through muscle contractions. However, these muscle motions are not easy to achieve in a mathematical or virtual model. Motors are used to make joints move. Certain joints in the model can be set as passive joints to reduce the number of actuators to be controlled.

2.3 Utilizing Chrono::Engine in the simulation environment

Chrono::Engine is utilized as the physics engine which will take care of the dynamics of the quadruped robot. Chrono::Engine is a physics engine which can perform dynamical, kinematical and static analyses for virtual mechanisms built of parts such as actuators, motors, springs and dampers. Chrono::Engine is able to provide simulation features in C++ projects. Together with Microsoft Visual C++, which acts as a compiler, a simulated environment can be built with the Chrono::Engine. Chrono::Engine has been used to solve engineering problems which include large cone complementarity problems, simulating large granular flows and solving large-scale, nonsmooth, rigid body dynamics [55 - 57].

2.3.1 Setting up the simulation environment

A simple simulation of a pendulum is set up to test the feasibility of the physics engine. Several aspects can be tested through the simulation of a simple pendulum system such as actuation, friction, collision and etc. Chrono::Engine provides several libraries which users can use to build the model. Table 2.1 shows the description of the most commonly used Chrono::Engine function in the simulation. Rigid body is the basic building block of every solid body needed in the simulation, including the quadruped's body and ground. Table 2.2 shows the description and initialization of commonly used rigid body in Chrono::Engine. Rigid bodies with different size and shape act as the components for the quadrupedal configuration. The joint connects links together to form the quadrupedal configuration and actuator is applied on each active joint. Fig.2.1 shows the user interface when working on the simulation. User will write and edit the programming code in the user workspace while the data feedback window and graphic interface will appear after the code is compiled. The data feedback window will show the needed information set by the user in numeric format while the graphic interface shows the actual model that is built and the models dynamic is updated from time to time showing the model moving.

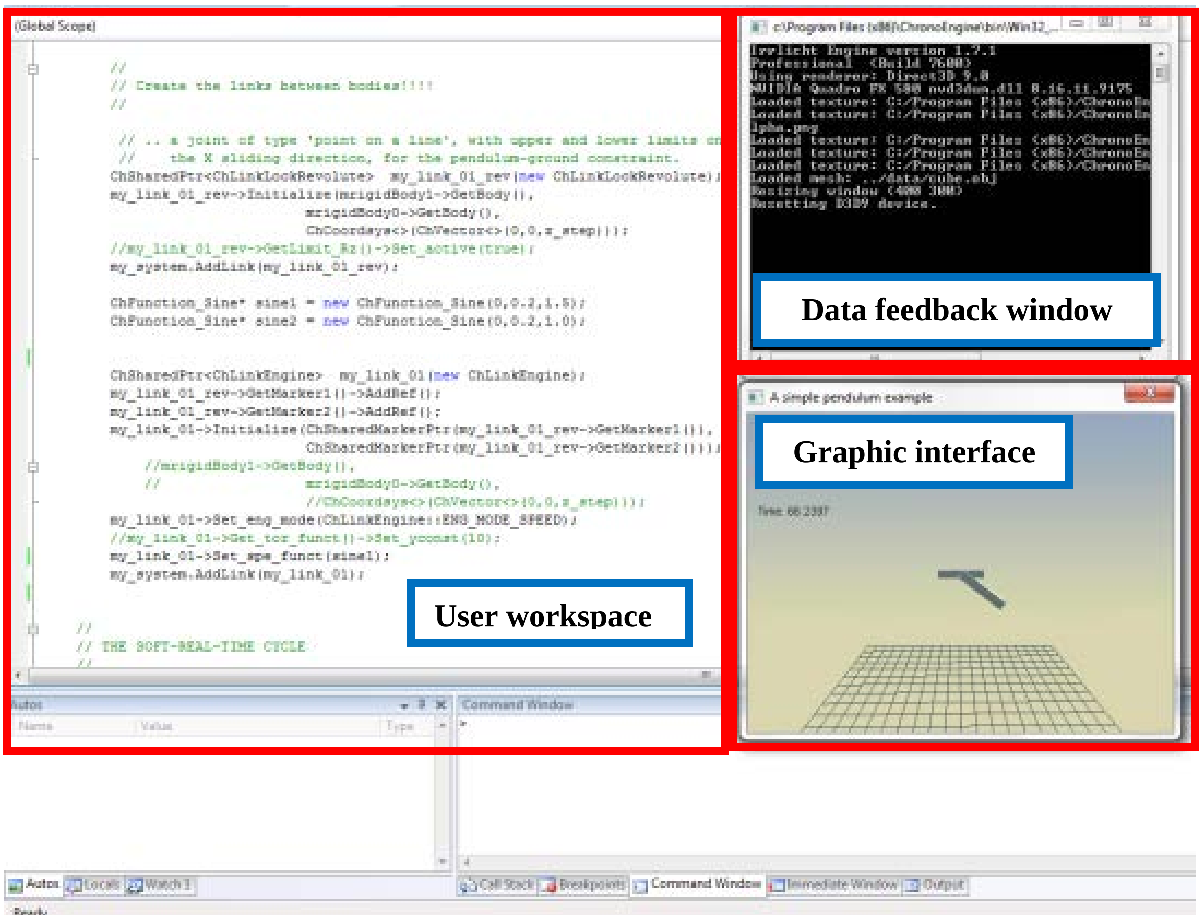
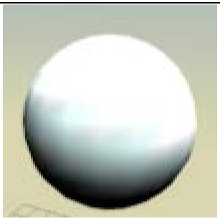


Figure 2.1: Multibody simulation with Chrono::Engine

Table 2.1: Description of common used Chrono::Engine function

	Library function	Description
Rigid body	ChBody	Different shape of rigid body can be created such as box, sphere, cylinder and etc. In Chrono::Engine, the position, size, dynamics can be defined.
Joint	ChLinkLock	Joint is created between different links to form a multibody system. Different type of joint can be created such as revolute joint, prismatic joint and etc. In Chrono::Engine, the position and orientation of the joint can be defined. Joints created are passive.
Actuator	ChLinkEngine	In Chrono::Engine, engine is applied on the joint created so that the joint can be actuated. Different engine control type can be defined such as torque and speed.

Table 2.2: Description and initialization of common used Chrono::Engine rigid body

Body type	Image	Initialization	Description
Box		Set position	Set the initial position of each body created. Position is defined by specifying the value in 3D Cartesian coordinate.
		Set orientation	Set the initial rotation of the body created. It is only needed in box and cylinder shape body. The orientation is defined by specifying the quartenion.
Sphere		Set friction	To set the kinetic and static friction of the body created.
		Set body fix	To set whether the body created is always fix in a specific location or is movable throughout the simulation.
Cylinder		Set collision	To set the body whether it is rigid or hollow.
		Set inertia tensor	Set the inertia tensor of the body created. If not specified, the initial value will be $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$.
		Set size	Set the size of the body created. For box and cylinder, the size is defined by specifying the value of x, y and z element. For sphere, the radius of the sphere will define the size created.

2.3.2 Assessment on multibody modelling: simple pendulum system

In the complete quadruped system, many joints will be in action and therefore the simulation software needs to provide ease in actuating every joint. Thus a simple pendulum system is built to test out certain aspect in the simulation software including rigid body creation, creating joint, actuation and collision model. Fig.2.2 shows a simple pendulum system created in Chrono::Engine. The simple pendulum consists of a support and a pendulum in which both of them are connected to each other using a revolute joint. The center of the joint is located on the center of the support which is represented using a blue circle in the figure. The collision model of the support is disabled as the friction and collision between the support and the pendulum system will cause inaccuracy in the simulation results. Therefore, some links which are not contacting with ground have the collision model disabled to avoid unintended error in the simulation.

A joint can be actuated by applying an engine on it. There is a variation of actuator type that is available in the Chrono::Engine and the most commonly used is the torque and speed actuation. Torque engine works by controlling the torque applied on the joint to move the pendulum. However, the drawback of torque engine is that large torque will break the joint and collapse the simulation. Speed engine in the other way is easier to be used and the appropriate torque required to be applied on the joint is calculated by the system.

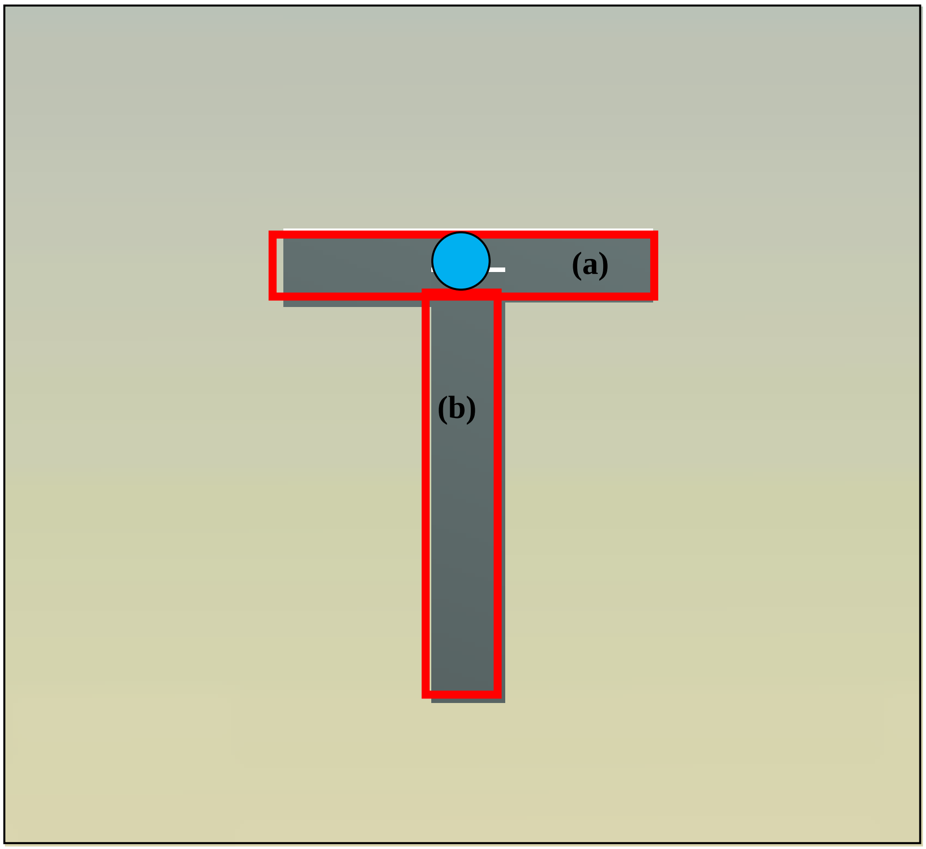


Figure 2.2: Simple pendulum system created in Chrono::Engine (a) support (b) pendulum

2.4 Basic Concepts of Genetic Algorithm

Genetic Algorithm (GA) is a search heuristic inspired by the natural evolution which is usually used to search for an optimal solution in a given problem [58, 59]. The idea of GA is to generate a population of individuals (chromosomes) that represents the candidate solutions and evolve the population by crossover and mutation over a number of generations until the stopping criteria is met. The evolution lies in selecting the fittest individuals in current generation as the parents of the next generation. The crossover operator is done by combining two or more parents to form one or more offspring while the mutation operator is done by making small random changes to an offspring. The purpose of crossover is to produce offspring population which is different from the parent while the purpose of mutation is to ensure diversity in the solution. Fig.2.1 shows a simple flow chart of GA.

The designing of GA for a given problem involves selection of following steps:

1) **Initialization**: Initialization in GA is the process of encoding the candidate solutions into a form that can be utilized in a computer. Individuals in the candidate population is named as chromosomes and each individual can have several bits named as genes. Each gene holds the encoded information of candidate solution to be optimized. The encoded populations can be in the form of binary, integer, float, character or strings depending on the need or type of the problem. The initial populations are usually created randomly. However, some researchers use good sets of results in the first evolutionary run to seed the initial population [60].
2) **Selection**: The purpose of selection operation is to emphasize the fitter individuals in a population so that offspring will have higher fitness [61]. There are several different selection methods. For example, in the roulette-wheel selection, fitness of each individual is evaluated and sorted in descending order. The normalized accumulated fitness for current generation is calculated (fitness value of each individual in current generation is summed up and normalized to unity) and a random number r in the range of 0 and 1 is selected. The first individual with accumulated fitness value greater than r is selected. Another example of selection method is the tournament selection where a random subset of individuals is chosen from the population to compare fitness values. The fittest individual is retained and the procedure is repeated until there is enough individuals chosen.
3) **Elitism**: During the process of selection, the fittest individual have the highest chance to be selected for next generation. However, there is a possibility that the fittest individuals can be dropped. Therefore, elitism is implemented to retain some best individual in the current generation so that the performance of GA can be promising.

4) **Crossover**: The main idea of crossover operator is to produce an offspring from more than one parents after selection. This process can be seen as exchanging information among different individuals in a population. There are many crossover techniques and examples include single point crossover, multipoint crossover and uniform crossover. Single point crossover is the most commonly used operator where a single crossover position is chosen randomly. Multipoint crossover occurs by evaluating each bit (gene) in an individual with a crossover probability *p*. When crossover is triggered, a particular bit is swapped between the two parents. As for uniform crossover, each bit in the parents has 50% chance to swap with each other.
5) **Mutation**: The mutation operator is to ensure diversity in population and avoid local minima. Depending on the encoding of chromosomes, different mutation technique can be used. However, a mutation probability which is not large is set so that mutation does not occur too often which can prevent GA to search for an optimal solution. In the binary encoding of chromosome, the bit where mutation is triggered is flipped. As for real value encoding, a small variation of the value is added to the bit.
6) **Termination**: The evaluation of GA is terminated when the exit criteria is met. The commonly used exit criteria are (a) meeting a fixed number of generations, (b) candidate solution yields a fitness value that satisfies the minimum criteria, (c) manual inspection by user and (d) a combination of several different exit criteria.

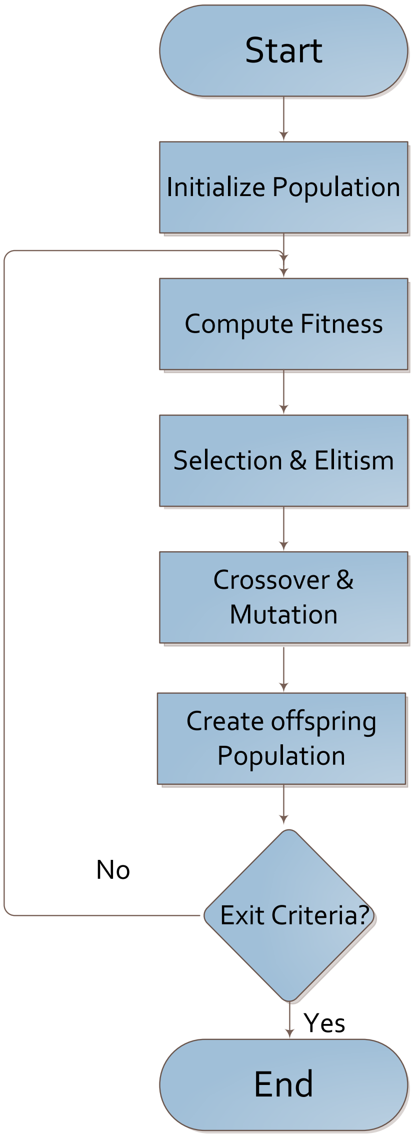


Figure 2.3: Simple flow chart of GA

2.4.1 Quadruped Simulation: Deviation from Numerical analysis

Quadruped standing up and walking are simulated in a virtual world abiding the laws of the physical world. The physics enginee namely 'Chrono::Engine' is used to manage the laws of the physical world. The simulated model's kinematic and dynamic are handled by the simulation software and the physics engine. The focus of the research in this thesis is to obtain stable walking gaits through an evolutionary process. In the numerical analysis, each joint's motion is calculated by a set of formula. In this research, a set of joint functions is generated and the candidate solution is evolved based on certain cost functions. Joint function is a specific function that is being applied on a joint to define the motion of the joint. There are torque function and speed function that can be applied on a joint. Torque function define the torque profile of the joint over time while speed function define the speed profile of the joint over time. By applying joint function on a joint, the joint will start moving to meet the specific function's profile. The focus in this work is cost function to evolve candidate solutions through GA in a virtual world abiding by the laws of the nature, which is a deviation from the standard mathematical model based simulation.

By utilizing relevant cost functions, GA searches for optimal joint function parameters. Joint functions utilized include pulse, sinusoidal and quasi-sinusoidal functions. The fitness of a candidate solution is evaluated in simulated virtual world.

2.4.2 Genetic Algorithm Initialization

The initialization of GA population is the process of encoding the candidate solution so that it can be processed by GA. In simulation, joint functions are applied to active joints and the parameters of the join functions are searched for. The candidate solution will have optimal parameters for the joint functions. In the Chrono::Engine, different joint functions are called from a library and applied to active joints where different joint functions have different initializations. Fig.2.4 shows an example of chromosome encoding. Joint functions considered include pulse, sinusoid and quasi-sinusoid. These joint functions are defined with the parameters amplitude, frequency and phase difference. Each joint function is encoded into more than one gene and each chromosome holds information for several joint functions.

Initialization of the population includes defining the population size and gene encoding method. The population size is set to 100 while each gene is initialized as a randomized real number. A random real number between -1 and 1 with a resolution of 0.01 is utilized to represent a joint function. The normalized join functions are calibrated with relevant multiplication factors to obtain the joint functions for the quadruped.

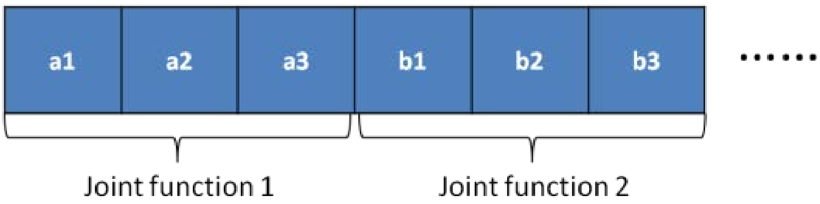


Figure 2.4: Example of chromosome encoding for GA

Selection is the process of selecting individuals from current population to breed a new generation. Fitter individuals are more likely be chosen and thus a fitter generation is bred. Tournament selection is used as the selection method where two random individuals are chosen from the current population (parent population) and the individuals with better fitness are selected into the offspring population. Selection is repeated until the offspring population is full. As the size of population is set to 100, tournament selection is terminated when the offspring population reaches 100. In order to avoid a fittest individual to be dropped out during selection, elitism is implemented. The parent population is sorted in descending order where individuals with better fitness are placed at top of the list. The top 5 individuals with highest fitness are chosen to the offspring population.

During the crossover process, two adjacent individuals are chosen from the offspring population to exchange information. Fig. 2.5 shows how an individual is chosen for crossover. Crossover rate is chosen as 85% which means that two chosen individuals will perform crossover with a chance of 85%. When crossover is performed, every gene from individual A has 50% chance to swap with the gene in same position of individual B. Fig. 2.6 shows the pseudo code of the crossover operator.

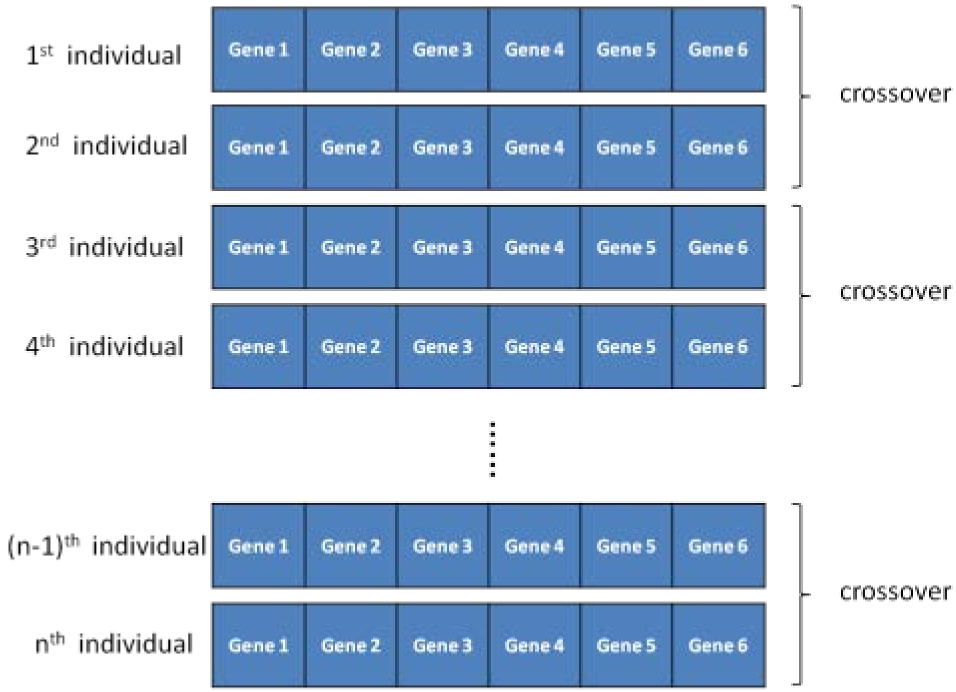


Figure 2.5: Crossover operator

```
REPEAT
   Select individual i and individual i + 1
   Select a random number between 0 to 1
   IF random number < 0.85
        REPEAT
                Swap gene j of individual i with gene j of individual i + 1 with 50% chance
        UNTIL every gene in the individual is evaluated
   i = i + 2 ( proceed to next set of crossover )
UNTIL i equal to population size
```

Figure 2.6: Pseudo code of crossover operator

Every gene in each individual is evaluated and mutation occurs with a chance of 5%. If mutation occurs, a small random real number is added to the respective gene. Equation 3.1 shows how a gene is mutated.

$$gene = gene + (random\ number\ between\ -0.25\ to\ 0.25)/(generation + 1) \tag{2.1}$$

The value of the gene mutated is diminished proportional to the generation number. This is to prevent large changes in gene value in later generations which can prevent GA from achieving optimal results. As the gene is initialized in the range of -1 to 1, whenever a gene is mutated to a value outside of the range, wrapping occurs. For an example, if the mutated gene has a value of 1.1, the wrapped value becomes -0.9. Lastly, GA is terminated when it reaches the maximum generation.

2.4.3 Case Study: Applying GA in multibody simulation

A simple problem is defined to study the feasibility of implementing GA in a simulated environment adhering to the laws of physics and investigate the effect of cost functions on fitness evaluations for each candidate solution. A simple multibody model with two links and a compressible spring is created as a suspension between the foot and the links. Fig. 2.7 and 2.8 shows the isometric and Chrono::Engine views of the two link configuration. The model has two active rotational joints and one passive prismatic joint while the rotational joint is speed actuated. Each rotational joint is actuated using a simple sine wave function which is defined by specifying its amplitude, angular frequency and phase difference. The sine wave function is as shown below:

$$y(t) = A \cdot \sin(\omega t + \emptyset) \tag{2.2}$$

where $y(t)$ is the joint function, A is the amplitude, ω is the angular frequency and $\emptyset$ is the phase difference. The scenario is created such that when each active joint is moving, the spring is to compress not shorter than 30% of its initial length (when the compliance is in relax state, its length is 100%). The resting length of the spring is set to unity. The candidate solution is a chromosome with the parameters of joint functions. Each joint function has 3 parameters, and the candidate solution has a total of 6 parameters to optimize.

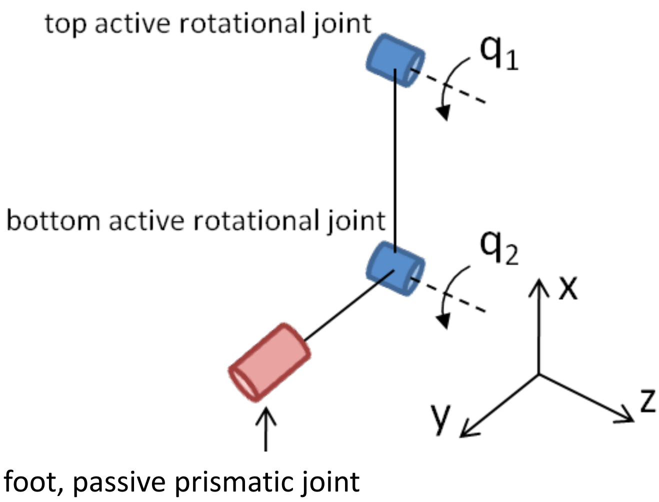


Figure 2.7: Isometric view of the two link model

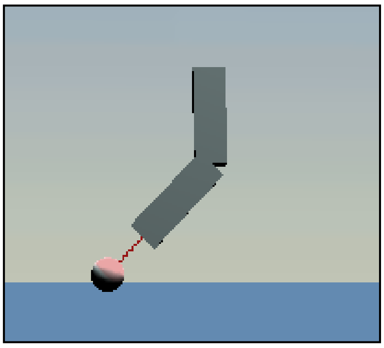


Figure 2.8: Chrono::Engine view of the two link model (side view)

Fig. 2.9 shows the flowchart of the two link multibody problem. The population size is set to 100. In each generation, there are 100 evaluations to perform and the stopping criterion is when GA reaches 50^{th} generation. In order to search for an optimal solution, the fitness of each candidate solution is calculated after 6s of simulation based on the compliance spring length. The fitness is unity if the spring is compressed such that the shortest spring length throughout the evaluation reaches 30% of its initial length at any time instance during the course of simulation. The fitness for each individual is calculated as below:

$$Set\ x = shortest\ spring\ length$$

$$fitness = \begin{cases} 1.3 - x, if\ x > 0.3 \\ 0, if\ x < 0.3 \end{cases} \tag{2.3}$$

A lower limit and a upper limit is applied on every parameter. The limits are applied on joint amplitude to avoid overshoot while limits are applied on joint frequency and phase difference to avoid wrap around value. Table 2.3 shows the parameter units and the searching range for each gene in the candidate solution. Table 2.4 shows the GA parameters for the case study.

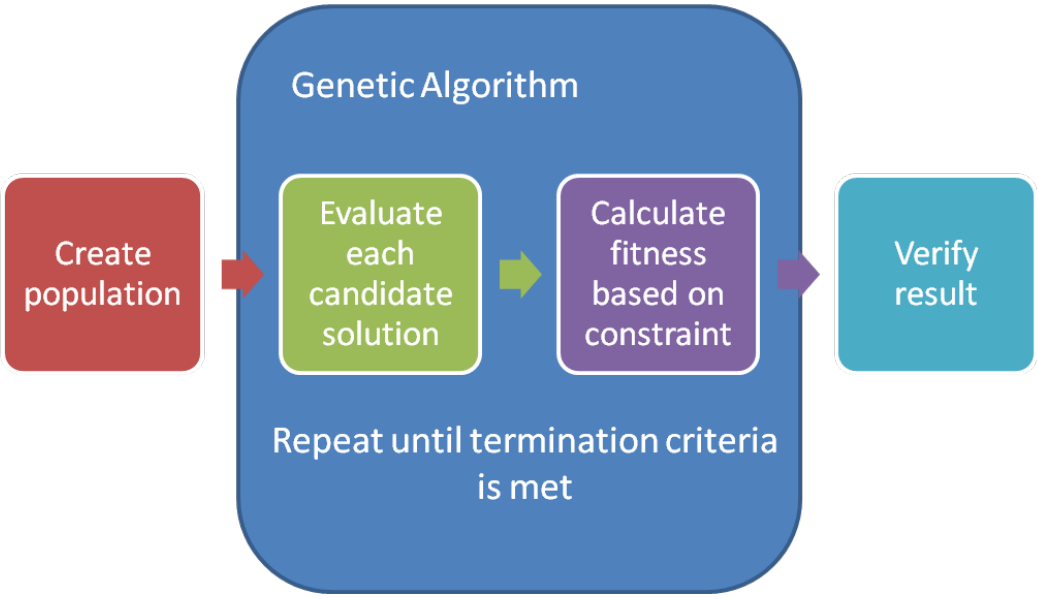


Figure 2.9: Flow Chart of the two link multibody problem

Table 2.3: GA parameter units and searching range

Parameter	Units	Lower limit	Upper limit
Top joint amplitude		-2.0	2.0
Top joint frequency	rad/s	0	1.6
Top joint phase difference	rad	0	1.6
Bottom joint amplitude		-2.0	2.0
Bottom joint frequency	rad/s	0	1.6
Bottom joint phase difference	rad	0	1.6

Table 2.4: GA parameters for the case study

Population	100
Chromosome Length	6
Exit Criteria	5000 evaluations
Crossover rate	85%
Mutation rate	5%

The fitness trend for this problem is shown in Fig. 2.10. The population of candidate solution is able to achieve unity fitness after 8^{th} generation. Unity fitness is the highest value for the fitness and this shows that GA is able to search for the best solution for this problem. The motion sequence of the double pendulum is shown in Fig. 2.11. The maximum spring compression occurs at the 5^{th} instance. It is shown that the spring is never fully compressed throughout the simulation and this shown that GA is able to achieve the objective. The overall computation time for the simulation is about 2000 seconds.

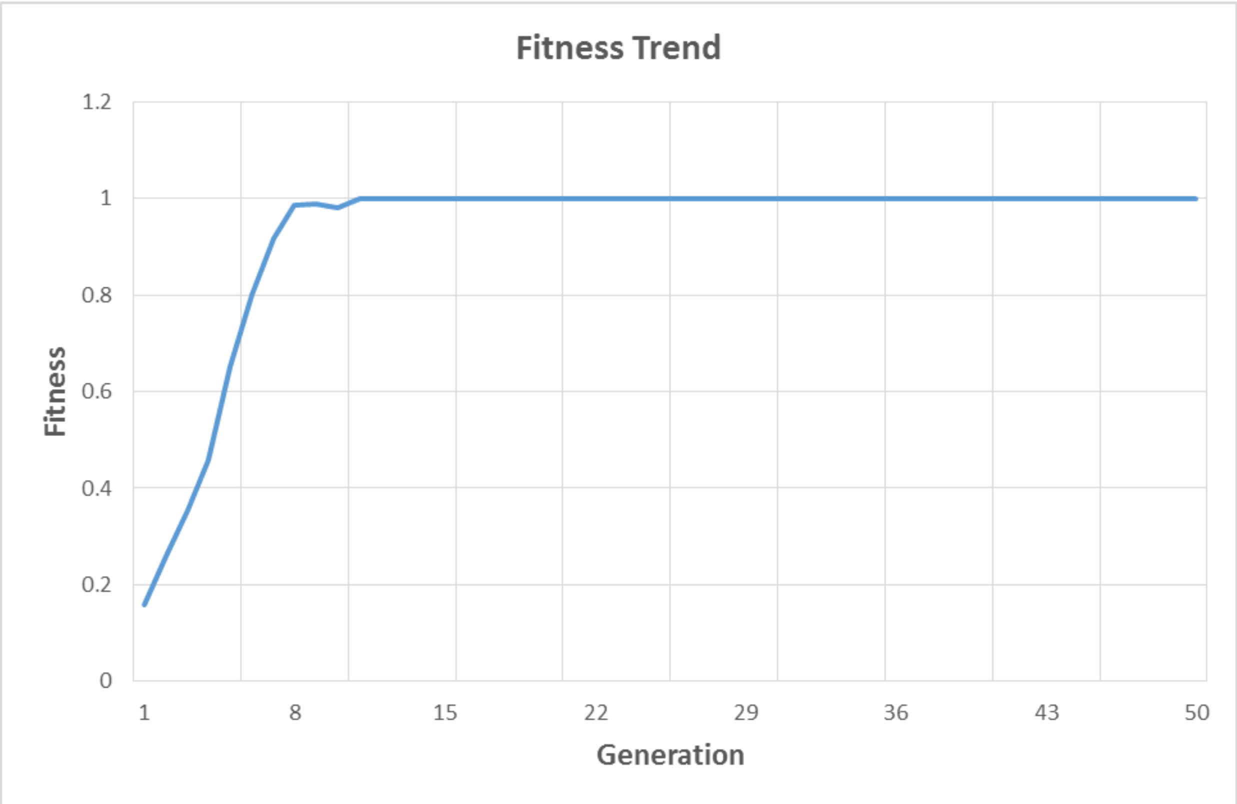


Figure 2.10: GA Fitness trend for two link inverted pendulum

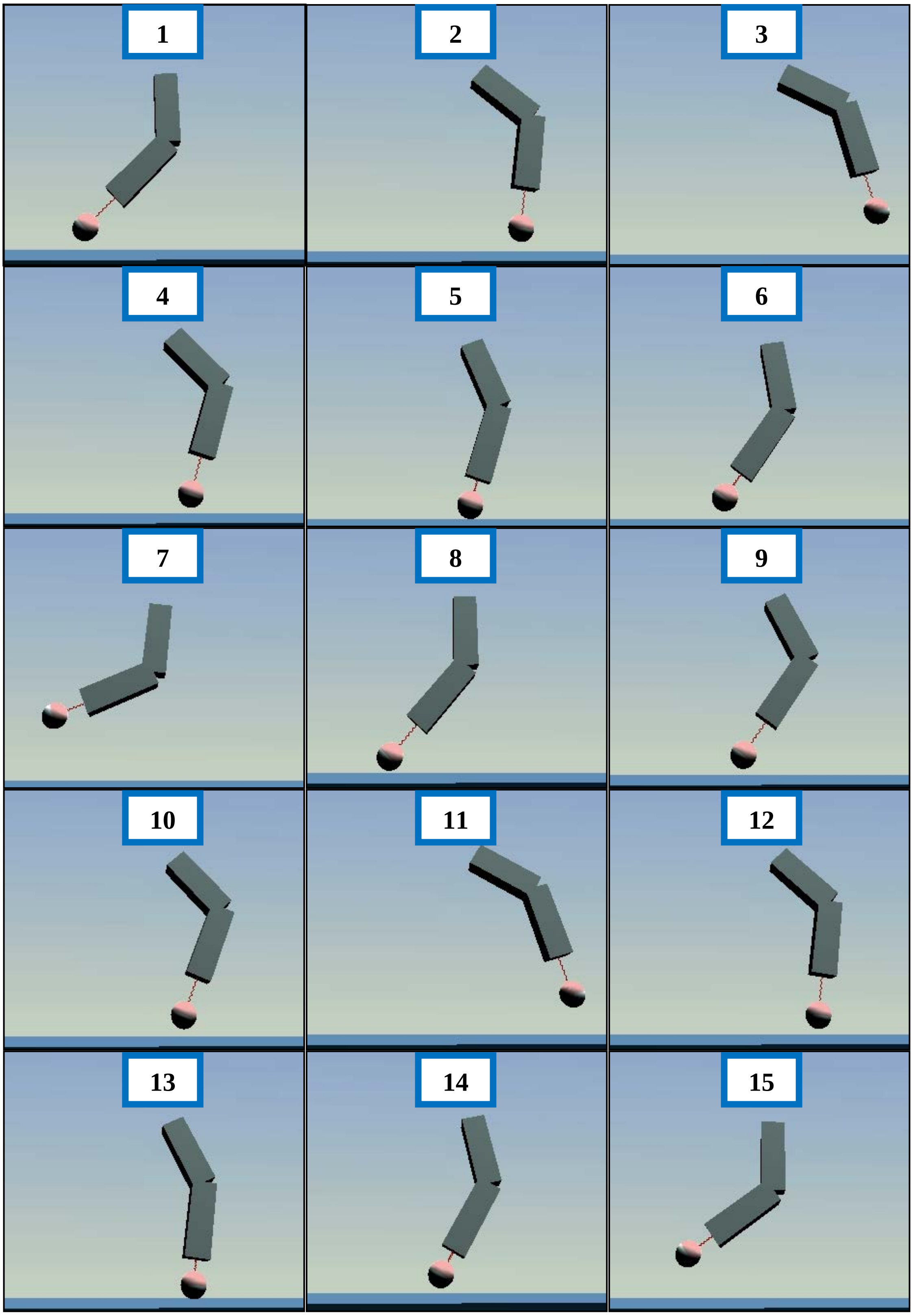


Figure 2.11: Motion sequence of the double pendulum case study

2.5 Summary

In this chapter, Chrono::Engine is introduced as the physics engine used to take care of the dynamic behavior of the simulation. GA is utilized in conjunction with Chrono::Engine to search for the optimal solution for a given problem. As the objective is to generate a walking gait, GA is to search for the optimized joint functions. A simple double pendulum problem is defined to test the capability of Chrono::Engine in conjunction with GA. The result shows that GA is able to evaluate and evolve the candidate solution to an optimal solution in the Chrono::Engine environment.

Chapter 3

Simulation of single quadruped limb

3.1 Introduction

In this chapter, the design of a single quadruped limb configuration is discussed. A single quadruped limb configuration is simulated and GA is utilized to search for optimal gait sequences to enable the single quadruped limb configuration to stand up from resting posture.

3.2 Single quadruped leg configuration inspired from nature

The single quadruped limb configuration is based on the forelimb of a dog (Fig. 3.1). In order to reduce the complexity of the leg configuration, an approximate leg configuration with reduced links and joints is defined. The approximate configuration of the single limb is shown in Fig. 3.1 (b) in which the skeleton configuration of the dog is approximated as a rectangular cube [63]. In the dog anatomy, the scapula is a bone which is attached only by muscle to the body torso and is extended laterally from the torso [62]. This leads to the fact that the scapula-body joint (joint that connect the body and scapula) and shoulder joint (joint that connect the scapula and humerus) do not lie in the same sagittal plane. Therefore, the scapula is retained in the approximate configuration. The foot of the single forelimb configuration acts as a suspension for the forelimb. The scapula and humerus are two single links each while the radius, ulna and metacarpus are approximated as a single link. The rationale is to limit the number of DOF of the limb to reduce the complexity of the configuration. Therefore, the forelimb is considered as a 3 links configuration.

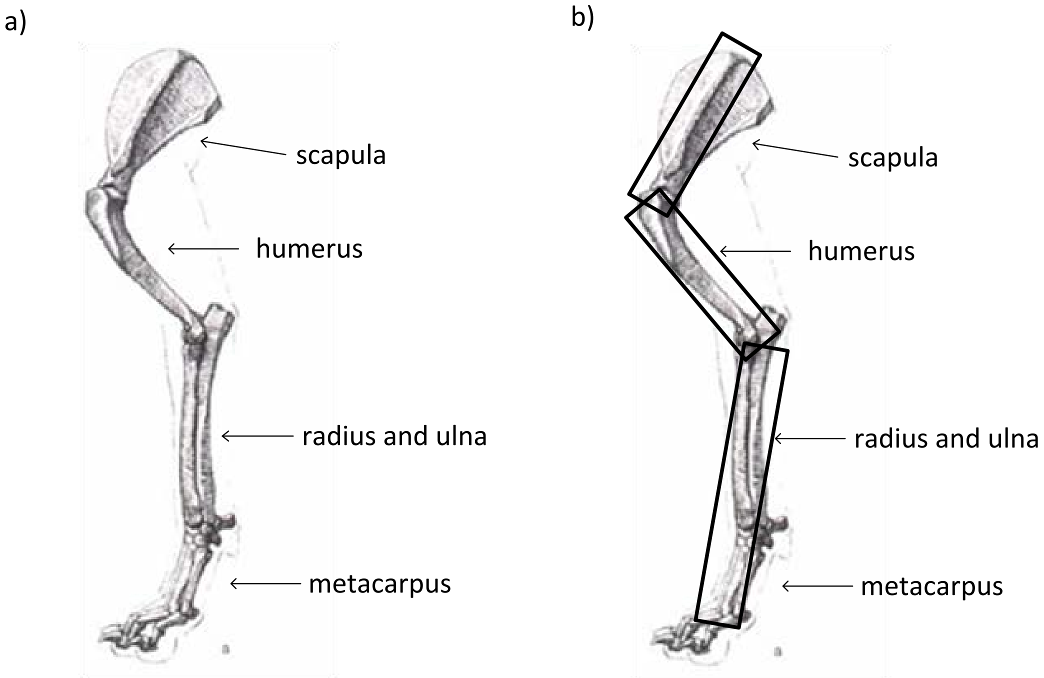


Figure 3.1: The skeletal configuration of the single forelimb (a) the skeletal configuration (b) approximate configuration to be simulated

3.2.1 Single quadruped leg configuration

In the simulation, each link is created as a rectangular cube and the foot as a sphere. Fig. 3.2 shows the configuration of a single forelimb in side view and front view where (i) is an active revolute scapula-body joint, (ii) is an active revolute shoulder joint, (iii) is an active revolute elbow joint, and, (iv) is a passive translational joint. Fig. 3.3 shows the isometric view of the single forelimb, where L_1 is the scapula link length, L_2 is the shoulder-elbow link length, L_3 is the elbow-foot link length and θ is the slant angle. A suspension spring is added between the foot and the elbow-foot link.

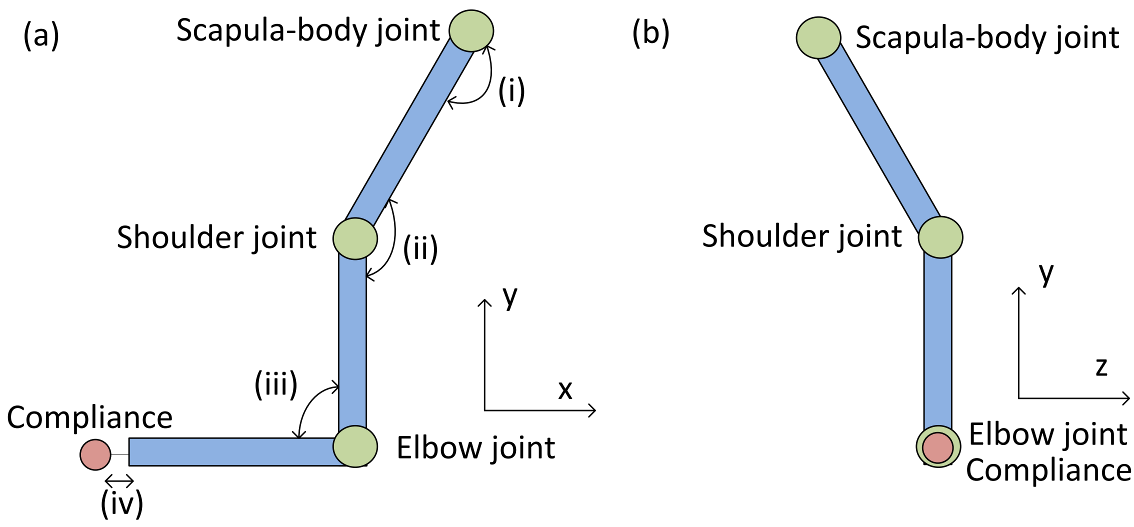


Figure 3.2: Chrono::Engine model of a single forelimb in rest posture (a) side view (b) front view

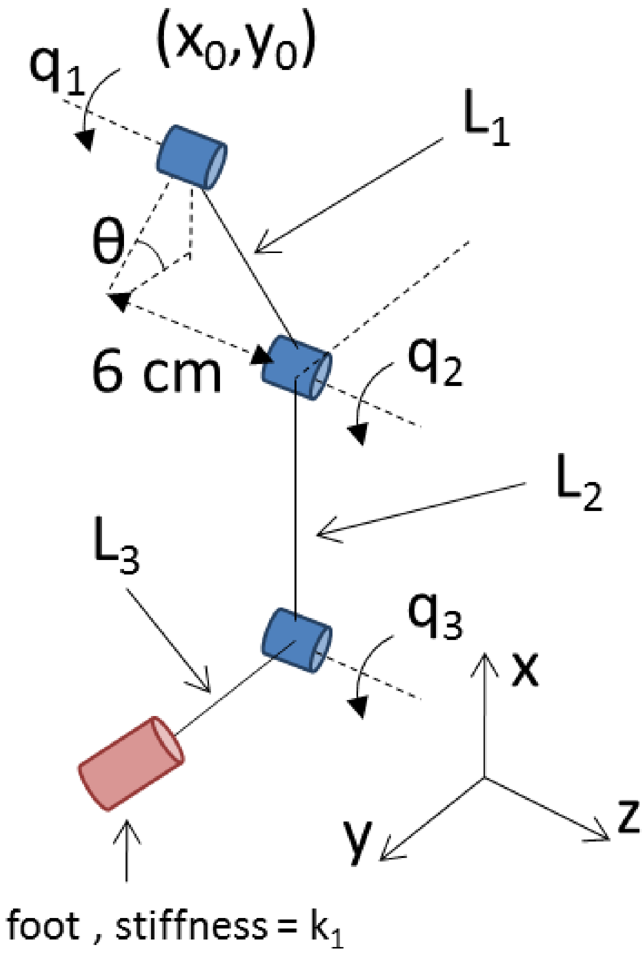


Figure 3.3: Isometric view of a single forelimb

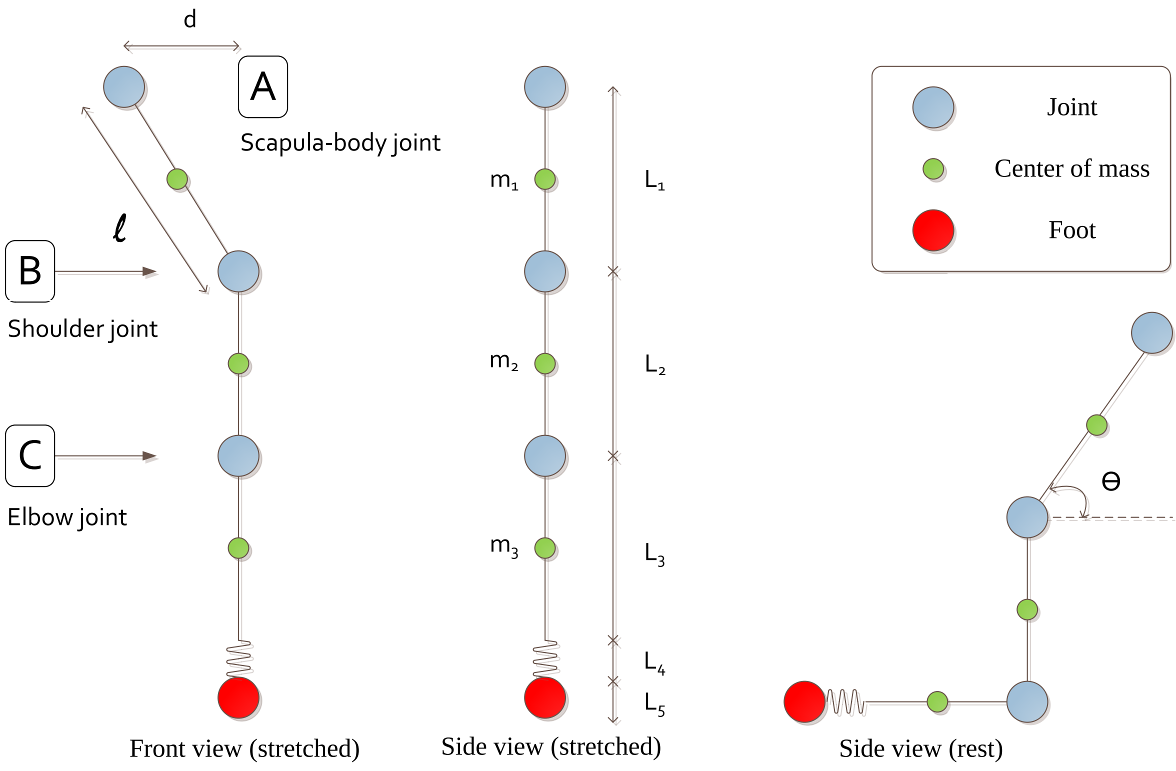


Figure 3.4: Single leg configuration: mass distribution

Fig 3.5 shows the single limb configuration and simulation framework. To start with, the single forelimb configuration is simulated. The single forelimb configuration has the mass distribution and link lengths as shown in Fig. 3.4. The parameter values of the single forelimb configuration are shown in Table 3.1. For simulation, the center of masses are considered at the middle of each link and the joints are considered as massless. θ is the slant angle that provides the initial resting posture. Joint A is the scapula-body joint, joint B is the shoulder joint and joint C is the elbow joint. These joints are able to do pitch rotation. Joint A which is the scapula-body joint is able to move freely in sagittal plane. The objective of the single forelimb simulation is to optimize the parameters of the joint function which enable the single forelimb configuration to stand up and maintain its standing posture for a period of time.

Table 3.1: Parameters of the single limb configuration

L_2	0.1 Meter
L_3	0.1 Meter
L_4	0.01 Meter
L_5	0.01 Meter
d	0.06 Meter
m_1	0.25 Kg
m_2	0.25 Kg
m_3	0.25 Kg

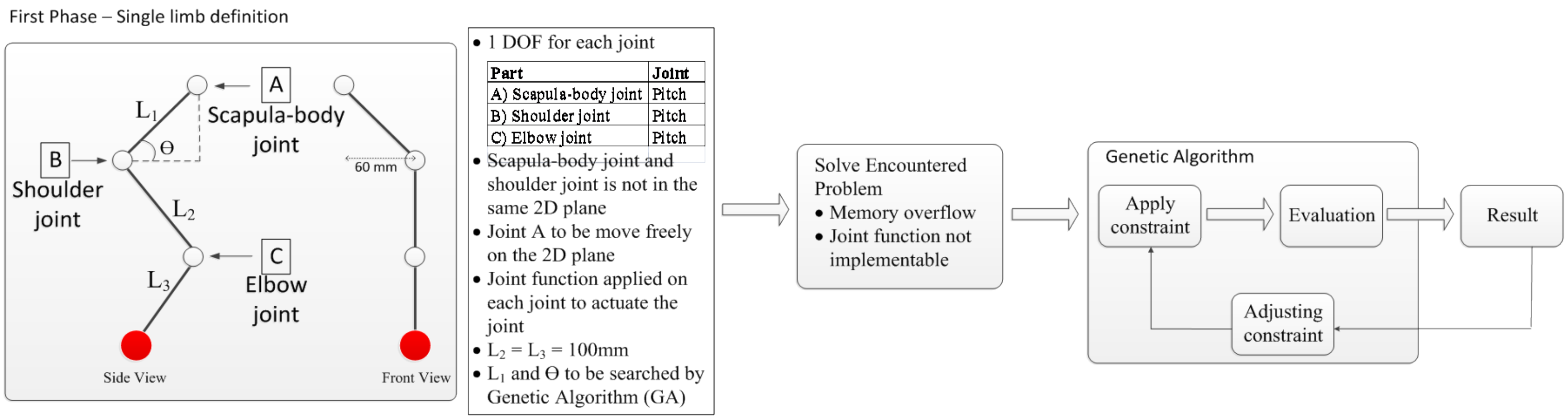


Figure 3.5: Framework of first phase – single limb configuration

A simple support is made at point (x_0, y_0) as shown in Fig. 3.3 which can hold the single forelimb model during the course of standing. The objective is to justify that the single forelimb model is able to stand up and maintain its standing posture for a reasonable duration. The lengths of the shoulder-elbow link and elbow-compliance link are set as 0.1m. The distance between the scapula-body joint and shoulder joint is set to 0.1m in the X-Y plane and 0.06m in Y-Z plane. The scapula-body joint and shoulder joint are not in the same sagittal plane. The advantages of such a design are the actuator at the body requires lesser torque to rotate the limb and the quadruped that formed with four such limbs design will be more stable. Similar to reptiles, the support polygon formed by the feet is larger so that the quadruped is more stable when its COG shifts.

As stated in section 2.4.3, joint functions are the function that applied on the joints to actuate the joints. The joints for the single forelimb is speed-controlled and the joint function considered are as shown in Fig. 3.6. The speed function is defined by amplitude and duration where a represents the amplitude and t represents time. The duration between t_{start} to t_{1i} is the same as the duration between t_{2i} to t_{end}. By using such joint functions, the joints rotate only for a short period and the optimal joint functions will cause the joint to stop moving after the configuration achieve standing posture. The joint functions are waveforms with smooth gradients which can avoid calculation errors in simulation (differentiating abruptly changing functions provides large or infinite values which can cause error in calculations).

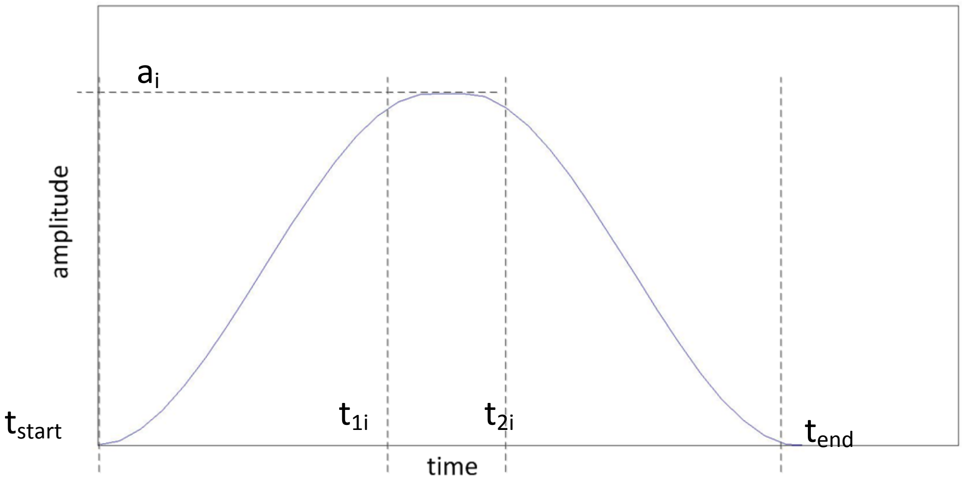


Figure 3.6: Joint function applied on single forelimb model

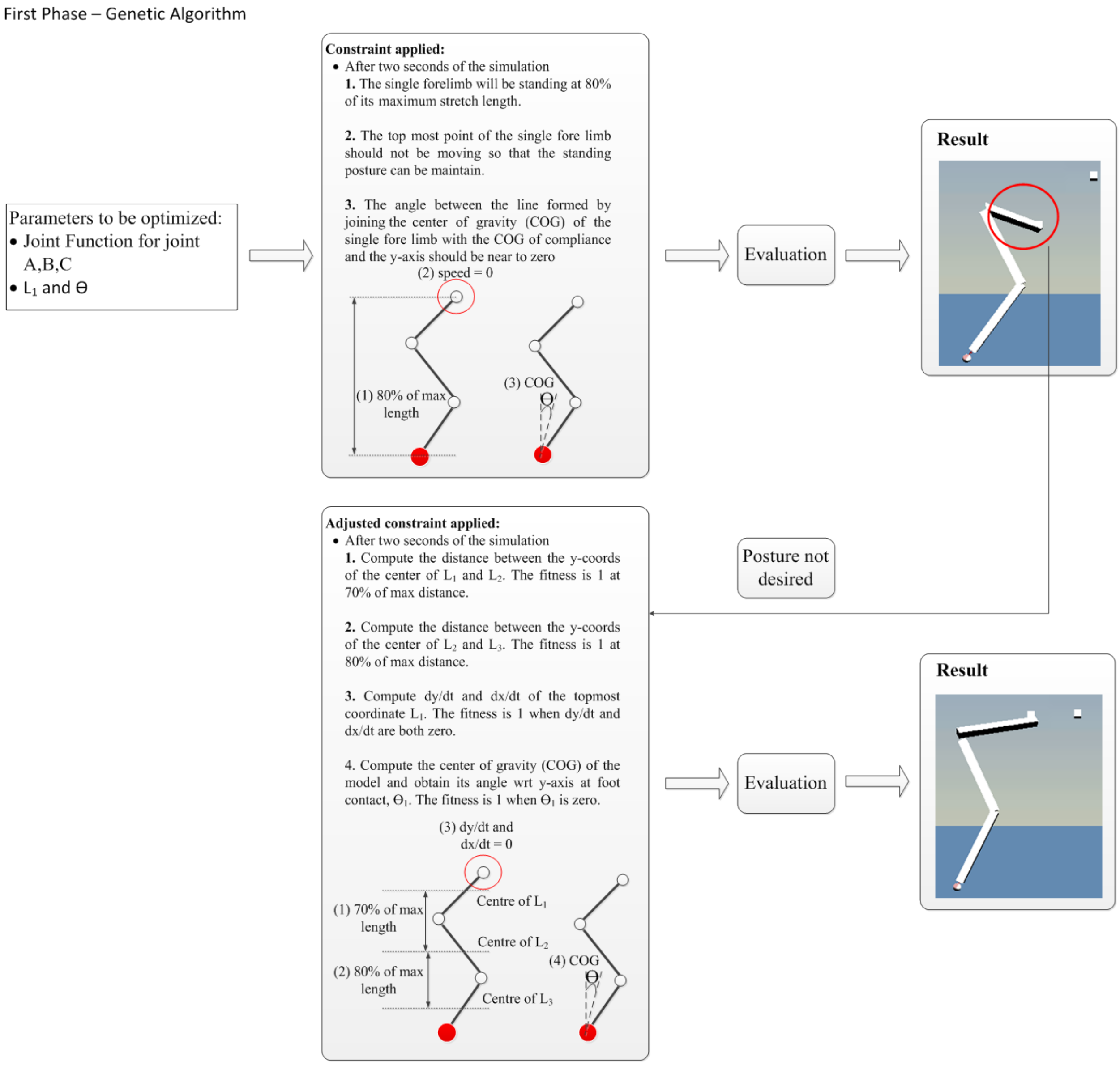


Figure 3.7: Framework of first phase – Genetic Algorithm

3.3 Genetic Algorithm and cost functions

The slant angle θ and the length of scapula link ℓ are not known initially and are to be optimized using GA. The objective of the GA is to search for the optimal slant angle of the scapula-shoulder link and the joint function for the single forelimb configuration to stand up from a resting posture. The speed function that applied to the joint of the single forelimb configuration is defined by three parameters which are amplitude a, time t_{1i} and t_{2i} (as $t_{start} = 0$ and $t_{end} = t_{1i} + t_{2i}$). A single forelimb has three joints and a total of 9 parameters for joint speed, 1 parameter for scapula link length and 1 slant angle which are 11 parameters in total to optimize. The population size is 100 with 11 genes. The evaluation is done by applying the parameters of a candidate solution to the single forelimb configuration and let the simulation to run for 2 seconds. The fitness of the respective candidate solution is then calculated based on certain criteria.

The fitness criteria is set to evaluate a candidate solution whether it is able to let the single forelimb configuration to move to standing posture after 2 seconds of simulation time. The process of determining the fitness function is shown in Fig. 3.7. The initial fitness function is based on 3 criteria, namely a) the stretch length of the standing posture, b) the limb must stop moving while remain in the standing posture at the end of 2 seconds and c) the leg must be standing. The stretch length of the standing posture is set to 80% of the maximum length. The stretch length is set as a criteria with several reasons. Firstly, this is to ensure that in the standing posture, the limb should not collapse and maintain a certain length. Secondly, this enables the limb to have 20% flexibility to extend the length in moving from the standing posture. The single forelimb configuration must stop moving while remain in the standing posture at the end of the simulations. This criteria is set to avoid the overshoot to occur and render the single forelimb configuration to collapse. This is done by computing the speed of the topmost coordinate of L_1 (scapula link). The last criteria is to ensure the single forelimb configuration to achieve the standing posture. This can be done by calculating the COG of the configuration. The single forelimb configuration is in standing posture if the line that pass through the COG of the configuration and the center of the foot is perpendicular with the ground.

The GA evolves the candidate solutions based on these criteria and able to achieve a result. However, the result is not desirable as the posture of L_1 (scapula link) is not desired (Fig. 3.6). With such posture, much stress is applied on the scapula-body joint as this joint needs to support both the body and the scapula link to maintain such posture. The problem is identified for the insufficient information of the criteria which defines the stretch length of the standing posture. As the evaluation is done based on the overall stretch length of the standing posture, it does not ensure the posture of the single forelimb configuration. The posture of the single forelimb configuration can be in any posture as long as it meets the criteria. To solve this problem, more information is given for the fitness calculation by evaluating the stretch length of different parts of the single forelimb configuration individually. By doing this modification, the desired result is achieved.

The fitness of the population is evaluated after 2 seconds of simulation based on the following criteria:

1. Compute the distance between the y-coordinate of the center of L_1 (scapula link) and L_2 (shoulder-elbow link). The fitness is unity at 70% of maximum length.

2. Compute the distance between the y-coordinate of the center of L_2 (shoulder-elbow link)and L_3 (elbow-foot link). The fitness is unity at 80% of maximum length.

3. Compute the speeds along x and y axis (dy/dt and dx/dt) of the topmost coordinate of L_1 (scapula link). The fitness is unity when the speeds (dy/dt and dx/dt) are both zeros.

4. Compute the COG of the configuration and obtain its angle θ_1 with respect to the Y-axis at foot contact. The fitness is unity when θ_1 is zero.

The total fitness is taken as the average of the above four fitness values which is also shown in Fig. 3.8. Table 3.2 shows the parameters for setting up GA to solve the single forelimb standing problem. The GA is terminated after 5000 evaluations (50 generations with 100 evaluations in each generation) on the population. The fitness of the population is found to lie within a range of 0.8 to 0.9. The scapula-shoulder link length is chosen as 0.115 m while the slant angle is 54.7°. The outcome of the single limb configuration and simulation are indicating that the single limb configuration is able to stand up. The standing up motion sequence of the single forelimb configuration is shown in Fig. 3.9. The next step is to build a two-legged model using the single limb configuration as a prototype.

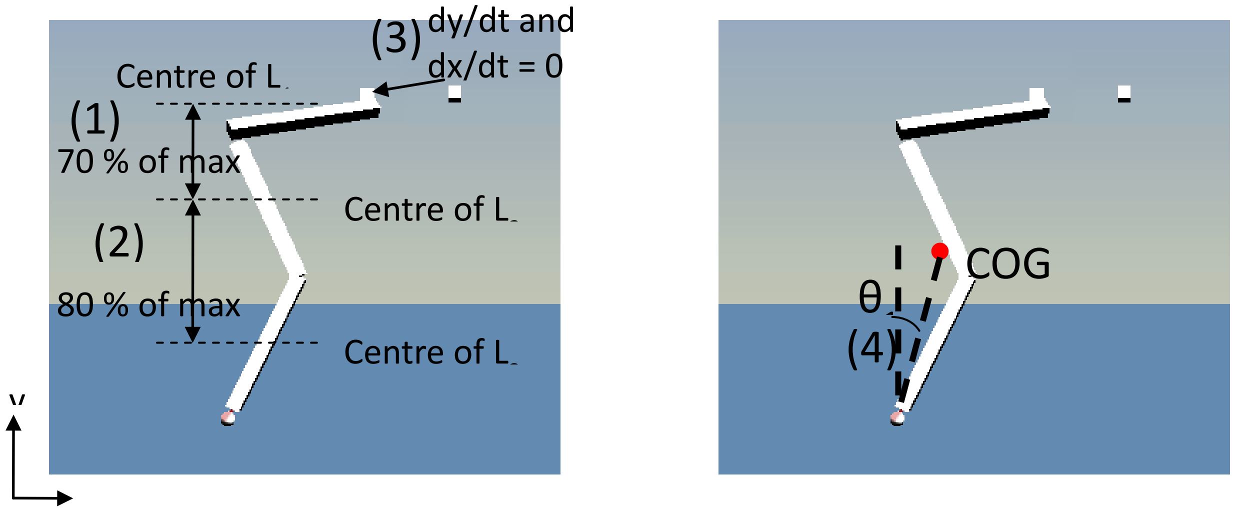


Figure 3.8: Fitness calculation for GA. (1),(2),(3) and (4) indicates the corresponding cost functions

Table 3.2: Parameters for setting up GA to solve single forelimb problem

Population	100
Chromosome Length	11
Exit Criteria	5000 evaluations
Crossover rate	85%
Mutation rate	5%

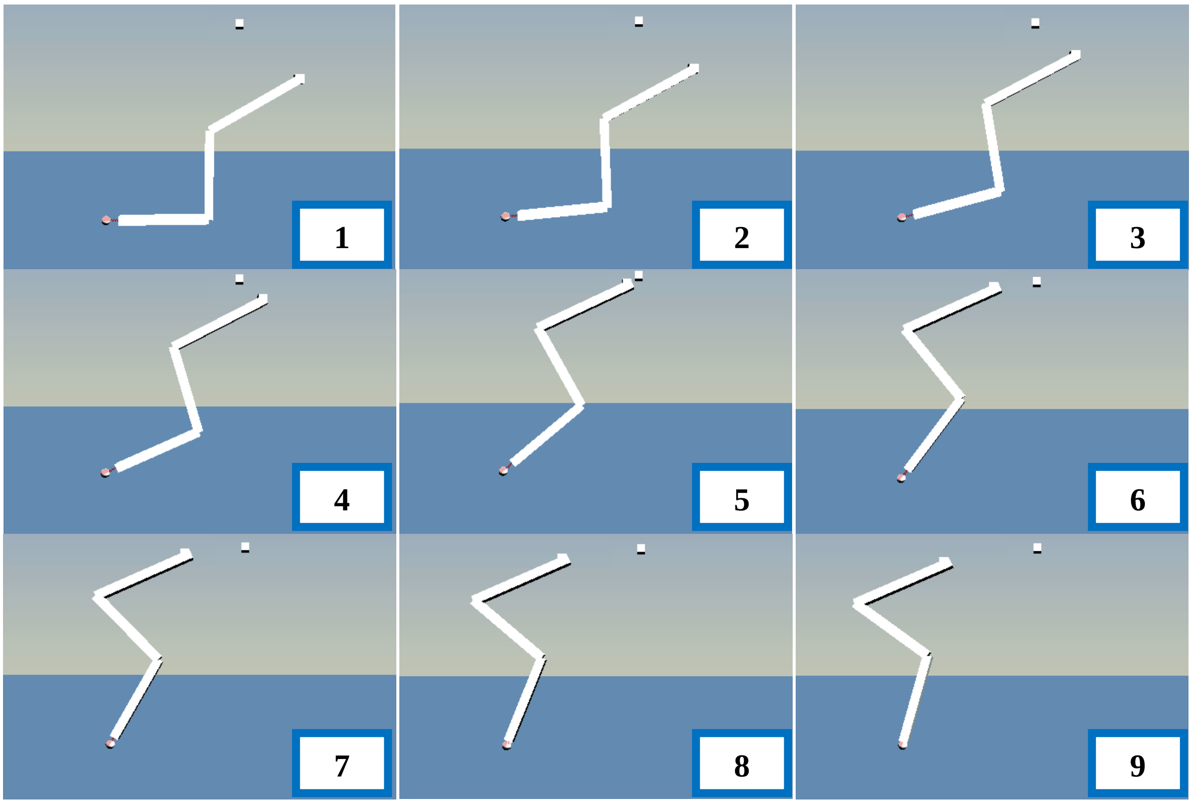


Figure 3.9: Motion sequence of the single forelimb configuration standing up

3.4 Summary

In this chapter, the design of the single forelimb configuration is discussed. The single forelimb configuration is design based on the approximation model of an actual dog's leg. This single forelimb configuration will be the basic building blocks for the quadruped in which every leg is formed by this single forelimb configuration. Therefore, the objective of this chapter is to check whether this single forelimb configuration is able to stand up from a resting posture. GA is able to search for the optimized solution to enable the single forelimb configuration to stand up from the resting posture.

Chapter 4

Simulation of double frontal quadruped limbs

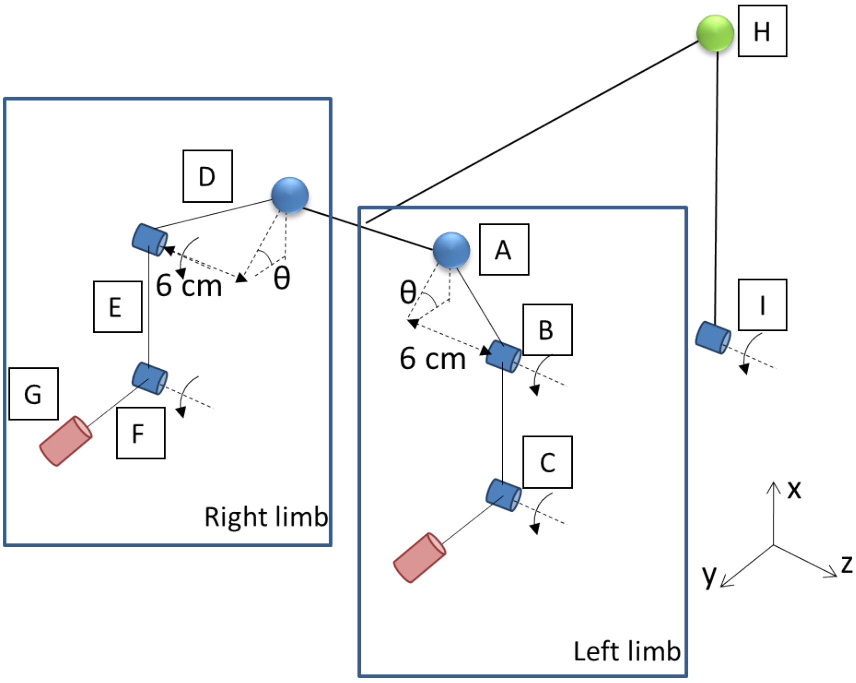
4.1 Double frontal quadruped limb configuration

A double frontal quadruped limbs configuration is built with both the limbs having the same design as the single quadruped limb configuration discussed in Chapter 3. The double frontal quadruped limbs configuration is simulated and the objective is to examine whether the optimized design of the single quadruped limb configuration is able to achieving simple walking motion. As the double frontal quadruped limbs configuration cannot support on its own, a support system is simulated to act as the hind part of the configuration.

4.1.1 Double frontal quadruped limb with support structure

Fig. 4.1 shows the isometric view of a double frontal quadruped limbs configuration with passive rotational support. The double frontal quadruped limbs are homogenous in configuration except for the scapula link. The passive rotational support in Fig. 4.1 is simulated as a four wheel structure providing support and mobility for the configuration.

The moment of inertia tensor for the entire configuration is derived mathematically instead of using the default value generated by the physics engine. The moment of inertia tensors for the basic buildings blocks used in the configuration are shown in Table 4.1.



A	Scapula-body joint
B	Shoulder joint
C	Elbow joint
D	Scapula link
E	Shoulder-elbow link
F	Elbow-foot link
G	Foot
H	Passive spherical joint
I	Passive rotational joint

Figure 4.1: Isometric view of double frontal quadruped limbs configuration with passive rotational support

Table 4.1: Moment of inertia tensor for cuboid, cylinder and sphere

Shape and description	Moment of Inertia Tensor
Solid cuboid of width *w*, height *h*, depth *d*, and mass *m*	$I = \begin{bmatrix} \frac{1}{12}m(h^2+d^2) & 0 & 0 \\ 0 & \frac{1}{12}m(w^2+d^2) & 0 \\ 0 & 0 & \frac{1}{12}m(w^2+h^2) \end{bmatrix}$
Solid cylinder of radius *r*, height *h* and mass *m*	$I = \begin{bmatrix} \frac{1}{12}m(3r^2+h^2) & 0 & 0 \\ 0 & \frac{1}{12}m(3r^2+h^2) & 0 \\ 0 & 0 & \frac{1}{2}mr^2 \end{bmatrix}$
Solid sphere of radius *r* and mass *m*	$I = \begin{bmatrix} \frac{2}{5}mr^2 & 0 & 0 \\ 0 & \frac{2}{5}mr^2 & 0 \\ 0 & 0 & \frac{2}{5}mr^2 \end{bmatrix}$

Fig.4.2 shows the simulated configuration of the double frontal quadruped limbs configuration with a passive support structure. Unlike the single quadruped limb configuration, the scapula-body joint is a 3 DOF joint and in Fig.4.2, joint (i) consists of active revolute scapula-body pitch, yaw and roll joints while (ii) consists of active revolute shoulder pitch joint, (iii) consists of active revolute elbow pitch joint, and, (iv) consists of passive translational joint.

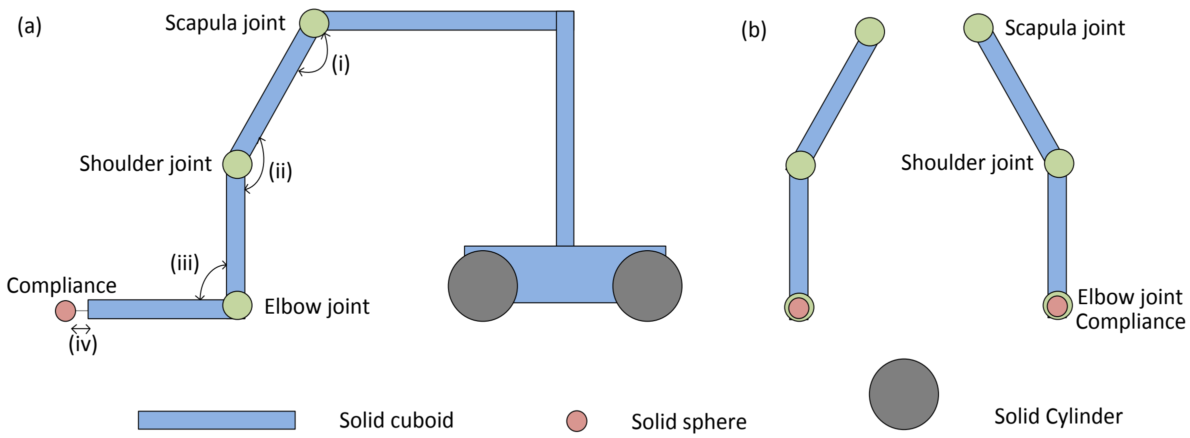


Figure 4.2: Chrono::Engine configuration of a double frontal quadruped limbs configuration in rest mode (a) side view (b) front view

The mass of the support structure is set to 2 kg with each wheel having a mass of 0.4 kg and the rest of the structure having 0.4 kg. The mass is evenly distributed across the body and wheels of the support structure. The mass of the double frontal quadruped limbs are 2 kg. The mass of the support structure is chosen in such a way that it is comparable to the mass of the double frontal quadruped limbs. If the support structure is heavier, the double frontal quadruped limbs will require higher power for walking as energy is wasted to offset the support's mass. On the other hand, if the support structure is lighter, the movement of the support structure can be unstable during the course of motion.

A small gap is inserted in between the body and wheels of the support structure (Fig.4.3 (c)). The gap is required to avoid any body collisions leading to simulation errors in the Chrono::Engine. If two bodies being simulated are too close to each other, the Chrono::Engine collision model for each body may overlap leading to calculation error rendering the simulation unsuccessful. In real world, if two bodies are placed too close, the friction between them will draw more power in order for each body to move. The length x width x height of the support structure's body is considered as 0.14m x 0.1m x 0.02m. Each wheel is considered as a cylinder with a diameter of 0.04m and width of 0.01m.

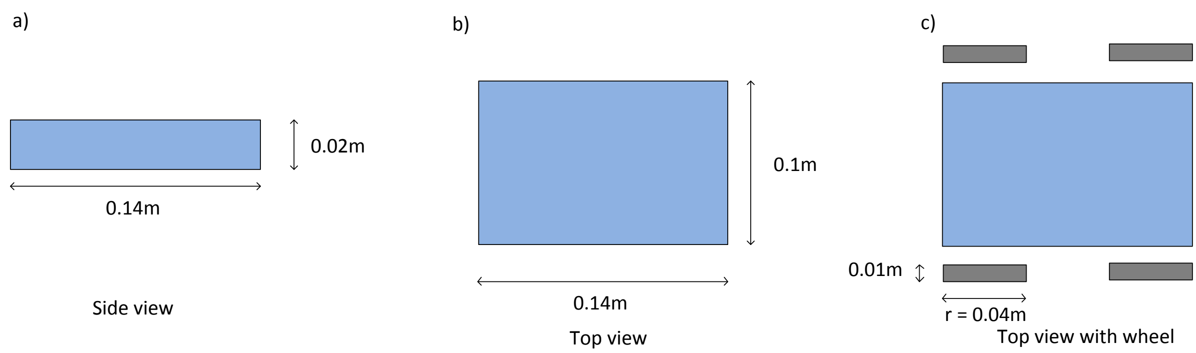


Figure 4.3: Body of the support car – (a) side view, (b) top view, (c) top view with wheels

4.1.2 Standing up, intermediate and walking phases

Fig.4.4 shows the process pipeline for the simulation of the double frontal quadruped limbs configuration. There are three phases in the simulation namely standing phase, intermediate phase and walking phase. Among these three phases, the standing phase is the phase which the double frontal quadruped limbs configuration moves from resting posture to standing posture while the intermediate phase is the phase to prepare for walking. Lastly, the walking phase is the phase when the double frontal quadruped limbs configuration walks. The joint functions and the cost functions for the GA in each phase are different.

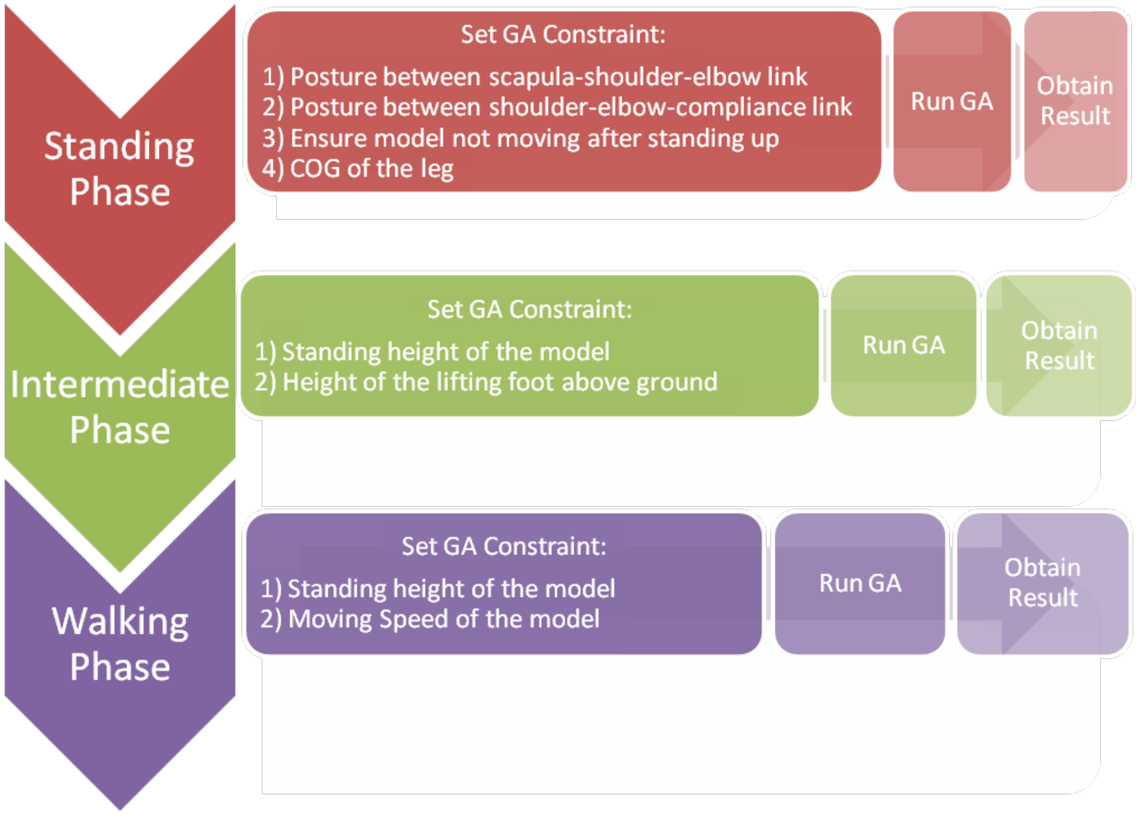


Figure 4.4: Process Pipeline for double forelimb configuration

4.2 Standing up phase: Genetic Algorithm and Cost functions

In chapter 3 the single forelimb configuration is made to stand up where only the pitch joint is utilized. On similar grounds, in the standing phase of the double frontal quadruped limbs configuration only pitch joint is active and, the roll and yaw joints are considered as inactive. During the standing up phase, 3 active pitch joints are active in the frontal limbs. As both left and right limbs are symmetrical, same joint functions are applied to both the left and right limbs, i.e. left shoulder joint and right shoulder joint are utilizing the same joint functions. Speed function with the shape shown in Fig. 3.5 is used. The speed function has 3 parameters defining its shape. Nine parameters are associated with the three active pitch joints. The support structure provides the pushing force to aid the configuration in the course of standing up. The support structure wheels are made to move with a constant speed to act as a forward thrust which aids the double frontal quadruped limbs configuration to stand up.

The candidate solution for GA to solve the standing problem will have 9 genes for the 9 parameters. Fig. 4.5 shows the chromosome encoding the parameters associated with the double limb standing up phase in which :

a_1	random floating point in between -5 to 0 with a resolution of 0.025
a_2	random floating point in between -10 to 0 with a resolution of 0.05
a_3	random floating point in between 0 to 10 with a resolution of 0.05
t_{1i}	random floating point in between 0.1 to 1 with a resolution of 0.0045
t_{2i}	random floating point in between 1 to 3 with a resolution of 0.01

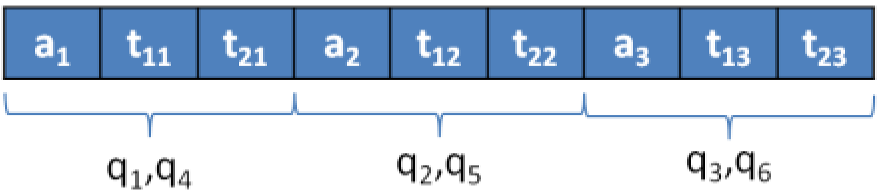


Figure 4.5: Chromosome encoding for double frontal quadruped limbs standing up

Table 4.2: Parameters for setting up GA to solve double frontal quadruped limbs standing

Population	100
Chromosome Length	12
Exit Criteria	3000 evaluations
Crossover rate	85%
Mutation rate	5%
Elitism	5%

Table 4.2 shows the parameters for the setup of GA to solve the double frontal quadruped limbs standing up phase. The cost functions to calculate the fitness of each population are the same as in the single forelimb configuration (Fig. 3.7). As both the limbs have symmetrical motion in the standing up phase, the same parameters for speed functions are utilized for the both limbs.

4.2.1 Standing up phase: Results for double frontal limbs

The GA simulation is carried out for 30 iterations to ensure convergence. Thirty iterations are chosen instead of 50 in the single limb configuration simulation as the configuration is complex for double limbs and the calculation of cost functions requires more computational effort. Fig.4.6 shows the fitness trend for double limbs standing up simulation and the average fitness of the population exceeded 0.9 at the 5th iteration and convergence was observed. The set of parameters available at the end of the GA iterations is utilized in the virtual simulation of the double frontal quadruped limbs configuration to stand up from rest. Fig.4.7 shows the sequence of motions from resting posture to standing posture of the forelimbs. The double quadruped limbs configuration is able to stand up in 2s in the simulation.

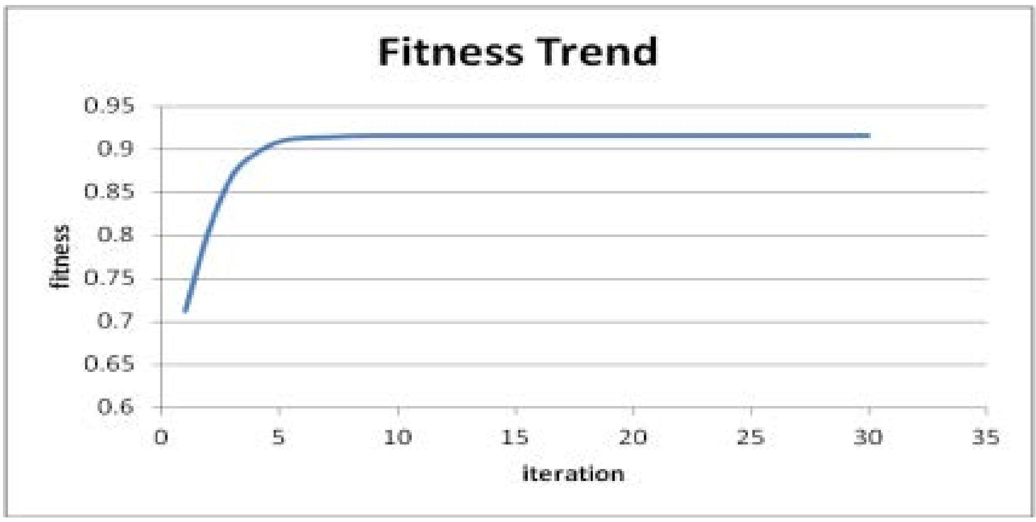


Figure 4.6: GA Fitness Trend for double forelimb standing up simulation

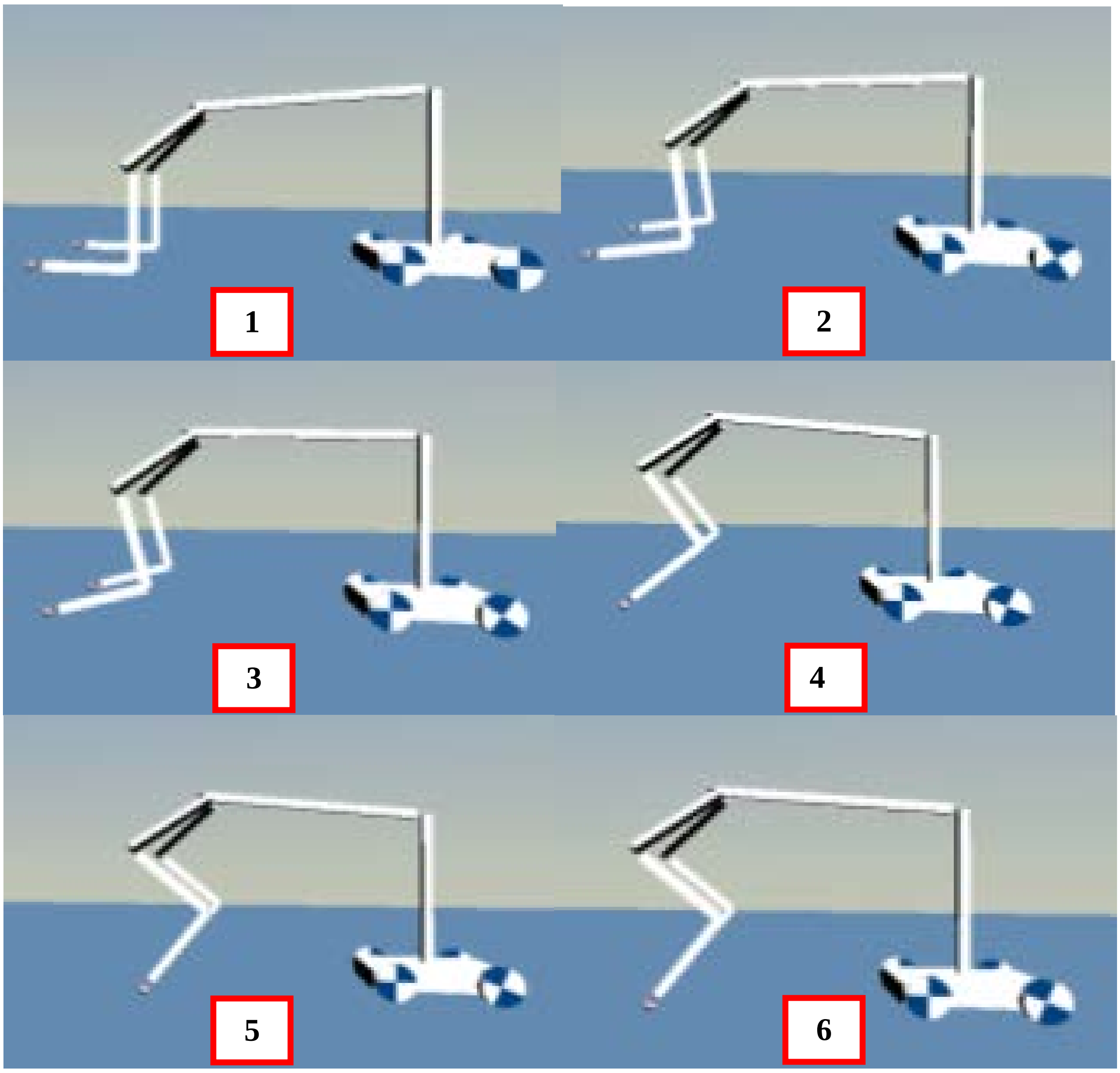


Figure 4.7: The sequence of motion of the double frontal quadruped limbs configuration from resting posture to standing posture

In the process of standing up, the shoulder joint does the most significant work (Fig. 4.8). The power consumed by the shoulder joint is highest. The double frontal quadruped limbs configuration depended greatly on the shoulder joint movement in standing up phase. Even though the elbow joint is allowed to move, the GA output provided zero motion for the elbow joint, which is indicating that the configuration is able to standup without any motion in the elbow joint. Minimal movement is noticed for the scapula-body joint during the early phase of standing up. A spike in the joint torque is observed during the start of the movement for every joint which is an indication of the sudden change in joint speeds at that point. Step or sudden changes in joint speeds are not desirable which will lead to higher torques being created at the joints.

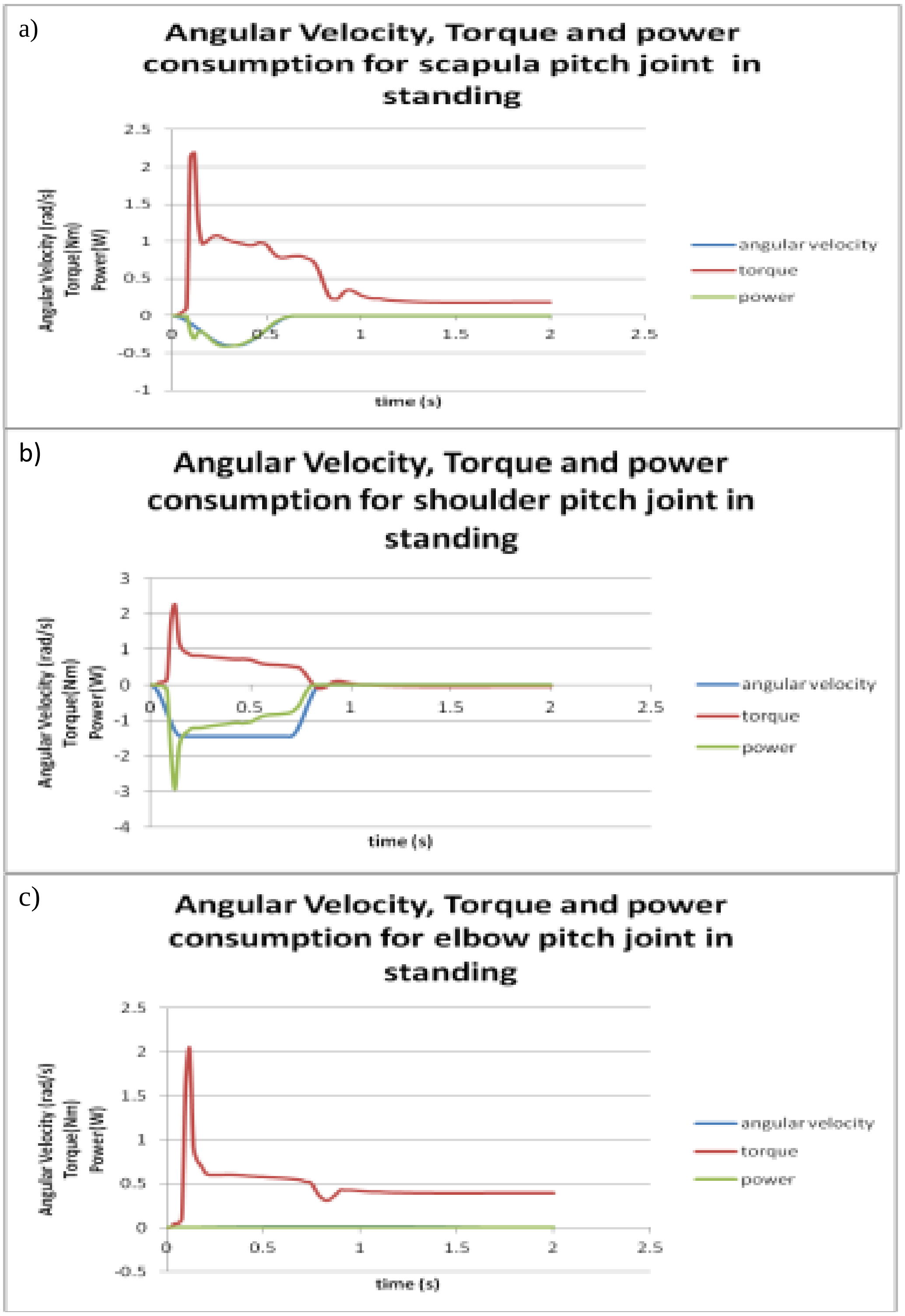


Figure 4.8: The angular velocity, torque and power graph for the double frontal quadruped limbs configuration. a) scapula-body joint b) shoulder joint c) elbow joint

4.3 Intermediate phase: GA and cost functions

The intermediate phase is the phase before walking and its role is to ensure the start of periodic movements. If this phase is skipped and go directly into the walking phase, the double frontal limbs configuration will only perform stepping on the spot instead of walking. The joint motion direction of the left frontal limb in intermediate phase is shown in Fig. 4.9, while the right forelimb has an opposite motion sequence. As the role of the intermediate phase is to ensure the start of periodic movements, periodical joint functions are desired. One of the most common periodical functions are sine waveform function. In the Chrono::Engine, a sine waveform function is called from the library and its parameters, amplitude, frequency and phase difference are variables. In the simulation, the sine waveform function frequency is set to 0.8Hz while the phase difference is ignored. The frequency is chosen to be 0.8Hz as this value is achievable in the real world. The amplitude of the sine function is the parameter to optimize in GA and, the left and right limbs have amplitudes of opposite signs (left limb is 180° out of phase with respect to the right limb). There are a total of 3 amplitude parameters for three different joints to optimize in the intermediate phase.

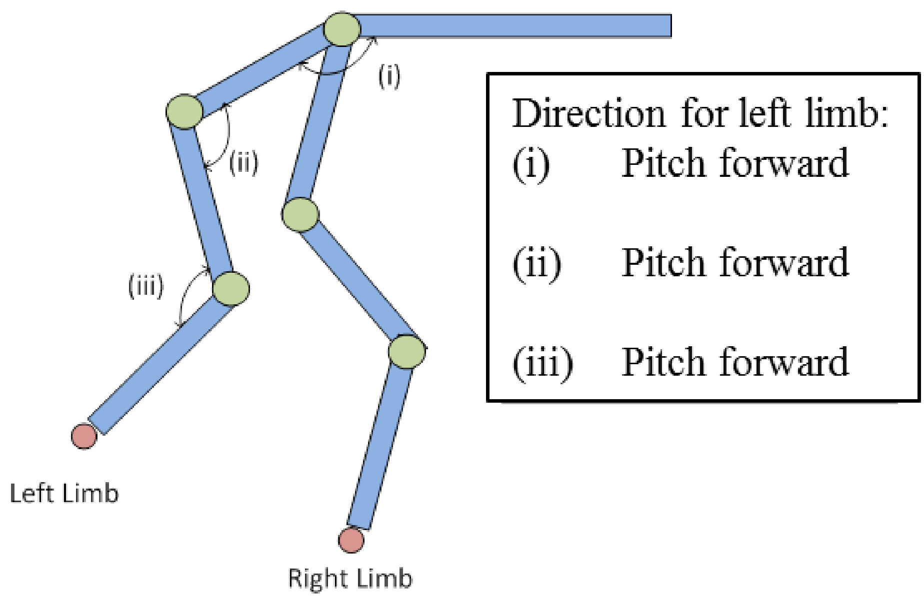


Figure 4.9: Joint motion direction of the left limb in intermediate phase

As the intermediate phase just occurs for a short period of time, the fitness of a candidate solution is evaluate after 0.4 second of simulation, which is approximate one third of the joint function's period. The fitness is evaluated based on two criteria, namely a) the standing height at the end of the intermediate phase and b) the lifted foot height. It is desired that after the intermediate phase, the double frontal quadruped limbs configuration does not collapse and the best case situation is it maintain the height of the standing posture. The fitness is calculated after 0.4 s of simulation based on the following cost functions:

1. The standing height of the forelimb configuration. The fitness is set to unit when the height achieved is the standing height.

2. The lifted height of the foot. The fitness is set to unity when the lifted foot is at 15% of the standing height.

The total fitness is taken as the average of the above two fitness values which is shown in Fig.4.10. Table 4.3 shows the GA parameters.

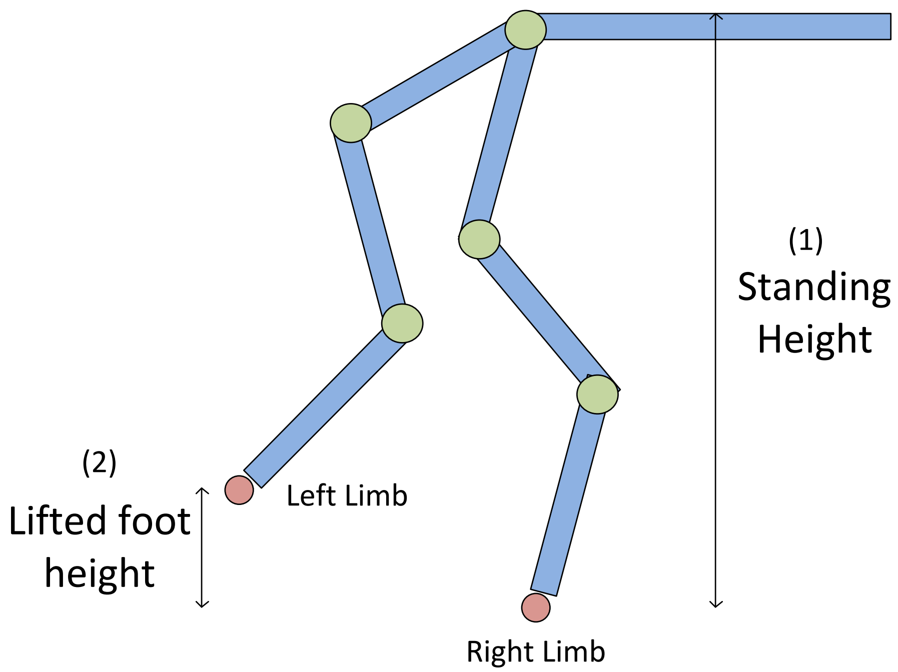


Figure 4.10: Fitness calculation in intermediate phase

Table 4.3: GA parameters in the intermediate phase for double forelimb configuration

Population	100
Chromosome Length	3
Exit Criteria	3000 evaluations
Crossover rate	85%
Mutation rate	5%
Elitism	5%

4.3.1 Intermediate phase: Results for double frontal quadruped limbs configuration

The simulation achieved an average fitness of over 0.95 after 30 iterations and every population converged to a solution. The final set of parameters is chosen to validate the standing up phase motion of the double forelimb. The fitness trend is shown in Fig. 4.11.

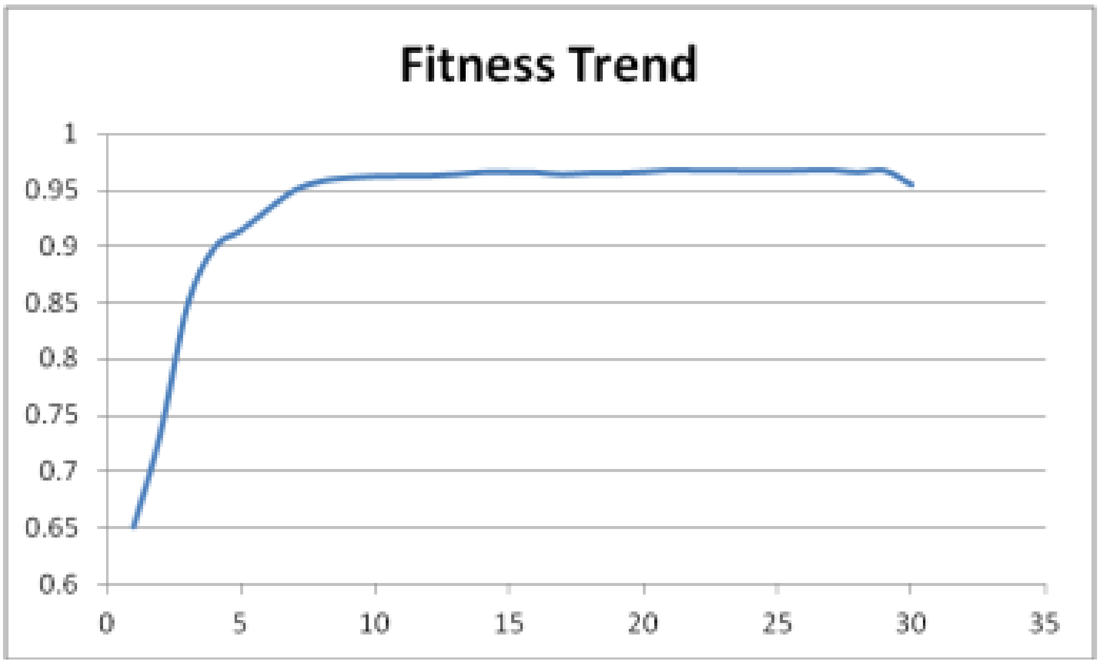


Figure 4.11: GA fitness trend for the double frontal quadruped limbs configuration intermediate phase

In the intermediate phase, the left limb is lifted up while the right limb acts as a support limb. It is observed from Fig.4.12 and Fig 4.13 that the power consumed is higher by the right limb, especially by the right scapula and right elbow pitch joints. The shoulder pitch joints in both limbs did not move much during the intermediate phase. The swing of left limb is completed mostly by the motion of the left scapula-body joint. As opposed to the standing phase where more power is consumed by the shoulder joint, in the intermediate phase, the shoulder joint is moved less significantly.

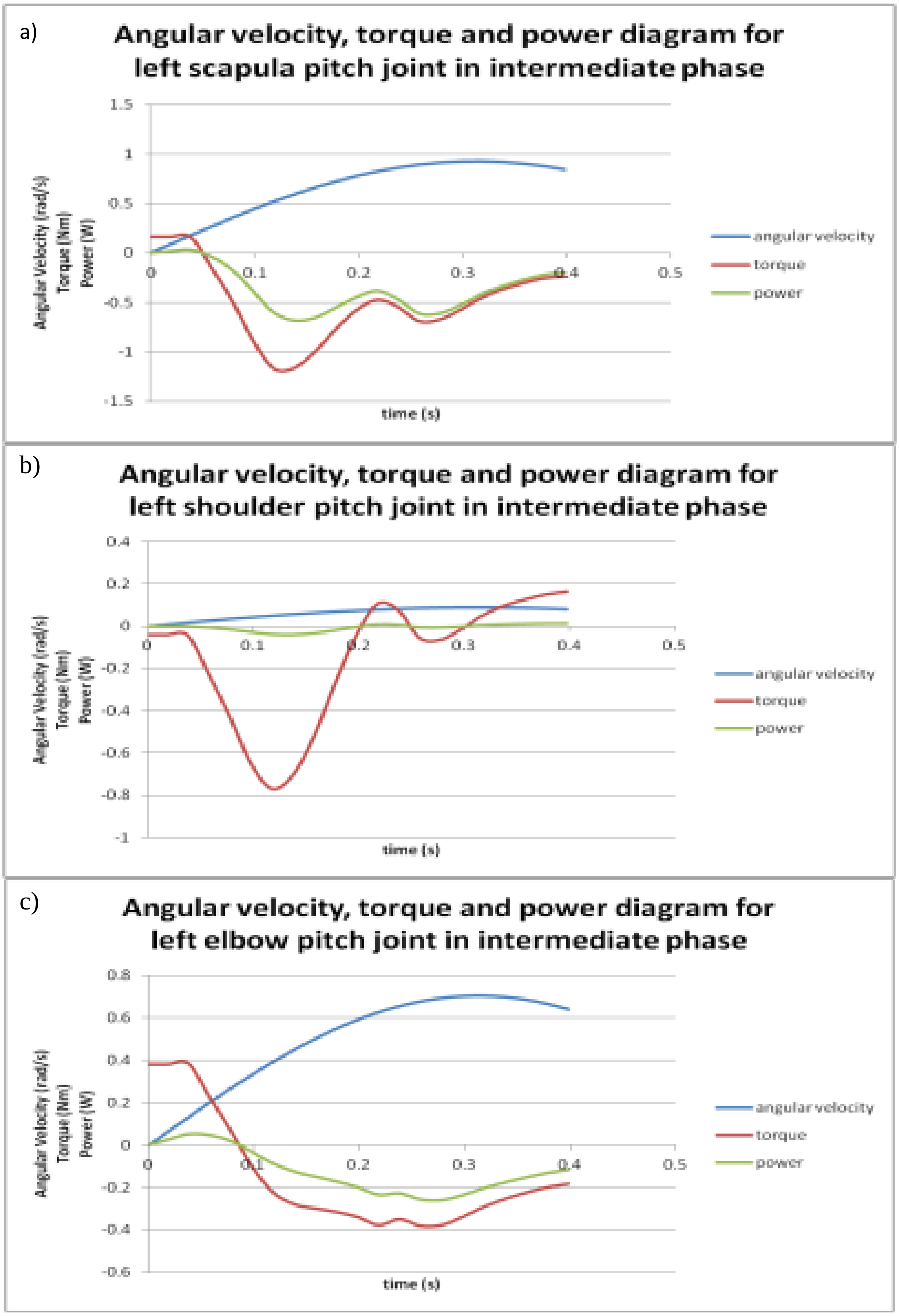


Figure 4.12: The angular velocity, torque and power graph for the left frontal limb configuration in intermediate phase. a) scapula-body joint b) shoulder joint c) elbow joint

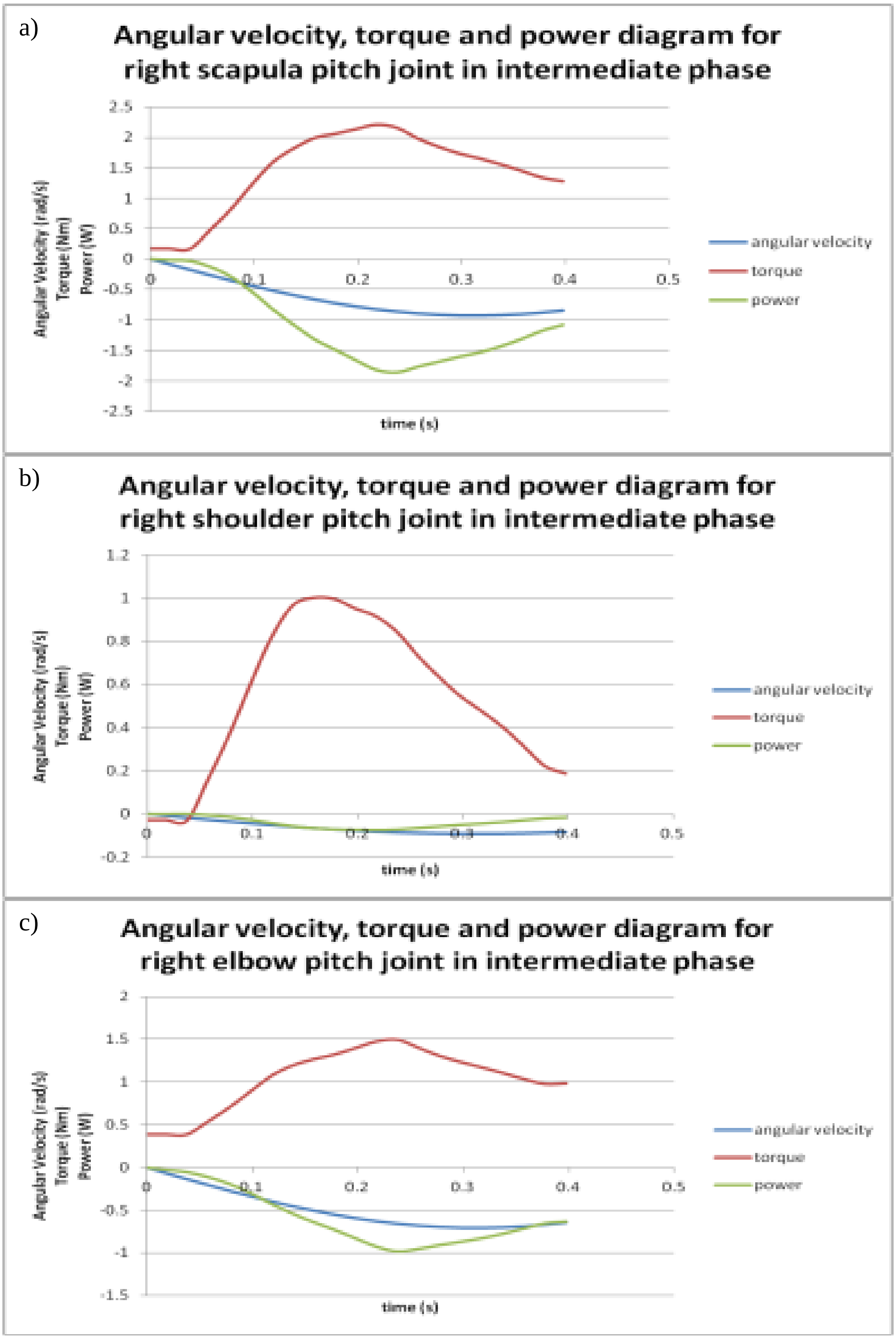


Figure 4.13: The angular velocity, torque and power graph for the right frontal limb configuration in intermediate phase. a) scapula-body joint b) shoulder joint c) elbow joint

4.4 Summary

In this chapter, the standing phase and intermediate phase of the double frontal quadruped limbs configuration are discussed. As the double limbs configuration cannot self-support on its own, a support structure with passive rotational capability is introduced into the configuration. The standing phase is same as discussed in chapter 3 with difference of applying to a double limbs configuration instead of single limb configuration. The intermediate phase is the phase before walking phase and its role is to ensure the periodical movement of the double limbs configuration. The optimal solution for both phases are achieved by utilizing GA.

Chapter 5

Walking with double frontal quadruped limbs configuration

5.1 Introduction

The walking sequences for a double forelimb configuration are discussed in this chapter. The walking phase is after the intermediate phase. Four cases of the double forelimb configuration walking phases are discussed. The four cases are categorized into two sub categories, namely walking with and without disturbance. For walking with disturbance, a forward speed is applied to the support structure at the rear. Two different scapula-body joint configurations are utilized in the study. The two configurations are namely the scapula-body joint with pitching and scapula-body joint with rolling, pitching and yawing. The objective is to evaluate the walking ability of the double forelimb configuration in different scenarios. There are four different cases that are considered in the walking phase namely, a) scapula-body pitch joint without forward thrust, b) scapula-body pitch joint with forward thrust, c) scapula-body roll, pitch and yaw joint without forward thrust, and d) scapula-body roll, pitch and yaw joint with forward thrust.

5.2 Walking phase: Genetic Algorithm and cost functions

The walking phase is achieved by applying a periodic speed function to the joints of the double forelimb. The starting sequence of walking is to return the lifted foot to the ground. A simple sine speed function is used for the scapula-body joint and shoulder joint while a periodic function shown in Fig. 5.1 is applied to the elbow joint. The period function in Fig. 5.1 completes the half cycle faster than a simple sine speed function so that the elbow joint can move to its desired position faster than the scapula and shoulder joints.

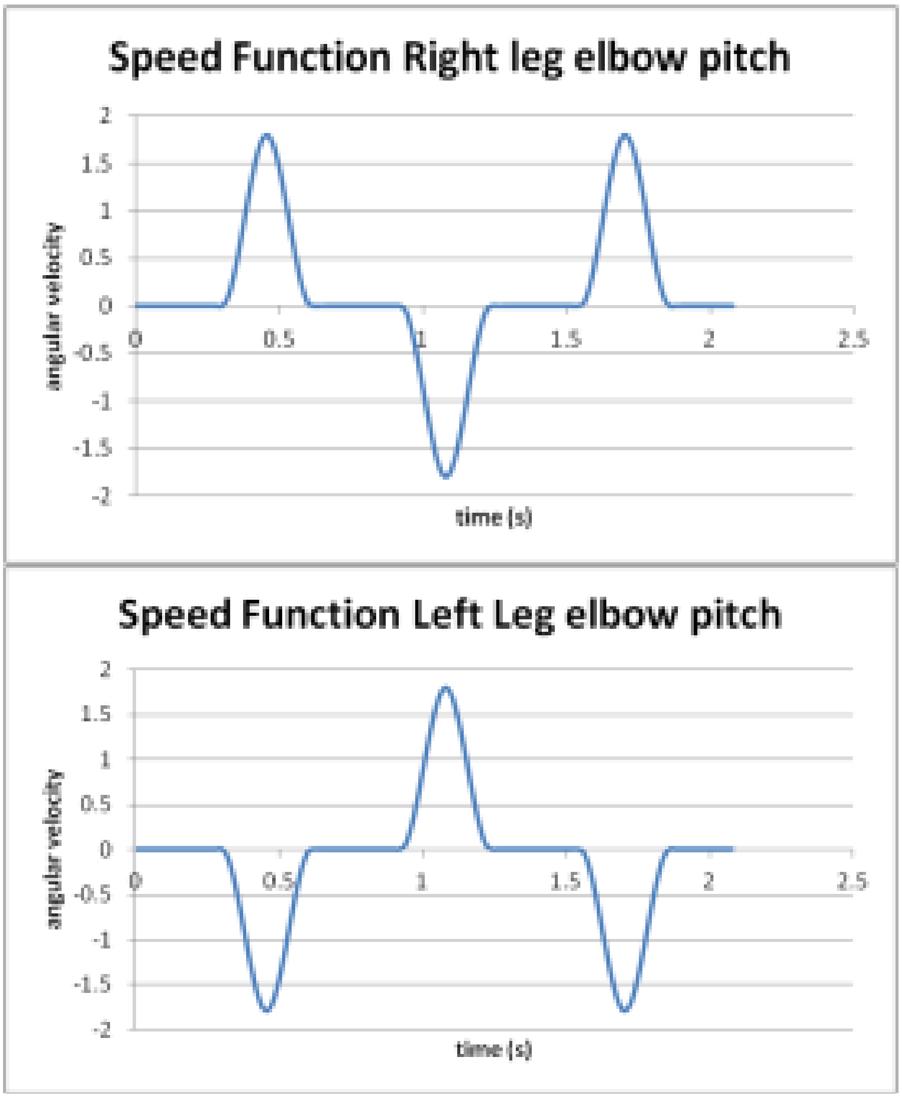


Figure 5.1: Self-defined periodic speed function for elbow joint

Different cases are considered for walking phase. In each case, the frequency of the speed functions is fixed at 0.8 with zero phase difference. The amplitude of the speed function applied to the left limb is opposite in sign with that of the right limb. Only the amplitude of the speed function is optimized. The fitness is evaluated on the candidate solution after letting the double limb configuration to walk for a relatively long period, which is set as 6 seconds in every cases. The fitness is evaluated based on two criteria which is a) the standing height of the double limbs configuration at the end of 6 seconds and b) the walking speed of the double limbs configuration. The walking speed is evaluated by comparing with a reference speed. The reference speed is calculated based on the walking Froude number [64, 65] which is given as,

$$Fr = \frac{V^2}{gl},$$

in which, V is the velocity of the quadruped, g is the acceleration due to gravity and l is the height from hip to the ground (from the scapula-body joint to ground). For walking, Froude number is set to 0.01, l = 0.24 m, the reference speed calculated is 0.15 m/s.

The same fitness function is utilized for all cases and it is calculated after 6 s of the simulation based on the following two cost functions:

1. The standing height of the double frontal quadruped limbs configuration. The fitness is set to unity when the height achieved is the standing height.
2. The forward moving speed of the configuration. The fitness is set to unity when the moving speed of the configuration is same as the reference moving speed (0.15 m/s).

5.3 Double frontal quadruped limbs configuration walking phase – Case 1: Scapula-body pitch joint without forward thrust

In this case, only pitch joints are actuated, namely pitch joint for scapula, shoulder and elbow. The chromosome length for individuals in GA is 3. Fig. 5.2 shows the chromosome encoding for GA. The support structure at the rear is not actuated, meaning that it is in passive mode so that no disturbance is applied.

Figure 5.2: Chromosome encoding for double forelimb walking phase

in which,

a_1 random floating point in between 0 to 2.0 with a resolution of 0.01

a_2 random floating point in between 0 to 2.0 with a resolution of 0.01

a_3 random floating point in between 0 to 3.0 with a resolution of 0.015

a_1, a_2 and a_3 is the amplitude for scapula-body, shoulder and elbow pitch respectively. A wider searching range is applied on the elbow pitch joint function because the joint function for the elbow pitch joint is different from scapula-body and shoulder joint. The total fitness is taken as the average of the above two fitness values.

Table 5.1: Double forelimb configuration walking phase GA Parameters for pitching scapula-body joint and without disturbance

Population	100
Chromosome Length	3
Exit Criteria	3000 evaluations
Crossover rate	85%
Mutation rate	5%
Elitism	5%

Table 5.2: Comparison of results for walking phase with pitching scapula-body joint and without forward thrust for 3 different runs

Parameter	1st run	2nd run	3rd run
a_1 (scapula)	0.94	1.31	1.05
a_2 (shoulder)	0.05	0.24	0.46
a_3 (elbow)	2.415	2.34	1.88
Speed	0.087m/s	0.105m/s	0.092m/s

The GA simulation stops at 30 iterations and the time taken is 34000s which is 9 hr and 29 min. Table 5.1 shows the GA parameters. The average fitness value at the 30th iteration is 0.861794. The fitness trend is as shown in Fig. 5.3. Table 5.2 shows the comparison of GA results for different simulation cycles. It is observed that the shoulder join speed peak amplitude, a_2, is small and the motor for the shoulder joint can be selected accordingly. Table 5.2 also shows that when the a_1 parameter corresponding to the scapula-body joint is larger, the walking speed is higher. With a larger value of a_1, the stride length is larger too.

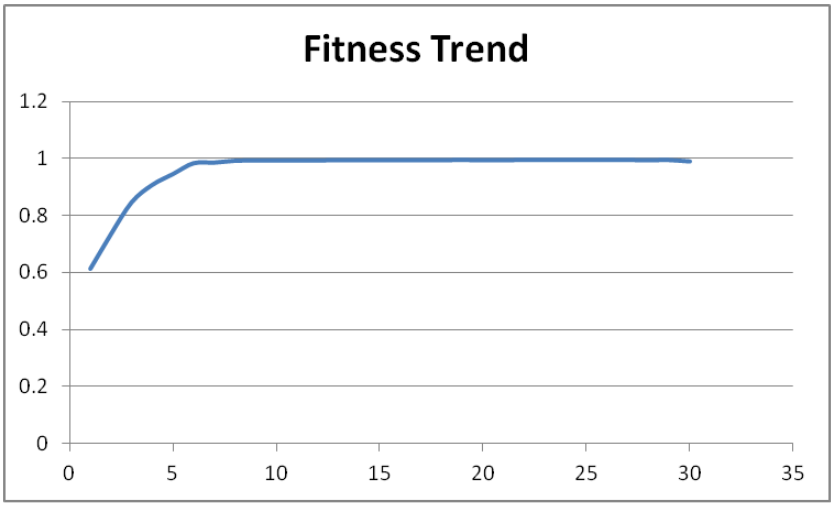


Figure 5.3: Fitness trend of double forelimb walking phase

The torque function, angular velocity and power consumption for the double forelimb configuration is shown in Fig. 5.4 and Fig. 5.5. It is observed that for both the limbs, the elbow joint consumes the most power during walking. This is because the elbow joint acts as the pushing joint during walking. When the foot hits the ground, the actuated elbow joint causes a reaction force on the body thus making the configuration to move forward. From Fig. 5.4 and Fig. 5.5, the shoulder joint speeds are not significant during walking.. The scapula-body joints consume most power when the elbow joints consume lesser peak power (Fig. 5.4). When one limb is lifted, the other limb acts as the support and the scapula body joint consumes higher power.

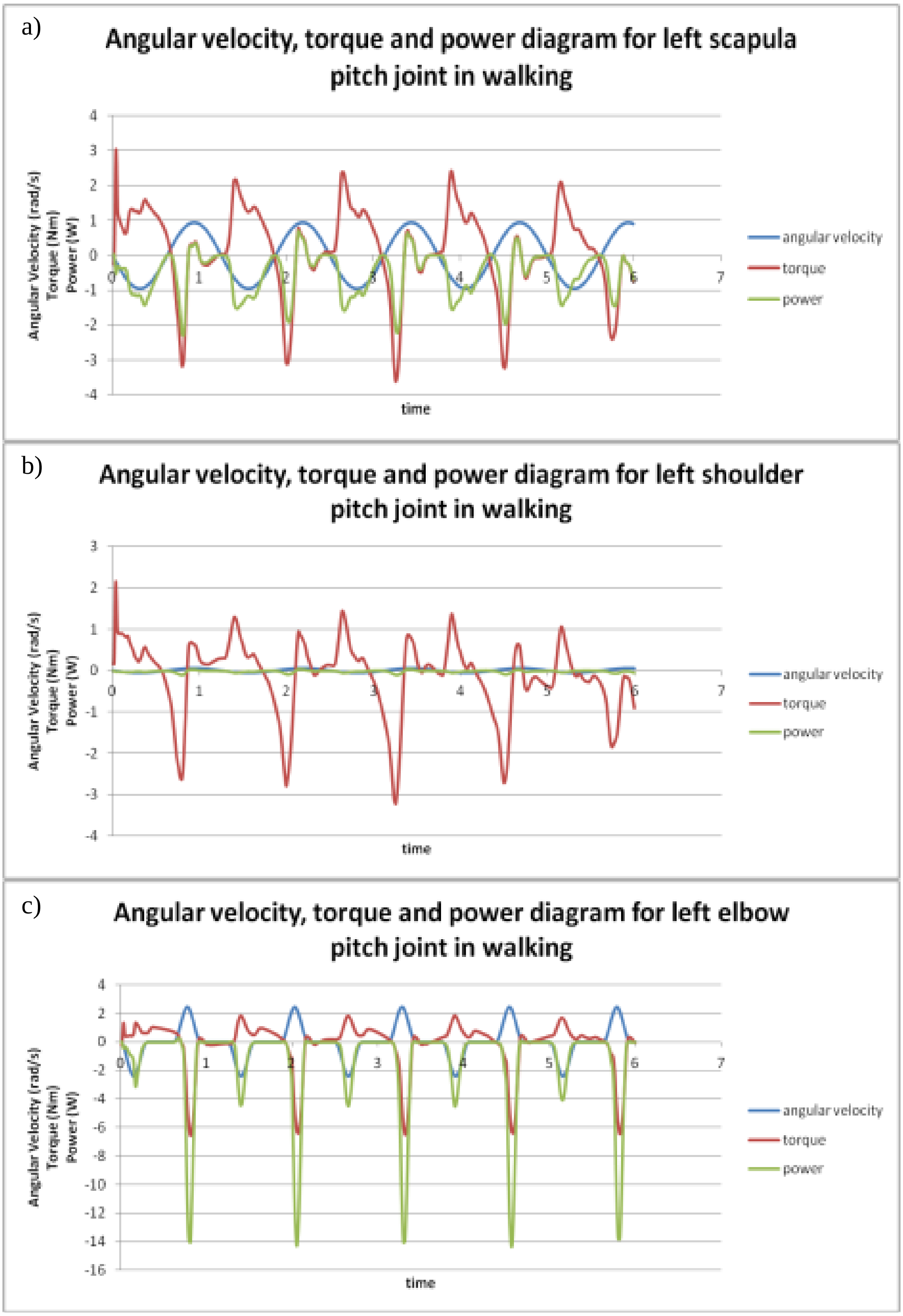


Figure 5.4: Torque function, angular velocity and power for left frontal limb in walking phase without forward thrust. a) scapula-body joint b) shoulder joint c) elbow joint

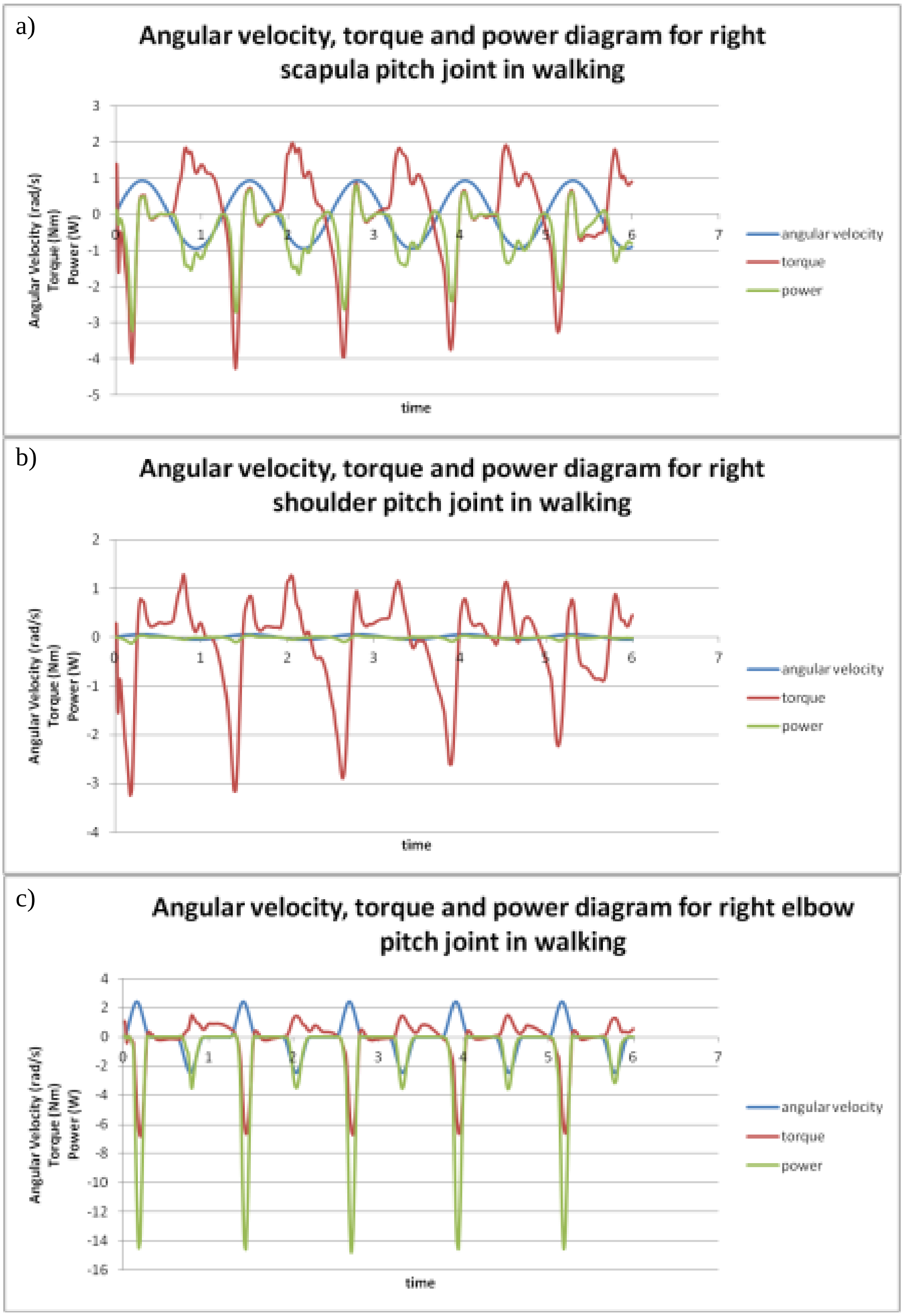


Figure 5.5: Torque function, angular velocity and power for right frontal limb in walking phase without forward thrust. a) scapula-body joint b) shoulder joint c) elbow joint

Fig. 5.6 shows the joint rotation angles over time. The joint rotation angles are in radian units. Over time, the joint angles eventually deviated due to foot friction against the ground. As the joint functions applied to the limbs are in open loop control, the disturbance due to foot friction on the configuration during the course of walking cannot be eliminated. The configuration starts to fall forward after several seconds of simulation.

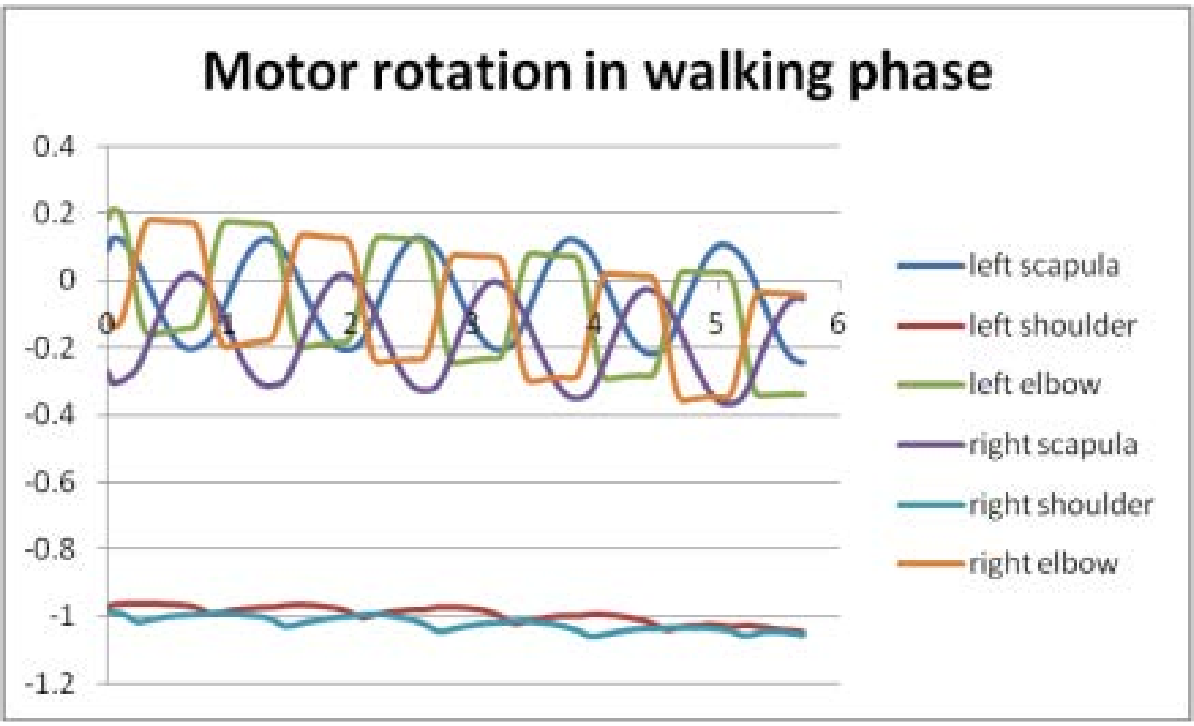


Figure 5.6: The change in the joint rotation angles over time

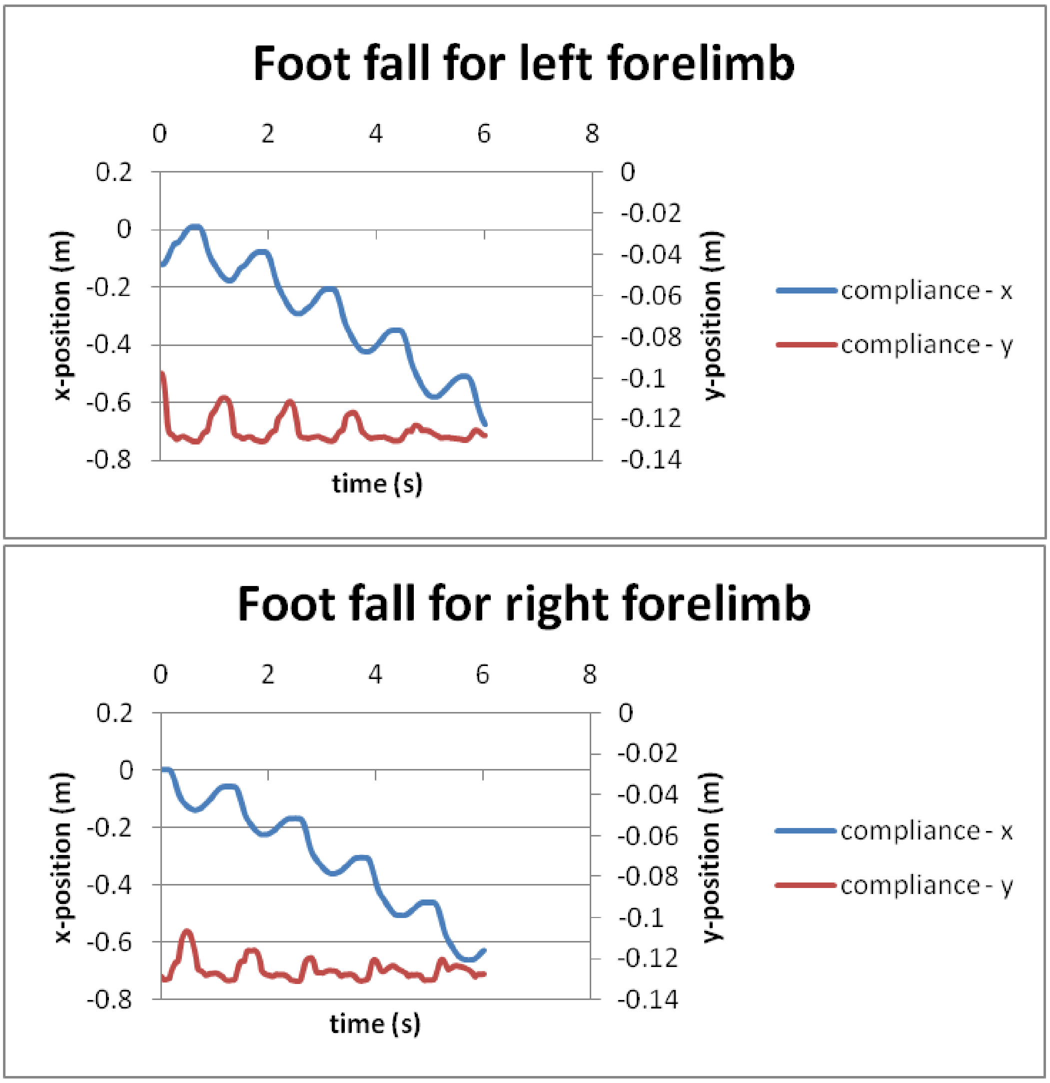


Figure 5.7: Footfall for the double forelimb configuration in 6 second. (a) left frontal limb (b) right frontal limb

Fig. 5.8 shows the gait graphs of the double forelimb configuration. During walking, foot on the ground has a longer timespan on the ground. When animals walk, the duration of the foot on ground is longer and there are instances where both the limbs are on the ground for a period of time. By increasing the moving speed or moving pattern (gait), the duration of the foot on the ground decreases. The frequency of the joint function is set to 0.8 Hz and the joint period is 1.25 second. Each leg takes 1.25 seconds to complete one movement cycle.

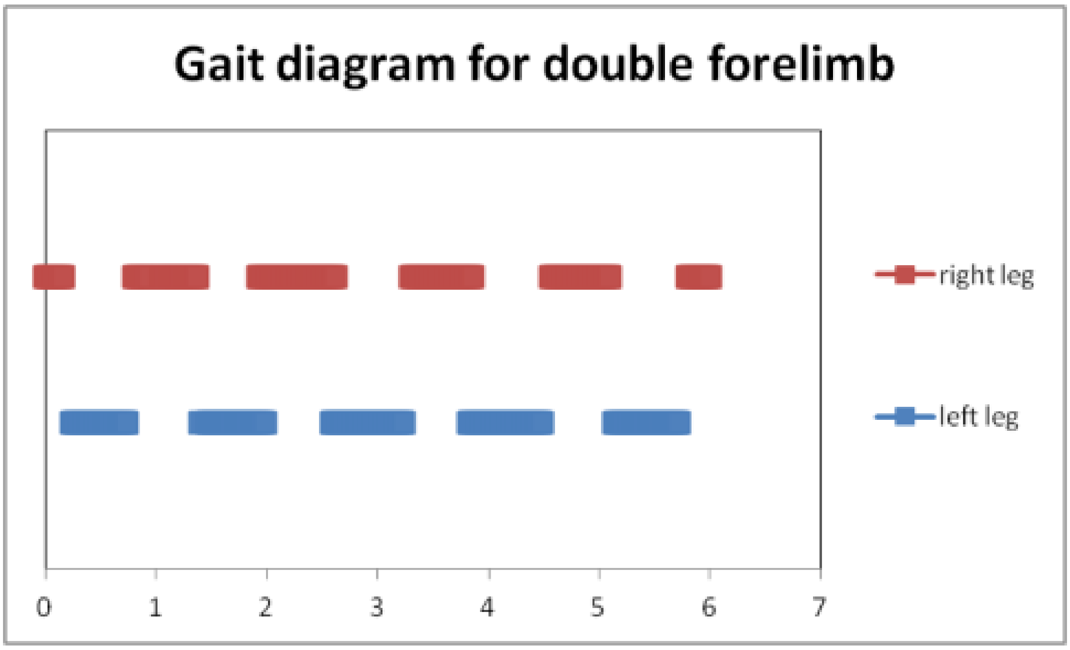


Figure 5.8: Gait graph for the double forelimb in walking with pitching scapula joint and with forward thrust

5.4 Double frontal quadruped limbs configuration walking phase – Case 2: scapula-body pitch joint with forward thrust

In this case, a forward thrust is injected by actuating the supporting structure and the rolling speeds of the supporting structure's wheels are set to 3.5 rad/s. The forward thrust added is to verify the reliability of the forelimb in walking motion when the hind limbs are implemented. In a quadruped the forelimbs provide steering while the hindlimbs provide driving force. As in case 1, only the pitching joints, scapula, shoulder and elbow, are actuated. The chromosome population is set to 100 while each chromosome has 3 genes representing the joint function parameters. The chromosome encoding is shown in Fig. 5.2.

Table 5.3 shows the GA parameters in this phase. GA simulation is stopped in 30 iterations and the time taken is 36640.3s (10 hours and 10 min). The average fitness value at the 30thiteration is 0.871833. Table 5.4 shows the optimized parameters. The fitness trend is shown in Fig.5.9. With the forward thrust, the moving speed of the configuration is faster which is reasonable as with the forward thrust at the support structure adds up to the forward speed of the configuration making it move faster.

Table 5.3: Parameters for setting up GA for optimized walking phase with pitching scapula joint and with forward thrust for the double frontal quadruped limbs configuration

Population	100
Chromosome Length	3
Exit Criteria	3000 evaluations
Crossover rate	85%
Mutation rate	5%
Elitism	5%
Walking Speed	0.107m/s

Table 5.4: GA optimized parameters for walking phase with forward thrust

Allele	Parameter	Corresponding value	Value
0	-0.1	a_1 (scapula pitch)	0.9
1	-0.91	a_2 (shoulder pitch)	0.09
2	0.99	a_3 (elbow pitch)	2.985

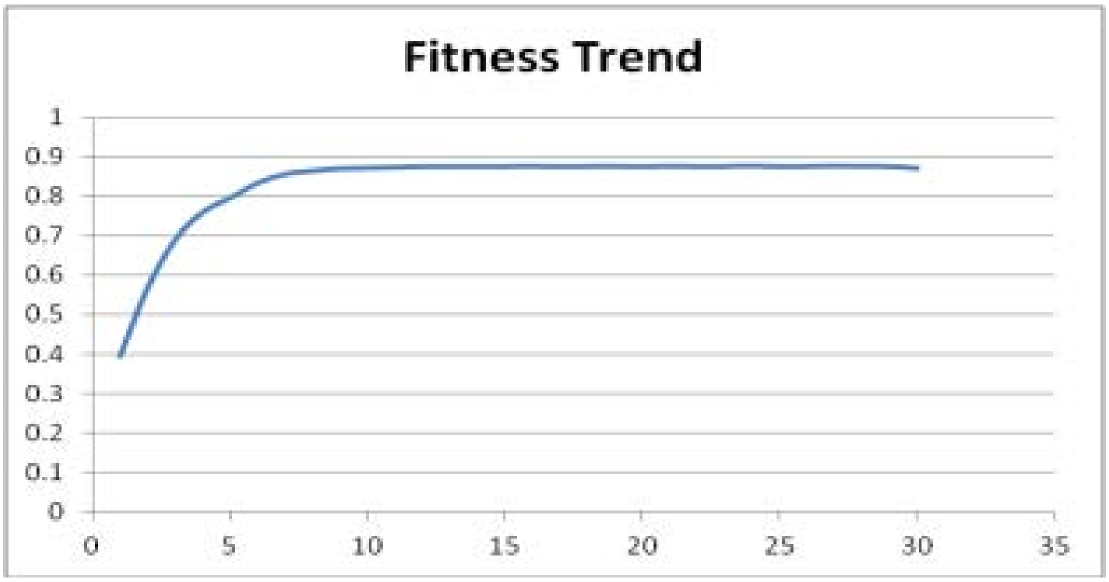


Figure 5.9: GA fitness trend for walking phase with pitching scapula body joint and with forward thrust

The torque function, angular velocity and power consumption for the double forelimb configuration is shown in Fig. 5.10 for left forelimb and Fig. 5.11for right forelimb. The torque, velocity and power profiles are similar in nature to the profiles for walking without forward thrust. The difference between the two cases (with and without forward thrust) is for the elbow pitch joint where the power consumed for walking with forward thrust is higher. This is due to fact that with the forward thrust added to the system, a higher reaction torque occurs on the elbow joint to offset the same. Therefore, it can be noted that when the hindlimbs are added, the power consumed by the forelimb will increase.

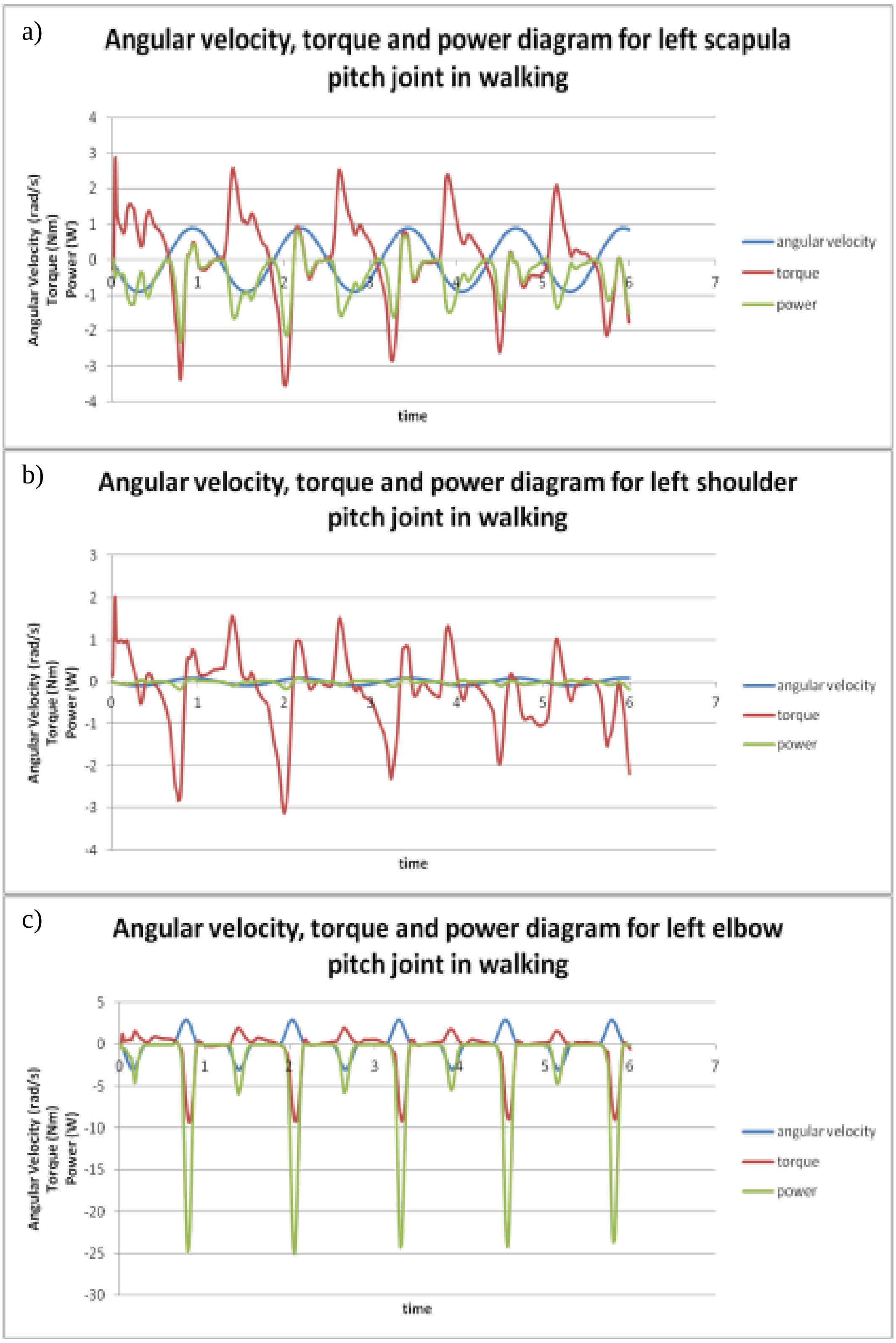


Figure 5.10: Torque function, angular velocity and power for left frontal limb in walking with forward thrust. a) scapula-body joint b) shoulder joint c) elbow joint

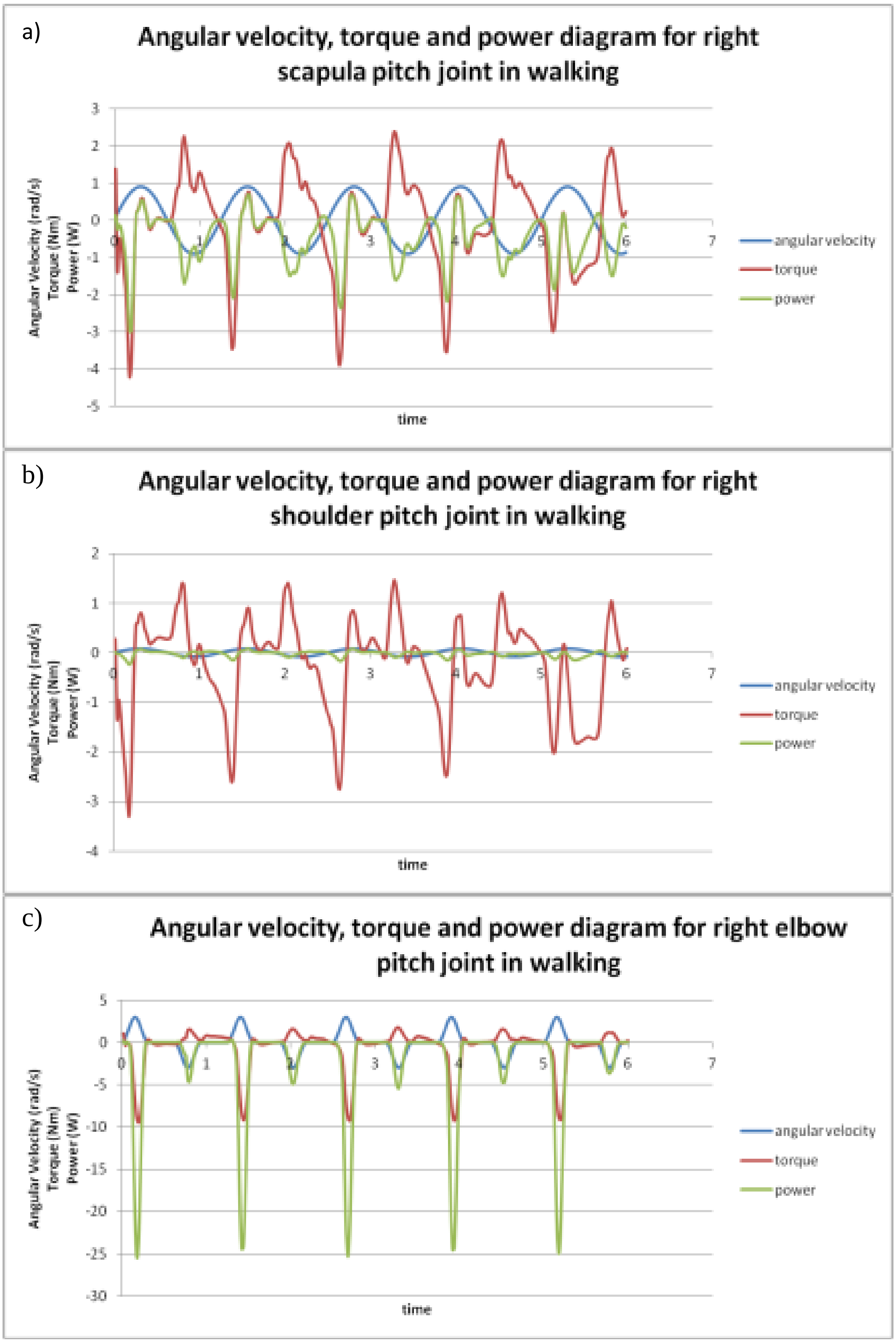


Figure 5.11: Torque function, angular velocity and power for right frontal limb in walking with forward thrust. a) scapula-body joint b) shoulder joint c) elbow joint

5.5 Double frontal quadruped limbs configuration walking phase-Case 3: scapula-body roll, pitch and yaw joint without forward thrust

In this case, the roll, pitch and yaw joints in scapula-body joint are activated and they play a role in the course of walking. The effect of roll and yaw joints are investigated in this case. Each roll and yaw joints are actuated with different sine-wave speed functions. Each roll and yaw joints functions consist of 3 parameters to optimize. For both the left and right forelimbs, there are additional 4 parameters (roll and yaw joints on left and right limbs have different amplitudes).With the 3 parameters from pitch joints add up to a total of 7 parameters to optimize. These 7 parameters are correspond to the amplitudes of the sine-wave speed functions, while the frequency and phase difference of the sine-wave speed functions are set to 0.8 and 0.0 respectively.

Table 5.5 shows the GA parameters used. The GA simulation stops after 30 iterations of evaluations, making a total of 3000 evaluations and the average fitness at the 30^{th}iteration is 0.977164. Table 5.6 shows the optimized parameters using GA. Fig.5.12 shows the fitness trend for the double frontal quadruped limbs with roll, pitch and yaw joints activated in scapula-body joint and without the forward thrust. The average fitness exceeds 0.8 at the 5^{th}iteration and converge is observed in 9th iteration.

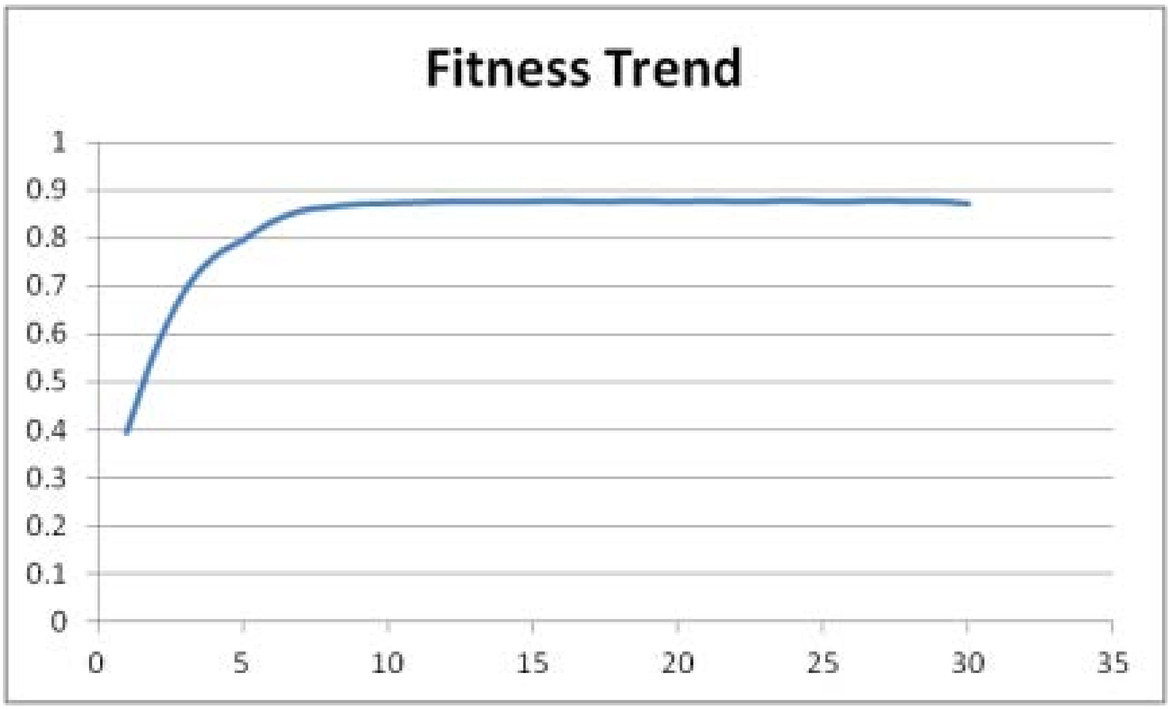


Figure 5.12: Fitness Trend for double frontal quadruped limbs configuration walking phase with roll, pitch, yaw joint and without forward thrust

Table 5.5: Parameters for setting up GA to optimized walking phase with roll, pitch roll and without forward thrust for double forelimb configuration

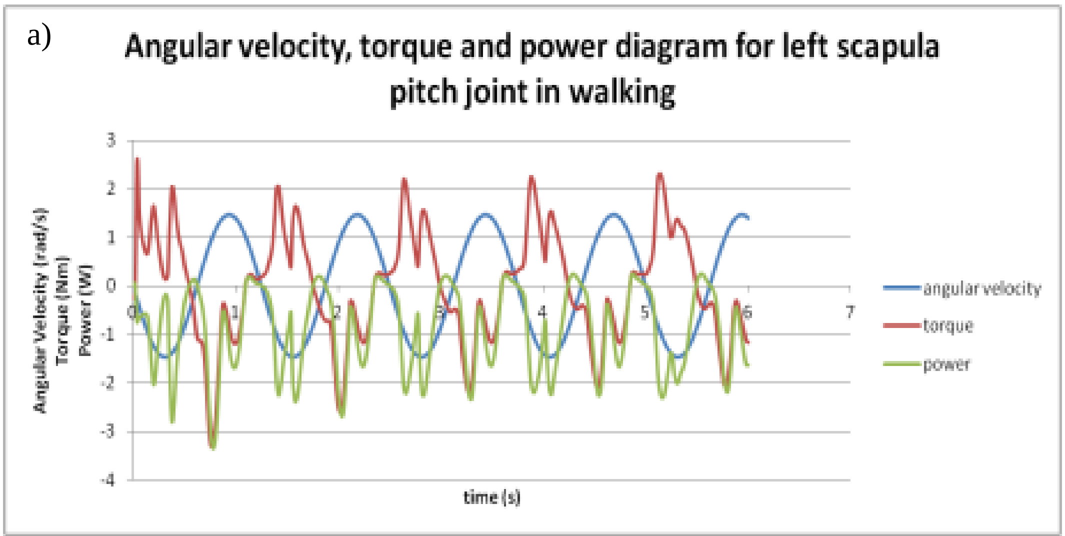
Population	100
Chromosome Length	7
Exit Criteria	3000 evaluations
Crossover rate	85%
Mutation rate	5%
Elitism	5%
Walking Speed	0.078m/s

Table 5.6: Optimized parameters using GA for walking phase with roll, pitch roll and without forward thrust

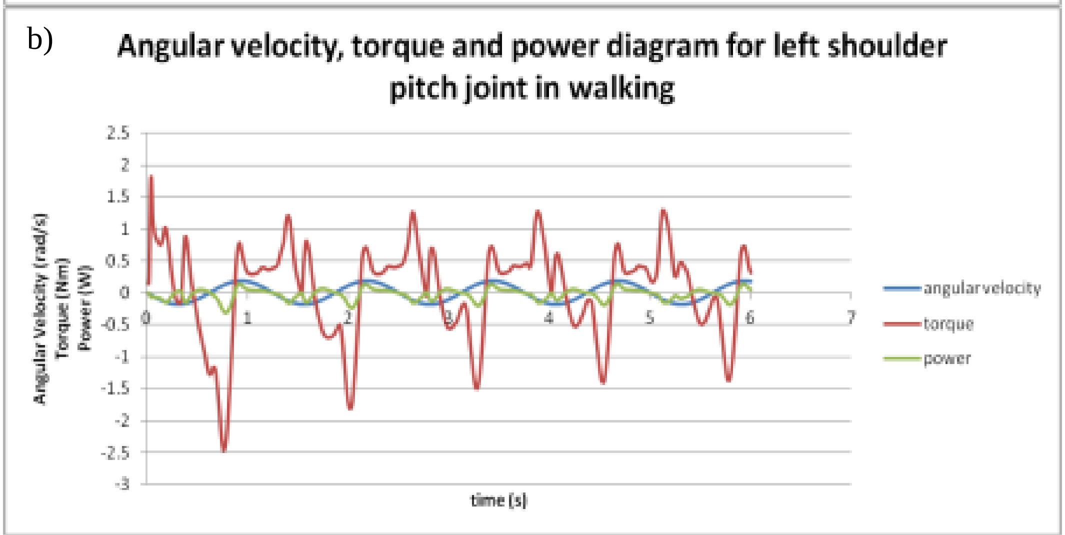
Allele	Parameter	Corresponding value	Value
0	0.47	a_1 (scapula pitch)	1.47
1	-0.82	a_2 (shoulder pitch)	0.18
2	-0.05	a_3 (elbow pitch)	1.425
3	0.0	a_4 (left scapula roll)	0.0
4	-0.19	a_5 (right scapula roll)	-0.28
5	-0.45	a_6 (left scapula yaw)	-0.9
6	-0.71	a_7 (right scapula yaw)	-1.42

The torque function, angular velocity and power consumption for the double forelimb configuration is shown in Fig. 5.13 for the left forelimb and in Fig. 5.14 for the right forelimb. Fig. 5.13 (d) shows that the roll joint for left scapula consumes no power and it is not needed to be actuated. Fig. 5.13 (b) shows that the movement of shoulder joint is minimal when the double forelimb configuration is walking, thus its power consumption is less as compared to other joints. The impact on shoulder joint is the same as the result for double forelimb without roll, pitch and yaw actuated. The scapula pitch joint consumes more power in this case (Fig. 5.13 (a)). This implies that when the roll and yaw joints are actuated, the scapula-body joint consumes more power in overall, as compared with Case 1 (Fig. 5.4 (a)). With the roll and yaw joints activated, the power consumed by the elbow joint is lesser in comparison to Case 1 (Fig. 5.4 (c)). While activating the roll and yaw of scapula-body joints, the movements of these two joints enable a side shift of the double forelimb which reduces the stress on the elbow joint. The optimal results show that the yaw joint is moving more than the roll joint.

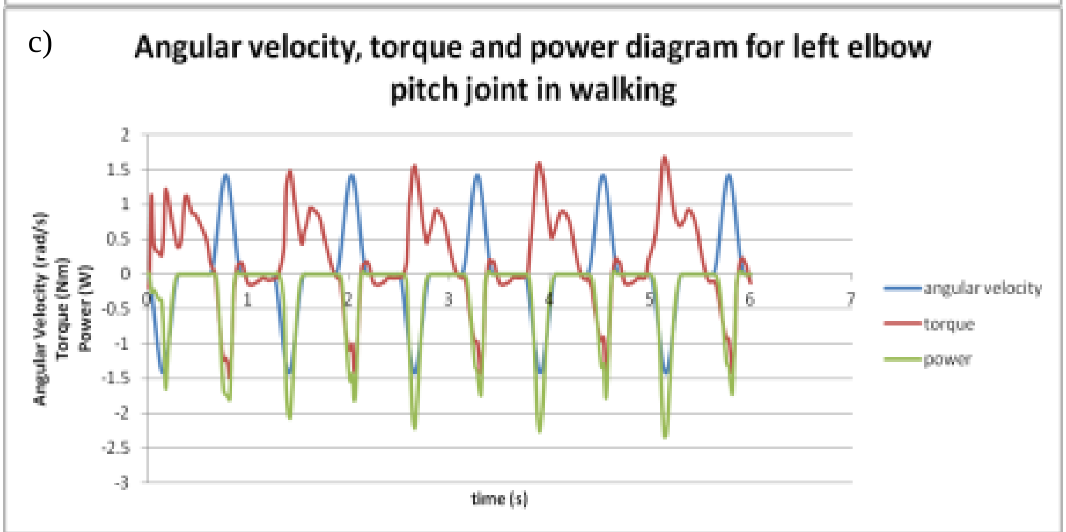
a)


Angular velocity, torque and power diagram for left scapula pitch joint in walking
3
2
1
0
-1
-2
-3
-4
0 1 2 3 4 5 6 7
Angular Velocity (rad/s)
Torque (Nm)
Power (W)
time (s)
angular velocity
torque
power

b)


Angular velocity, torque and power diagram for left shoulder pitch joint in walking
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2.5
-3
0 1 2 3 4 5 6 7
Angular Velocity (rad/s)
Torque (Nm)
Power (W)
time (s)
angular velocity
torque
power

c)


Angular velocity, torque and power diagram for left elbow pitch joint in walking
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2.5
-3
0 1 2 3 4 5 6 7
Angular Velocity (rad/s)
Torque (Nm)
Power (W)
time (s)
angular velocity
torque
power

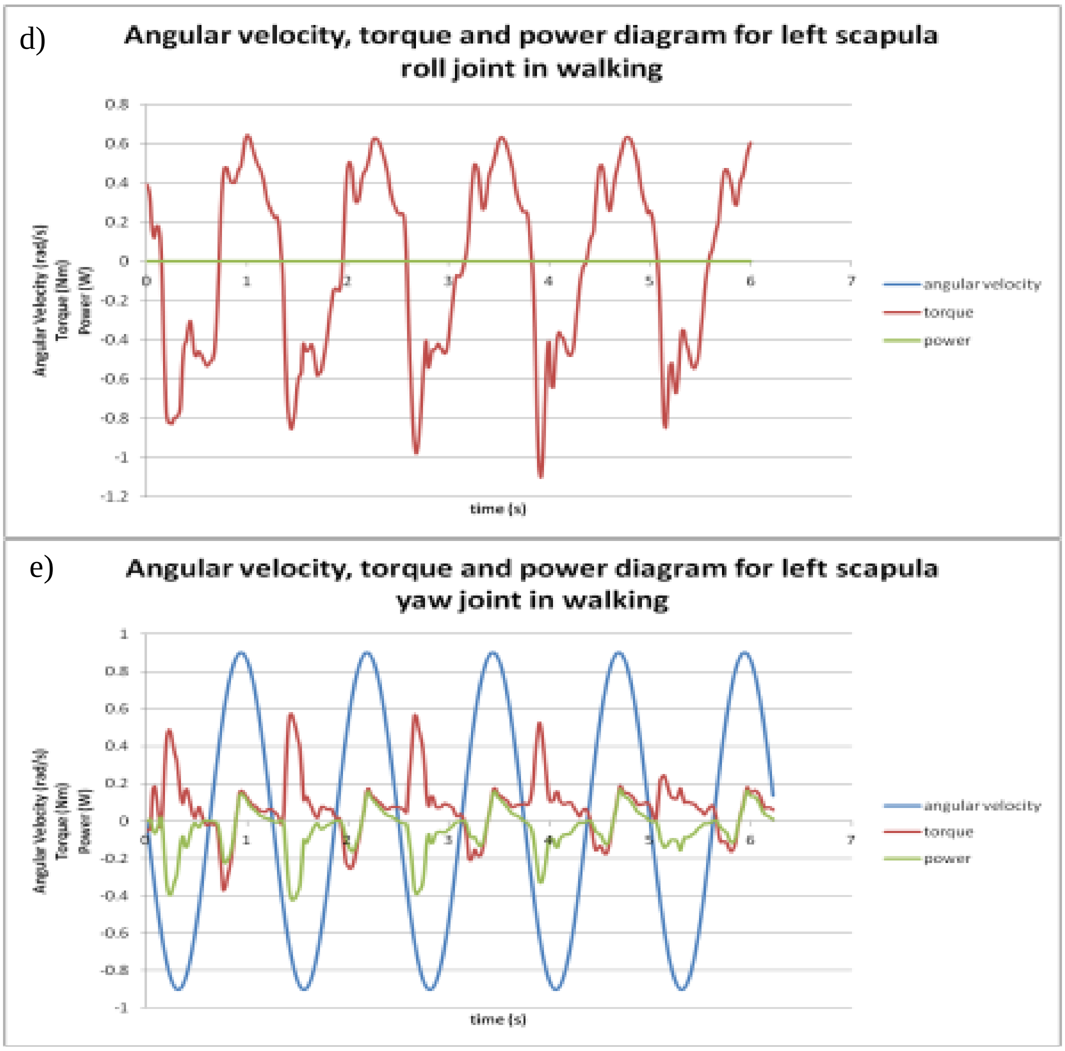
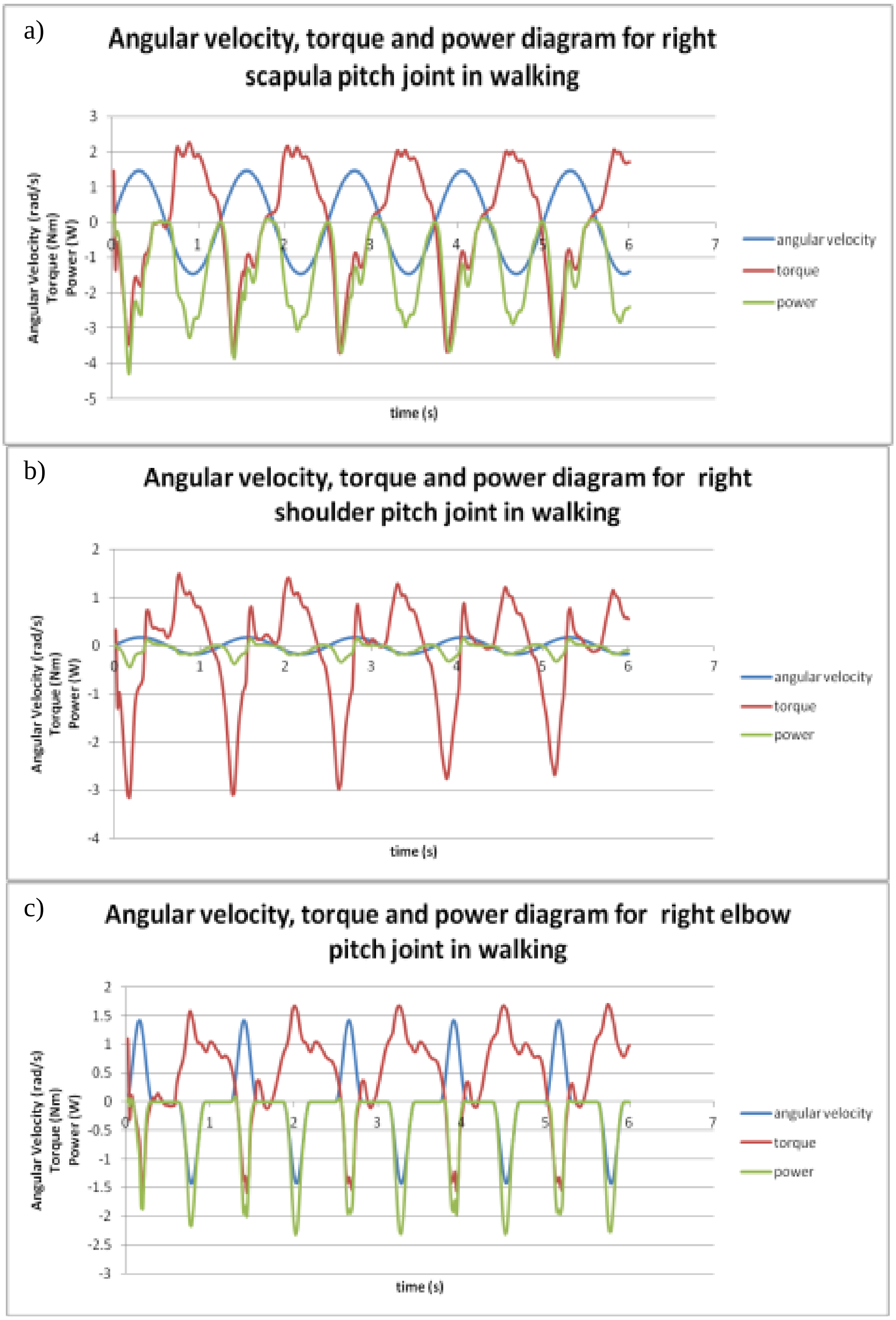


Figure 5.13: Torque function, angular velocity and power for left frontal limb in walking with roll, pitch yaw activated for scapula-body joint and without forward thrust. a) Scapula pitch joint b) shoulder joint c) elbow joint d) scapula roll joint e) scapula yaw joint

For right forelimb, the overall effect is similar to the left frontal limb. The only difference is that the roll, pitch and yaw of scapula-body joints in right frontal limb consume more power than its counterpart on the left frontal limb. This is due the non-synchronized of the left and right frontal limb's roll, pitch, yaw joint. With the difference power consumed in the roll, pitch and yaw of the scapula-body, the configuration is not walking in a straight line.


a)
Angular velocity, torque and power diagram for right scapula pitch joint in walking
Angular Velocity (rad/s)
Torque (Nm)
Power (W)
time (s)
angular velocity
torque
power
b)
Angular velocity, torque and power diagram for right shoulder pitch joint in walking
Angular Velocity (rad/s)
Torque (Nm)
Power (W)
time (s)
angular velocity
torque
power
c)
Angular velocity, torque and power diagram for right elbow pitch joint in walking
Angular Velocity (rad/s)
Torque (Nm)
Power (W)
time (s)
angular velocity
torque
power

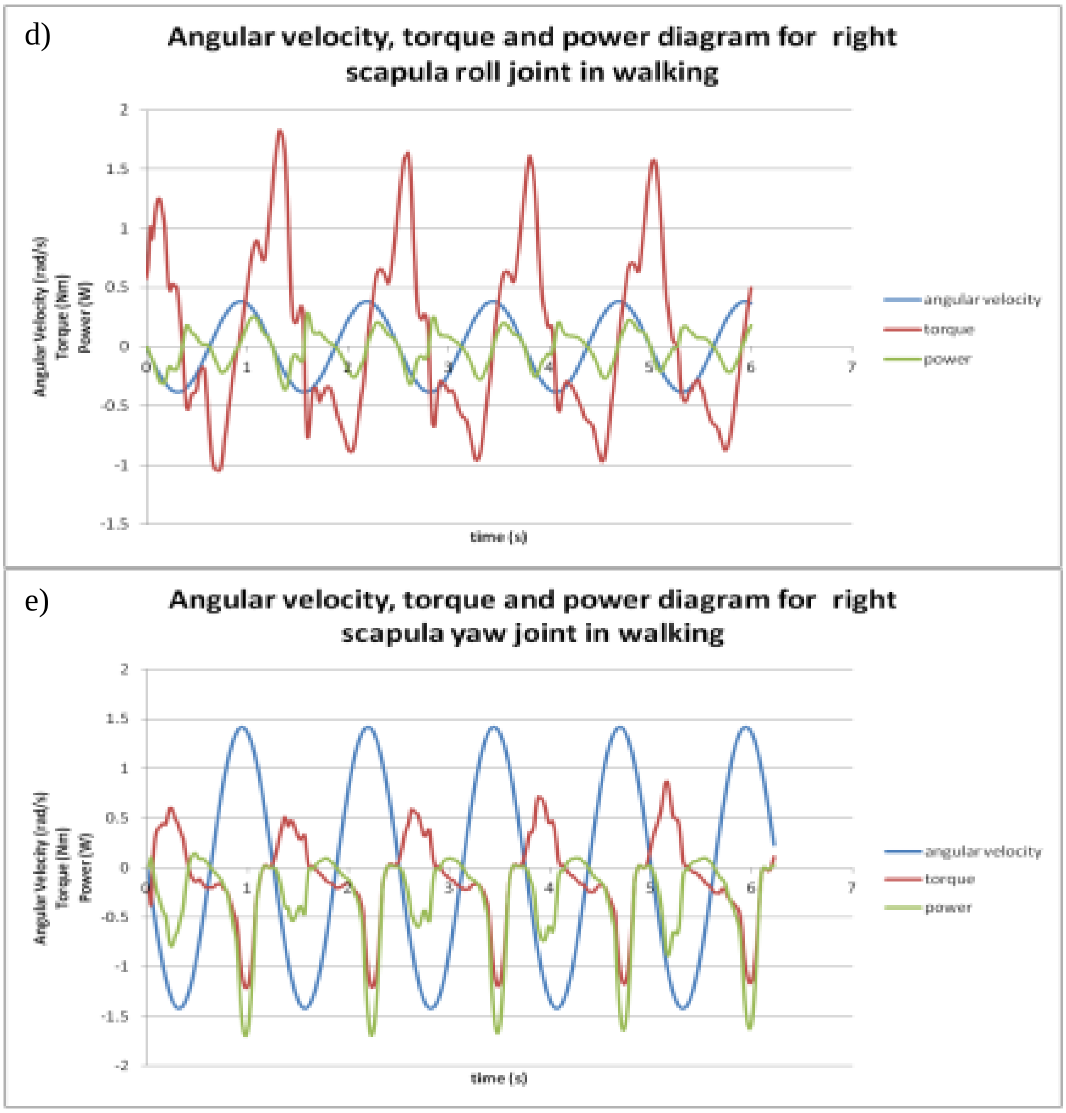


Figure 5.14: Torque function, angular velocity and power for right forelimb in walking with roll, pitch yaw activated for scapula-body joint and without forward thrust. a) Scapula pitch joint b) shoulder joint c) elbow joint d) scapula roll joint e) scapula yaw joint

For ideal case of walking, the double forelimb configuration has to follow a straight line along the x-axis. When roll and yaw joints are activated, it is found that the change in z direction for the configuration is 0.133m causing the configuration to drift away from its original path. This is due to the non-synchronized motions of the roll and yaw joints on the left and right sides. By changing the values of the roll and yaw joints in the scapula, the configuration can turn to a desired direction and which shows that the steering of the configuration is effected by the roll and yaw joints. Fig. 5.15 shows the orientation of the double frontal quadruped limbs configuration. The red colored cross, x, in figure indicates the middle point between right and left forelimbs which is used to calculate the orientation of the configuration. The orientation is calculated by taking both the initial and final positions of the red colored cross, x, in the Fig. 5.15.

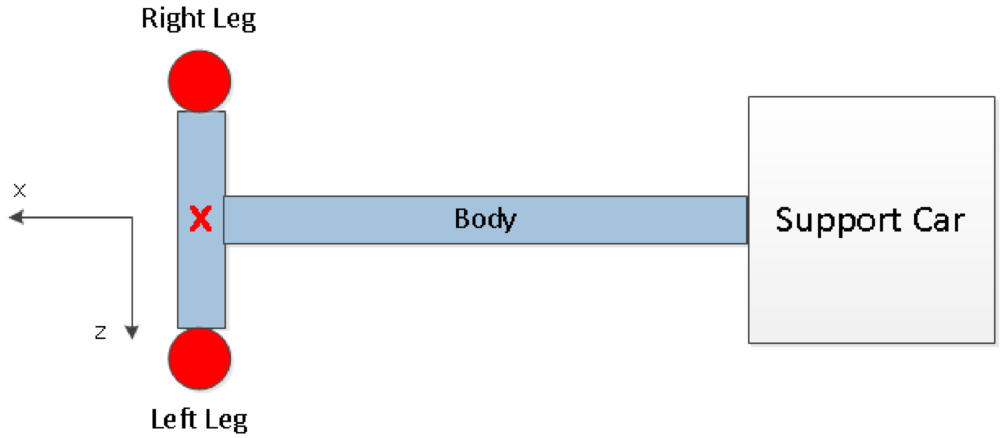


Figure 5.15: Orientation of the double frontal quadruped limbs configuration

5.6 Double frontal quadruped limbs configuration walking phase – Case 4: scapula-body roll, pitch and yaw joint with forward thrust

In this case, a forward thrust is injected by actuating the supporting structure. Similar to case 3, the roll and yaw in the scapula-body joint are activated. There are 7 parameters for GA to optimize. These 7 parameters are the amplitudes of the sine-wave speed functions. The frequency and phase difference of the sine-wave speed functions are set to 0.8Hz and 0.0 respectively.

Table 5.7 shows the GA parameters used. The GA simulation is stopped after 30 iterations of evaluations. The average fitness at the 30thiteration is 0.963387. Table 5.8 shows the GA optimized parameters. Fig.5.16 shows the fitness trend for the double forelimb with roll, pitch and yaw activated in scapula-body joint with forward thrust from the rear.

Table 5.7: GA parameters to optimize the walking phase for double frontal quadruped limbs configuration with roll, pitch and yaw with forward thrust

Population	100
Chromosome Length	7
Exit Criteria	3000 evaluations
Crossover rate	85%
Mutation rate	5%
Elitism	5%
Walking Speed	0.108m/s

Table 5.8: Optimized parameters using GA for walking phase with roll, pitch roll and with forward thrust

Allele	Parameter	Corresponding value	Value
0	-0.04	a_1 (scapula pitch)	0.96
1	-0.38	a_2 (shoulder pitch)	0.62
2	0.41	a_3 (elbow pitch)	2.115
3	-0.19	a_4 (left scapula roll)	-0.38
4	0.12	a_5 (right scapula roll)	0.24
5	0.21	a_6 (left scapula yaw)	0.42
6	-0.29	a_7 (right scapula yaw)	-0.58

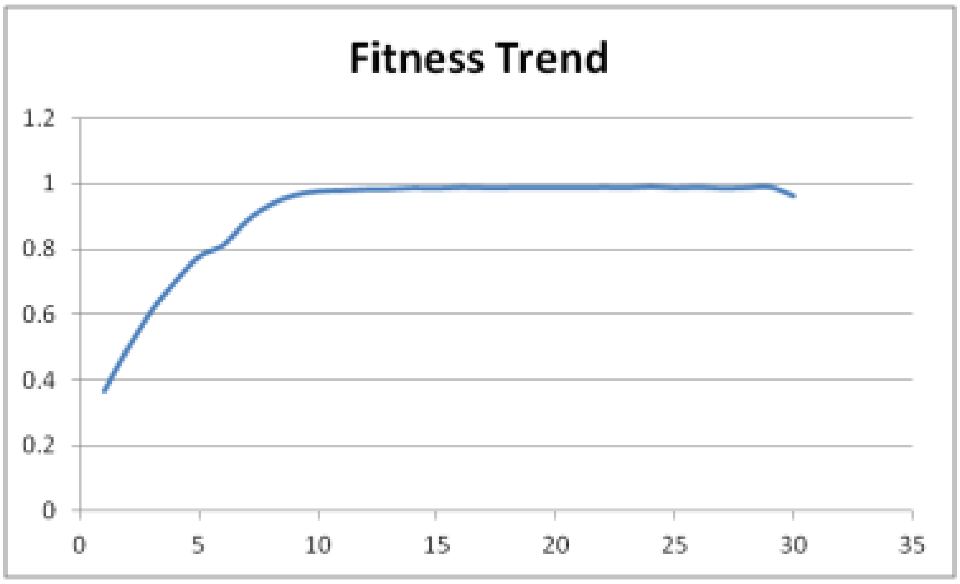
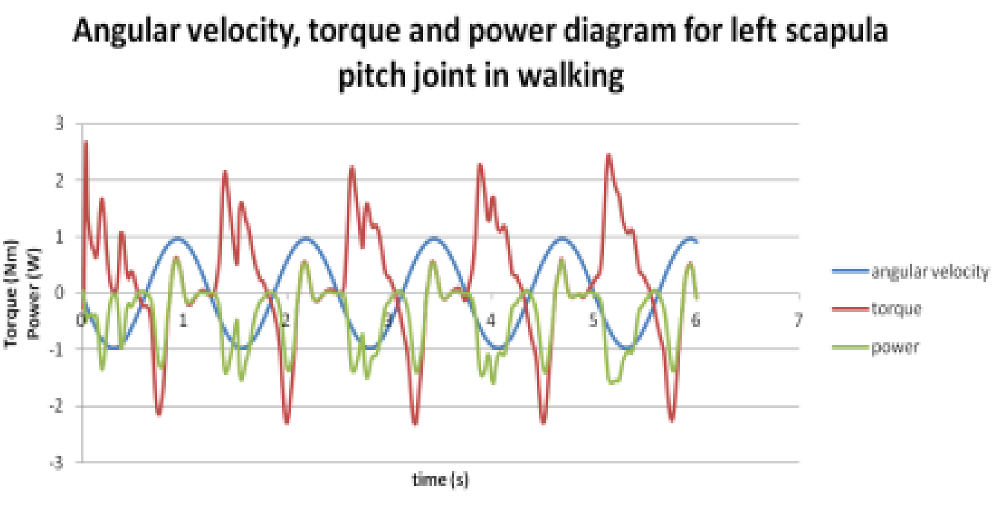


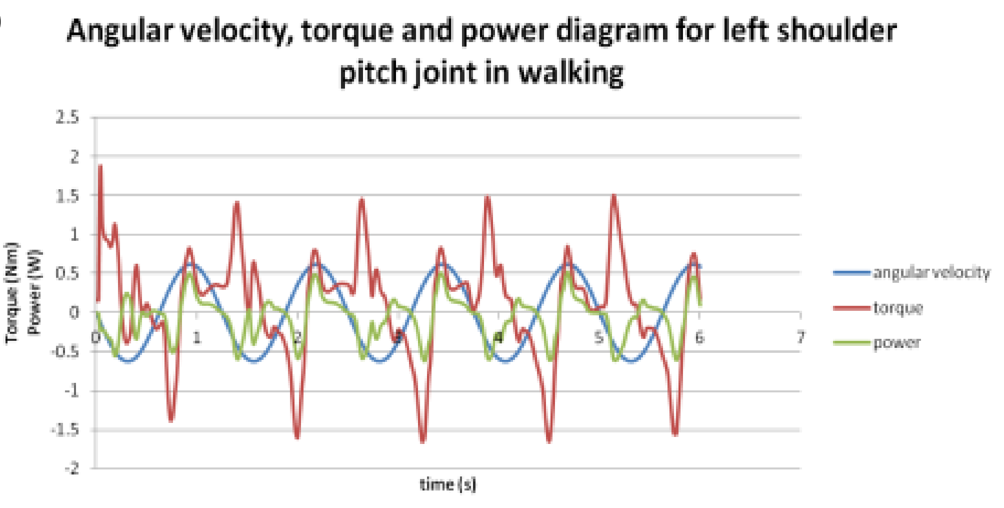
Figure 5.16: The fitness trend for walking phase with roll, pitch and yaw joints on the scapula-body joint with forward thrust

The torque function, angular velocity and power consumption for the double forelimb configuration are shown in Fig. 5.17 for left frontal limb and Fig. 5.18 for right frontal limb. In reaction to the forward thrust, the elbow joints in forelimbs consume more power (Fig. 5.17 (c)). However, due to the increase in the DOF in the scapula-body joints, with roll and yaw, the power consumed in the elbow joint is significantly lesser than the case without roll and yaw. In Case 4, the shoulder joint move significantly to overcome both the forward thrust and to stabilize the motion of the roll and yaw scapula-body joints. From Fig. 5.17 (d) and Fig. 5.17 (e), it is observed that the motion of both the roll and yaw joints are significant than the motion without the forward thrust (Case 3). This can be seen as the reaction to the forward thrust introduced. However, such motion greatly reduces the power consumed in elbow joint.

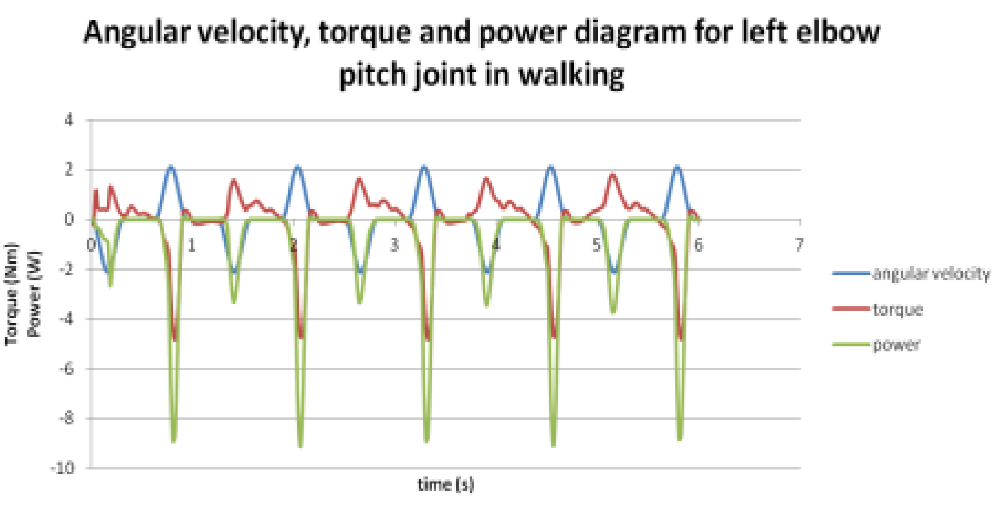
a)


Angular velocity, torque and power diagram for left scapula pitch joint in walking
Angular Velocity (rad/s)
Torque (Nm)
Power (W)
3
2
1
0
-1
-2
-3
0
1
2
3
4
5
6
7
time (s)
angular velocity
torque
power

b)


Angular velocity, torque and power diagram for left shoulder pitch joint in walking
Angular Velocity (rad/s)
Torque (Nm)
Power (W)
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
0
1
2
3
4
5
6
7
time (s)
angular velocity
torque
power

c)


Angular velocity, torque and power diagram for left elbow pitch joint in walking
Angular Velocity (rad/s)
Torque (Nm)
Power (W)
4
2
0
-2
-4
-6
-8
-10
0
1
2
3
4
5
6
7
time (s)
angular velocity
torque
power

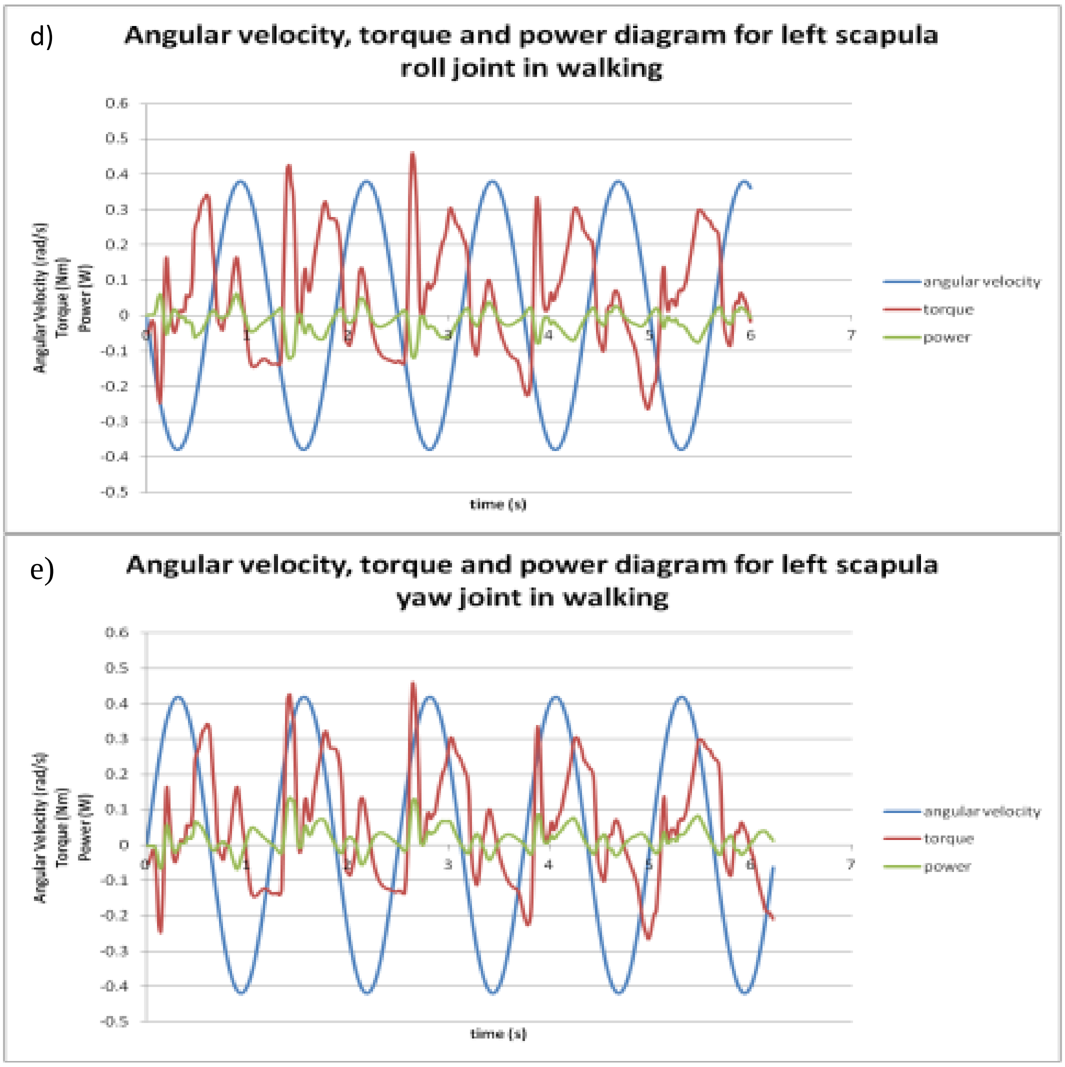
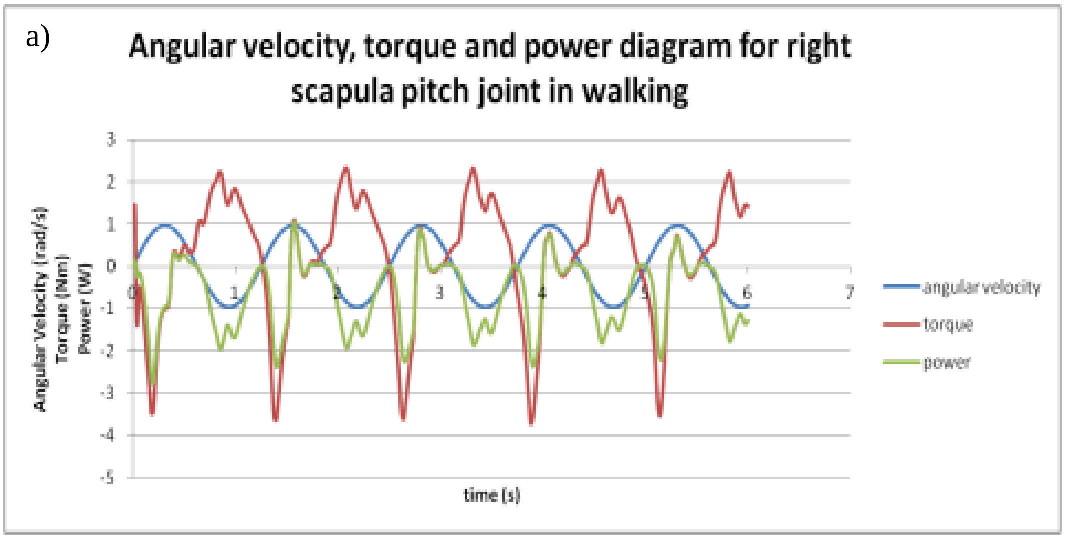
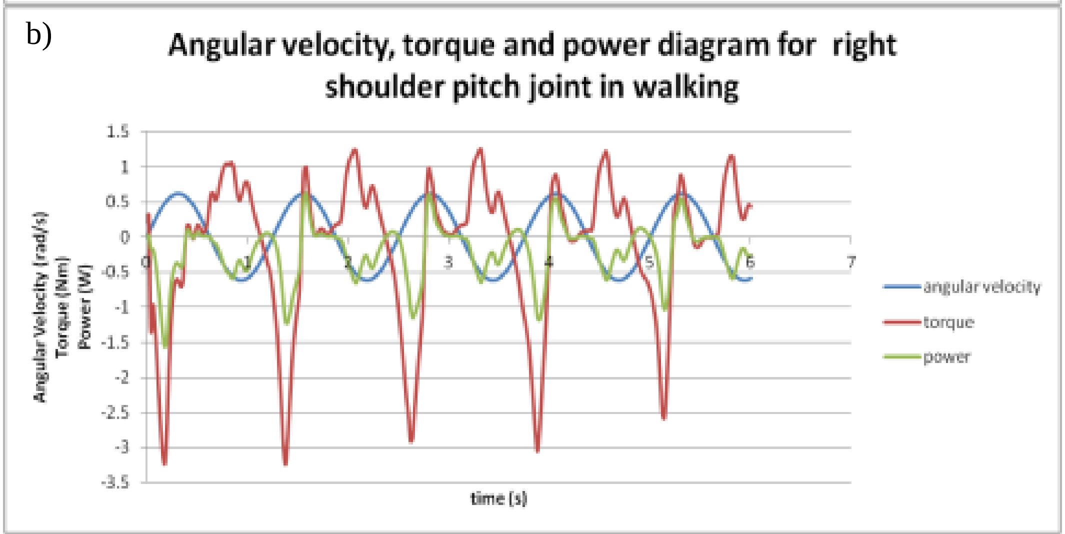
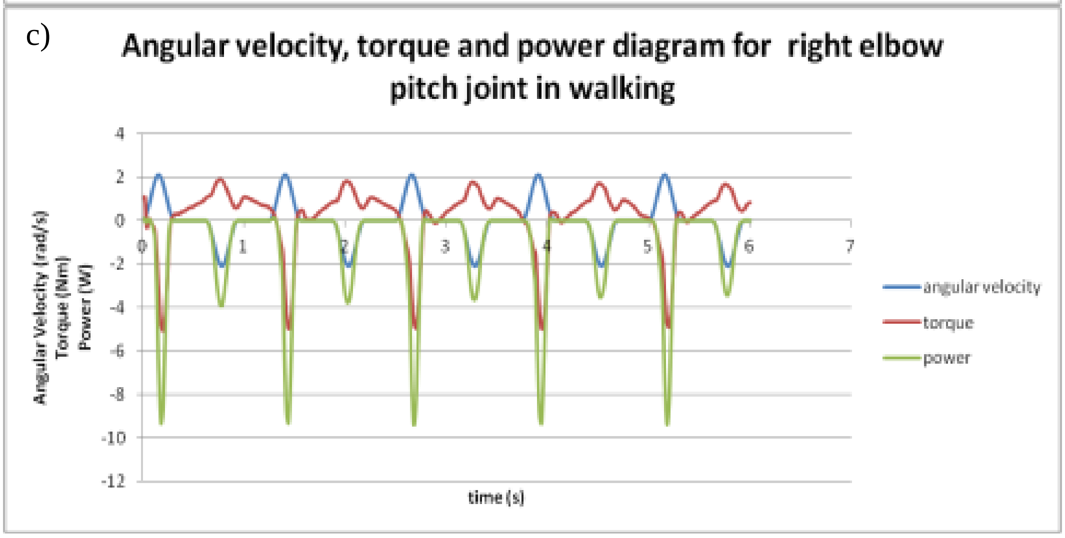


Figure 5.17: Torque function, angular velocity and power for left frontal limb in walking with forward thrust. (a) Scapula pitch joint (b) shoulder joint (c) elbow joint (d) scapula roll joint (e) scapula yaw joint

The response of the right forelimb is similar to the left forelimb. The main difference is that there is higher reaction torque on scapula roll joint. The torque is highest when the right forelimb is going into support stance and the roll joint needs more torque to complete the motion.


a)
Angular velocity, torque and power diagram for right scapula pitch joint in walking
Angular Velocity (rad/s)
Torque (Nm)
Power (W)
3
2
1
0
-1
-2
-3
-4
-5
0
1
2
3
4
5
6
7
time (s)
angular velocity
torque
power


b)
Angular velocity, torque and power diagram for right shoulder pitch joint in walking
Angular Velocity (rad/s)
Torque (Nm)
Power (W)
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2.5
-3
-3.5
0
1
2
3
4
5
6
7
time (s)
angular velocity
torque
power


c)
Angular velocity, torque and power diagram for right elbow pitch joint in walking
Angular Velocity (rad/s)
Torque (Nm)
Power (W)
4
2
0
-2
-4
-6
-8
-10
-12
0
1
2
3
4
5
6
7
time (s)
angular velocity
torque
power

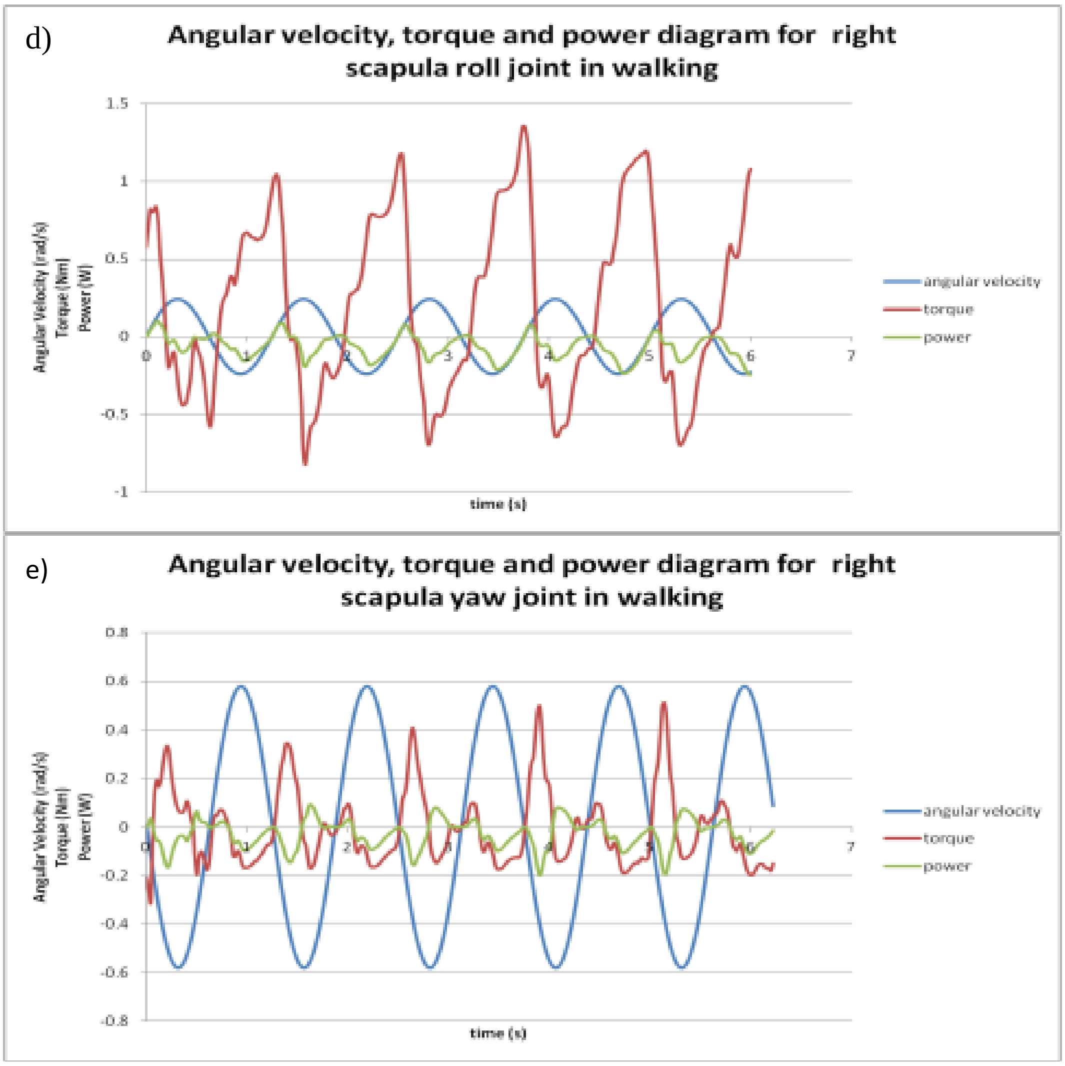


Figure 5.18: Torque function, angular velocity and power for right frontal limb in walking with forward thrust. (a) Scapula pitch joint (b) shoulder joint (c) elbow joint (d) scapula roll joint (e) scapula yaw joint

The main difference between the four cases lies on the power consumption by the elbow joint. In Fig. 5.19, when roll, pitch and yaw in scapula-body joint are activated, the power consumption in elbow joint is reduced significantly as compared to the pitching scapula-body joint alone. The power consumption in the elbow joint is higher when there is a forward thrust from rear. This implies that in a quadruped system, the elbow joints in forelimbs consume more power as the hindlimbs provide forward thrust.

5.7 Summary

In this chapter, four different cases of walking phase is discussed. GA is able to search for the optimal joint functions for these four cases and walking is achieved in these four cases. The different cases show the reaction of the double frontal quadruped limbs configuration with and without roll, yaw joints together with and without forward thrust from the support structure. The results give us some foresight of the performance in quadruped configuration's walking.

Chapter 6

Towards the Quadruped System: The Quadruped Configuration

6.1 Introduction

To complete the quadruped configuration, the support car in double forelimb configuration is replaced with two hind limbs. As the design of the quadruped configuration is modular (each limb has the same design and dimensions), the only difference between the front limbs and hind limbs is in the initial limp positions and limp posture.

6.2 Simulating the quadruped configuration

The hindlimbs are simulated in a different resting posture as compared to forelimbs. The resting posture of hind limbs follows the posture of a dog at rest. Fig. 6.3 shows the quadruped configuration with hind limbs. With this design, the hind limbs provide a pushing force when the configuration during the course of standing up.

The centers of the scapula links are to be shifted from C_1 to C_2 (Fig. 6.1) specifically for the resting posture of the hind limbs. C_1 in Fig.6.1 shows the mirrored center of the link while C_2 shows the desired new position. The position of C_1 and the orientation of the scapula-shoulder link in C_1 are taken as the mirrored values of the forelimbs' scapula-shoulder links. From a 3D point of view, C_1 and C_2 are in the same x-y plane. The rotation from C_1 to C_2 forms an arc in XY plane and a cone in YZ plane (Fig. 6.2).

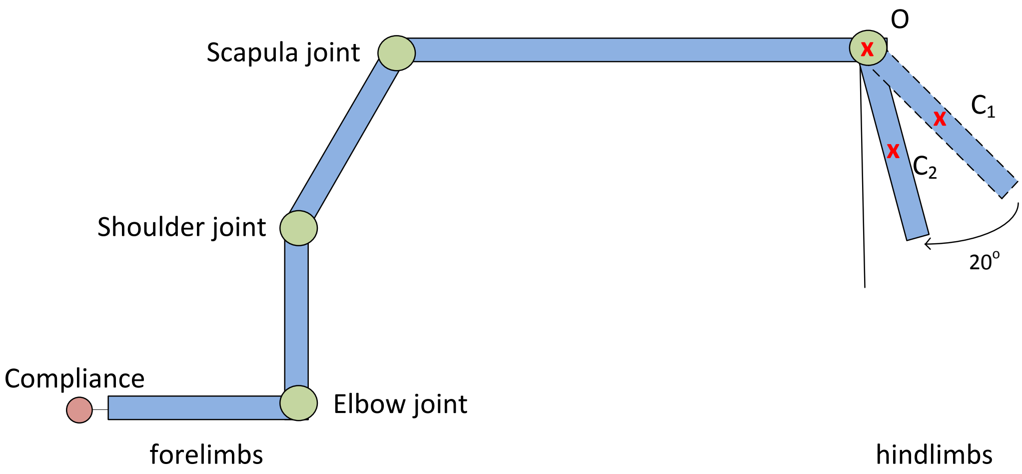


Figure 6.1: Change in initial posture for the scapula link of hindlimbs

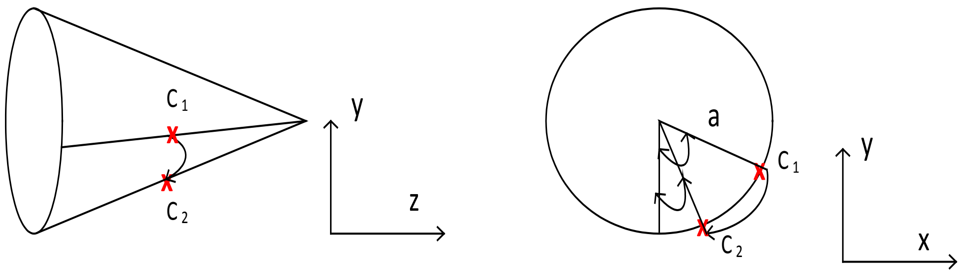


Figure 6.2: Rotation from C_1 to C_2 which is in a cone shape

The objective is to move the hindlimbs' scapula-shoulder link for certain degree (20° is arrived at this study) in clockwise direction. The length of the scapula link is L while *a* is the distance from the center of the cone to the x-y plane containing C1 and C2 (Fig. 5.2). As C1 and C2 are in the same x-y plane, the value of *a* should remain the same after rotation and *a* is calculated using equation 6.1.

$$a = \sqrt{{x_{c1}}^2 + {y_{c1}}^2} = \sqrt{{x_{c2}}^2 + {y_{c2}}^2}$$

$$\theta_1 = \tan^{-1}\frac{x_{c1}}{y_{c1}}$$

$$\theta_2 = \theta_1 - \theta_a$$

$$x_{c2} = a \sin\theta_2 \ , \qquad y_{c2} = a\cos\theta_2 \qquad (6.1)$$

From equation 6.1, the new center for the scapula link is obtained. θ_a is a constant which is 20 degree in this research. The coordinate for the shoulder joint is thus (x_{c2}, y_{c2}, z) where z is always a fixed value. There is a limit length for resting posture, which is the height of O (Fig.6.1) from the ground which is fixed at h = y_{c1} + 0.1m + compliance length/2. The posture of the elbow-compliance link is defined before defining the shoulder-elbow link to ensure that the hind limb foot touches the ground at the initial posture.

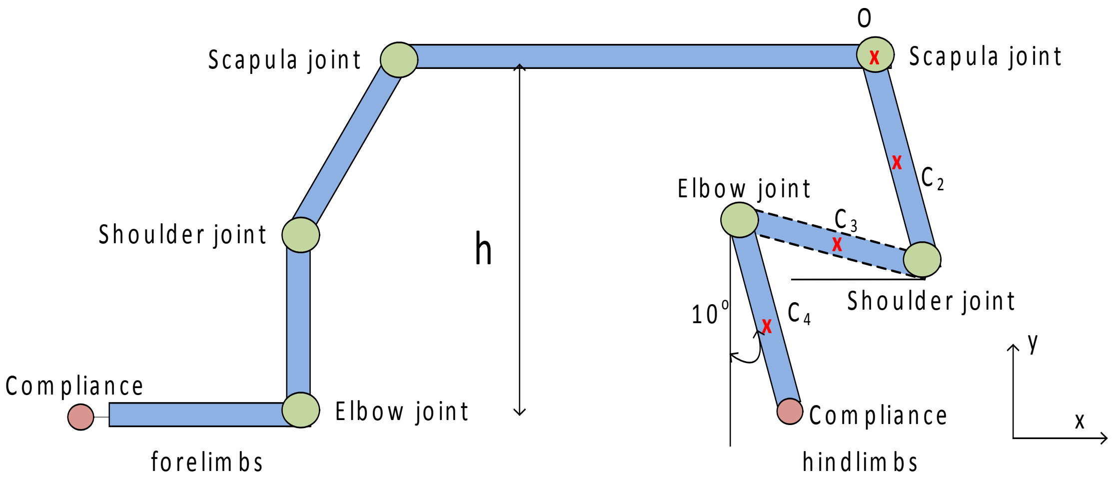


Figure 6.3: Expected position of the hindlimbs' elbow-compliance links

From equation 6.1, the position of the shoulder joint is known.

$$x_{shoulder} = 2x_{c2} = 2a\sin\theta_2 \ , \qquad y_{shoulder} = 2y_{c2} = 2a\cos\theta_2$$

In Fig 6.3, C_3 is the center of the elbow-compliance link and the length of the link is l = 0.1m. The compliance height is set to h_c = 0.01m. The y-position of the elbow joint can thus be calculated, which is done by deducting the slanted link length and compliance size from h. θ_b is the elbow-compliance angle as reference to y-axis which is 10 degree here.

$$y_{elbow} = h - (l\cos\theta_b + h_c\ \cos\theta_b) \tag{6.2}$$

With y_{elbow} known, the angle $\theta_{shoulder}$ and center of the shoulder-elbow link are calculated with (6.3).

$$y_{c3} = (y_{elbow} - y_{shoulder})/2$$

$$\theta_{shoulder} = \sin^{-1}\frac{2y_{c3}}{0.1}$$

$$x_{c3} = x_{shoulder} - \frac{(l\cos\theta_{shoulder})}{2}$$

$$x_{elbow} = x_{shoulder} - (l\cos\theta_{shoulder}) \tag{6.3}$$

The C_4position can be calculated when the position of the elbow joint is obtained.

$$y_{c4} = y_{elbow} - \frac{(l\cos\theta_b)}{2}$$

$$x_{c4} = x_{elbow} + \frac{(l\sin\theta_b)}{2}$$

Finally, the positions of the hindlimbs' compliance is calculated.

$$y_{compliance} = y_{elbow} - (l\cos\theta_b + \frac{h_c\cos\theta_b}{2})$$

$$x_{compliance} = x_{elbow} + (l\sin\theta_b + \frac{h_c\sin\theta_b}{2})$$

With the position and rotation coordinates of every link in the hind limbs are calculated, the quadruped configuration is built and initialized in resting posture.

6.2.1 Quadruped: From standing to walking

The scapula roll, pitch and yaw joints, and, t shoulder pitch and elbow pitch joints are controlled by speed functions. In standing motion, the roll and yaw of the scapula-body joints act as passive joints. Only the pitch joints for the limbs are actuated using speed function. The left and the right limbs are symmetrical and thus the speed function used to generate the standing motion is the same, which means that the left and right scapula-body pitch joints use the same speed functions and the same holds for the shoulder and elbow pitch joints. The forelimbs use the parameters obtained for the double forelimb configuration, therefore only the hind limbs' parameters need to be optimized. The chromosome population is set to 100 while each chromosome has 9 genes representing the unknown parameters.

6.2.2 Genetic Algorithm and cost functions to optimize in standing phase

The objective of the hind limbs is not only to stand up but also to provide a pushing force to the front limbs in the course of standing. The fitness function for GA to search for hindlimbs' joints function in the standing up phase is the same as front limbs as shown in section 3.3. The only difference is an addition criteria of the standing height. After the quadruped stood up, the standing height for both the hindlimbs and forelimbs are best at the same height. Therefore, the fitness function is calculated after 2 s of simulation based on the following five cost functions.

1. Distance between the y-coordinates of the centers of L_1 and L_2. The fitness is set to unity at 70% of max distance.
2. Distance between the y-coordinates of the centers of L_2 and L_3. The fitness is set to unity at 80% of max distance.
3. Compute the speeds along y and x axes (dy/dt and dx/dt) of the scapula-body joint. The fitness is set to unity when both the velocities (dy/dt and dx/dt) are zero.
4. Compute the COG of the configuration and obtain its angle with respect to y-axis at foot contact, θ_1. The fitness is set to unity when θ_1 is zero.
5. Compute the difference between the forelimb height and hind limb height. The fitness is set to unity when both standing limbs are at the same height.

The total fitness is taken as the average of the above five fitness' values. The coordinates and standing posture are shown in Fig.6.4(1st to 4thcost functions) and Fig.6.5 (5thcost function).

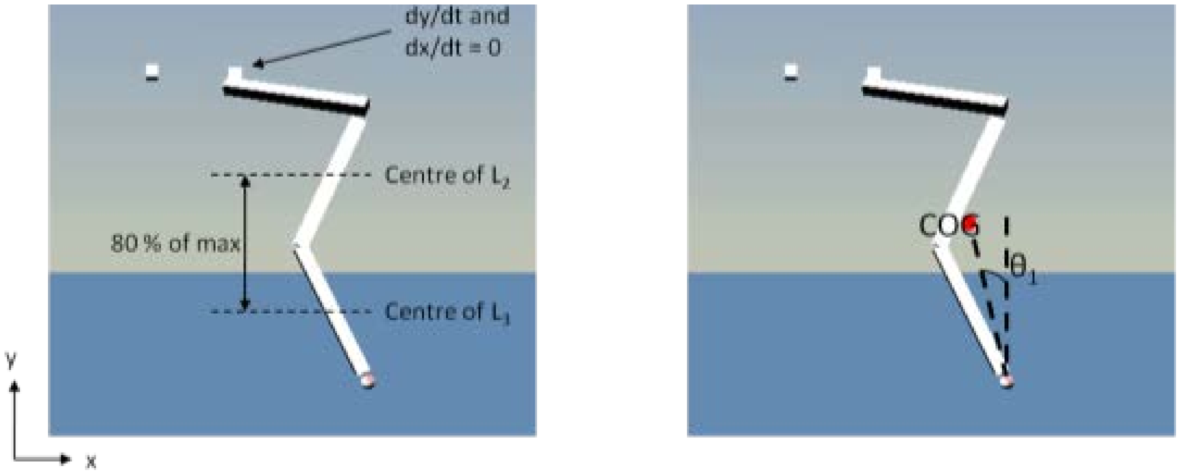


Figure 6.4: The coordinates for fitness calculation and configuration’s standing posture

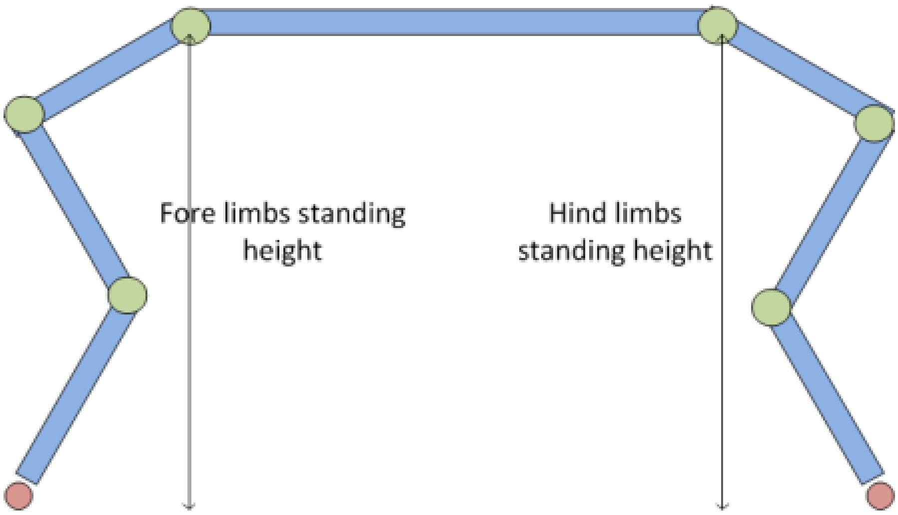


Figure 6.5: The 5th cost functions for standing up motion

The GA is stopped after 30 iterations. Fig. 6.6 shows the stabilized standing posture for quadruped configuration. GA is able to achieve optimized joints function that allowed the quadruped configuration to stand up.

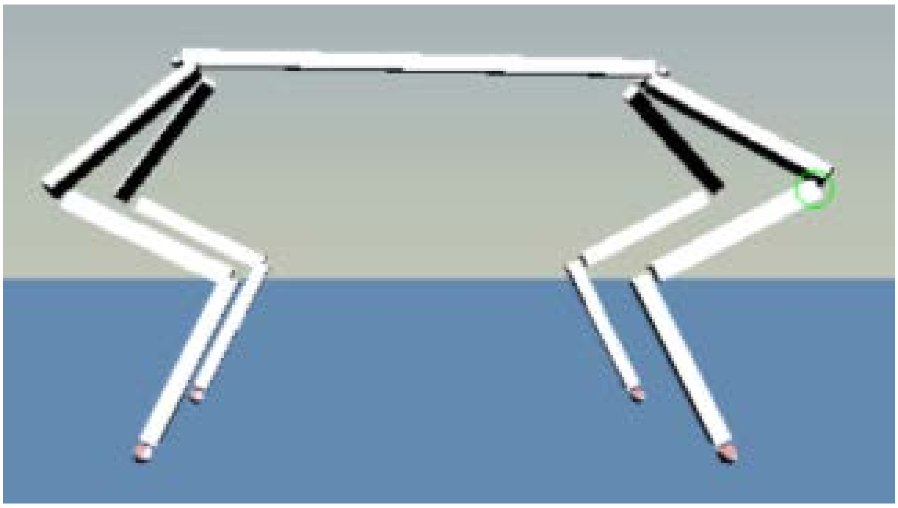


Figure 6.6: Stabilized standing posture for the quadruped

6.2.3 Genetic Algorithm and cost functions to optimize in walking phase

Unlike the approach taken in single forelimb and double forelimb configurations, an intermediate phase is not utilized for the quadruped to achieve walking motion. With the hind limbs inaction, the cost functions in GA to achieve a stable intermediate phase doubles as opposed to double forelimb configuration and thus it is more difficult for GA to search for an optimal result. Due to which, instead of going through an intermediate phase, the approach in quadruped configuration is to directly go into walking phase. In order to achieve a stable walking motion, the sequences of each joint need to be known. The walking pattern to achieve is the trot gait and the associated gait graph is shown in Fig.6.7.

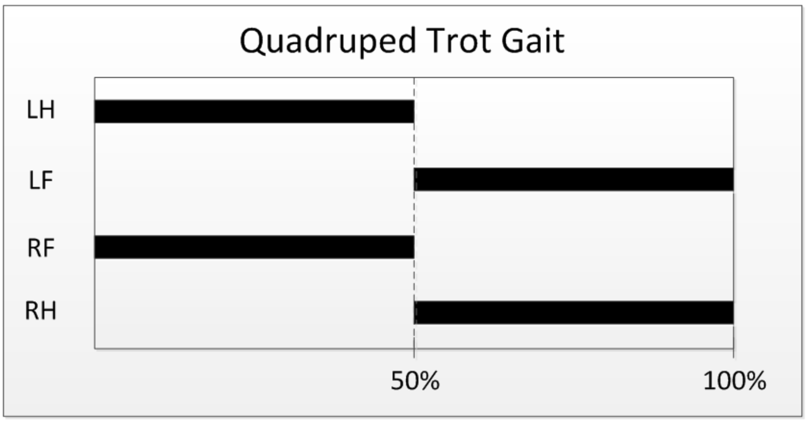


Figure 6.7: Gait graph for trot gait

Fig.6.8 shows the sequences of each foot for the trot gait. Two diagonal feet are lifted up while the remaining two diagonal feet are on the ground. For example, when the right front foot and left hind foot (blue coloured in Fig. 6.8) are lifted up, the left front foot and right hind foot (green coloured in Fig. 6.8) must be on the ground. This is the basic sequence for a trotting gait and the feet on the ground support the whole body's weight thus the reaction forces on these feet can be highest. Vice versa, the feet that are lifted up have zero reaction forces. Therefore, there are two different phases in the trot gait and to make it easy to be differentiated, the phase where the feet are on the ground supporting the weight is named as the support phase while the phase where the feet are lifted up is named as the swing phase.

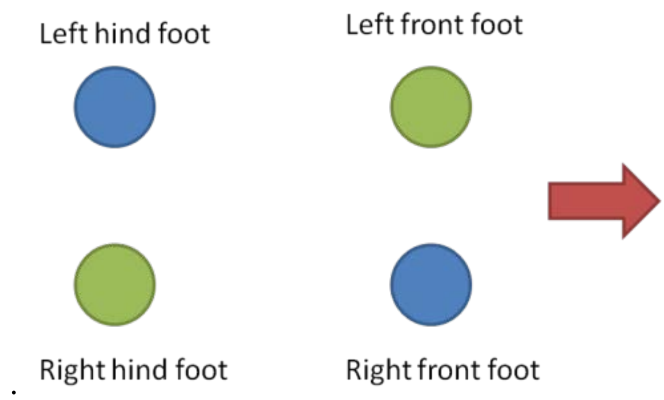


Figure 6.8: Top view of foot placement for quadruped trot gait

Fig. 6.9 shows the joint distributions and joint types for the quadruped configuration. Each limb has five joints and the sequence of each joint motion need to be determined in order to achieve a walking motion. The sequence of each joint is shown in Table 6.1 which consists of four sub tables. Each of the sub table shows the sequences of joint motions in respective limbs. The column in yellow represents the support phase while the non-coloured column represents the swing phase. When the right front limb is in support phase, the left front limb is in opposite phase which is the swing phase. In the support phase of the right front limb, the lower and upper limb pitch joints are pitching back, and the upper limb roll and yaw joints respectively are rolling right and yawing back. The exact motion is described as shown in Fig. 6.10. The sequences for the middle limb pitch joint is not determined because the effect of its direction of motion is not able to predict. Due to which, GA is used to search for the optimal sequence for the middle limb pitch joints. These joint sequences are determined by determining the correct postures of the limbs in corresponding phases, and the movements required for standing posture is achieved.

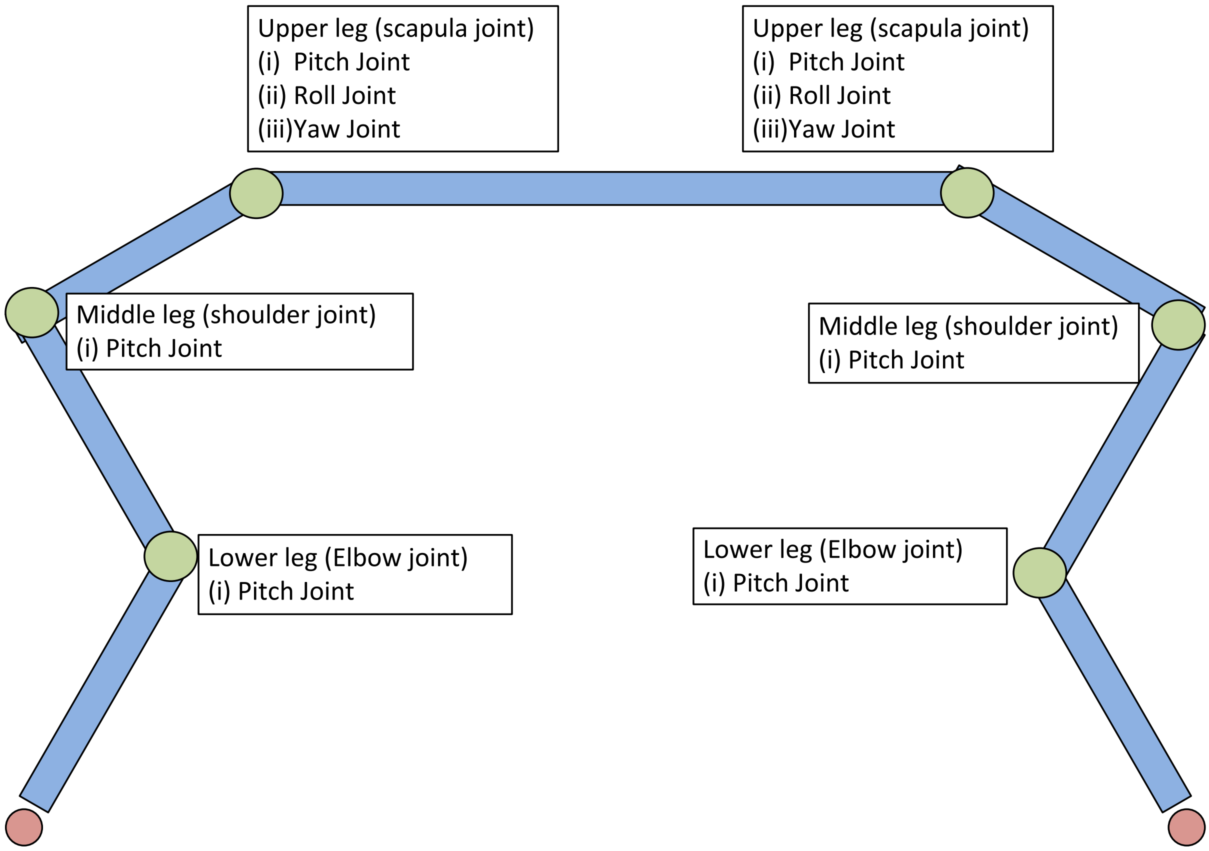


Figure 6.9: Joint distributions and types for the quadruped configuration

Table 6.1: The sequences joint motions in different phases (a) right front limb (b) left front limb (c) right back limb (d) left back limb

(a)	Limb	Right Front				
Time	Joints	lower limb pitch	middle limb pitch	upper limb pitch	upper limb roll	upper limb yaw
support phase		pitch back	Not determined	pitch back	roll right	yaw back
swing phase						
support phase						
swing phase		pitch forward	Not determined	pitch forward	roll left	Yaw front

(b)	limb	Left Front				
Time	Joints	lower limb pitch	middle limb pitch	upper limb pitch	upper limb roll	upper limb yaw
support phase						
swing phase		pitch forward	Not determined	pitch forward	roll right	Yaw front
support phase		pitch back	Not determined	pitch back	roll left	yaw back
swing phase						

(c)	limb	Right Back				
Time	Joints	lower limb pitch	middle limb pitch	upper limb pitch	upper limb roll	upper limb yaw
support phase						
swing phase		pitch back	Not determined	pitch forward	roll left	Yaw front
support phase		pitch forward	Not determined	pitch back	roll right	yaw back
swing phase						

(d)	limb	Left Back				
Time	Joints	lower limb pitch	middle limb pitch	upper limb pitch	upper limb roll	upper limb yaw
support phase		pitch forward	Not determined	pitch back	roll left	yaw back
swing phase						
support phase						
swing phase		pitch back	Not determined	pitch forward	roll right	Yaw front

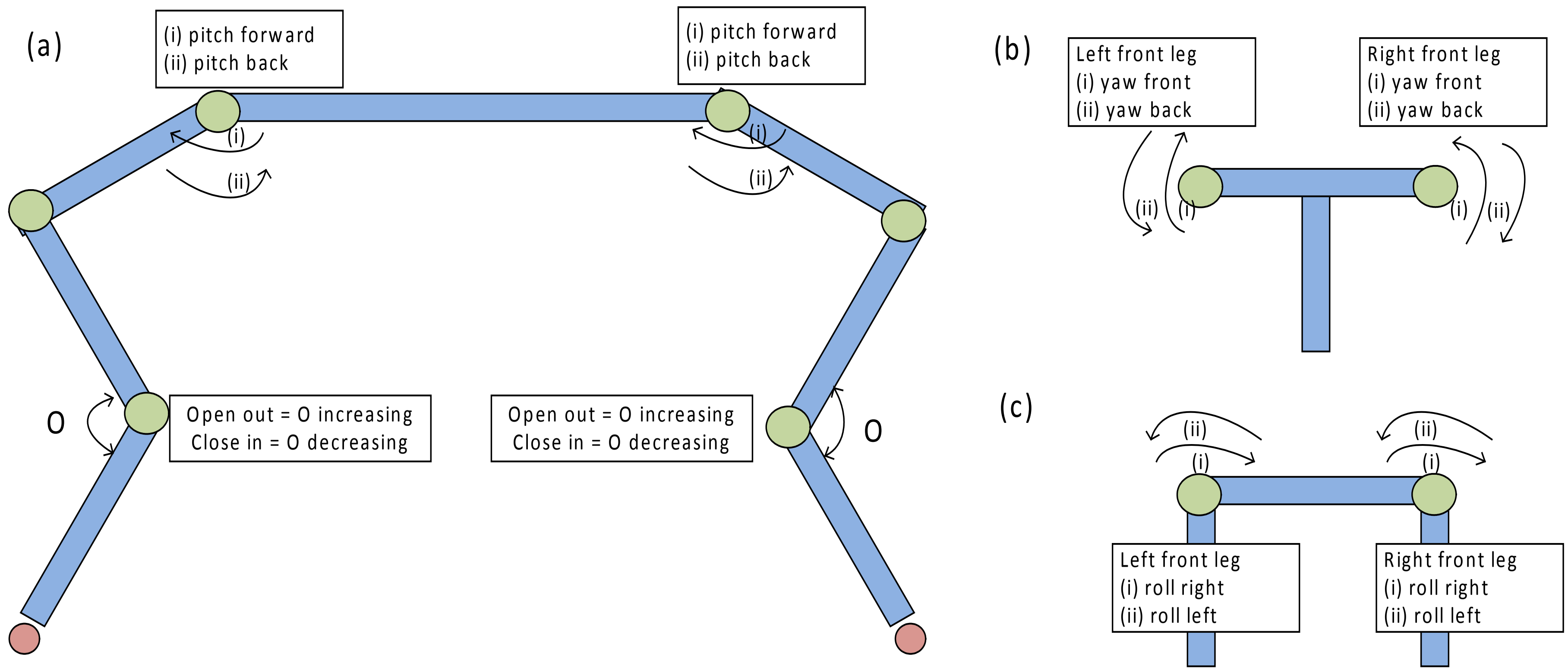


Figure 6.10: Motion directions of joints (a) side view, (b) top view, (c) back view

To make the quadruped configuration to move, optimal joint functions are needed. Each joint is speed controlled and to let the joint to move periodically for continuous walking motion, the sine function is used as the speed function for each joint. As the sequence motion of each joint is determined, the phase difference of each joint is known. For example, for joints having opposite motion sequences, the phase difference is always 180° and otherwise 0°. Besides, to ease the effort in GA search, the joint function frequency is chosen as 0.8 Hz which is just slightly below 1 Hz which is reasonable with respect to real world motor reaction times. Therefore, the parameters that needed to be optimized are the amplitudes of the sine functions.

As shown in Fig. 6.9, each limb has five joints and thus each limb has 5 parameters to optimize. To ensure that the quadruped configuration can walk properly, same amplitude is applied to the left and right limbs. For example, the right front limb lower pitch joint and left front limb lower pitch joint have same amplitude magnitudes but with different signs as the directions of motion are different. This further reduces the number of parameters to optimize to 5 for the front limb and hind limb totaling to 10.

Besides, to ensure that the lower pitch joint (elbow joint) move to its desired destination faster than the scapula-body and shoulder joints, a periodic function shown in Fig.5.12 is applied to the elbow joint. Such function completes a half cycle faster than a simple sine speed function so that the elbow joint can move to its desired position faster than the scapula and shoulder joints.

The fitness is calculated after 2.0 s of the simulation based on the following cost functions:

1. Standing height of the quadruped configuration. The fitness is set to unity when the height is equal the desired standing height.

2. Speed of the quadruped configuration. The fitness is set to unity when the moving speed of the configuration is same as the reference speed. The reference speed is considered as 0.15m/s.

Table 6.2 shows the GA initialization to search for optimal parameters for walking gait. The time taken for the simulation is approximate 17 hours.

Table 6.2: GA parameters to optimize the quadruped configuration walking phase

Population	100
Chromosome Length	10
Exit Criteria	3000 evaluations
Crossover rate	85%
Mutation rate	5%
Elitism	5%
Walking Speed	0.083m/s

6.3 Walking phase: Results for quadruped configuration

The GA simulation is stopped after 30 iterations. Thirty iterations are chosen instead of 50, because the configuration has become more complex and the calculation of fitness consumes more computational effort. Thus by reducing the simulation iteration, the total time consumed is reduced. The average fitness of the entire population is obtained as 0.761484 and it is observed that every population converges into the same set of parameter values. This set of parameter values is chosen to validate the objectives. The fitness trend is as shown in Fig. 6.11 and Fig. 6.12 shows the sequence of the trot motion.

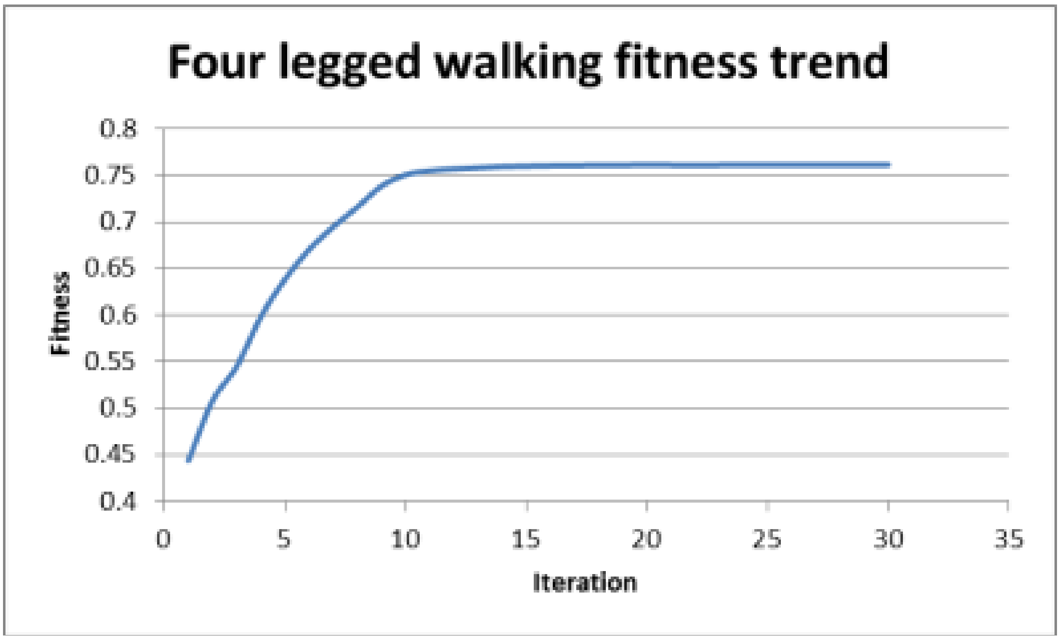


Figure 6.11: Fitness trend for quadruped configuration walking

The torque function, angular velocity and power consumption for the quadruped configuration are shown in Fig.6.13 for the left forelimb and Fig.6.14 for the left hind limb. The response of the left front limb (Fig 6.13) is the same as compared to the double front limb configuration Case 4 (Section 5.6). The highest power consumption occurs on both scapula-body and elbow pitch joints. The high power consumption on the elbow joint is in reaction to the forward thrust produced by the hind limbs. In the hind limb, the elbow joint has the highest power consumption. This is due to the fact that the hind limbs provide driving force during walking. When a limb changes from swing phase into support phase, the friction force that occurs between the foot and ground generates a forward thrust for the quadruped. The reaction force is applied on the elbow joint.

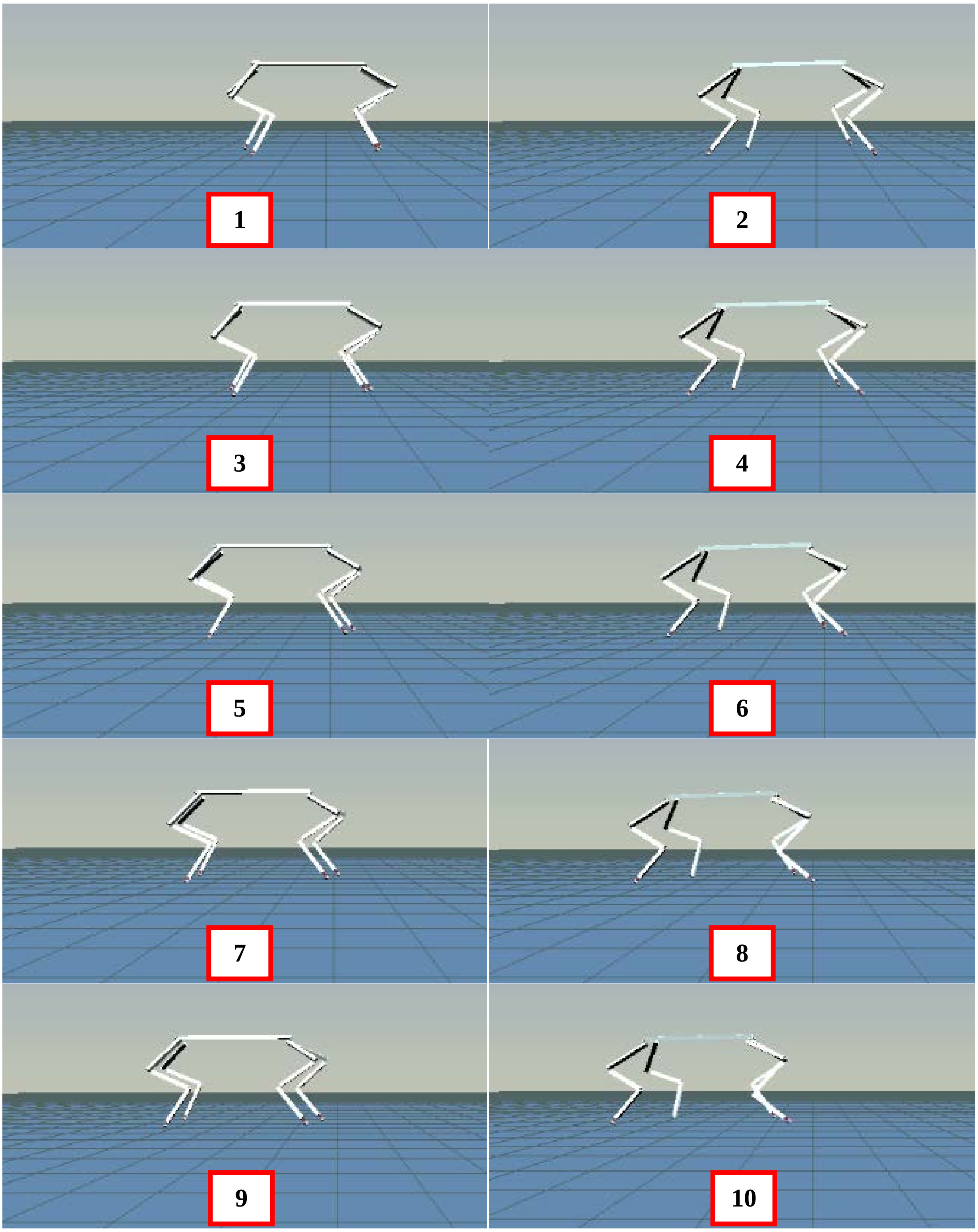
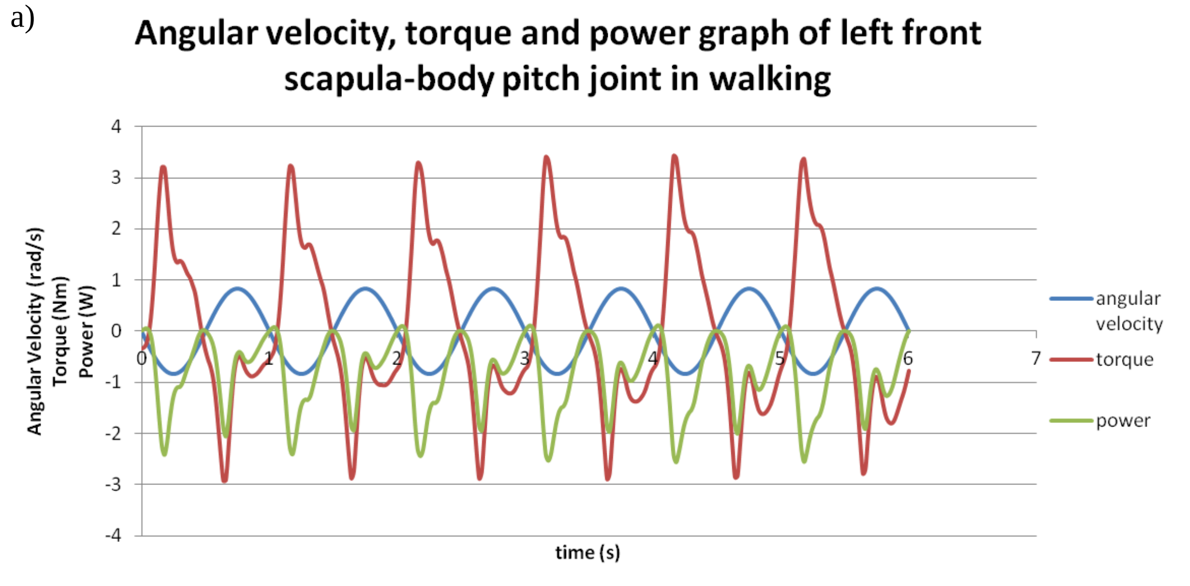


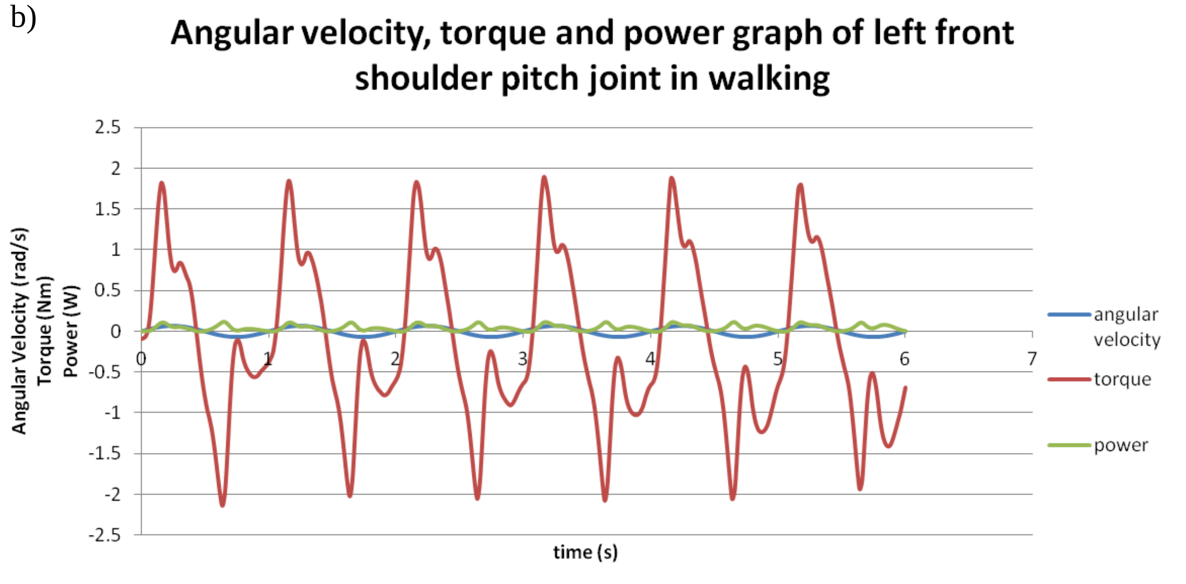
Figure 6.12: Motion sequences of the quadruped configuration trotting motion

a)



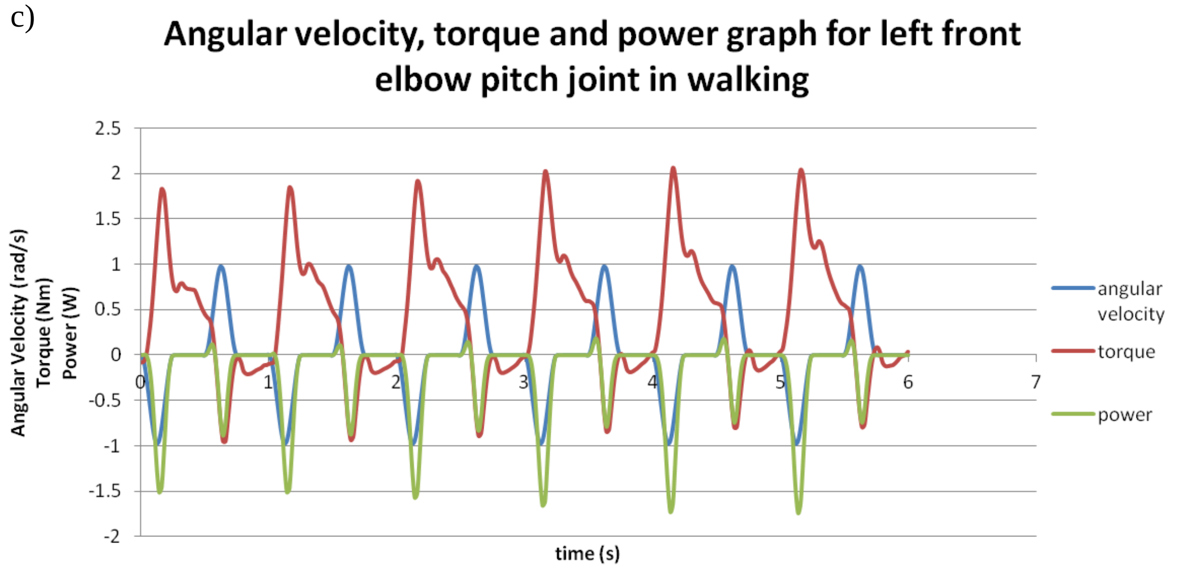
Angular velocity, torque and power graph of left front scapula-body pitch joint in walking
Angular Velocity (rad/s)
Torque (Nm)
Power (W)
4
3
2
1
0
-1
-2
-3
-4
0
1
2
3
4
5
6
7
time (s)
angular velocity
torque
power

b)



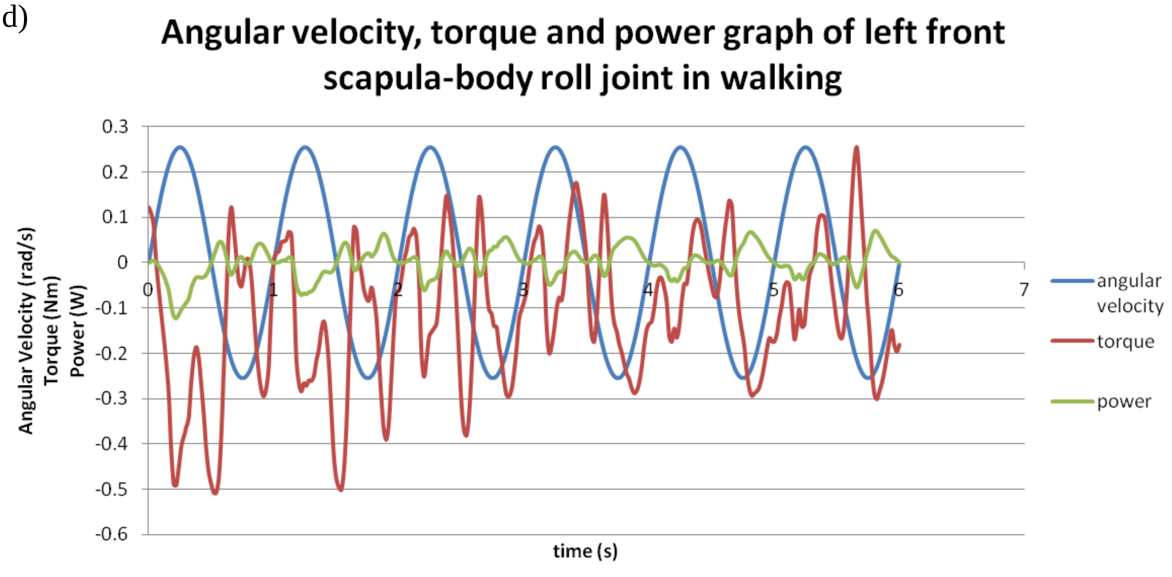
Angular velocity, torque and power graph of left front shoulder pitch joint in walking
Angular Velocity (rad/s)
Torque (Nm)
Power (W)
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2.5
0
1
2
3
4
5
6
7
time (s)
angular velocity
torque
power

c)



Angular velocity, torque and power graph for left front elbow pitch joint in walking
Angular Velocity (rad/s)
Torque (Nm)
Power (W)
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
0
1
2
3
4
5
6
7
time (s)
angular velocity
torque
power

d)



e)

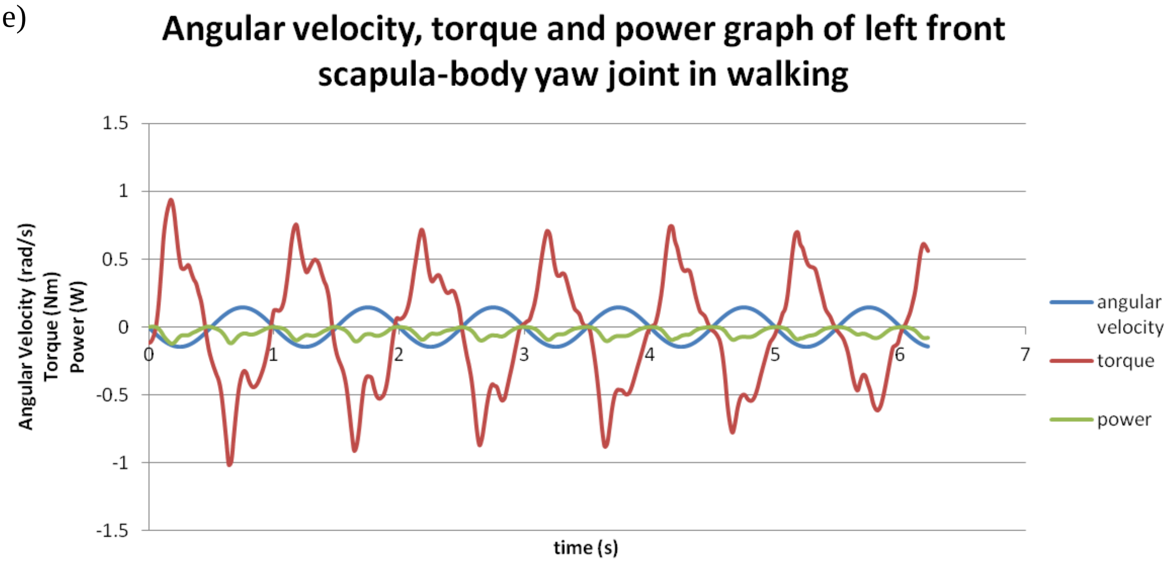
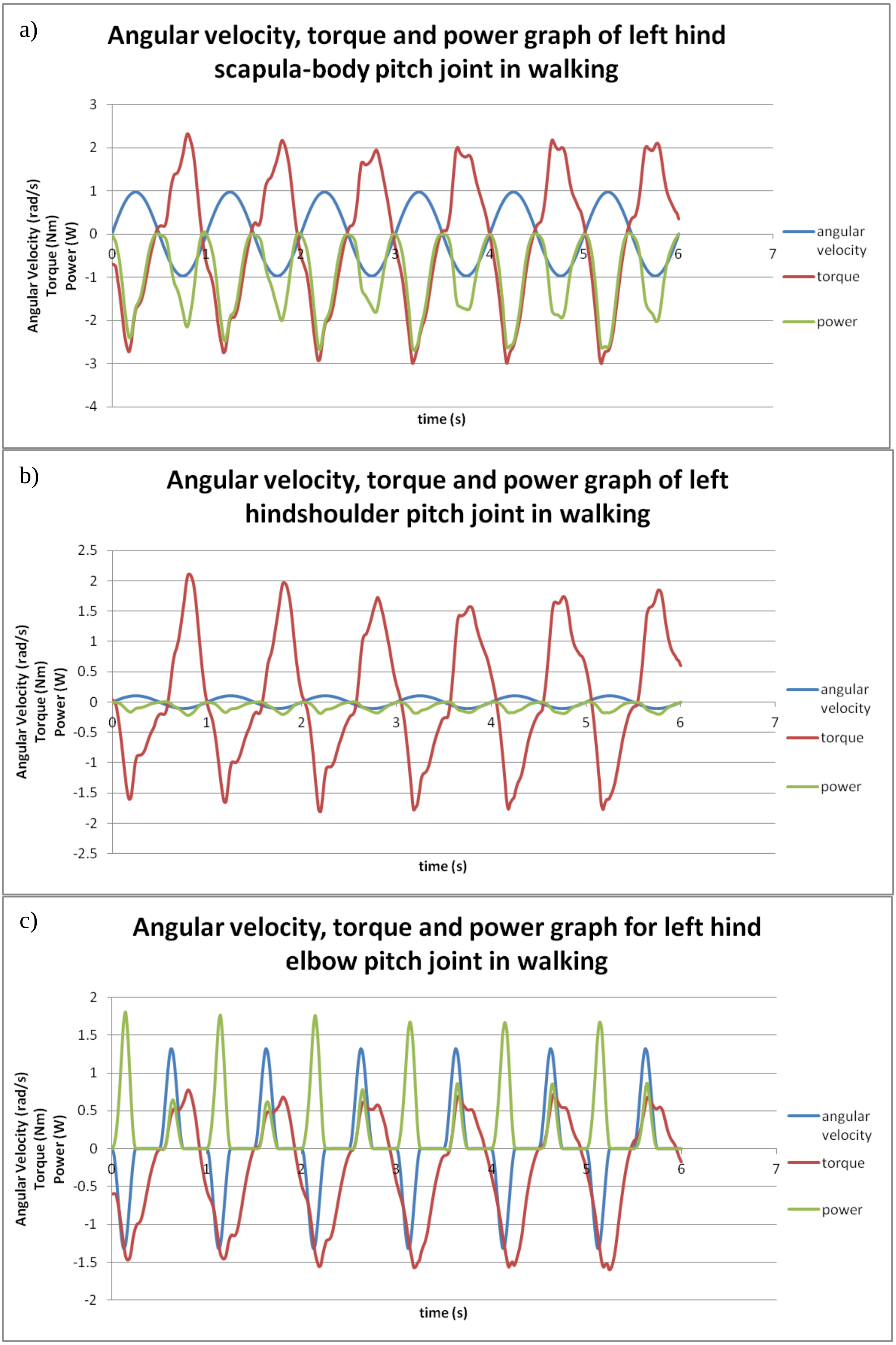
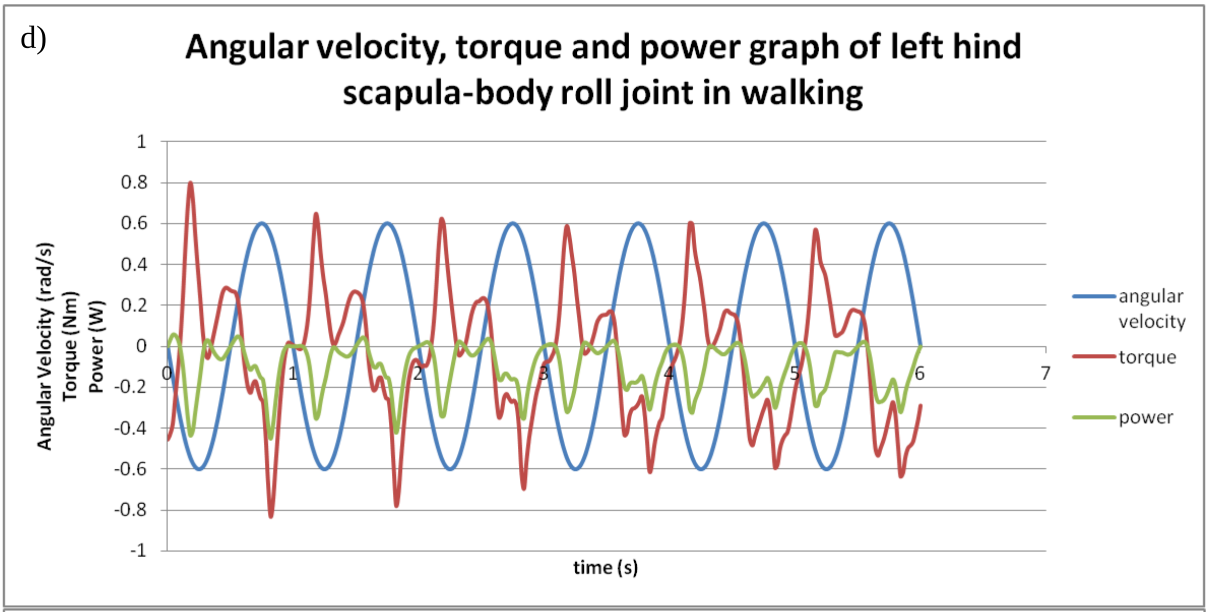


Figure 6.13: Torque function, angular velocity and power consumed for left forelimb (a) Scapula-body pitch joint (b) shoulder joint (c) elbow joint (d) scapula-body roll joint (e) scapula-body yaw joint


a)
Angular velocity, torque and power graph of left hind scapula-body pitch joint in walking
Angular Velocity (rad/s) Torque (Nm) Power (W)
3
2
1
0
-1
-2
-3
-4
0
1
2
3
4
5
6
7
time (s)
angular velocity
torque
power
b)
Angular velocity, torque and power graph of left hindshoulder pitch joint in walking
Angular Velocity (rad/s) Torque (Nm) Power (W)
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2.5
0
1
2
3
4
5
6
7
time (s)
angular velocity
torque
power
c)
Angular velocity, torque and power graph for left hind elbow pitch joint in walking
Angular Velocity (rad/s) Torque (Nm) Power (W)
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
0
1
2
3
4
5
6
7
time (s)
angular velocity
torque
power



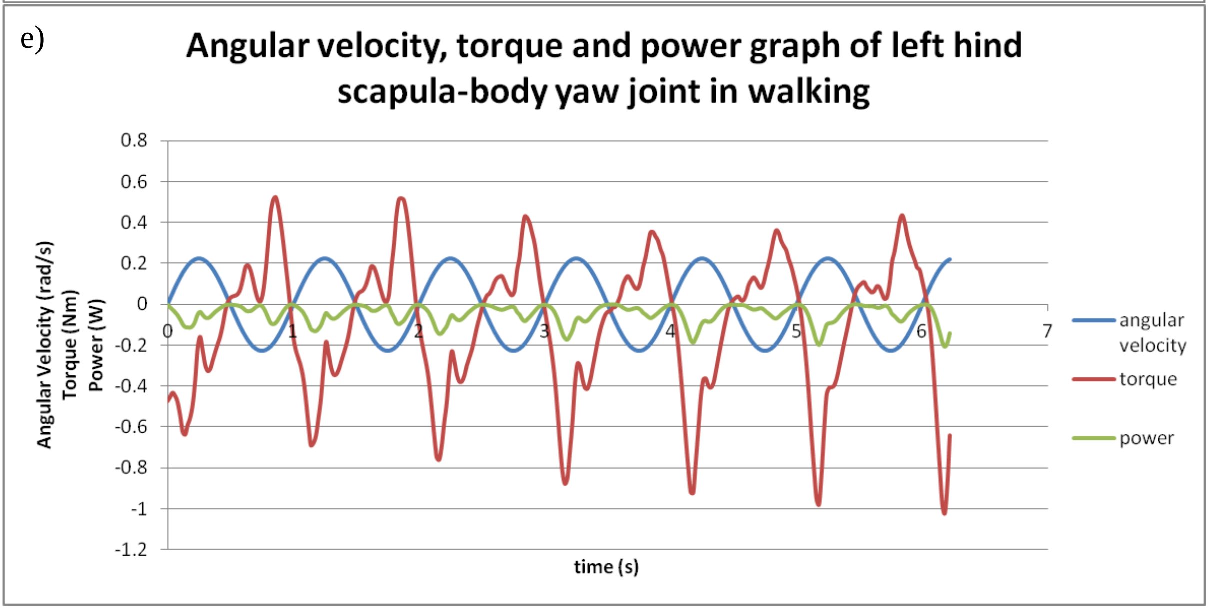


Figure 6.14: Torque function, angular velocity and power for left hind limb in walking. (a) Scapula-body pitch joint (b) shoulder joint (c) elbow joint (d) scapula-body roll joint (e) scapula-body yaw joint

The advantage of the quadruped configuration is the homogeneous configuration. The quadruped is both left-right and back-front symmetrical. With this configuration, the quadruped can walk backward by just reversing the direction of the joint motion. This saves the time required for the quadruped to make a u-turn in order to reach the target behind.

6.4 Summary

The walking of the quadruped configuration is achieved. The process of achieving the desired rest posture of the quadruped configuration is arrived at. The desired rest posture of the hind limbs has the ability to provide the forward thrust needed on both standing up and walking phases. GA is able to search for optimal joint functions for walking in the 20 DOF quadruped configuration. By using the same joint functions but by inverting the motion direction of the joints, the quadruped is able to move backward.

Chapter 7

Conclusions and Future Directions

7.1 Conclusions

In this thesis, we achieved the quadrupedal walking gait generation by utilizing Genetic Algorithm (GA) in a simulated environment. To simulate a quadruped system in a virtual environment, the dynamic of the system needs to be defined. In this dissertation, a physics engine, Chrono::Engine is introduced to take care of the dynamic behaviors of the system. A simple double pendulum problem is defined to test the eligibility of GA to work with Chrono::Engine in searching an optimal solution given a problem.

Chrono::Engine is proven to be able to work along with GA. The next part of the dissertation discussed on the simulating a single forelimb configuration. This single forelimb configuration acts as a basic building block for the quadruped configuration. The quadruped configuration is aimed to have four limbs with the same design. GA is able to search for the optimal design of the single forelimb configuration and by utilizing the appropriate fitness functions, the optimal joint functions that enable the single forelimb configuration to stand up is achieved.

A double frontal quadruped limbs configuration is simulated based on the single limb configuration. A support wheeled structure is introduced to the hind of the double frontal quadruped limbs configuration to support the double frontal limbs. There are three phases in the double frontal quadruped limbs configuration namely standing up, intermediate and walking phase. GA is utilized to search for the optimal joint functions that can achieve the objective in these three phases. The walking phase is further broken down into four different cases which include a) scapula-body pitch joint without forward thrust, b) scapula-body pitch joint with forward thrust, c) scapula-body roll, pitch and yaw joint without forward thrust, and d) scapula-body roll, pitch and yaw joint with forward thrust.

Finally, the quadruped configuration is discussed. The hind limbs are introduced into the configuration and replace the support wheel structure from the double frontal quadruped limbs configuration. The hindlimbs provide the forward thrust for the movement while the front limbs provide steering. In order to ensure the quadruped configuration to walk, the elbow joint is made to complete its motion cycle faster than other joints. With such setting, a forward thrust will occur when the foot strike the ground. A set of optimal joint functions which enable the quadruped configuration to walk is achieved.

7.2 Future Directions

The concept of utilizing GA for quadruped gait generation in a virtual world with physics laws is useful and able to achieve the objectives. Some possible works that can be done in the future include:

- Running the simulation with design parameters (e.g. link mass, link length etc) as parameters to be optimized instead of a fixed value.
- Running the simulation by setting energy optimization as an objective to achieve by GA. Thus the optimized parameter can generate a walking gait that is energy optimized.
- Introducing a controller for joint function to minimize the effect of disturbance on joint rotation. Currently, the joint functions that applied on the joints, which enable the configuration to walk, are set points. Overtime, errors accumulate leading to

deviation of joint positions (Section 5.3 Fig. 5.6). A closed loop control that eliminates the errors can maintain proper walking posture for a long period.

- Building an actual quadruped robot which is based on the simulation results and applies the joint functions to the robot. The simulation can easily reproduce to fit in different robots by changing the definition for the simulated model including the mass, links length, joints position and etc.
- Applying the same method to solve problem for walking of configuration with more limbs including hexaped and octoped configuration. Different configuration will have different resting posture. Therefore, different GA fitness function is needed to search for the optimal joint functions. By changing the GA fitness function accordingly, the walking gait of the new configuration can be achieved.

References

[1] Cheetah vs horse (2012). Retrieved 10 October 2012, from cheetahspot.com: http://www.cheetahspot.com/horseche.php

[2] A. E. Minetti, L. P. Ardigo, E. Reinach& F. Saibene, “The relationship between mechanical work and energy expenditure of locomotion in horses”, *The Journal of Experimental Biology*, vol. 202, pp. 2329–2338, 1999.

[3] 4-Legged Speed (2004). Retrieved 11 October 2012, from OricomTechnology :http://www.oricomtech.com/projects/leg-sped.htm

[4] J. W. Chung, I. W. Park & J. H. Oh, “On the Design and Development of a Quadruped Robot Platform”, *Advanced Robotics*, vol. 24, pp. 277-298, 2010.

[5] Quadruped climbing (2004). Retrieved 11 October 2012, from Oricom Technology :http://www.oricomtech.com/projects/leg-clmb.htm

[6] R. M. Alexander, Principles of Animal locomotion, Princeton University Press, 2003.

[7] D. Golubovic & H. Hu, “Evolving locomotion gaits for quadruped walking robots”, *Industrial Robot: An International Journal*, vol. 32, iss 3, pp. 259-267, 2005.

[8] M. Raibert, K. Blankespoor, G. Nelson & R.Playter, “BigDog, the Rough-Terrain Quaduped Robot”, *Proceedings of the 17th World Congress The International Federation of Automatic Control*, pp. 10822-10855, 2008.

[9] M. Raibert, “Dynamic legged robots for rough terrain”, *2010 10th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pp. 1, 2010.

[10] D.Wooden, M.Malchano, K.Blankespoor, A.Howardy, A.A.Rizzi&M. Raibert, “Autonomous navigation for BigDog”, *2010 IEEE International Conference on Robotics and Automation (ICRA 2010)*, p 4736-41, 2010.

[11] R. Playter, M. Buehler &M. Raibert.“Bigdog”, *In Proc. SPIE*, volume 6230, 2006.

[12] LittleDog - The Legged Locomotion Learning Robot (2012). Retrieved 11 October 2012, from Boston Dynamics :http://www.bostondynamics.com/robot_littledog.html

[13] J. Z. Kolter& A. Y Ng, “The Stanford LittleDog: A learning and rapid replanning approach to quadruped locomotion”, *The International Journal of Robotics Research*, vol. 30, no. 2,pp. 150- 174, 2011.

[14] H. Kimura, Y. Fukuoka & A. H. Cohen, “Biologically inspired adaptive walking of a quadruped robot”, *Philosophical Transactions of The Royal Society*, vol. 365, no. 1850,pp. 153-170, 2007.

[15] Y. Fukuoka, H. Kimura, Y. Hada & K.Takase, “Adaptive dynamic walking of a quadruped robot 'Tekken' on irregular terrain using a neural system model”, *IEEE International Conference on Robotics and Automation*, vol. 2, pp. 2037 – 2042, 2003.

[16] H. Kimura, Y. Fukuoka, Y. Hada& K.Takase, “Three-dimensional adaptive dynamic walking of a quadruped - rolling motion feedback to CPGs controlling pitching motion”, *IEEE International Conference on Robotics and Automation*, vol. 3, pp. 2228 – 2233, 2002.

[17] Y. Fukuoka & H. Kimura, “Dynamic locomotion of a biomorphic quadruped ‘Tekken’ robot using various gaits: walk, trot, free-gait and bound”, *Applied Bionics and Biomechanics*, vol. 6, iss.1, pp. 63- 71, 2009.

[18] Y. Fukuoka, H. Katabuchi& H. Kimura, “Dynamic Locomotion of Quadrupeds Tekken 3&4 Using Simple Navigation”, *Journal of Robotics and Mechatronics*, vol. 22, no.1, pp. 36-42, 2010.

[19] I. Markelic&K.Zahedi, “An evolved neural network for fast quadrupedal locomotion”, *International Conference on Climbing and Walking Robots and the Support Technologies for Mobile Machines*, pp.65-72, 2007.

[20] J. Zhang, M. Tomizuka, Q. Chen &C. Liu, “Dynamic Walking of AIBO with Hopf Oscillators”, Chinese Journal of Mechanical Engineering, vol. 24, no. 4, pp. 1-6, 2011.

[21] M.Mahdaviani, N. de Freitas, B. Fraser & F.Hamze, “Fast Computational Methods for Visually Guided Robots”, *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pp. 138-143, 2005.

[22] G. S. Hornby, M. Fujita, S. Takamura, T. Yamamoto, & O. Hanagata, “Evolving Robust Gaits with AIBO”, *IEEE International Conference on Robotics and Automation*, pp. 3040-3045, 2000.

[23] K. D. Budras, P. H. McCarthy, W. Frice, & R. Richter, Anatomy of the dog, Hannover : Schlütersche, 2007.

[24] P. T. Doan, H. D. Vo, H. K. Kim & S. B. Kim, “A New Approach for Development of Quadruped Robot Based on Biological Concepts”, *International Journal of Precision Engineering and manufacturing*, vol. 11, no. 4, pp. 559-568, 2010.

[25] H. D. Vo, T. P. Nguyen, S. M. Yoon, H. K. Kim& S. B. Kim “A New Approach for Designing Quadruped Robot”, *ICM2007 4th IEEE International Conference on Mechatronics*, pp. 1-5, 2007

[26] A.Spröwitz, A. Tuleu, M. Vespignani, M. Ajallooeian, E. Badri&A. J.Ijspeert, “Towards dynamic trot gait locomotion: Design, control, and experiments with Cheetah-cub, a compliant quadruped”, *The International Journal of Robotics Research*, vol. 32, num. 8, pp. 933 – 951, 2013.

[27] X. Li, J. Gao, X. Duan, Q. Huang, H. Li, H. Liu, W. Zhao, Z. Xu, Y. Liu & W. Sun, “Mechanical Design of the Legs of Hydraulically Actuated Quadruped Bionic Robot”, *International Conference on Complex Medical Engineering*, pp. 626 – 632, 2013.

[28] S. Hirose, Y. Fukuda, K. Yoneda, A. Nagakubo, H. Tsukagoshi, K. Arikawa, G. Endo, T. Doi&R. Hodoshima, "Quadruped walking robots at Tokyo Institute of Technology", *IEEE Robotics & Automation Magazine*, pp. 104-114, 2009.

[29] S. Talebi, I. Poulakakis, E. Papadopoulos & M. Buehler, "Quadruped Robot Running With a Bounding Gait", *In proceeding of: Experimental Robotics VII*, 2000.

[30] Z. Bien, M.G. Chun &H.S. Son, "An Optimal Turning Gait for a Quadruped Walking Robot", *IEEE/RSJ International Workshop on Intelligent Robots and Systems*, pp. 1511-1577, 1991.

[31] D. Pongas, M. Mistry & S. Schaal, "A Robust Quadruped Walking Gait for Traversing Rough Terrain", *IEEE International Conference on Robotics and Automation*, pp. 1474-1479, 2007.

[32] G. Antonelli, "Stability Analysis for Prioritized Closed-Loop Inverse Kinematic Algorithms for Redundant Robotic Systems", *IEEE Transactions on Robotics*, vol. 25, no. 25, pp.985-994, 2009.

[33] L. Sun,Y. Zhou, W. Chen, H. Liang & T. Mei, "Modeling and Robust Control of Quadruped Robot", *Proceedings of the 2007 International Conference on Information Acquisition*, pp.356-360, 2007.

[34] L. Yu, Y. Wang & W. Tao, "Gait Analysis and Implementation of a Simple Quadruped Robot", *International Conference on Industrial Mechatronics and Automation*, pp. 431-434, 2010.

[35] J. Z. Kolter& A. Y. Ng, "The Stanford LittleDog: A learning and rapid replanning approach to quadruped locomotion", *The International Journal of Robotics Reseach*, vol. 30, no.2, pp. 150-174, 2011.

[36] I. Poulakakis, E. Papadopoulos & M. Buehler, "On the Stability of the Passive Dynamics of Quadrupedal Running with a Bounding Gait", The International Journal of Robotics Research, vol. 25, no. 7, pp. 669 – 687, 2006.

[37] F. Iida & R. Pfeifer, ""Cheap" Rapid Locomotion of a Quadruped Robot: Self-Stabilization of Bounding Gait", *Intelligent Autonomous Systems*, vol. 8. 2004.

[38] A. A. Shabana, "Dynamics of Multibody Systems", Cambridge University Press, 2005.

[39] R. E. Roberson, R. Schwertassek, "Dynamics of Multibody Systems", Springer, 1988.

[40] R. V. Schwerin, "Multibody System Simulation: Numerical Methods, Algorithm, and Software", Springer, 1999.

[41] M. H. Miller & K. Hirsch, "Multibody system dynamics and mechatronics", *76th Annual GAMM Conference*, vol. 86, no. 2, pp. 87-109, 2006.

[42] L. Zlajpah, "Simulation in robotics", *Mathematics and Computers in Simulation*, vol. 9, no. 4, pp. 879-897, 2008.

[43] S. Dubowsky & E. Papadopoulos, "The Kinematics, Dynamics, and Control of Free-Flying and Free-Floating Space Robotic Systems", *IEEE Transactions on Robotics*, vol. 9, no. 5, pp. 531 - 543, 1993.

[44] D. P. Aguirre & J. J. M. Molina, "Mutibody Modelization of a Biped Robot", *2012 16th IEEE Mediterranean Electrotechnical Conference (MELECON)*, pp. 985 - 988, 2012.

[45] Z. Nagy, R. I. Leine & D. R. Frutiger, "Modeling the Motion of Microrobots on Surfaces Using Nonsmooth Multibody Dynamics", *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1058 - 1068, 2012.

[46] Z. G. Zhang, Y. Fukuoka & H. Kimura, "Adaptive Running of a Quadruped Robot Using Delayed Feedback Control", *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pp. 3739 - 3744, 2005.

[47] Z. G. Zhang, T. Masuda, H. Kimura & K. Takase, "Towards Realization of Adaptive Running of a Quadruped Robot Using Delayed Feedback Control", *2007 IEEE International Conference on Robotics and Automation*, pp. 4325 - 4330, 2007.

[48] Z. G. Zhang & H. Kimura, "Rush: a simple and autonomous quadruped running robot", *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 223, no. 3, pp. 323 - 336, 2008.

[49] D.P. Krasny, & D.E. Orin, "Generating high-speed dynamic running gaits in a quadruped robot using an evolutionary search", IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, vol. 34, iss. 4, pp. 1685 - 1696, 2004.

[50] R. M. Murray, Z. Li & S. S. Sastry, "A Mathematical Introduction to Robotic Manipulation", CRC Press, 1994.

[51] S. Carpin, M. Lewis, J. Wang & S. Balakirsky, "USARSim: a robot simulator for research and education", *IEEE International Conference on Robotics and Automation,* pp. 1400 – 1405, 2007.

[52] M. Zaratti, M. Fratarcangeli & L. Iocchi "A 3D Simulator of Multiple Legged Robots Based on USARSim", Springer Berlin Heidelberg, 2007.

[53] Microsoft Robotic Studio (2012). Retrieved 08 November 2013, from Microsoft Robotics: http://www.microsoft.com/robotics/

[54] J. Jackson, "Microsoft robotics studio: A technical introduction", *IEEE Robotics & Automation Magazine*, vol. 14, no. 4, pp. 82 – 87, 2007.

[55] M. Anitescu & A. Tasora, "An iterative approach for cone complementarity problems for nonsmooth dynamics", *Computational Optimization and Applications,* vol. 27, pp. 207 - 235, 2008.

[56] M. Anitescu & A. Tasora, “A matrix-free cone complementarity approach for solving large-scale, nonsmooth, rigid body dynamics”, *Computer Methods in Applied Mechanics and Engineering*, vol. 200, pp. 439 - 453, 2011.

[57] M. Anitescu & A. Tasora, “A convex complementarity approach for simulating large granular flows”, Journal of Computational and Nonlinear Dynamics, vol. 5, pp. 1 - 10, 2010.

[58] M. D. Vose, The simple Genetic algorithm : foundations and theory, The MIT Press, 1999.

[59] D. Ginat, Genetic algorithm : a function optimizer, University of Maryland, 1988.

[60] G. S. Hornby, S. Takamura, T. Yamamoto & M. Fujita, “Autonomouse Evolution of Dynamic Gaits with Two Quadruped Robots”, *IEEE Transactions on Robotics*, vol. 21, no. 2, pp. 402-410.

[61] M .Mitchell, “An Introduction to Genetic Algorithm”, The MIT Press, 1998.

[62] P. Vadakkepat, S. J. Tan & C. Y. Hong, “Model Validation for a Pair of Biologically Abstracted Quadruped Forelimb through the Design of Experiment (DOE) Technique”, *Transactions of the Institute of Measurement and Control*, vol. 34, no. 2/3, pp 213 – 235, 2012.

[63] A Study in Canine Anatomy. Retrieved 20 September 2012, from website : http://tailweaver2.faithweb.com/inuyasha/anatomy/inuanatomy.html.

[64] R. M. Alexander, “The Gaits of Bipedal and Quadrupedal Animals”, *The International Journal of Robotics Research*, vol. 3, no. 2, pp. 49-59, 1983.

[65] T. Pfau, E. Hinto, C.Whitehead, A. Wiktorowicz-Conroy & J. R. Hutchinson, “Temporal gait parameters in the alpaca and the evolution of pacing and trotting locomotion in the Camelidae”, *Journal of Zoology*, vol. 283, pp. 193–202, 2011.