



Dissertation

Walking Control of a Humanoid Robot

ausgeführt zum Zwecke der Erlangung des akademischen Grades eines
Doktors der technischen Wissenschaften unter der Leitung von

Em.o.Univ.Prof. Dipl.-Ing. Dr.techn. Dr.h.c.mult. Peter Kopacek

E325/A6

Institut für Mechanik und Mechatronik

eingereicht an der Technischen Universität Wien

Fakultät für Maschinenwesen und Betriebswissenschaften

Von

Mohsen Mohamadi Daniali

Matrikelnummer: 1128821

Wien, November 2013

Kurzfassung

Die Arbeit beschäftigt sich mit der Steuerung und Regelung eines am Institut in Entwicklung befindlichen Roboters. Der derzeit vorhandene Unterkörper besitzt 12 Freiheitsgrade.

Das dafür neu entwickelte Steuerungs- und Regelungssystem basiert auf einer dezentralen Struktur. Bei dieser wird jeder Freiheitsgrad unabhängig geregelt, so dass es möglich ist sowohl die vorgegebenen Trajektorien zu realisieren als auch Störungen zu kompensieren. Die Regelungsstruktur ist kaskadenförmig aufgebaut, wobei für die inneren Kreise digitale zeitdiskrete PI Regler und für die äußeren Kreise ebensolche PI Regler eingesetzt werden.

Die dreidimensionalen Schrittrajektorien wurden vom menschlichen Gang abgeleitet. Bei dem vorgeschlagenen Verfahren können auf einfache Art und Weise durch Änderung der Parameter diese geändert und optimiert werden. Für die Bestimmung der Roboter-Koordinaten fand die Methode der inversen Kinematik Verwendung und wurde in C++ implementiert.

Abschließend wurden an der bestehenden Hardware durch eine Vielzahl von Versuchen diese Methode getestet und die Ergebnisse mit den berechneten verglichen. Weiters war es dadurch möglich, die optimalen Parameter der Schrittrajektorien zu bestimmen.

Abstract

In this dissertation, the design and implementation of a control system for stable walking of a biped robot is presented. The biped robot used as a test bed for walking experiments is called Archie that was designed and constructed in Vienna University of Technology. We also improved the robot hardware for walking by adding a joint to each ankle. The improved biped robot has 12 degrees of freedom totally, i.e. each leg has 6 joints.

The proposed control system is based on decentralized control method. In this strategy, each joint's rotation angle is controlled independently and the dynamic effects of manipulator links to each other are considered as disturbances. Therefore the independent joint controller is designed such that not only the output tracks the reference trajectory but also reject the disturbance. Since harmonic drive with high gear ratio is used in each joint to transmit the torque from the motor to the link, the independent joint controller can reject the effect of the nonlinear disturbance by utilizing cascaded control system. Thus the proposed independent joint controller consists of a inner velocity loop which is cascaded with a outer position loop. For the inner loop a PI controller is used while for the outer loop P controller is employed. The controller gains are tuned based on the step response for each joint motor.

In order to imitate the human walking, the three dimensional trajectories of the feet and the torso are developed. For constructing the trajectories, first, motion constraints during walking are derived based on the analysis of human walking pattern. Then cubic spline interpolation is used to find the smooth trajectories for the feet and the torso in both single and double-support phase. The trajectories generated by walking pattern generator can be redesigned easily by changing the walking parameters.

The closed-loop solution of inverse kinematics is developed to convert the desired trajectories from the operational space to the joint space. The closed-loop solution of the inverse kinematics is superior with respect to the iterative solution due to the less computation time. In addition, a kinematic simulation is developed to illustrate the robot configuration before implementation.

For implementation, a C++ program is developed to generate the reference joint angle trajectories. This program convert these trajectories to digital number and put them in Position-Time (PT) table. These data are sent to each joint controller under CAN message format. In this manner, the controller actuate the motor by generating proper voltage to synchronize the motion of the robot joints.

In order to realize the biped walking in the sense of static stability, the robot's center of gravity should be located on the above of the support foot area. Therefore, many experiments have been done to find the optimal values of walking parameters. Finally, stable walking realized for the biped robot with speed up to 0.076 km/h .

Acknowledgment

First of all, I would like to express my sincere gratitude to my advisor, Professor Peter Kopacek, for supporting and guiding me during this project. I would like also to thank Mr. Peter Unterkreuter for helping us to construct and improve the robot.

My sincere thanks also goes to my friend, Siavash Dezfouli, for his great cooperation during this project.

I would like to thank my parents, brother and sister who have given me infinite love and great support throughout my life. Undoubtedly, without their support I could not pass this stage of my life.

Last but not the least, my special thanks goes to my wife, Pantea Pezeshkpour, for giving me love and support and for her great patience at all time.

Table of Contents

Kurzfassung	II
Abstract	III
Acknowledgment.....	IV
Table of Contents	V
List of Figures.....	VII
List of Tables.....	XII
Chapter 1 Introduction.....	1
1.1 Overview of walking robots	2
1.2 Problem statement	7
Chapter 2 Archie description.....	9
2.1 Mechanical structure	9
2.2 Electric motors	9
2.2.1 Brushless DC motor	9
2.2.2 DC motor	11
2.3 Transmission.....	12
2.3.1 Harmonic drive	13
2.3.2 planetary gear	13
2.4 Encoder.....	14
2.5 Electric drive	14
Chapter 3 Forward and inverse kinematics	16
3.1 Forward kinematics	17
3.1.1 Left leg	17
3.1.2 Right leg.....	21
3.2 Inverse kinematics.....	23
3.2.1 Left leg	23
3.2.2 Right leg.....	29
3.3 Inverse kinematics validation.....	34
Chapter 4 Human-like walking pattern	39
4.1 Gait analysis.....	39
4.2 Planning walking pattern.....	42
4.2.1 Gait cycle	42

4.2.2 Foot trajectories	44
4.2.3 Torso trajectories	48
4.3 Walking trajectories	54
4.4 Kinematic simulation.....	69
Chapter 5 Control design.....	77
5.1 Decentralized control	79
5.2 Velocity versus torque control	82
5.3 Independent joint control	86
5.4 Biped walking control system	89
Chapter 6 Implementation and results	92
6.1 Implementation.....	92
6.1.1 PT table:.....	93
6.1.2 Controller tuning:	94
6.2 Walking realization	101
Chapter 7 Conclusions and future works	117
References.....	120

List of Figures

Fig. 1.1 The biped robots WL-5 and WL-10RD	3
Fig. 1.2 The biped robot developed at MIT Leg Lab called Spring Flamingo	4
Fig. 1.3 The biped robot Johnnie.....	4
Fig. 1.4 Developed prototypes of ASIMO (www.asimo.honda.com , 2013).....	5
Fig. 1.5 Photos of HRP-1, HRP-3, HRP-4C and HRP-4	6
Fig. 1.6 WABIAN 2 photo.....	6
Fig. 2.1 The existing biped robot, Archie.....	10
Fig. 2.2 Brushless DC motor components (www.maxonmotor.de , 2013)	10
Fig. 2.3 Block commutation of brushless DC motor (www.maxonmotor.de , 2013).....	11
Fig. 2.4 DC motor components (www.faulhaber.com , 2013).....	12
Fig. 2.5 Planetary gear components (www.faulhaber.com , 2013)	13
Fig. 2.6 Encoder chip and its housing to connect to the link	14
Fig. 2.7 The electric drive (www.elmocom.com , 2013).....	15
Fig. 2.8 Electric drive Architecture (www.elmocom.com , 2013).....	15
Fig. 3.1 Schematic figure of the left leg and its attached coordinates and dimensions.....	18
Fig. 3.2 Schematic figure of the right leg and its attached coordinates and dimensions.....	22
Fig. 3.3 Vector relation between frames {0} ,{5} and {6} for the left leg	25
Fig. 3.4 Position of the point c (center of frame {3}, {4} and {5}) when first joint rotates in the left leg.....	26
Fig. 3.5 Vector relation between frames {0}, {5} and {6} for the right leg.....	30
Fig. 3.6 Position of the point c (center of frame {3}, {4} and {5}) when first joint rotates in the right leg.....	31
Fig. 3.7 Schematic diagram for verification of inverse kinematics validation	35
Fig. 3.8 Comparison between actual and calculated joint angle of the right frontal ankle.....	35
Fig. 3.9 Comparison between actual and calculated joint angle of the right lateral ankle	35
Fig. 3.10 Comparison between actual and calculated joint angle of the right knee	36
Fig. 3.11 Comparison between actual and calculated joint angle of the right frontal hip	36
Fig. 3.12 Comparison between actual and calculated joint angle of the right lateral hip.....	36
Fig. 3.13 Comparison between actual and calculated joint angle of the right transversal hip.....	36
Fig. 3.14 Comparison between actual and calculated joint angle of the left frontal ankle.....	37

Fig. 3.15 Comparison between actual and calculated joint angle of the left lateral ankle	37
Fig. 3.16 Comparison between actual and calculated joint angle of the left knee	37
Fig. 3.17 Comparison between actual and calculated joint angle of the left frontal hip	38
Fig. 3.18 Comparison between actual and calculated joint angle of the left lateral hip	38
Fig. 3.19 Comparison between actual and calculated joint angle of the left transversal hip .	38
Fig. 4.1 Standard primary plane for the human body movement	39
Fig. 4.2 Cyclic pattern of human gait describing eight sequence of walking (Vaughan et al., 1992).....	40
Fig. 4.3 Eight main events of gait cycle (Vaughan et al., 1992).....	41
Fig. 4.4 Distance parameters based on footprints (Vaughan et al., 1992)	41
Fig. 4.5 Model of biped robot in sagittal plane (Huang et al., 2001)	43
Fig. 4.6 Walking cycle (Huang et al., 2001)	43
Fig. 4.7 Walking parameters in the sagittal plane (Huang et al., 2001).....	44
Fig. 4.8 Walking parameters in the transversal plane (top view)	45
Fig. 4.9 Position and orientation of the frames attached to the left foot and the torso with respect to the base frame	52
Fig. 4.10 Position and orientation of the frames attached to the right foot and the torso with respect to the base frame	54
Fig. 4.11 Trajectories of θ_a for right foot (blue) and left foot (red).....	56
Fig. 4.12 Right foot (blue), left foot(red) and torso (green) trajectories in x direction	56
Fig. 4.13 Right foot (blue), left foot(red) and torso (green) trajectories in x direction	57
Fig. 4.14 Modified trajectory of torso in comparison with Fig. 4.13	58
Fig. 4.15 Right (blue) and left (red) foot trajectory for upward movement	58
Fig. 4.16 Trajectory of upward movement of torso	59
Fig. 4.17 Trajectory of left foot (red), right foot (blue) and torso (green) in y direction.....	60
Fig. 4.18 Traversed trajectory of the right foot (blue) and the left foot (red) in the z-x plane	61
Fig. 4.19 Traversed trajectory of the torso in the z-x plane during the double-support phase (red) and during the single-support phase (green).....	62
Fig. 4.20 Traversed trajectory of the torso in the z-x plane during the double-support phase (red) and during the single-support phase (green).....	63
Fig. 4.21 Position trajectory of the right foot (blue) and the left foot (red) and the torso (green) in the y-x plane	64

Fig. 4.22 Position trajectory of the right foot (blue) and the left foot (red) and the torso (green-black-yellow) in the z-y plane	65
Fig. 4.23 Position trajectory of the right foot (blue), left foot (red) and the torso (green) in three dimension	65
Fig. 4.24 Frontal ankle joint angles for right leg (blue) and left leg (red)	66
Fig. 4.25 Lateral ankle joint angles for right leg (blue) and left leg (red).....	67
Fig. 4.26 Knee joint angles for right leg (blue) and left leg (red)	67
Fig. 4.27 Lateral hip joint angles for right leg (blue) and left leg (red)	68
Fig. 4.28 Frontal hip joint angles for right leg (blue) and left leg (red).....	68
Fig. 4.29 Transversal hip joint angles for right leg (blue) and left leg (red).....	69
Fig. 4.30 Schematic diagram of the proposed program for kinematic simulation of the biped walking.....	70
Fig. 4.31 Robot simulation at time $t = 0$	70
Fig. 4.32 Robot simulation at time $t = Td/2$	71
Fig. 4.33 Robot simulation at the time $t = Td$	71
Fig. 4.34 Robot simulation at the time $t = Tmax = (Tc + Td)/2$	72
Fig. 4.35 Robot simulation at the time $t = Tc$	72
Fig. 4.36 Robot simulation at the time $t = Tc + (Td/2)$	73
Fig. 4.37 Robot simulation at the time $t = Tc + Td$	73
Fig. 4.38 Robot simulation at the time $t = Tc + (Tc + Td)/2$	74
Fig. 4.39 Robot simulation at the time $t = 2 Tc$	74
Fig. 4.40 The right leg simulation during one stride walking	75
Fig. 4.41 The left leg simulation during one stride walking	76
Fig. 5.1 General schematic diagram of joint space control scheme	77
Fig. 5.2 General schematic diagram of operational space control scheme.....	78
Fig. 5.3 Block diagram for equation of motion of manipulator and drive(Siciliano et al., 2009)	81
Fig. 5.4 General torque-speed curves of DC motor	83
Fig. 5.5 Block diagram of DC motor.....	83
Fig. 5.6 Block diagram of the manipulator and motor system as voltage-controlled system(Siciliano et al., 2009).....	84
Fig. 5.7 General block diagram of feedback control system for SISO system.....	86

Fig. 5.8 General block diagram of cascaded control method for independent joint control ..	87
Fig. 5.9 Block diagram of feedforward compensation added to position, velocity and acceleration feedback controller	88
Fig. 5.10 Overall control system of the biped robot	90
Fig. 5.11 Block diagram of independent joint control for the <i>ith</i> joint	91
Fig. 5.12 Block diagram of the motor and linear part of manipulat	91
Fig. 6.1 Control program algorithm.....	93
Fig. 6.2 PT motion decisions flowchart	95
Fig. 6.3 Step response for first selected <i>KP</i> and <i>KI</i>	97
Fig. 6.4 Velocity step response for different values of <i>KP</i> (<i>V.sec/rad</i>) and <i>KI</i> (<i>V/rad</i>) ...	97
Fig. 6.5 Step response of velocity closed loop when <i>KP</i> = 30 <i>V.sec/rad</i> and <i>KI</i> (<i>V/rad</i>) is varying	98
Fig. 6.6 Best step response for optimal values of <i>KP</i> and <i>KI</i>	100
Fig. 6.9 Position step response of the cascaded close loop system.....	100
Fig. 6.8 Configurations of the robot in z-x plane at times $t = 0, T_d, T_c$	102
Fig. 6.9 Configurations of the robot in z-y plane at times $t = 0, T_d$	102
Fig. 6.10 Experimental tests to find the optimal value of <i>wm</i>	103
Fig. 6.11 Right foot (blue), left foot(red) and torso (green) trajectories in x direction	106
Fig. 6.12 Right (blue) and left (red) foot trajectory for upward movement z_a	106
Fig. 6.13 Trajectory of upward movement of torso z_h	106
Fig. 6.14 Trajectory of left foot (red), right foot (blue) and torso (green) in y direction.....	107
Fig. 6.15 Traversed trajectory of the right foot (blue) and the left foot (red) in the z-x plane	107
Fig. 6.16 Traversed trajectory of the torso in the z-x plane during the double-support phase (red) and during the single-support phase (green).....	108
Fig. 6.17 Position trajectory of the right foot (blue) and the left foot (red) and the torso (green) in the y-x plane	108
Fig. 6.18 Position trajectory of the right foot (blue) and the left foot (red) and the torso (yellow: first double-support, black: second double-support, green: both two single-support phases) in the z-y plane.....	109
Fig. 6.19 Position trajectory of the right foot (blue) and the left foot (red) and the torso (yellow: first double-support, black: second double-support, green: both two single-support phases) in the z-x plane	109

Fig. 6.20 Frontal ankle joint angles for the right leg (blue) and the left leg (red).....	111
Fig. 6.21 Lateral ankle joint angles for the right leg (blue) and the left leg (red)	111
Fig. 6.22 Knee joint angles for the right leg (blue) and the left leg (red).....	111
Fig. 6.23 Lateral hip joint angles for the right leg (blue) and the left leg (red).....	112
Fig. 6.24 Frontal hip joint angles for the right leg (blue) and the left leg (red)	112
Fig. 6.25 Transversal hip joint angles for the right leg (blue) and the left leg (red)	112
Fig. 6.26 Archie configurations in z-x plane according to the resulted joint angles for optimal walking parameters (all dimensions are <i>cm</i>).....	113
Fig. 6.27 Archie configurations in z-y plane according to the resulted joint angles for optimal walking parameters (all dimensions are <i>cm</i>).....	114
Fig. 6.28 Complete cycle of walking depicted at nine important specified time.....	116

List of Tables

Table 2.1 Brushless DC motor characteristics	11
Table 2.2 DC motor characteristics (<i>www.faulhaber.com</i> , 2013).....	12
Table 3.1 D-H parameters for the left leg	20
Table 3.2 D-H parameters for the right leg	21

Chapter 1 Introduction

There are different types of locomotion (the ability of body movement from one place to another) in nature such as flying, swimming and walking. In each of these locomotion, the body is manipulated with respect to the environment. These different locomotion are optimal for various application in diverse environments. By inspiration from nature, human beings have also created similar objects such as airplane, ship and car for movement on the air, water and ground, respectively.

"In the case of environments with discontinuous ground support, such as a rocky slope, a flight of stairs, or the rungs of a ladder, it is arguable that the most appropriate and versatile means for locomotion is legs. Legs enable the avoidance of support discontinuities in the environment by stepping over them. Moreover, legs are an obvious choice for locomotion in environments designed for human walking, running, and climbing." (Westervelt et al., 2007).

On the other hand the fact that the intelligent robots are attractive for people leads that industry has begun building and selling robots for personal and entertainment purpose in the last decades. An important, but so far rather expensive market segment is being filled by humanoid robots: robots that are designed to imitate human being behavior (Duindam & Stramigioli, 2006).

Although the traditional mobile robots that their locomotion is based on rolling wheels are more easier to control and highly efficient for moving on hard terrain, humanoid robots with biped walking locomotion are useful in rocky or soft terrain and it can be efficient as well. This is due to the fact that humans have very articulate legs that can cross high obstacle, whereas the wheels on cars cannot. Therefore using a design that has been under development for millions of years is more easier than reinvent the wheel.

In fact, "The motivation for studying bipedal robots in particular arises from diverse sociological and commercial interests, ranging from the desire to replace humans in hazardous occupations (de-mining, nuclear power plant inspection, military interventions, etc.), to the restoration of motion in the disabled (dynamically controlled lower-limb prostheses, rehabilitation robotics, and functional neural stimulation)" (Westervelt et al., 2007).

Furthermore, humanoid robots are designed to help people especially elderly for their daily works, e.g. work inside a house. In this manner, since the humanoid robot are shaped and

articulated like humans, they have ability to move and walk up the stairs and grab objects from different part of a house.

The most important and interesting aspect of human being that humanoid robot should imitate is its locomotion technique: biped walking. "A biped is an open kinematic chain consisting of two subchains called legs and a subchain called torso, all connected at a common point called hip" (Westervelt et al., 2007). The end link of each leg that is in contact with the ground is called foot. "Walking can be defined as a locomotion gait of a biped, in which the feet are lifted alternately, while at least one foot is on the ground at all times" (Duindam & Stramigioli, 2006). On the other hand, when both feet leave the ground for even short period of time, the gait is called running. The stance leg is referred to the contacting leg in the case when only one leg is in contact with the ground while the other leg is called swing leg.

Many researches have been done to realize the biped walking. The major challenges to overcome this problem are as follows:

1. Redundancy: biped robots are usually mechanisms with high degrees of freedom (DOF). But on the other hand, the task of walking that is typically transportation of the robot's center of mass from one point to another is inherently a low DOF task. Therefore the biped mechanism has more DOF than required for doing the walking task. This kind of manipulator is called redundant manipulator. Consequently, there is no unique solution for limb coordination (robot configuration) such that the robot's center of mass is placed on the desired position. In this case, finding one solution among other is a difficult task. It can be more difficult if the best solution is needed among all.

2. Hybrid dynamics: walking inherently consists of multiple phase that is due to the presence of impacts and the different nature of contact conditions of the foot with the environment during walking cycle. Therefore, the model of walking robot is a hybrid one. Control of such a hybrid dynamic system is difficult.

3. Underactuation: contrary to traditional robot manipulators that are securely attached to the environment, biped robots are designed to move with respect to the environment. For fully actuated walking robots, the number of DOF can be actuated via electric motors are equal the number of joints. But the other 6 DOF of robot that relates to the position and orientation of the robot with respect to the ground cannot be controlled directly. The control of underactuated system is more difficult.

1.1 Overview of walking robots

Walking locomotion have been investigated from many years ago. Various fields are involved in legged locomotion researches such as dynamics, control, computer science, physiology and robotics. In spite of such great interest, it can be claimed that there are almost no

walking robot in use today, and those in use are for entertainment. Although, in the past 40 years hundreds of prototype have been constructed to improve different aspects of walking robot such as stability, speed, weight reduction, energy efficiency and so on, the real walking robot are not being employed to perform real work in real environment. In order to specify the development in research of walking locomotion, some biped prototype are introduced in this section.

Generally the biped robots that have been designed and investigated can be divided into two main groups: passive and powered biped robots. In the passive robots, the passive joints are used and the robot movement is based on the gravitational force, while powered biped robots use actuators in their joints to produce required energy for walking. This review only covers powered (fully actuated) robots.

The Japanese researchers are pioneers in constructing biped robots. The first reported biped robot that can walk is the WL-5, constructed at Waseda University in Japan in 1972 (Kato & Tsuiki, 1972). As shown in Fig. 1.1, WL-5 is a three-dimensional with 11 DOF. This group also improved this robot to construct the WL-10RD with 12 DOF and weight of 80 kg (Takanishi et al., 1985).

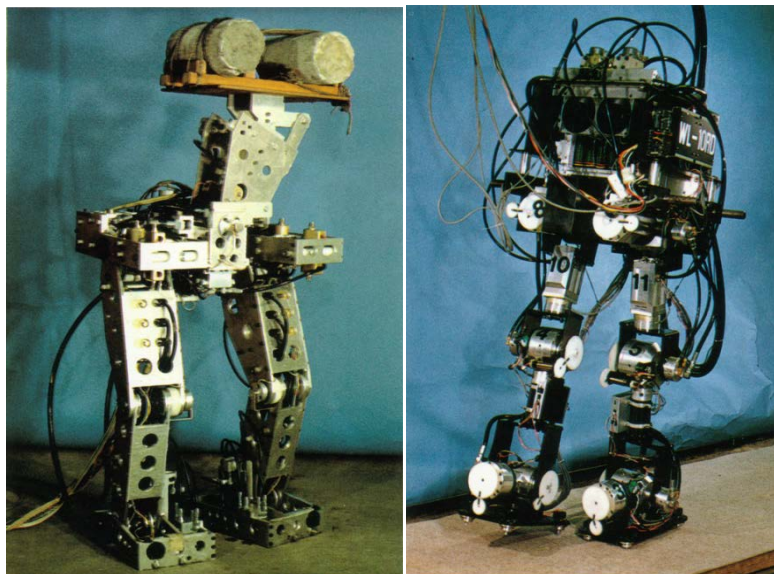


Fig. 1.1 The biped robots WL-5 and WL-10RD

A Planar biped robot with 6 DOF was built in the 1990s that its height was 0.45 m and its weight was 4.7 kg (Kajita & Tani, 1996 ; Kajita et al., 1992). It was able to walk at a speed of 0.2 m/s. Another biped robot was built at the MIT Leg Lab which is a seven-link planar biped robot with 1.2 m in height and 14 kg in weight (J. E. Pratt, 2000 ; J. Pratt et al., 2001). This robot was called Spring Flamingo and was ability to walk at 1.2 m/s even in sloped terrain, see Fig. 1.2.

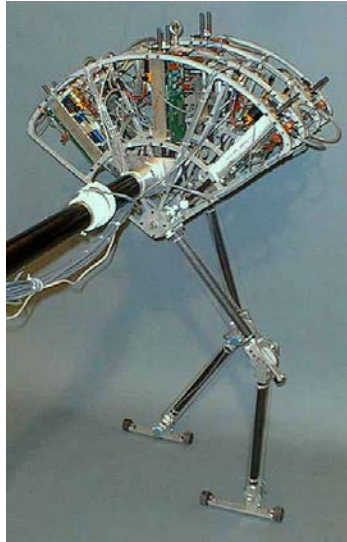


Fig. 1.2 The biped robot developed at MIT Leg Lab called Spring Flamingo

Another important biped robot built in 1990s was Johnnie. Johnnie was a 23-DOF walking robot weighting 40 kg and measuring 1.8 m in height (Löffler et al., 2003). It was constructed at the Technical University of Munich. After Johnnie, another biped robot developed at this university in 2009, named LOLA, with 25 DOF for high speed walking (target speed of 5 km/h) (Lohmeier & Buschmann, 2009a ; 2009b).

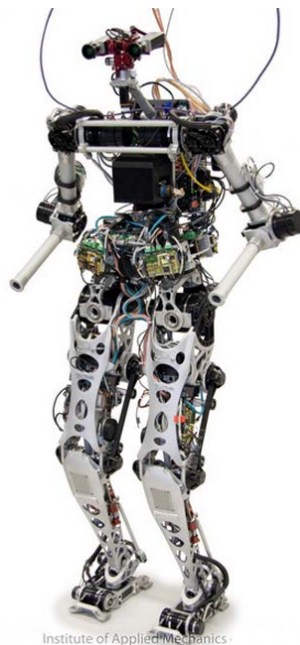


Fig. 1.3 The biped robot Johnnie

Without a doubt, the most famous humanoid robot is ASIMO (standing for Advanced Step in Innovation MObility) constructed by the Honda Corporation. Although, the ASIMO development was first reported at 1998 (Hirai et al., 1998), the development of ASIMO began in mid-1980s and continues to the present time (Chestnutt et al., 2005). ASIMO is an

autonomous three-dimensional biped robot with 26-DOF weighting 43 kg and measuring 1.2 m in height (Sakagami et al., 2002). It can walk at 0.3 m/s on level ground as well as climbing and descending stairs. As shown in Fig. 1.4, the development has involved ten generations of prototypes, named E0 to E6 and P1 to P3, and has cost hundreds of millions of dollars.



Fig. 1.4 Developed prototypes of ASIMO (www.asimo.honda.com, 2013)

Following HONDA's success, the Humanoid Robot Project (HRP) was started by government of Japan to grow Japan's service robot part. This project that supported by the National Institute of Advanced Industrial Science and Technology (AIST) started with three Honda P3 robot bought from Honda. The HRP-1 constructed in 1997 has 28 DOF and weighting 130 kg and measuring 1.6 m in height (Yokoi et al., 2003). In 2007, HRP-3 was constructed with 42 DOF and weighting 68 kg and measuring 1.6 m in height (Kaneko et al., 2008). Recently, HRP-4C and HRP-4 were constructed with 42 and 34 DOF, respectively (Kaneko et al., 2009 ; 2011). Fig. 1.5 shows the photos of mentioned HRP biped robots.

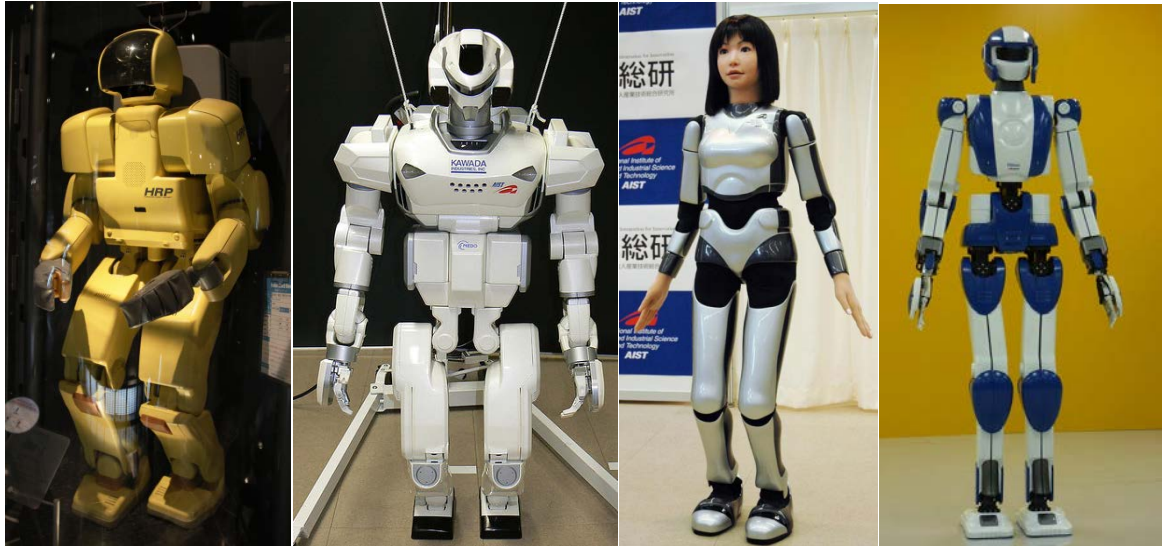


Fig. 1.5 Photos of HRP-1, HRP-3, HRP-4C and HRP-4

WABIAN is a three dimensional biped robot with 52 DOF that was developed at the Humanoid Robotics Institute in Waseda University in 2000. Another version of this robot was constructed in 2006, named WABIAN 2, see Fig. 1.6. Its weight is 64.5 kg and its height is 1.53 m (Aikawa et al., 2006).

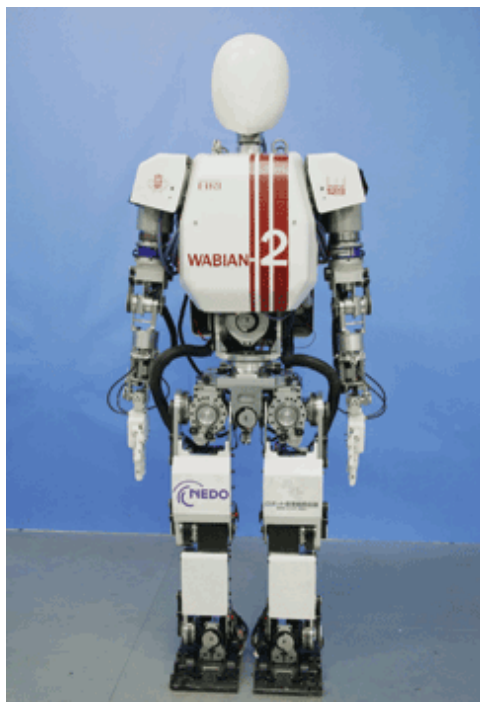


Fig. 1.6 WABIAN 2 photo

1.2 Problem statement

Walking realization for humanoid robot has been attracting great attention among robotic scientists specially in the two last decades. In this manner, many prototypes were constructed to realize biped walking such as robots mentioned in preceding section. Most of the humanoid robots were constructed in well-founded companies with expensive budget. Therefore, constructing humanoid robot (specially in the human size) have been limited to small group of big companies and well-known universities. In addition, the developed robots in these institutes are very expensive. This matter causes that these robots only have been used in high level research but not in real condition.

In order to develop a cost-oriented humanoid robot to assist people in daily life, a project was started in mid of 2000s at Vienna University of Technology. The project aim was to construct a cost-oriented humanoid robot, named Archie, for daily use. In the first stage of this project, the lower body of Archie was constructed (Byagowi, 2010 ; Baltes et al., 2009 ; Dezfouli et al., 2011).

Without a doubt, walking is the most important aspect of each humanoid robot. This feature enable the robot to move in the environment to reach desired position to perform other duties. Our project aim was specified to design and implement controller to realize stable walking of Archie.

In order to reach this goal, first some improvements in structure and electric part were required. First the electric parts (consists of motors, encoders and communication cables) were improved to make the robot reliable as a test bed for the proposed controller in this project. Second, one DOF was increased to each leg by adding a frontal ankle joint. After that, a controller was proposed to stably control the walking of Archie. This controller consists of a trajectory planner to create trajectory for the feet and the torso during walking. Then closed-form solution of inverse kinematics were derived to convert the generated trajectory in the Cartesian space into the joint space. The desired joint angles generated by inverse kinematics are used as the reference joint angles for independent joint controllers. The proposed controller was designed based on the decentralized cascaded control strategy. In this scheme, each joint are controlled to track the reference joint angle independently by a velocity closed-loop cascaded by position closed-loop system. The outline of thesis chapters are as follows:

- Chapter1 : Humanoid robot and biped walking are introduced and then an overview of famous walking robots is presented.
- Chapter2 : This chapter explains about the developed biped robot, Archie. Different parts of Archie such as mechanical structure, electric motor, transmissions and so on are introduced.

- Chapter3 : The forward kinematics of Archie is derived. Furthermore, the closed-form solution for inverse kinematics is developed. Finally, the validation of the derived inverse kinematics is approved.
- Chapter4 : The pattern generator are explained in this chapter. Also, the walking parameters that are selected by user to shape the desired walking shape are introduced.
- Chapter5 : The proposed controller based on the decentralized cascaded control method is presented in this chapter.
- Chapter6 : The implementation procedure consists of tuning controller parameters and synchronization is explained in this chapter. Then selecting optimal walking parameters for stable walking are discussed. Finally, walking realization results are presented.
- Chapter7 : The project is summarized and future works to improve Archie and its control system are proposed.

Chapter 2 Archie description

This chapter introduces the mechanical and electrical architecture of the biped robot, named Archie. Archie was designed as a humanoid robot with 31 degrees of freedom (DOF) (Byagowi, 2010). Currently, the lower body of Archie was constructed with 12 DOF. By adding a new ankle joint to each leg, this biped robot was improved to have totally 12 DOF which is the minimum DOF for the stable biped walking.

2.1 Mechanical structure

As shown in Fig. 2.1, Archie consists of two leg connecting with the torso link and a spinal column. The total height of Archie is 110 *cm* while each leg has 69.6 *cm*. Each leg has 6 joints; two ankle joints (frontal and lateral), one knee joint and three hip joints (frontal, lateral and transversal). With this 6 joints, the position and orientation of the torso can be specified completely in three dimensional space. These joints are connected via links that are constructed by Aluminum alloy. Since Aluminum has low weight with acceptable strength. The joint housing also made from Aluminum where the brushless DC motor and the harmonic drive are placed on it. The total weight of Archie including the structure and the electric motors is about 20 *kg*.

2.2 Electric motors

Two kind of electric motors are used in Archie: DC motors for transversal hip joints and brushless DC motors for the other joints.

2.2.1 Brushless DC motor

A flat brushless DC motor is used to actuate the most joints of the Archie. This motor is powerful and compact which is suitable to use in joints of biped robot. Fig. 2.2 shows the components of the brushless DC motor.

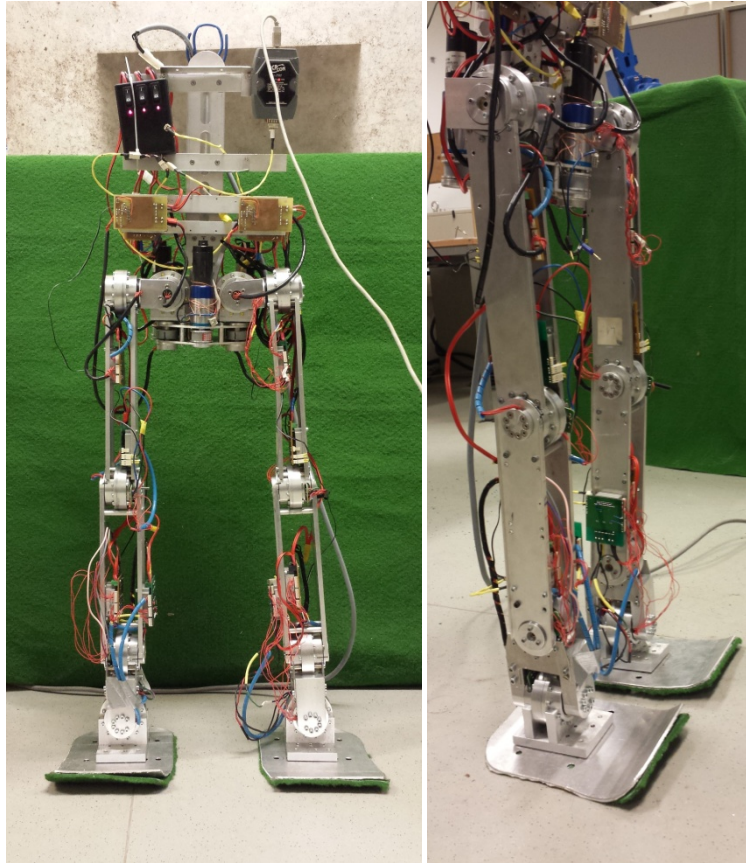


Fig. 2.1 The existing biped robot, Archie

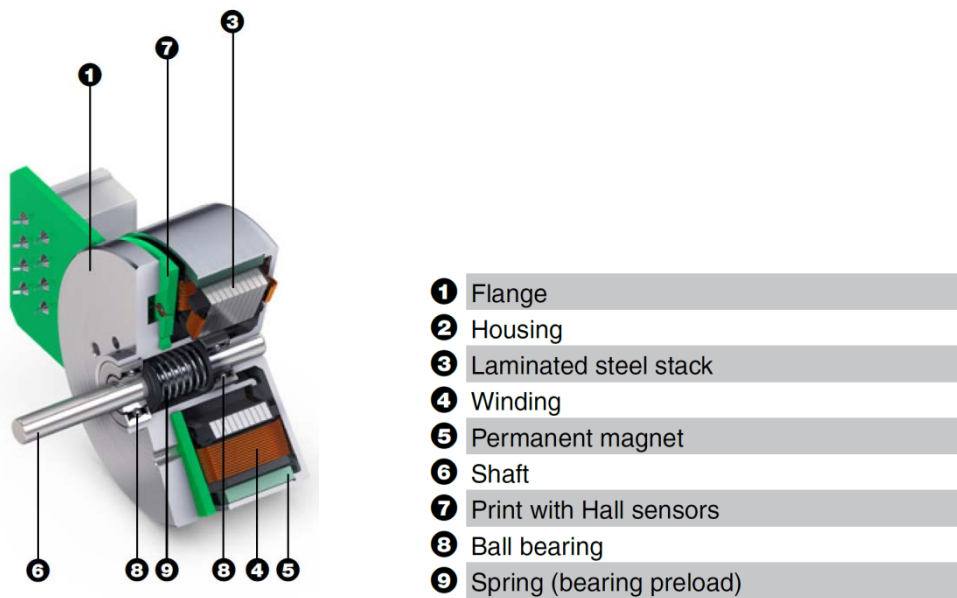


Fig. 2.2 Brushless DC motor components (www.maxonmotor.de, 2013)

The control of brushless DC motor are more complicated compared to the brushed DC motors. For commutation, three hall sensor are mounted in this motor to report the position

of the rotor to drive. Based on received information, the drive supply voltage to the proper wire winding to rotate the rotor, see Fig. 2.3.

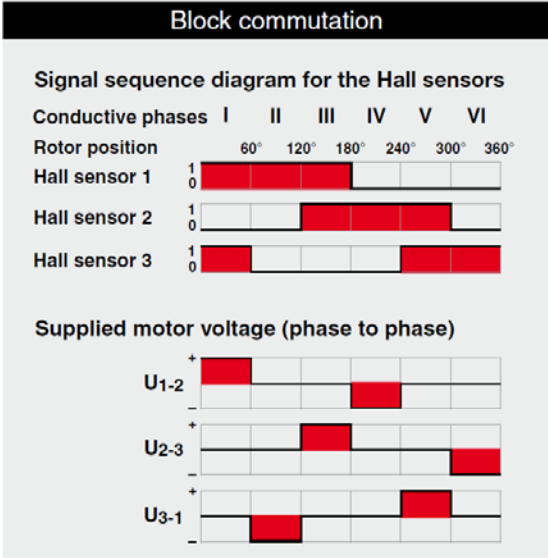


Fig. 2.3 Block commutation of brushless DC motor (*www.maxonmotor.de*, 2013)

The characteristics of the brushless DC motors are presented in table 2.1.

Nominal voltage	24 V
No load speed	6700 rpm
No load current	201 mA
Nominal speed	5260 rpm
Max. continuous torque	84.3 mNm
Max. continuous current	2.36 A
Stall torque	822 mNm
Starting current	24.5 A

Table 2.1 Brushless DC motor characteristics

2.2.2 DC motor

Due to small movement of the transversal joints, a brushed DC motor is utilized for these joints. The control of this motor is easier than brushless DC motor since the commutation is done by the built-in brush, see Fig. 2.4 for the DC motor components. On the other hand, its drawback is due to the corrosion of brush that leads to more maintenance service compared to brushless DC motor. The characteristics of the DC motor are summarized in table 2.2.

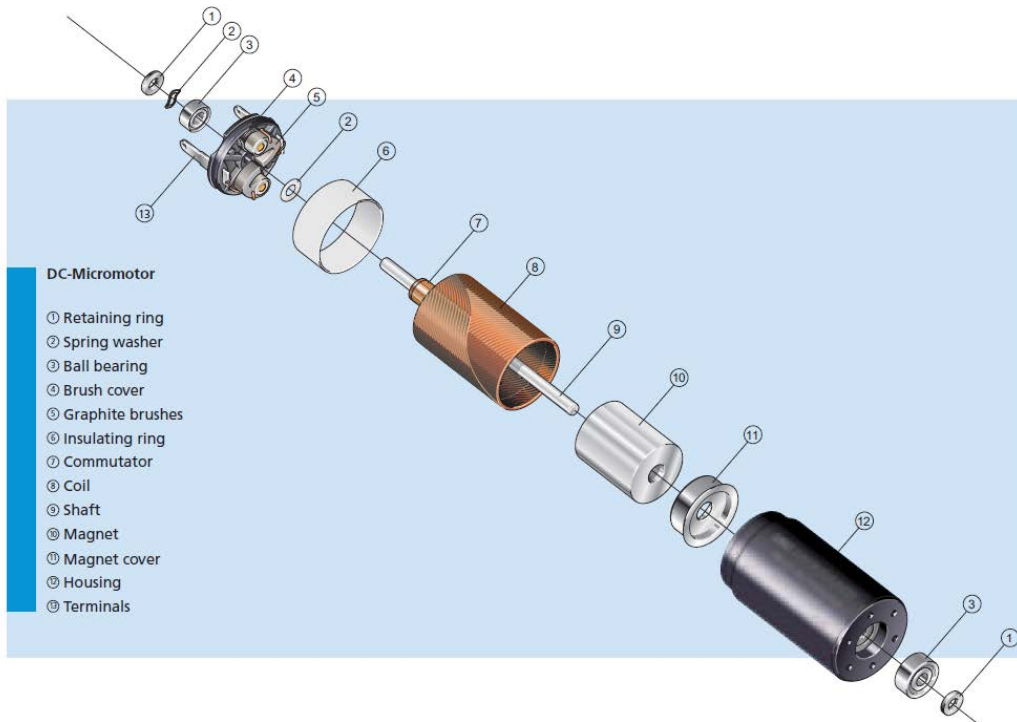


Fig. 2.4 DC motor components (www.faulhaber.com, 2013)

Nominal voltage	24 V
No load speed	5900 rpm
No load current	129 mA
Nominal speed	5000 rpm
Max. continuous torque	70 mNm
Max. continuous current	1.86 A
Stall torque	539 mNm
Starting current	14.3 A

Table 2.2 DC motor characteristics (www.faulhaber.com, 2013)

2.3 Transmission

Usually the joint of manipulator should actuate with high torques and low speeds. But the electric typically provide low torques with high speeds. Therefore, it is needed to use transmission (gear) to reduce the speed of shaft as well as increasing the joint torques. In Archie, two kind of gears are used for two kind of motors. For the brushless DC motors, harmonic drive is used while for the DC motors, planetary gears are utilized.

2.3.1 Harmonic drive

The harmonic drives are used as a transmission mechanism in the most joints of Archie due to the advantages such as low backlash, high torque transmission and compact size.

Generally the harmonic drive consists of three major parts: a rigid circular spline, a flexible flexspline and an elliptical wave generator. The wave generator is attached to the motor and is turned at high speed by the motor. The circular spline is attached to the link of manipulator. As the wave generator rotates it deforms the flexspline causing a number of teeth of the flexspline to mesh with the teeth of the circular spline. The low backlash and high torque of the harmonic drive results from the relative large number of teeth at any given time. However its drawback is the flexibility that is the principle of the harmonic drive. The effective gear ratio is specified by the difference in the number of teeth of the flexspline and circular spline. The gear ratio of selected harmonic drive for Archie is 160 and the maximum torque that it can tolerate is 76 Nm.

2.3.2 planetary gear

For DC motors, the planetary gear are used to transmit the power from the motor to the link. In this gear, 5 number of stages are used to reduce the speed and increase the torque by the ratio of 415. Fig. 2.5 shows the components of this planetary gear. The Maximum torque that this gear can transmit is 15 Nm. This gear designed for continuous speed of 4000 rpm and continuous torque of 10 Nm that is suitable for the case of transversal joints.

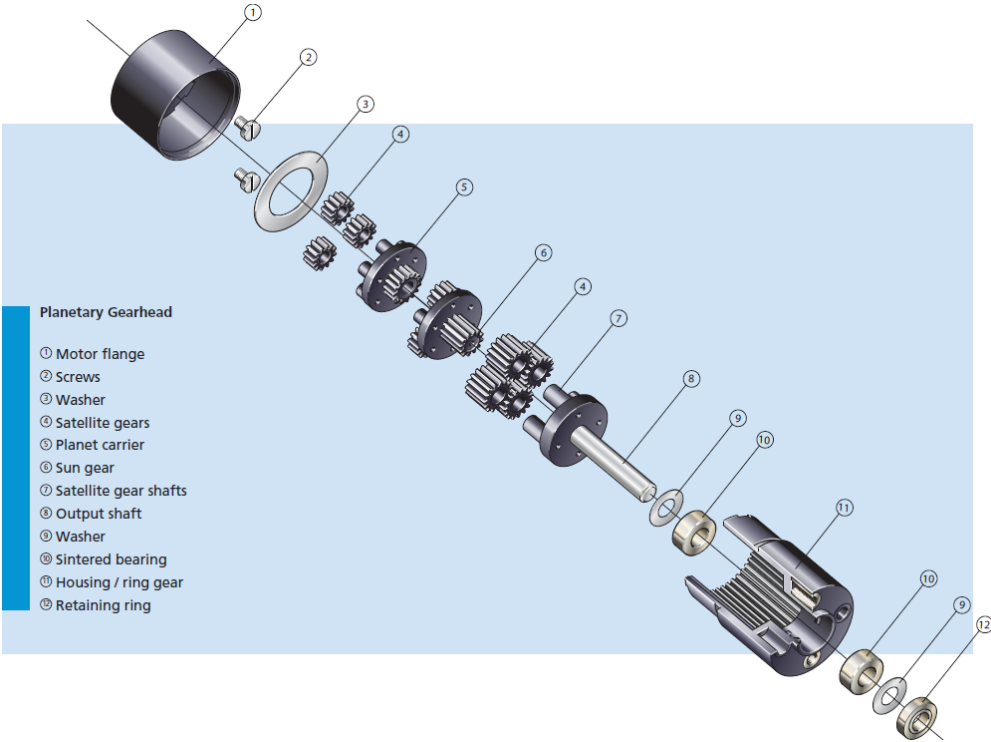


Fig. 2.5 Planetary gear components (www.faulhaber.com, 2013)

2.4 Encoder

In order to control the joint position in closed-loop system, the measurement of the joint angle is required. This measurement uses as a feedback for the joint controller. As shown in Fig. 2.6, a contactless magnetic rotary encoder is used for accurate angular measurement over a full turn of 360° . To measure the angle, a two-pole magnet is located on the shaft of the motor and the chip of encoder is attached to the manipulator link. Thus, based on the hall effect, the chip is able to measure the angle of the motor.

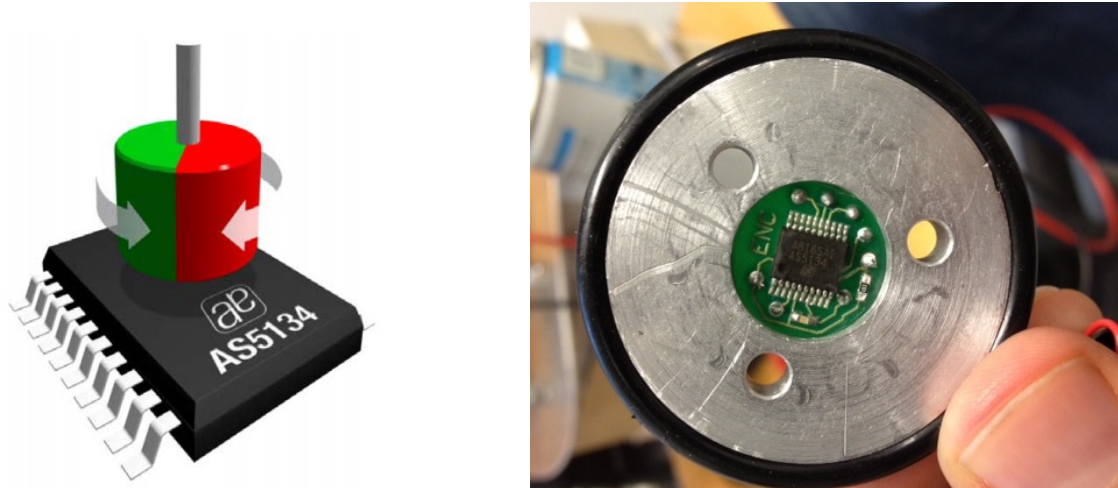


Fig. 2.6 Encoder chip and its housing to connect to the link

In fact this sensor is the incremental encoder that measure the angle of the motor with a resolution of 8.5 bit which is equal to 360 positions per revolution (1 degree).

2.5 Electric drive

In order to control the joint position based on the reference signal, an industrial drive consist of PI cascaded controller and power amplifier is used, see Fig. 2.7. This compact drive enable the user to tune select the three different mode; position, velocity and current modes. Then automatic or manual tuning of controller gain is done corresponding to the operation situation using the Composer software. The nominal voltage needed for electric drive is 50 V and its output power is 480 W. The system architecture of the electric drive is depicted in Fig. 2.8.

Two kind of communication are supported by Elmo motion controller; RS 232 and CAN communication. For Archie, CAN communication is used to transfer data between the main computer and the joints controller due to its high speed performance.



Fig. 2.7 The electric drive (www.elmopc.com, 2013)

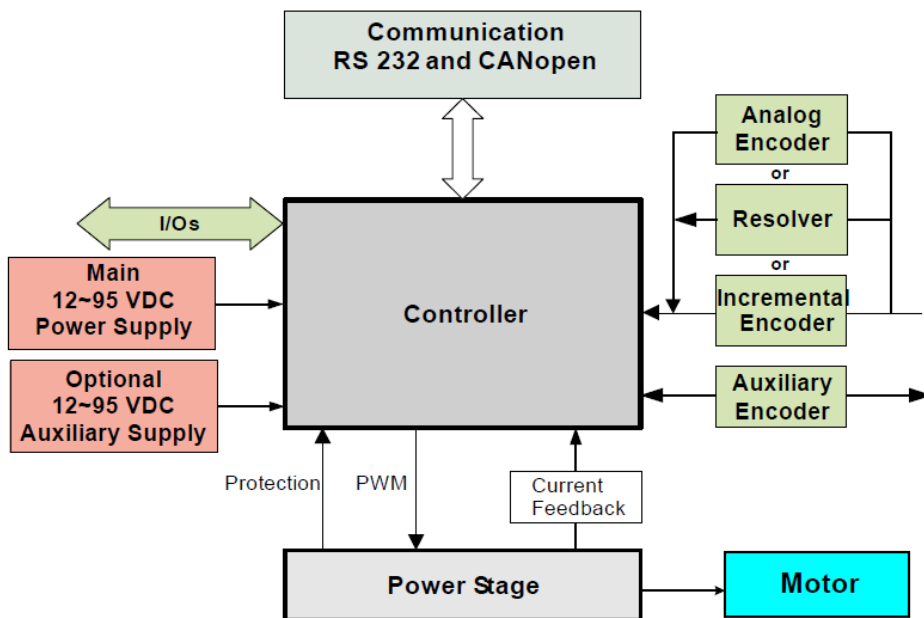


Fig. 2.8 Electric drive Architecture (www.elmopc.com, 2013)

Chapter 3 Forward and inverse kinematics

In this chapter, forward and inverse kinematic relations for Archie are developed. Generally kinematic problem is concerned with the relationship between the individual joints of the robot manipulator and the pose (position and orientation) of the end-effector. In forward kinematic problem, the aim is to find the pose of the end-effector, given the values of joint angles and geometric link parameters. Conversely, inverse kinematic problem is to determine the joint angles in terms of the end-effector pose. The more general definition of the inverse kinematic problem that is more applicable in the case of biped robot is stated as: find the values of all of the joint angles, given the relative positions and orientations of two members of a mechanism.

Solving the inverse kinematic problem is much more complicated than the forward kinematic problem due to the nonlinear equations to be solved. The number of equations equals the number of independently driven joints, i.e. degrees of freedom (DOF), of the robot and the number of unknowns is specified based on the desired movement of the end-effector. For instance, if the end-effector accepts spatial pose, the number of unknowns is equal to 6 (3 DOF for position and 3 DOF for orientation) and if the end-effector only moves in a plane, the number of unknowns is 3. When a manipulator has more independently driven joints than are necessary to define the desired pose, it is called kinematically redundant manipulator. Since, In this case, the equations are more than of the unknowns, the inverse kinematic problem has infinite solutions. In the case that spatial serial manipulator has 6 DOF, multiple solutions may exist. Choosing the best solution among all is another challenging part of the inverse kinematic problem. The number of solution decreases according to the structural joint limit.

Solving inverse kinematic problem can be classified into two main approaches: analytical and numerical. In the analytical approach the closed-form solution is developed while the numerical approach is based on the iterative process. The closed-form solution needs less computation time compared to the numerical method. This advantage is more critical in real time control. Furthermore, Having closed form solutions allows one to develop rules for choosing a particular solution among several. These advantages make the closed-form solution more desirable compared to the solution based on iterative method.

As mentioned before, the lower body of Archie consists of 12 DOF totally; Each leg has 6 DOF to enable the robot for the biped locomotion. In order to track a desired path for walking, the position of left and right feet of the biped robot should be specified. The stability of

walking also is achieved with proper positioning of the torso with respect to the stance leg(s). The trajectories of the feet and the torso based on human walking are developed in the Cartesian space in the next chapter.

Consequently, in order to find the desired joint angle values for walking, the desired position and orientation of the robot generated by pattern generator should be transformed from the Cartesian space into the joint space via inverse kinematic relation. In this manner, each leg is assumed as a separate manipulator with 6 DOF. Because the position and orientation of the torso ,i.e. end-effector for each leg, is determined relative to the foot, one can extract closed-form solution for inverse kinematics of each leg. The computed joint angles are used as reference to control each joint. In the following sections, the forward and inverse kinematics relations for each leg are derived. And finally the validation of inverse kinematics is verified.

3.1 Forward kinematics

The forward kinematic problem generally specifies relationship between the $(n \times 1)$ joint vector θ and the $(m \times 1)$ Cartesian vector x as

$$x = f(\theta) \quad 3-1$$

As above mentioned, in the case of our robot, kinematic relation of each leg is developed independently. In this manner, number of joint (n) is equal to 6 and the Cartesian vector is a (6×1) vector to determine position and orientation of the torso , i.e. the end-effector.

In order to develop the kinematics in universal language, the method proposed in (Spong et al., 2006) is utilized. This method simplifies the kinematic analysis considerably based on the Denavit-Hartenberg (D-H) convention. In the following subsection, the kinematic equations based on this method for left and right leg are derived.

3.1.1 Left leg

First, the kinematic relations of left leg are developed in this section. As illustrated in Fig. 3.1, the left leg consists of 6 joints connected to each other with 7 links. The joints are numbered from 1 to 6 starting from the base ,i.e. the frontal ankle joint. By this definition, joint i connects link $i - 1$ to link i . The location of joint i is assumed to be fixed with respect to link $i - 1$. When joint i is actuated, link i moves; Therefore, link 0 (the first link) is fixed, and does not move when the other joints are actuated. Furthermore, θ_i is the angle of rotation for the i^{th} joint .

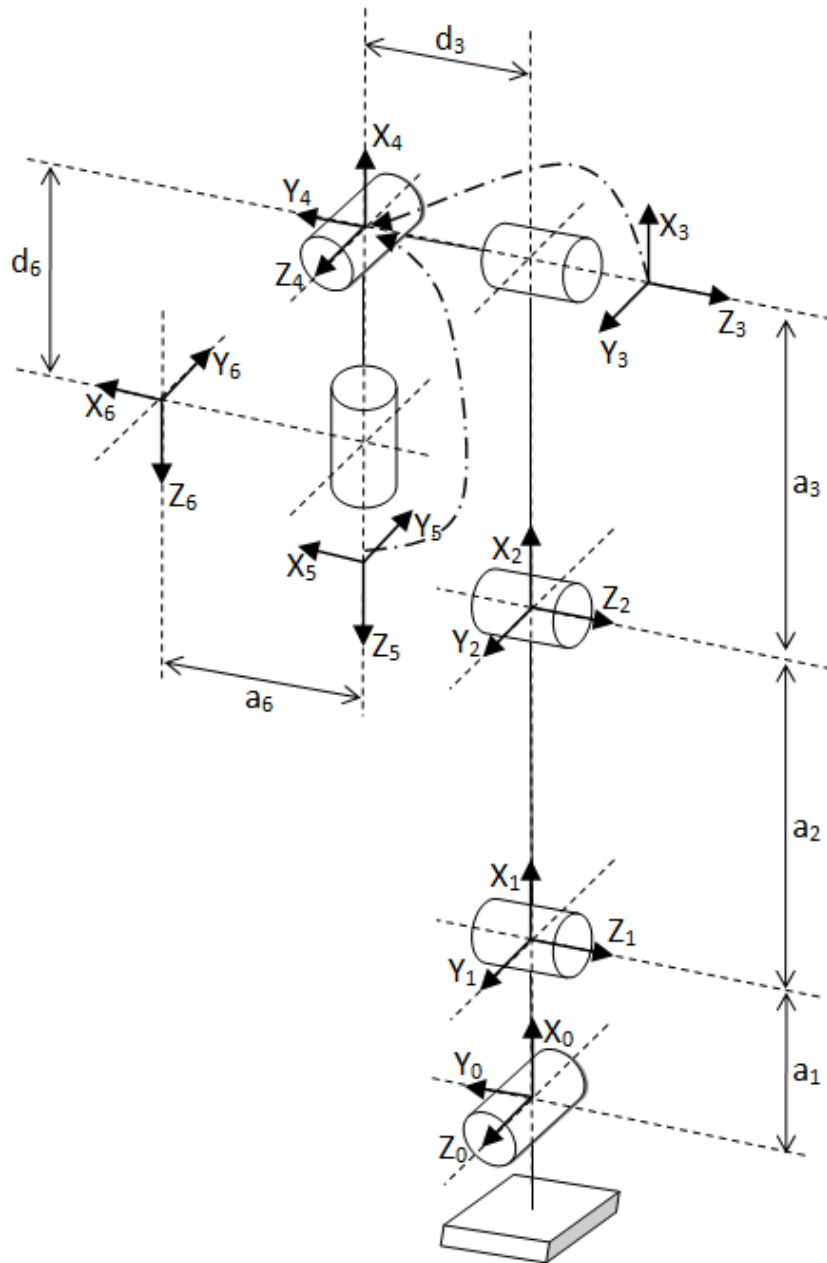


Fig. 3.1 Schematic figure of the left leg and its attached coordinates and dimensions

The first step in the kinematic analysis is to rigidly attach a coordinate frame to each link. In particular, the frame $\{0\}$, i.e. $X_0Y_0Z_0$, is attached to link 0. This means that, whatever motion the robot executes, the coordinates of each point on link i are constant when expressed in the i^{th} coordinate frame. In addition, when joint i is actuated, link i and its attached frame, $X_iY_iZ_i$, experience a resulting motion. The frame $X_0Y_0Z_0$, which is attached to the robot base, is referred to as the inertial frame. Fig. 3.1 illustrates the attached frames to the links for the left leg based on the D-H convention.

The homogenous transformation matrix T_i^{i-1} is defined such that expresses the position and orientation of the frame $X_iY_iZ_i$ with respect to the frame $X_{i-1}Y_{i-1}Z_{i-1}$. This matrix T_i^{i-1} is not fixed and is only a function of θ_i .

Each homogenous transformation matrix T_i^{i-1} is of the form of

$$T_i^{i-1} = \begin{bmatrix} R_i^{i-1} & O_i^{i-1} \\ 0 & 1 \end{bmatrix} \quad 3-2$$

where O_i^{i-1} is a (3×1) vector that specifies the position of the origin of the frame $\{i\}$ with respect to the frame $\{i-1\}$ and R_i^{i-1} is a (3×3) rotation matrix that determines the orientation of the frame $\{i\}$ with respect to the frame $\{i-1\}$. The general form of the transformation matrix can be obtained as

$$T_j^i = T_{i+1}^i T_{i+2}^{i+1} T_{i+3}^{i+2} \dots T_j^{j-1} \quad 3-3$$

Based on the D-H convention, each homogenous transformation matrix is represented as a product of four basic transformation

$$T_i^{i-1} = Rot_{z,\theta_i} Trans_{z,d_i} Trans_{x,\alpha_i} Rot_{x,\alpha_i} \quad 3-4$$

where Rot_{z,θ_i} means a rotation of θ_i about axis Z and $Trans_{z,d_i}$ means a translation of d_i along axis Z and so on. By expanding this formula and defining the following writing abbreviations

$$C\theta_i = \cos(\theta_i) \quad \text{and} \quad S\theta_i = \sin(\theta_i) \quad 3-5$$

we have

$$T_i^{i-1} = \begin{bmatrix} C\theta_i & -S\theta_i & 0 & 0 \\ S\theta_i & C\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & C\alpha_i & -S\alpha_i & 0 \\ 0 & S\alpha_i & C\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} C\theta_i & -S\theta_i C\alpha_i & S\theta_i S\alpha_i & a_i C\theta_i \\ S\theta_i & C\theta_i C\alpha_i & -C\theta_i S\alpha_i & a_i S\theta_i \\ 0 & S\alpha_i & C\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad 3-6$$

where the four quantities θ_i , a_i , d_i , α_i are the parameters associated with the link i and the joint i . The four parameters a_i , α_i , d_i , and θ_i are generally given the names link length, link twist, link offset, and joint angle, respectively. These names derive from specific aspects of the geometric relationship between two coordinate frames. Since all the robot joints are revolute, only θ_i is joint variable and the others are constant with following definition:

a_i = distance along X_i from the origin of the frame $\{i\}$ to the intersection of the X_i and Z_{i-1} axes.

d_i = distance along Z_{i-1} from the origin of the frame $\{i-1\}$ to the intersection of the X_i and Z_{i-1} axes.

α_i = the angle between Z_{i-1} and Z_i measured about X_i .

θ_i = the angle between X_{i-1} and X_i measured about Z_{i-1} .

Link	a_i	α_i	d_i	θ_i	Initial angle
1	a_1	$\pi/2$	0	θ_1	0
2	a_2	0	0	θ_2	0
3	a_3	0	$-d_3$	θ_3	0
4	0	$-\pi/2$	0	θ_4	0
5	0	$-\pi/2$	0	θ_5	$\pi/2$
6	a_6	0	d_6	θ_6	0

Table 3.1 D-H parameters for the left leg

Based on the above definitions and the attached coordinate to each link shown in Fig. 3.1, the D-H parameters of each link are obtained and tabulated in table 3.1. The last column of this table also shows the initial joints angles for standing position of the robot.

By substituting the D-H parameters, the transformation matrices between all coordinates can be expressed as

$$T_1^0 = \begin{bmatrix} C1 & 0 & S1 & a_1 C1 \\ S1 & 0 & -C1 & a_1 S1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad 3-7$$

$$T_2^1 = \begin{bmatrix} C2 & -S2 & 0 & a_2 C2 \\ S2 & C2 & 0 & a_2 S2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad 3-8$$

$$T_3^2 = \begin{bmatrix} C3 & -S3 & 0 & a_3 C3 \\ S3 & C3 & 0 & a_3 S3 \\ 0 & 0 & 1 & -d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad 3-9$$

$$T_4^3 = \begin{bmatrix} C4 & 0 & -S4 & 0 \\ S4 & 0 & C4 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad 3-10$$

$$T_5^4 = \begin{bmatrix} C5 & 0 & -S5 & 0 \\ S5 & 0 & C5 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad 3-11$$

$$T_6^5 = \begin{bmatrix} C6 & -S6 & 0 & a_6 C6 \\ S6 & C6 & 0 & a_6 S6 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad 3-12$$

It is worth to mention that the following abbreviations are used in the above formulas:

$$C_i = \cos \theta_i, S_i = \sin \theta_i, C_{ij} = \cos(\theta_i + \theta_j), S_{ij} = \sin(\theta_i + \theta_j) \quad 3-13$$

Multiplying the transformation matrices leads to find the T_6^0 as:

$$T_6^0 = T_1^0 \cdot T_2^1 \cdot T_3^2 \cdot T_4^3 \cdot T_5^4 \cdot T_6^5 \quad 3-14$$

$$T_3^0 = T_1^0 \cdot T_2^1 \cdot T_3^2 = \begin{bmatrix} C1 C23 & -C1 S23 & S1 & a_3 C1 C23 + a_2 C1 C2 - d_3 S1 + a_1 C1 \\ S1 C23 & -S1 S23 & -C1 & a_3 S1 C23 + a_2 S1 C2 + d_3 C1 + a_1 S1 \\ S23 & C23 & 0 & a_3 S23 + a_2 S2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad 3-15$$

$$T_6^3 = T_4^3 \cdot T_5^4 \cdot T_6^5 = \begin{bmatrix} C4 C5 C6 + S4 S6 & -C4 C5 S6 + S4 C6 & -C4 S5 & a_6 C6 C4 C5 + a_6 S4 S6 - d_6 C4 S5 \\ S4 C5 C6 - C4 S6 & -S4 C5 S6 - C4 C6 & -S4 S5 & a_6 S4 C5 C6 - a_6 C4 S6 - d_6 S4 S5 \\ -S5 C6 & S5 S6 & -C5 & -a_6 S5 C6 - d_6 C5 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad 3-16$$

3.1.2 Right leg

The kinematic relations for the right leg are derived analogously to the left leg in this section with some differences. Fig. 3.2 shows the attached coordinate to each link based on D-H convention. Table 3.2 also shows the D-H parameters for the right leg.

Link	a_i	α_i	d_i	θ_i	Initial angle
1	a_1	$-\pi/2$	0	θ_1	0
2	a_2	0	0	θ_2	0
3	a_3	0	$-d_3$	θ_3	0
4	0	$\pi/2$	0	θ_4	0
5	0	$-\pi/2$	0	θ_5	$\pi/2$
6	$-a_6$	0	d_6	θ_6	0

Table 3.2 D-H parameters for the right leg

Substituting D-H parameters values leads to the transformation matrices between all coordinates as :

$$T_1^0 = \begin{bmatrix} C1 & 0 & -S1 & a_1 C1 \\ S1 & 0 & C1 & a_1 S1 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad 3-17$$

$$T_2^1 = \begin{bmatrix} C2 & -S2 & 0 & a_2 C2 \\ S2 & C2 & 0 & a_2 S2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad 3-18$$

$$T_3^2 = \begin{bmatrix} C3 & -S3 & 0 & a_3 C3 \\ S3 & C3 & 0 & a_3 S3 \\ 0 & 0 & 1 & -d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad 3-19$$

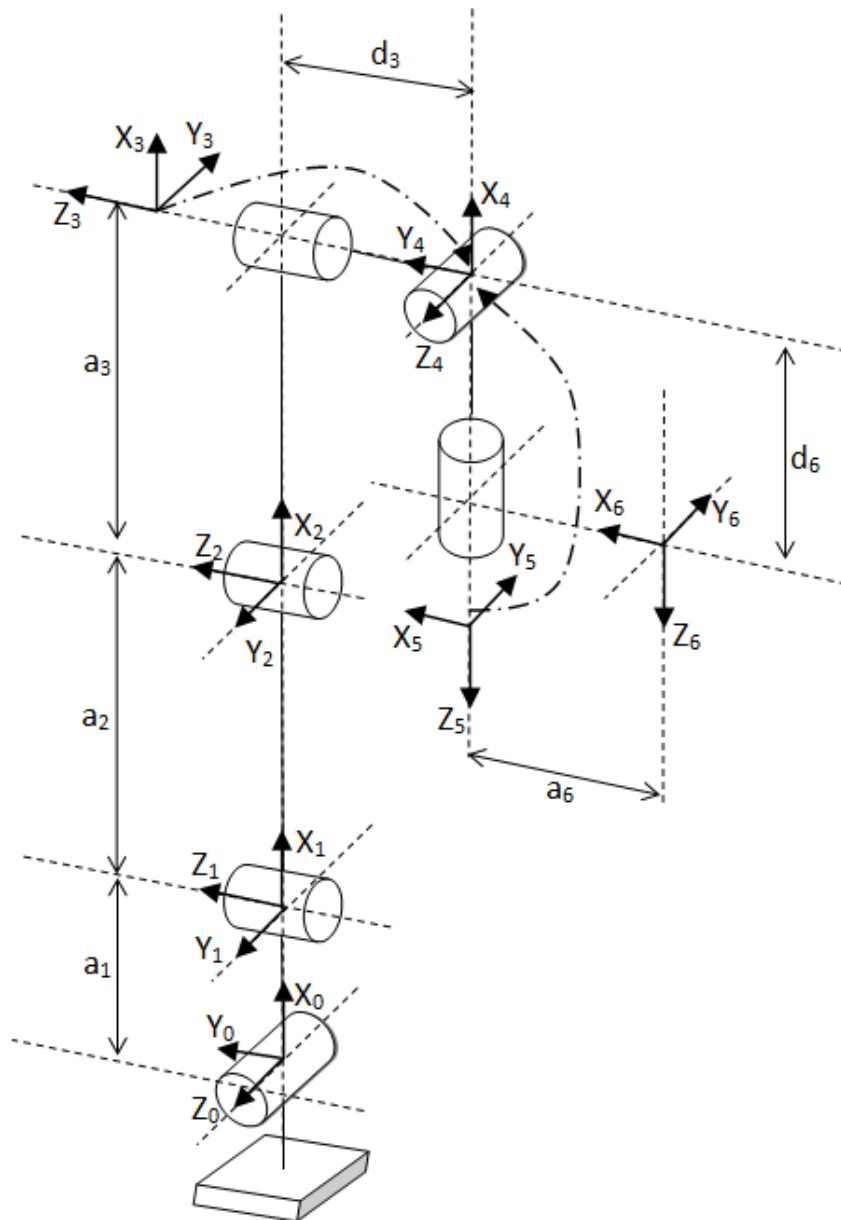


Fig. 3.2 Schematic figure of the right leg and its attached coordinates and dimensions

$$T_4^3 = \begin{bmatrix} C4 & 0 & S4 & 0 \\ S4 & 0 & -C4 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad 3-20$$

$$T_5^4 = \begin{bmatrix} C5 & 0 & -S5 & 0 \\ S5 & 0 & C5 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad 3-21$$

$$T_6^5 = \begin{bmatrix} C6 & -S6 & 0 & -a_6 C6 \\ S6 & C6 & 0 & -a_6 S6 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad 3-22$$

$$T_6^0 = T_1^0 \cdot T_2^1 \cdot T_3^2 \cdot T_4^3 \cdot T_5^4 \cdot T_6^5 \quad 3-23$$

$$T_3^0 = T_1^0 \cdot T_2^1 \cdot T_3^2 = \begin{bmatrix} C1 C23 & -C1 S23 & -S1 & a_3 C1 C23 + a_2 C1 C2 + d_3 S1 + a_1 C1 \\ S1 C23 & -S1 S23 & C1 & a_3 S1 C23 + a_2 S1 C2 - d_3 C1 + a_1 S1 \\ -S23 & -C23 & 0 & -a_3 S23 - a_2 S2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad 3-24$$

$$T_6^3 = T_4^3 \cdot T_5^4 \cdot T_6^5 = \begin{bmatrix} C4 C5 C6 - S4 S6 & -C4 C5 S6 - S4 C6 & -C4 S5 & -a_6 C6 C4 C5 + a_6 S4 S6 - d_6 C4 S5 \\ S4 C5 C6 + C4 S6 & -S4 C5 S6 - C4 C6 & -S4 S5 & -a_6 S4 C5 C6 - a_6 C4 S6 - d_6 S4 S5 \\ S5 C6 & -S5 S6 & -C5 & -a_6 S5 C6 + d_6 C5 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad 3-25$$

3.2 Inverse kinematics

The solution to the inverse kinematic problem is of the fundamental importance in order to transform the motion specifications, assigned to the end-effector in the operational space, into the corresponding joint space motions that allow execution of the desired motion.

3.2.1 Left leg

This section aims to find the joint angle trajectories based on the end-effector and base position and orientation which is called inverse kinematics. So the basic assumption is to know the position and orientation of the end-effector frame (here frame {6} attached to the torso) with respect to the coordinate {0} attached to the frontal ankle. It is noted that the planning of the trajectories for the feet and the torso will be presented in the next chapter.

In this manner the position of frame {6} with respect to the frame {0}, i.e. P_6^0 , and the orientation of the frame {6} with respect to the frame {0}, i.e. R_6^0 , are assumed to be specified. Thus T_6^0 is known and it is desired to find $\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6$.

$$T_6^0 = \begin{bmatrix} R_6^0 & P_6^0 \\ 0 & 1 \end{bmatrix} \quad 3-26$$

Although the general problem of inverse kinematics is quite difficult, for manipulators having six joints with the last three joints intersecting at a point, it is possible to decouple the inverse kinematics problem into two simpler problems, known respectively, as inverse position kinematics, and inverse orientation kinematics.

a. Inverse position

In order to decompose inverse kinematic problem into two separate problem, i.e. position and orientation, first the position of the frame {5} with respect to the frame {0} should be specified. As shown in Fig. 3.3, the geometric relation can be expressed as

$$P_5^0 = P_6^0 + R_6^0 \cdot P_5^6 \quad 3-27$$

where

$P_5^6 = \begin{bmatrix} -a_6 \\ 0 \\ -d_6 \end{bmatrix}$ is a position of the frame {5} stated in the frame {6}, see Fig. 3.1. To obtain this vector in frame {0}, it is multiplied by R_6^0 to change the observer frame from {6} to {0}.

So $P_5^0 = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix}$ will be found since the right side of the equation are all known.

Because origin of frame {3}, {4} and {5} located in a same point (see Fig. 3.1), we have

$$P_3^0 = P_4^0 = P_5^0 \quad 3-28$$

First of all, the geometrical approach is used to find θ_1 . Fig. 3.4 shows the location of point c (center of frame {3}, {4} and {5}) when first joint rotates. Using trigonometric relation and based on the Fig. 3.4 we have

$$\tan(\psi + \theta_1) = \frac{P_y}{P_x} \quad 3-29$$

$$\psi = \sin^{-1} \frac{d_3}{\sqrt{P_x^2 + P_y^2}} \quad 3-30$$

Thus

$$\boxed{\theta_1 = \text{Atan2}(P_y, P_x) - \psi} \quad 3-31$$

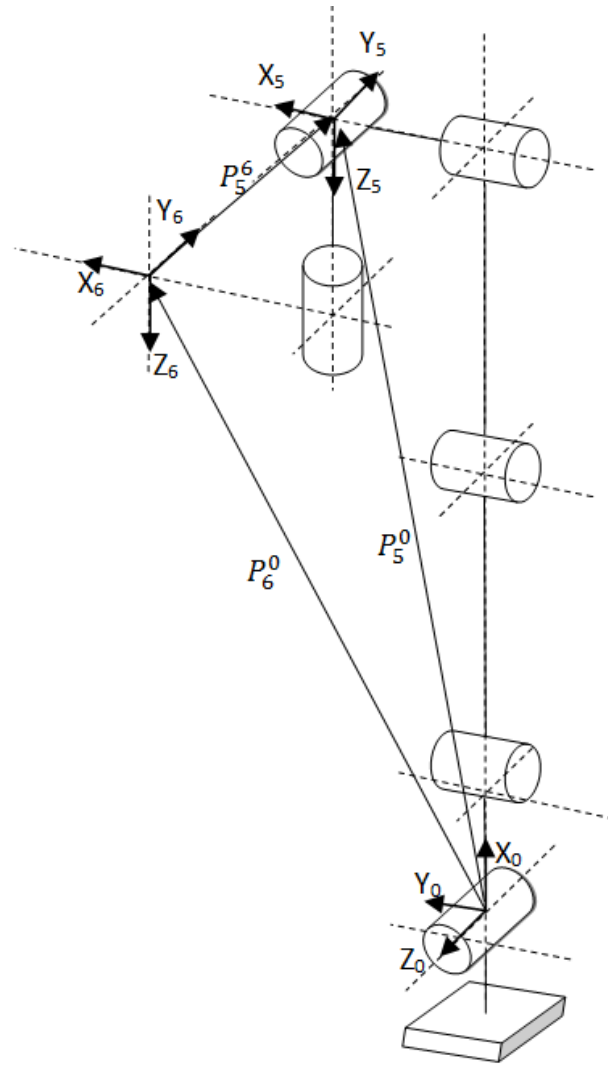


Fig. 3.3 Vector relation between frames {0} ,{5} and {6} for the left leg

From the forward kinematics it is known that

$$P_3^0 = \begin{bmatrix} a_3 C1 C23 + a_2 C1 C2 - d_3 S1 + a_1 C1 \\ a_3 S1 C23 + a_2 S1 C2 + d_3 C1 + a_1 S1 \\ a_3 S23 + a_2 S2 \end{bmatrix} = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} \quad 3-32$$

Then for each element of the matrices we have

$$a_3 C1 C23 + a_2 C1 C2 = P_x + d_3 S1 - a_1 C1 \quad 3-33$$

$$a_3 S1 C23 + a_2 S1 C2 = P_y - d_3 C1 - a_1 S1 \quad 3-34$$

$$a_3 S23 + a_2 S2 = P_z \quad 3-35$$

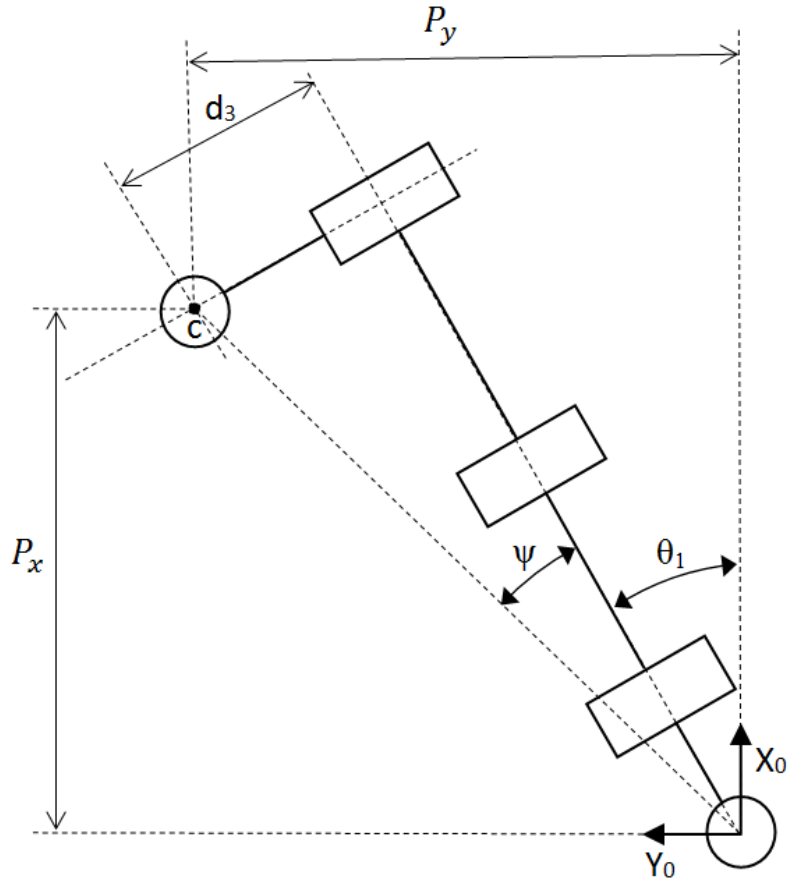


Fig. 3.4 Position of the point c (center of frame {3}, {4} and {5}) when first joint rotates in the left leg

By adding the squared of equations (3-33) , (3-34) and (3-35), we have

$$(3 - 33)^2 + (3 - 34)^2 + (3 - 35)^2 \Rightarrow$$

$$a_3^2 + a_2^2 + 2 a_2 a_3 C3 = P_z^2 + (P_x + d_3 S1 - a_1 C1)^2 + (P_y - d_3 C1 - a_1 S1)^2 \quad 3-36$$

Then C3 and S3 can be obtained as

$$C3 = \frac{P_z^2 + (P_x + d_3 S1 - a_1 C1)^2 + (P_y - d_3 C1 - a_1 S1)^2 - a_3^2 - a_2^2}{2 a_2 a_3} \quad 3-37$$

$$S3 = \pm \sqrt{1 - C3^2} \quad 3-38$$

Because the knee joint can only rotates backward and also there is joint movement limitation, the knee joint angle should conform following constraint:

$$-\frac{\pi}{2} \leq \theta_3 \leq 0 \quad 3-39$$

So S3 should be negative then

$$\theta_3 = \text{Atan2}(-\sqrt{1 - C3^2}, C3) \quad 3-40$$

To find θ_2 , (3-33) and (3-34) are squared and added.

$$(3-33)^2 + (3-34)^2 \Rightarrow$$

$$(a_3 C23 + a_2 C2)^2 = (P_x + d_3 S1 - a_1 C1)^2 + (P_y - d_3 C1 - a_1 S1)^2$$

$$\Rightarrow C2(a_3 C3 + a_2) - S2(a_3 S3) = \pm \sqrt{(P_x + d_3 S1 - a_1 C1)^2 + (P_y - d_3 C1 - a_1 S1)^2}$$

3-41

$$(3-35) \Rightarrow S2(a_3 C3 + a_2) + C2(a_3 S3) = P_z$$

3-42

Then let define ($r > 0$) and φ as

$$a_3 C3 + a_2 = r C\varphi$$

3-43

$$a_3 S3 = r S\varphi$$

3-44

Multiplying equation (3-43) by $C2$ and equation (3-44) by $S2$, then subtracted from each other :

$$\Rightarrow \begin{cases} r C\varphi C2 - r S\varphi S2 = \pm \sqrt{(P_x + d_3 S1 - a_1 C1)^2 + (P_y - d_3 C1 - a_1 S1)^2} \\ r C\varphi S2 + r S\varphi C2 = P_z \end{cases}$$

3-45

$$\Rightarrow \begin{cases} r \cos(\varphi + \theta_2) = \pm \sqrt{(P_x + d_3 S1 - a_1 C1)^2 + (P_y - d_3 C1 - a_1 S1)^2} \\ r \sin(\varphi + \theta_2) = P_z \end{cases}$$

3-46

$$\varphi = \text{Atan2}(a_3 S3, a_3 C3 + a_2)$$

3-47

$$\theta_2 + \varphi = \text{Atan2}(P_z, \pm \sqrt{(P_x + d_3 S1 - a_1 C1)^2 + (P_y - d_3 C1 - a_1 S1)^2})$$

3-48

As stated before the knee joint is constrained as $-\frac{\pi}{2} \leq \theta_3 \leq 0$ and a_3 and a_2 are positive values. Then based on the equation (3-48), one can obtain that :

$$-\frac{\pi}{2} \leq \varphi \leq 0$$

3-49

Also from movement limitation of the lateral ankle joint, it is known that

$$-\frac{\pi}{3} \leq \theta_2 \leq \frac{\pi}{3}$$

3-50

Thus

$$-\left(\frac{\pi}{3} + \frac{\pi}{2}\right) \leq \theta_2 + \varphi \leq \frac{\pi}{3}$$

3-51

So choosing positive sign for $\cos(\varphi + \theta_2)$ leads to satisfactory result as

$$\theta_2 = \text{Atan2}\left(P_z, \sqrt{(P_x + d_3 S1 - a_1 C1)^2 + (P_y - d_3 C1 - a_1 S1)^2}\right) - \varphi \quad 3-52$$

b. Inverse orientation

In the previous section the geometric and analytic approach was used to solve the inverse position problem. This gives the values of the first three joint variables corresponding to a given position of the axis intersection of three hip joints. The inverse orientation problem is now finding the values of the final three joint variables corresponding to a given orientation with respect to the frame {0}. So, suppose the orientation of the frame {6} with respect to the frame {0} is known. From the forward kinematics it is clear that:

$$R_6^0 = R_1^0 R_2^1 R_3^2 R_4^3 R_5^4 R_6^5 \quad 3-53$$

$$R_6^0 = R_3^0 R_6^3 \Rightarrow R_6^3 = R_3^{0^{-1}} R_6^0 \quad 3-54$$

From the definition of the rotation matrix we know:

$$R_3^{0^{-1}} = R_3^{0T} \quad 3-55$$

So

$$R_6^3 = R_3^{0T} R_6^0 \quad 3-56$$

By substitution of values of $\theta_1, \theta_2, \theta_3$ calculated from the previous section, R_3^0 can be found from forward kinematics as follows:

$$R_3^0 = \begin{bmatrix} C1 C23 & -C1 S23 & S1 \\ S1 C23 & -S1 S23 & -C1 \\ S23 & C23 & 0 \end{bmatrix} \quad 3-57$$

And all elements of $R_6^0 = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$ are known.

Let

$$R_3^{0T} R_6^0 = R' = \begin{bmatrix} r'_{11} & r'_{12} & r'_{13} \\ r'_{21} & r'_{22} & r'_{23} \\ r'_{31} & r'_{32} & r'_{33} \end{bmatrix} \quad 3-58$$

So all elements of R' can be obtained.

From the forward kinematics also R_6^3 are formulated as

$$R_6^3 = \begin{bmatrix} C4 C5 C6 + S4 S6 & -C4 C5 S6 + S4 C6 & -C4 S5 \\ S4 C5 C6 - C4 S6 & -S4 C5 S6 - C4 C6 & -S4 S5 \\ -S5 C6 & S5 S6 & -C5 \end{bmatrix} \quad 3-59$$

First suppose that not both of r'_{13} and r'_{23} are zero, then one can conclude that $S5 \neq 0$ and hence that not both of r'_{31} and r'_{32} are zero. Thus $r'_{33} \neq \pm 1$ and we have

$$C5 = -r'_{33} \quad 3-60$$

$$S5 = \pm \sqrt{1 - r'_{33}} \quad 3-61$$

Due to the joint movement limitation in joint 5, only positive value of $S5$ is acceptable. So

$$\theta_5 = \text{Atan2}(\sqrt{1 - r'_{33}}, -r'_{33}) \quad 3-62$$

$$\theta_4 = \text{Atan2}(-r'_{23}, -r'_{13}) \quad 3-63$$

$$\theta_6 = \text{Atan2}(r'_{32}, -r'_{31}) \quad 3-64$$

Another case is when $r'_{13} = r'_{23} = 0$, then the fact that R' is orthogonal implies that

$$r'_{33} = \pm 1 \quad 3-65$$

$$r'_{31} = r'_{32} = 0 \quad 3-66$$

So in this case $\theta_5 = 0$ or π which is out of this joint limit and is not acceptable result.

3.2.2 Right leg

a. Inverse position

As shown in Fig. 3.5, the position of the frame {5} with respect to the frame {0} is obtained by following relation:

$$P_5^0 = P_6^0 + R_6^0 \cdot P_5^6 \quad 3-67$$

where

$P_5^6 = \begin{bmatrix} a_6 \\ 0 \\ -d_6 \end{bmatrix}$ is a position of the frame {5} stated in the frame {6}, see Fig. 3.2. To obtain this vector seen from the frame {0}, it is multiplied by R_6^0 to change the observer frame from {6} to {0}.

So $P_5^0 = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix}$ will be found due to the fact that all parameters of the right side of the equation are known.

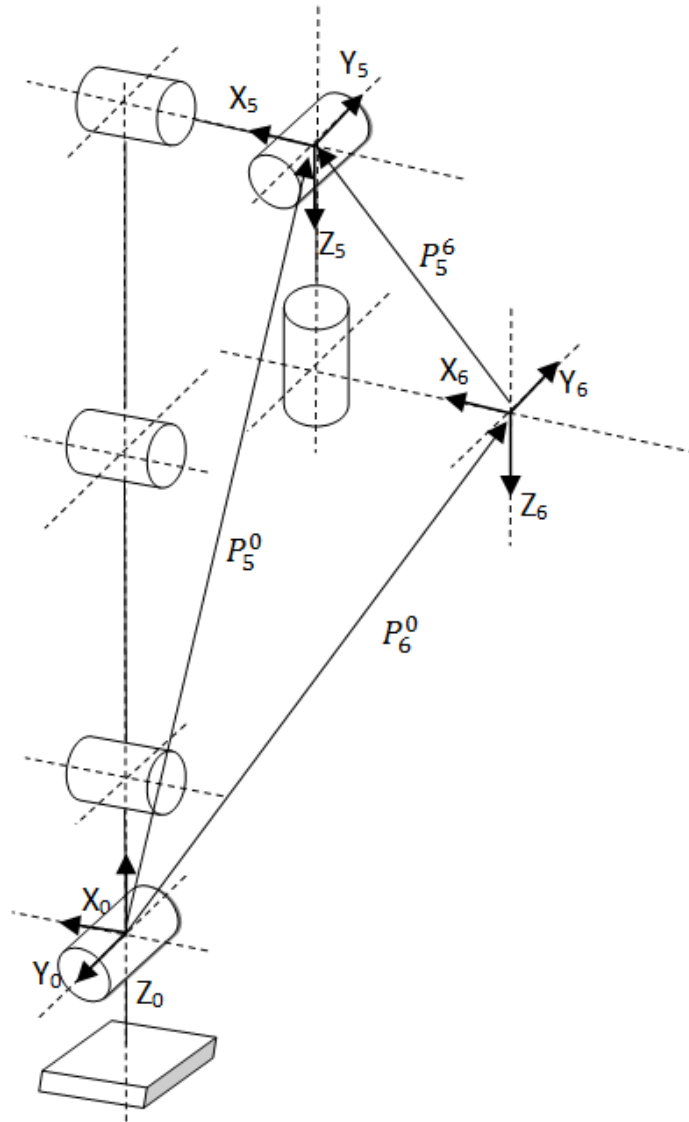


Fig. 3.5 Vector relation between frames {0}, {5} and {6} for the right leg

Fig. 3.6 shows the location of point c (center of frame {3}, {4} and {5}) for right leg when the first joint, i.e. frontal ankle, rotates. Using trigonometry relation depicted in Fig. 3.6, θ_1 can be obtained by

$$\tan(\theta_1 - \psi) = \frac{P_y}{P_x} \quad 3-68$$

$$\psi = \sin^{-1} \frac{d_3}{\sqrt{P_x^2 + P_y^2}} \quad 3-69$$

$$\boxed{\theta_1 = \text{Atan2}(P_y, P_x) + \psi} \quad 3-70$$

From the forward kinematics of right leg it is known:

$$P_3^0 = \begin{bmatrix} a_3 C1 C23 + a_2 C1 C2 + d_3 S1 + a_1 C1 \\ a_3 S1 C23 + a_2 S1 C2 - d_3 C1 + a_1 S1 \\ -a_3 S23 - a_2 S2 \end{bmatrix} = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} \quad 3-71$$

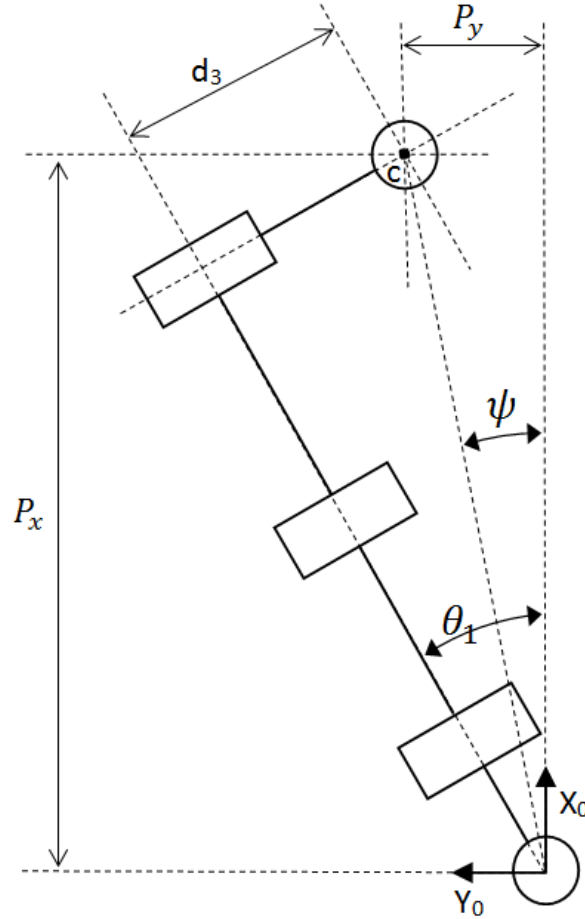


Fig. 3.6 Position of the point c (center of frame {3}, {4} and {5}) when first joint rotates in the right leg

Reordering the above matrix elements as:

$$a_3 C1 C23 + a_2 C1 C2 = P_x - d_3 S1 - a_1 C1 \quad 3-72$$

$$a_3 S1 C23 + a_2 S1 C2 = P_y + d_3 C1 - a_1 S1 \quad 3-73$$

$$a_3 S23 + a_2 S2 = -P_z \quad 3-74$$

Squaring and adding above relations :

$$(3-72)^2 + (3-73)^2 + (3-74)^2 \Rightarrow$$

$$a_3^2 + a_2^2 + 2 a_2 a_3 C3 = P_z^2 + (P_x - d_3 S1 - a_1 C1)^2 + (P_y + d_3 C1 - a_1 S1)^2 \quad 3-75$$

$C3$ and $S3$ can be found as

$$C3 = \frac{P_z^2 + (P_x - d_3 S1 - a_1 C1)^2 + (P_y + d_3 C1 - a_1 S1)^2 - a_3^2 - a_2^2}{2 a_2 a_3} \quad 3-76$$

$$S3 = \pm \sqrt{1 - C3^2} \quad 3-77$$

Because the right knee joint can only rotate backward and there is joint movement limitation, the knee joint angle should conform the following constraint:

$$0 \leq \theta_3 \leq \frac{\pi}{2} \quad 3-78$$

So $S3$ should be negative then

$$\theta_3 = \text{Atan2}(\sqrt{1 - C3^2}, C3) \quad 3-79$$

To find θ_2 for right leg, relations (3 – 72) and (3 – 73) are squared and added and reordered as:

$$\begin{aligned} (3 - 72)^2 + (3 - 73)^2 &\Rightarrow \\ (a_3 C23 + a_2 C2)^2 &= (P_x + d_3 S1 - a_1 C1)^2 + (P_y - d_3 C1 - a_1 S1)^2 \\ \Rightarrow C2(a_3 C3 + a_2) - S2(a_3 S3) &= \pm \sqrt{(P_x - d_3 S1 - a_1 C1)^2 + (P_y + d_3 C1 - a_1 S1)^2} \end{aligned} \quad 3-80$$

$$(3 - 74) \Rightarrow S2(a_3 C3 + a_2) + C2(a_3 S3) = -P_z \quad 3-81$$

Let define r and φ as

$$a_3 C3 + a_2 = r C\varphi \quad 3-82$$

$$a_3 S3 = r S\varphi \quad 3-83$$

Substituting into above equations we have

$$\Rightarrow \begin{cases} r C\varphi C2 - r S\varphi S2 = \pm \sqrt{(P_x - d_3 S1 - a_1 C1)^2 + (P_y + d_3 C1 - a_1 S1)^2} \\ r C\varphi S2 + r S\varphi C2 = -P_z \end{cases} \quad 3-84$$

$$\Rightarrow \begin{cases} r \cos(\varphi + \theta_2) = \pm \sqrt{(P_x - d_3 S1 - a_1 C1)^2 + (P_y + d_3 C1 - a_1 S1)^2} \\ r \sin(\varphi + \theta_2) = -P_z \end{cases} \quad 3-85$$

$$\varphi = \text{Atan2}(a_3 S3, a_3 C3 + a_2) \quad 3-86$$

$$\theta_2 + \varphi = \text{Atan2}(-P_z, \pm\sqrt{(P_x - d_3 S1 - a_1 C1)^2 + (P_y + d_3 C1 - a_1 S1)^2}) \quad 3-87$$

Because $0 \leq \theta_3 \leq \frac{\pi}{2}$ and $a_3, a_2 > 0$ then

$$0 \leq \varphi \leq \frac{\pi}{2} \quad 3-88$$

Also from movement limitation of the lateral ankle joint it is known that

$$-\frac{\pi}{3} \leq \theta_2 \leq \frac{\pi}{3} \quad 3-89$$

Thus

$$-\frac{\pi}{3} \leq \theta_2 + \varphi \leq \frac{\pi}{3} + \frac{\pi}{2} \quad 3-90$$

So choosing positive sign for $\cos(\varphi + \theta_2)$ leads to a satisfactory result.

$$\theta_2 = \text{Atan2}(-P_z, \sqrt{(P_x - d_3 S1 - a_1 C1)^2 + (P_y + d_3 C1 - a_1 S1)^2}) - \varphi \quad 3-91$$

b. Inverse orientation

Like left leg, here inverse orientation of right leg are developed to find the next three joints angles. We have the same formula

$$R_6^3 = R_3^{0T} R_6^0 \quad 3-92$$

And also by substitution of values of $\theta_1, \theta_2, \theta_3$ calculated from previous section, R_3^0 can be found from forward kinematics of the right leg as follows:

$$R_3^0 = \begin{bmatrix} C1 C23 & -C1 S23 & -S1 \\ S1 C23 & -S1 S23 & C1 \\ -S23 & -C23 & 0 \end{bmatrix} \quad 3-93$$

Let

$$R_3^{0T} R_6^0 = R' = \begin{bmatrix} r'_{11} & r'_{12} & r'_{13} \\ r'_{21} & r'_{22} & r'_{23} \\ r'_{31} & r'_{32} & r'_{33} \end{bmatrix} \quad 3-94$$

which is now completely known.

From forward kinematics of the right leg also R_6^3 are formulated as

$$R_6^3 = \begin{bmatrix} C4 C5 C6 - S4 S6 & -C4 C5 S6 - S4 C6 & -C4 S5 \\ S4 C5 C6 + C4 S6 & -S4 C5 S6 + C4 C6 & -S4 S5 \\ S5 C6 & -S5 S6 & C5 \end{bmatrix} \quad 3-95$$

First suppose that not both of r'_{13} and r'_{23} are zero, then one can conclude that $S5 \neq 0$ and hence that not both of r'_{31} and r'_{32} are zero. Thus $r'_{33} \neq \pm 1$ and we have

$$C5 = r'_{33} \quad 3-96$$

$$S5 = \pm\sqrt{1 - r'_{33}} \quad 3-97$$

Due to the joint movement limitation in the joint 5, only positive values of S5 are acceptable.

$$\theta_5 = \text{Atan2}(\sqrt{1 - r'_{33}}, r'_{33}) \quad 3-98$$

$$\theta_4 = \text{Atan2}(-r'_{23}, -r'_{13}) \quad 3-99$$

$$\theta_6 = \text{Atan2}(-r'_{32}, r'_{31}) \quad 3-100$$

Another case is when $r'_{13} = r'_{23} = 0$, then the fact that R' is orthogonal implies that

$$r'_{33} = \pm 1 \quad 3-101$$

$$r'_{31} = r'_{32} = 0 \quad 3-102$$

So in this case $\theta_5 = 0$ or π which are both out of the joint limit and are not acceptable results.

3.3 Inverse kinematics validation

In this section, the validation of the inverse kinematic relations derived in the previous section is verified. The verification of the inverse kinematic validity let us use them in controller design with complete reliability. As it is clear, the inverse kinematics are the inverse of the forward kinematic relations. Therefore in order to test the correctness of this inversion, the joints angles are selected to be any random periodic functions as an input trajectories discretized in equal sample intervals to the forward kinematics. Then the output of the forward kinematics relevant to the selected joint angle trajectories, is the transformation matrix of the frame {6} with respect to the frame {0} for both legs, including the position and orientation. Now the transformation matrix is substituted in the inverse kinematic relations. Then the joint angles resulted from the inverse kinematics are compared to the input joint angles selected as a random periodic function via visual drawing for all sample times. Fig. 3.7 shows the schematic diagram of the inverse kinematic validation test procedure.

The values of the link dimensions used in the kinematic relations of Archie are as follows (see Fig. 3.1 and 3.2) :

$$a_1 = 8 \text{ cm}, a_2 = 26 \text{ cm}, a_3 = 30 \text{ cm}, a_6 = 5.6 \text{ cm}, d_3 = 7.4 \text{ cm}, d_6 = 4 \text{ cm}$$

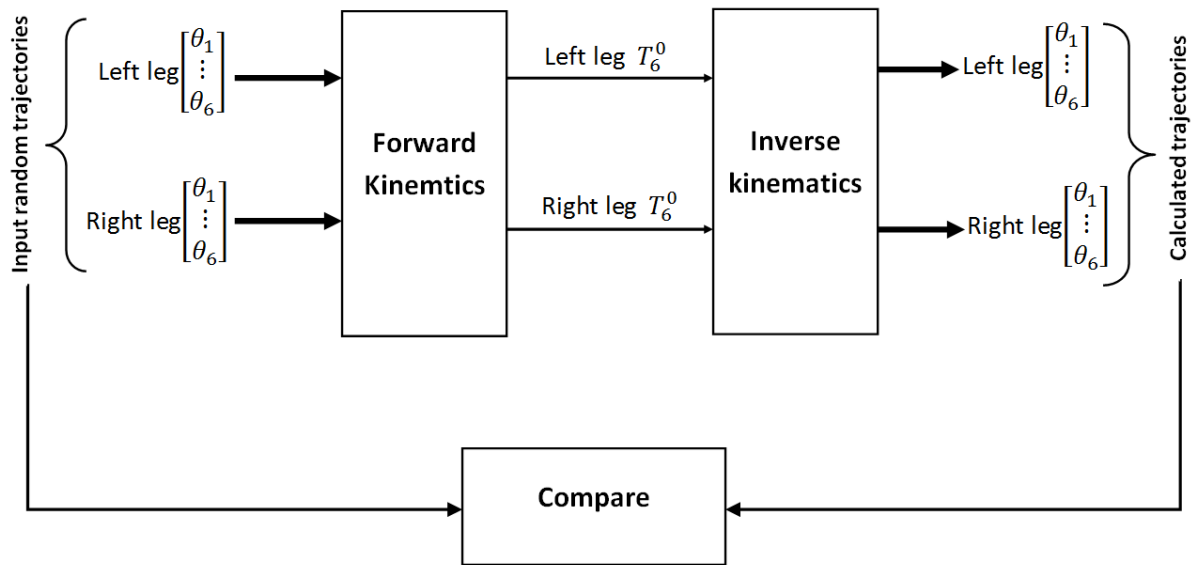


Fig. 3.7 Schematic diagram for verification of inverse kinematics validation

Fig. 3.8 to 3.13 shows the actual and calculated angle trajectories sampled in 200 points for θ_1 to θ_6 of the right leg, respectively. The actual and the calculated trajectories for all joints are completely match with each other. This implies that for any random signal, the inverse kinematic relation derived for right leg is a valid inverse of forward kinematics. Therefore, it is reliable to use inverse kinematic relations in our controller to convert the desired trajectories from the operational space to the joint space.

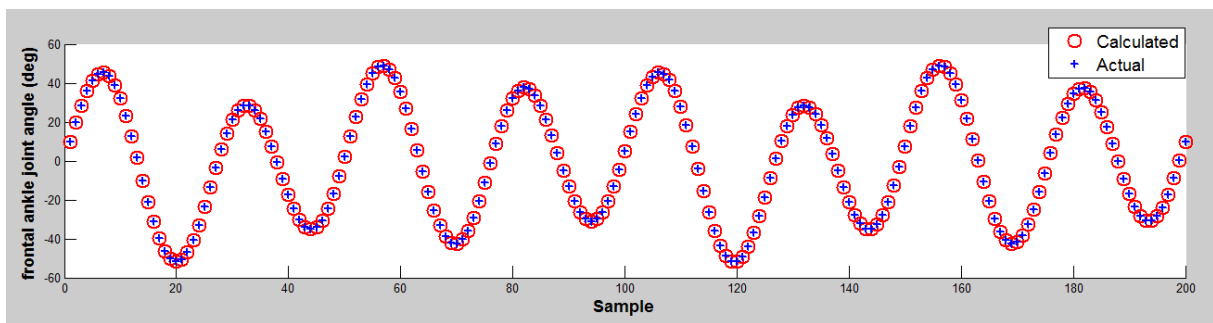


Fig. 3.8 Comparison between actual and calculated joint angle of the right frontal ankle

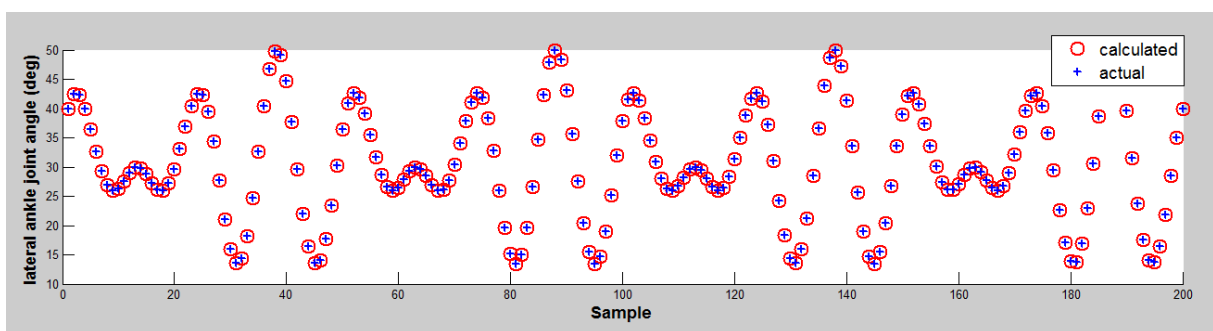


Fig. 3.9 Comparison between actual and calculated joint angle of the right lateral ankle

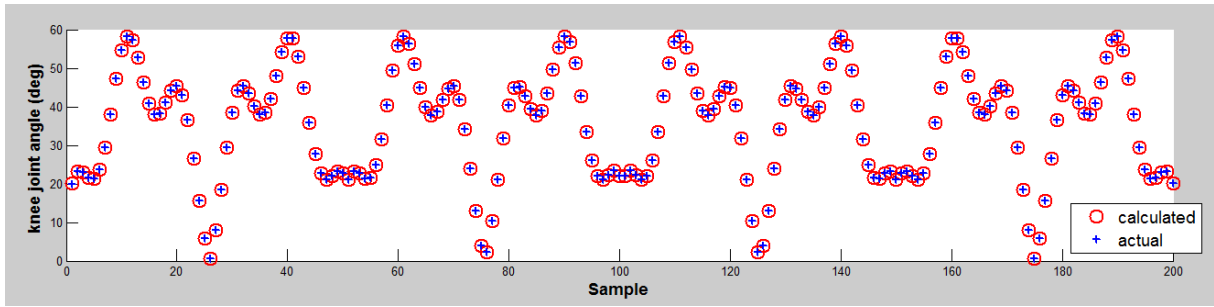


Fig. 3.10 Comparison between actual and calculated joint angle of the right knee

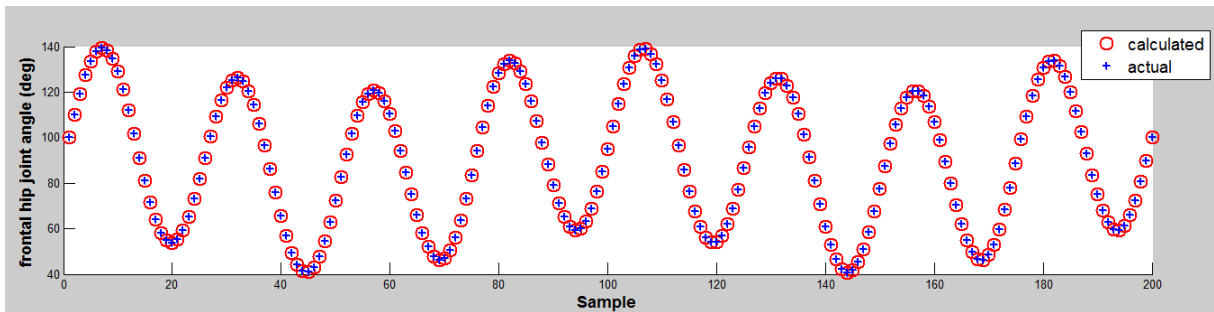


Fig. 3.11 Comparison between actual and calculated joint angle of the right frontal hip

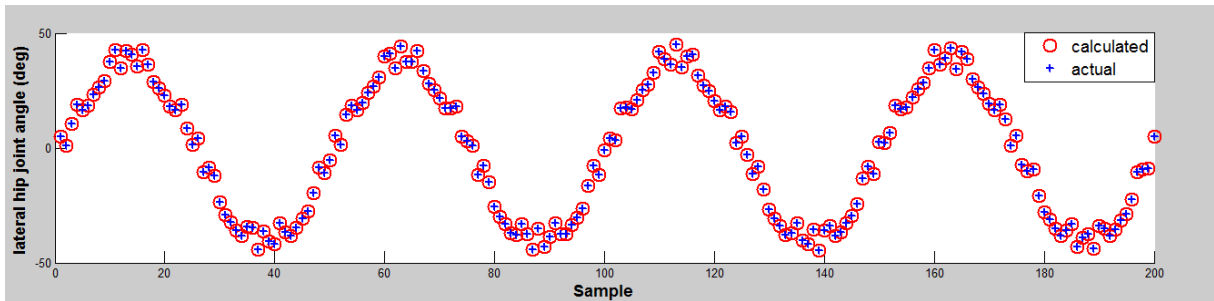


Fig. 3.12 Comparison between actual and calculated joint angle of the right lateral hip

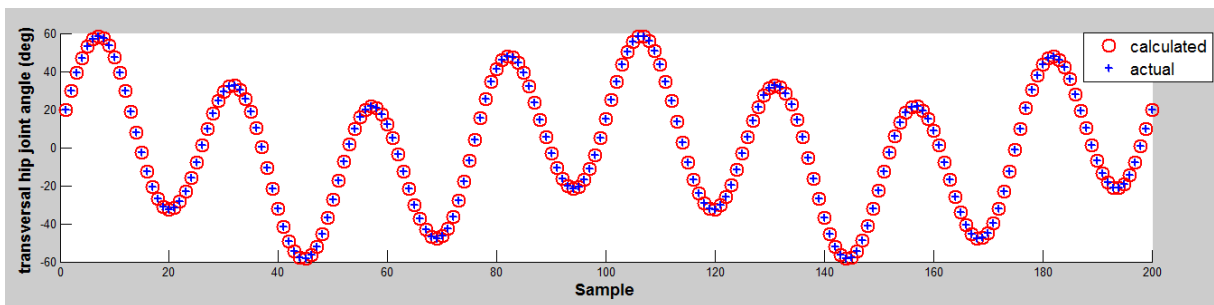


Fig. 3.13 Comparison between actual and calculated joint angle of the right transversal hip

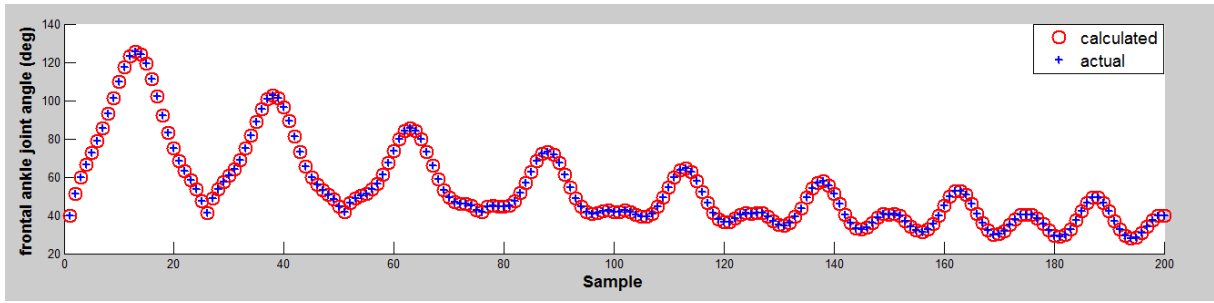


Fig. 3.14 Comparison between actual and calculated joint angle of the left frontal ankle

Similarly, Fig. 3.14 to 3.19 show the actual and calculated joint angles for the left leg. As it is clear in these figures, the random joint angles substituted in forward kinematics are matched in all points with the joint angles calculated from inverse kinematics. So the error between actual and calculated joint angles are zero and the inversion is completely valid. In this manner the correctness of the inverse kinematic relations derived for the left leg is verified.

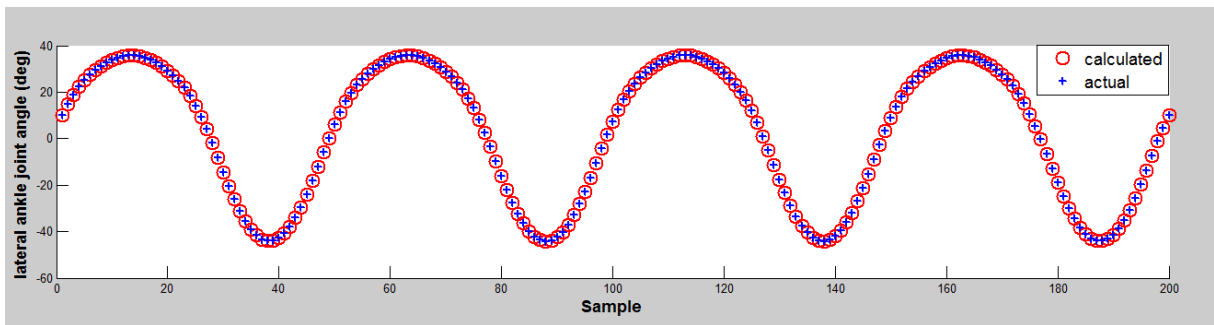


Fig. 3.15 Comparison between actual and calculated joint angle of the left lateral ankle

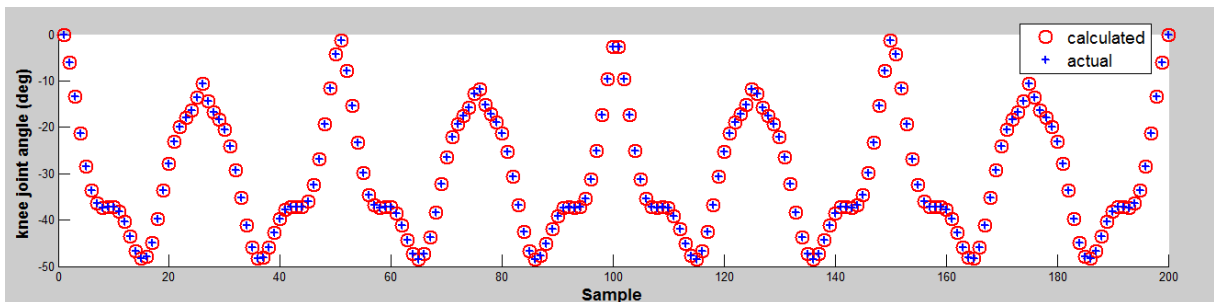


Fig. 3.16 Comparison between actual and calculated joint angle of the left knee

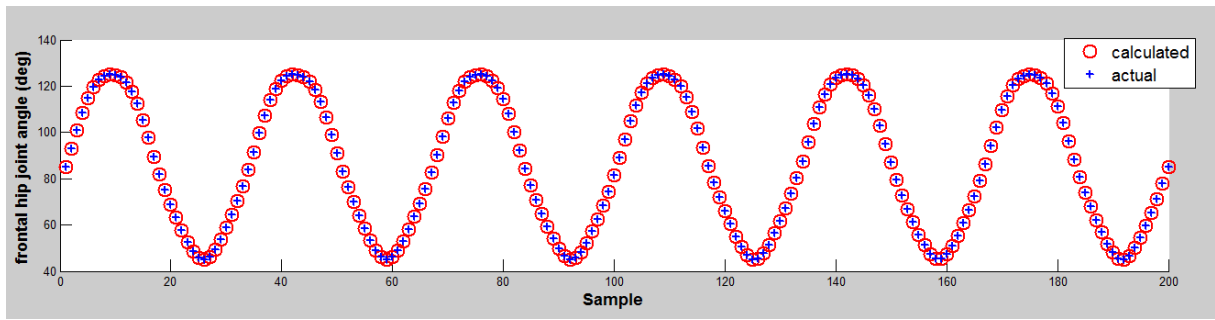


Fig. 3.17 Comparison between actual and calculated joint angle of the left frontal hip

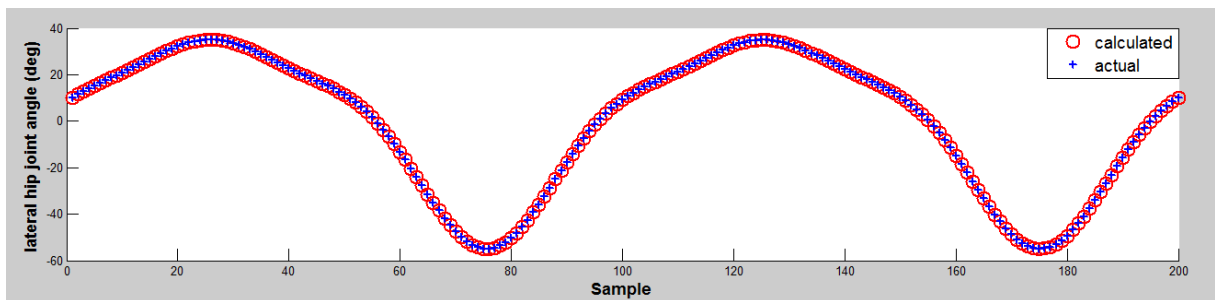


Fig. 3.18 Comparison between actual and calculated joint angle of the left lateral hip

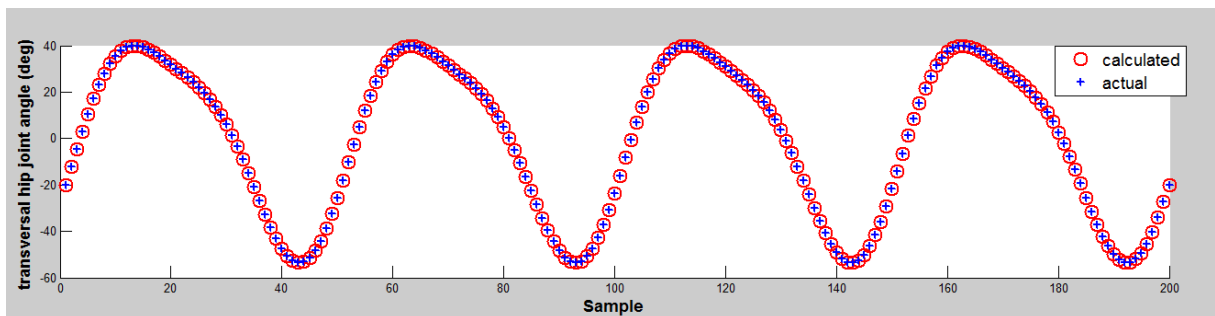


Fig. 3.19 Comparison between actual and calculated joint angle of the left transversal hip

In conclusion, the derived closed-form solution for the inverse kinematics of Archie was verified through a comparison between the selected random signals and the results of inverse and forward kinematics for each joint.

Chapter 4 Human-like walking pattern

In order to realize human-like walking for a biped robot, it is necessary to investigate walking process of human. This investigation leads to find the walking pattern suitable to implement in the biped robot. Many researches have been done based on different approaches to capture gait cycles such as (Pat Murray et al., 1964).

In this chapter, the gait analysis of human walking are reviewed. Then the foot and hip trajectories are generated based on the human-like walking pattern.

4.1 Gait analysis

First of all it is worth to mention some basic definition of gait cycle. For gait analysis, the space are divided into three primary planes: sagittal (lateral), frontal and transversal planes, see Fig. 4.1.

Due to cyclic nature of human gait, it is possible to draw only for one cycle that is repeated over and over, step after step. The assumption that successive cycles are all about the same is a reasonable approximation (Vaughan et al., 1992).

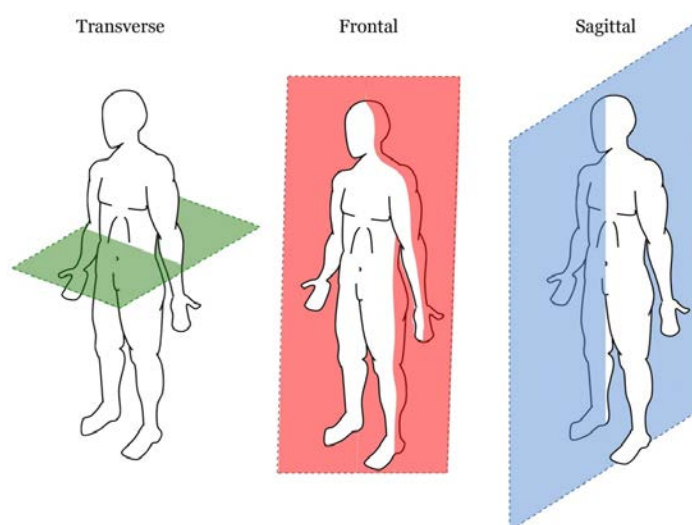


Fig. 4.1 Standard primary plane for the human body movement

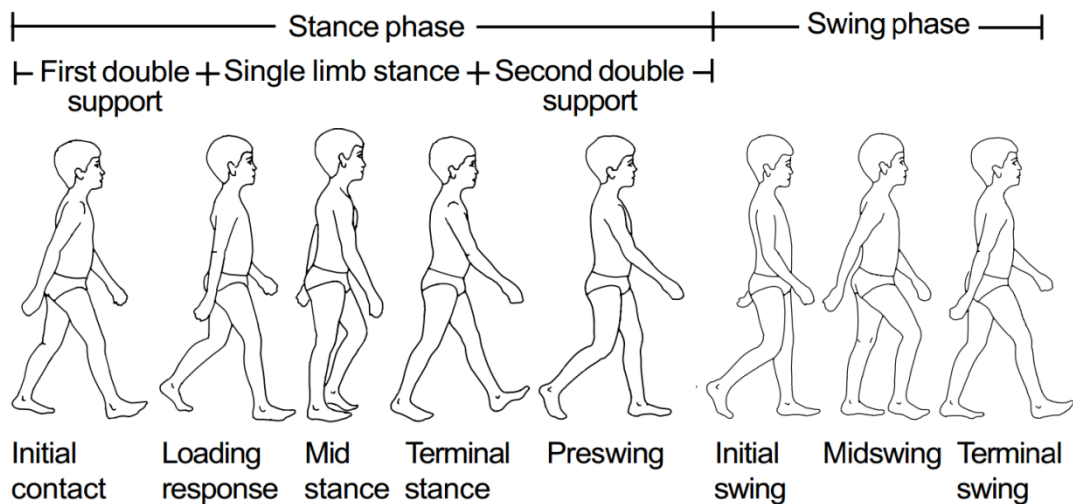


Fig. 4.2 Cyclic pattern of human gait describing eight sequence of walking (Vaughan et al., 1992)

As seen in Fig. 4.2, each foot realize two major phases: during stance phase, the foot is on the ground, whereas in swing phase that same foot is no longer in contact with the ground and the leg is swinging through in preparation for the next foot strike. Stance phase contains two double-support sub phase and one single-support phase. Double-support means both feet are in contact with the ground, whereas in the single-support phase, one foot is in contact with ground and the other is swinging.

Traditionally the gait cycle has been divided into eight events or periods, five during stance phase and three during swing phase, see Fig. 4.3. These events can be described as follows:

1. Heel Strike initiates the gait cycle, it is the moment in time where the heel of the front foot impacts with the ground, and it also represents the point at which the body's center of gravity is at its lowest position.
2. Foot-flat occurs when the leg rolls forward on the foot and the plantar surface of the foot touches the ground.
3. Mid-stance happens when the other foot swings by, the foot in focus is still flat on the ground.
4. Heel-off is a result of the upper body now being in front of the rear foot. Ground contact is maintained by lifting the heel and stand on the toes, in this phase the push-off is also initiated.
5. Toe-off terminates the stance phase as the foot leaves the ground.
6. Acceleration begins at the exact moment in time when the foot is lifted from the ground, and indicates that the foot accelerates forward.
7. Mid-swing occurs when the foot passes directly beneath the body, coincidental with mid-stance for the other foot.

8. Deceleration is the last motion in the swing phase, the swinging is slowed down and the knee is extended, and the foot is stopped just in time to have the proper stride length when heel strike occurs.

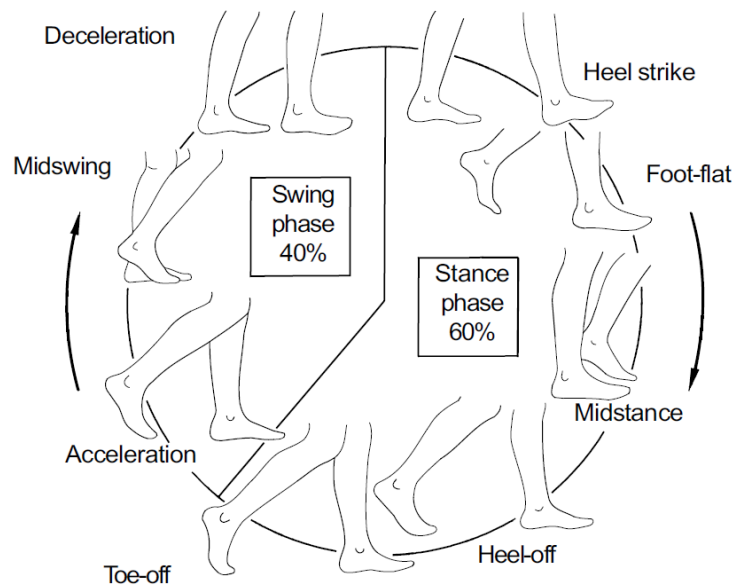


Fig. 4.3 Eight main events of gait cycle (Vaughan et al., 1992)

As shown in Fig. 4.3, The gross normal distribution of the floor contact periods is 60% for stance and 40% for swing. Timing for the phases of stance is 10% for each double-support interval and 40% for single limb support (Perry, 1992).

In order to measure walking movement in each cycle two phrases are important: step and stride length, see Fig. 4.4. Stride length is the distance travelled by a person during one stride (or cycle) and can be measured as the length between the heels from one heel strike to the next heel strike on the same side. Two step lengths (left plus right) make one stride length. Another useful parameter shown in Fig. 4.4 is step width, which is the mediolateral distance between the feet and has a value of a few centimeters for normal human. Finally, the angle of the foot relative to the line of progression can also provide useful information, documenting the degree of external or internal rotation of the lower extremity during the stance phase.

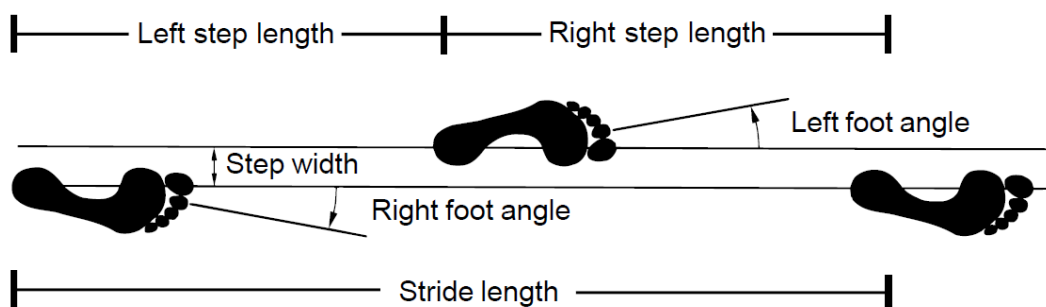


Fig. 4.4 Distance parameters based on footprints (Vaughan et al., 1992)

4.2 Planning walking pattern

In previous section gait analysis for human were reviewed and basic definition useful for pattern generation were presented. In this section, the human-like walking pattern are generated to adapt with different ground constraints. In this manner there are only few parameters like step length, step width, step height and so on to generate suitable walking pattern for biped robot. For example, a biped robot should able to lift its feet high enough to pass obstacles, or keep the support foot with suitable angles to match the roughness of the terrain.

In order to formulate the feet and torso trajectories based on the desired constraints, the method presented in (Huang, Yokoi, & Kajita, 2001) is used. In addition, this method is extend in this section to three dimensional space because the method was only presented for sagittal plane that is not appropriate for 3D walking.

As explained in chapter 3, knowing the foot and torso trajectories, one can find the joint angles based on inverse kinematics. So, The walking pattern can therefore be determined uniquely by the both feet trajectories and the torso trajectory.

4.2.1 Gait cycle

In a sagittal plane, each foot trajectory can be denoted by a vector $[x_a(t), z_a(t), \theta_a(t)]^T$, where $x_a(t), z_a(t)$ is the coordinate of the ankle position in sagittal plane, and $\theta_a(t)$ denotes the angle of the foot. The torso trajectory can be denoted by a vector $[x_h(t), z_h(t), \theta_h(t)]^T$, where $x_h(t), z_h(t)$ denotes the coordinate of the torso position in sagittal plane and $\theta_h(t)$ denotes the angle of the torso, see Fig. 4.5. Furthermore, for specifying 3D trajectories of feet while walking, the side motion of feet and torso $y_a(t), y_h(t)$ should be determined. Therefore, in transversal plane, feet trajectories can be determined by a vector $[x_a(t), y_a(t), \alpha_a(t)]^T$ where $x_a(t), y_a(t)$ is the coordinate of the ankle position in transversal plane, and $\alpha_a(t)$ denotes the angle of the foot in transversal plane. The torso trajectory can be denoted by a vector $[x_h(t), y_h(t), \alpha_h(t)]^T$, where $x_h(t), y_h(t)$ denotes the coordinate of the torso position in transversal plane and $\alpha_h(t)$ denotes the angle of the torso in this plane. Similarly, $[y_a(t), z_a(t), \beta_a(t)]^T$ and $[y_h(t), z_h(t), \beta_h(t)]^T$ are the vector determine the coordinate and the angle of the feet and the torso in frontal plane, respectively.

So, first of all, both foot trajectories should be specified to adapt various walking parameters and then torso trajectory are formulated with regards to stability criteria.

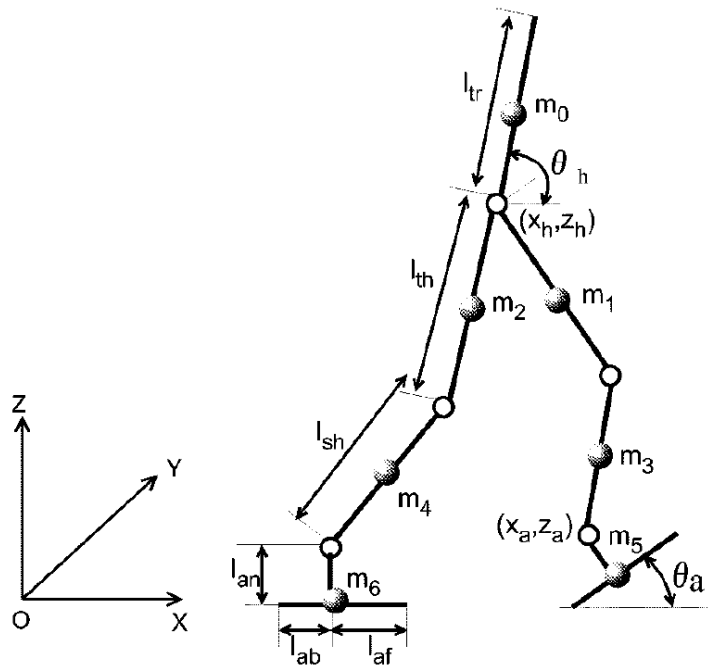


Fig. 4.5 Model of biped robot in sagittal plane (Huang et al., 2001)

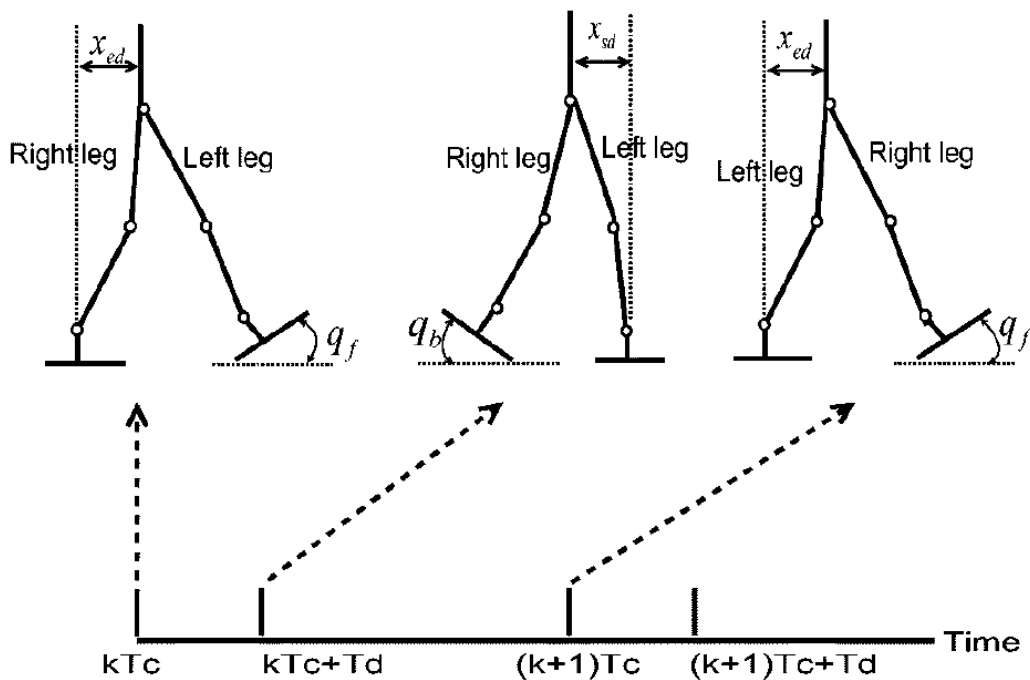


Fig. 4.6 Walking cycle (Huang et al., 2001)

As shown in Fig. 4.6 , if we assume that the time necessary for one walking step is T_c , the time of the k th step is from kT_c to $(k + 1)T_c$, $k = 0,1,2, \dots, k$ where k is the number of steps. The k th begins with the heel of the right foot leaving the ground at $t = kT_c$ and ends with the heel of the right foot making first contact with the ground $t = (k + 1)T_c$. T_d is the interval of the double-support phase. Based on the gait analysis of human, T_d is selected as

20% of T_c . Because walking is a periodic phenomenon, the feet trajectories can be formulated from $t = 0$ to $t = 2T_c$, i.e. start from left heel contact with the ground and end to the next left heel contact with the ground. Then these trajectories repeats till next changes in the walking parameters.

4.2.2 Foot trajectories

Unlike some researches that assume the feet are always level with the ground (i.e. $\theta_a(t) = 0$), for smooth walking, in this research, it is mentioned that the robot can touch the ground first by the heel of the forward foot and leave the ground finally by the toe of the rear foot. This is more compatible with natural human locomotion.

As shown in Fig. 4.7, Let q_b and q_f be the desired angles of the right or left foot as it leaves and lands on the ground, respectively. Assuming that the entire sole surface of the right foot is in contact with the ground at $t = 0$ and $t = 2T_c$, we have the following constraints:

$$\begin{cases} \theta_a(0) = 0 \\ \theta_a(T_d) = q_b \\ \theta_a(T_c) = -q_f \\ \theta_a(T_c + T_d) = 0 \\ \theta_a(2T_c) = 0 \end{cases} \quad 4-1$$

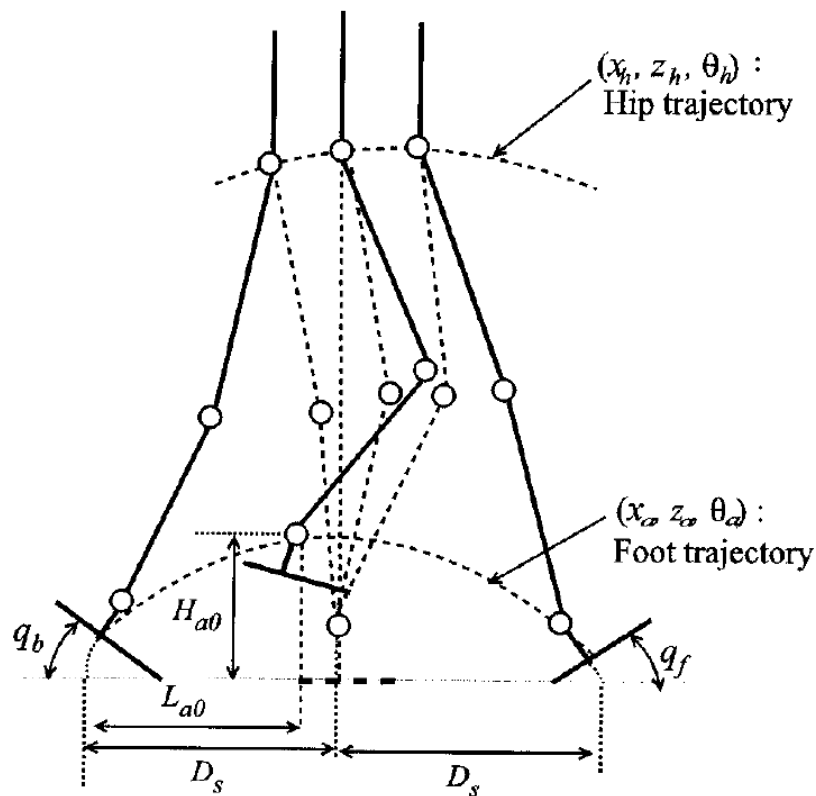


Fig. 4.7 Walking parameters in the sagittal plane (Huang et al., 2001)

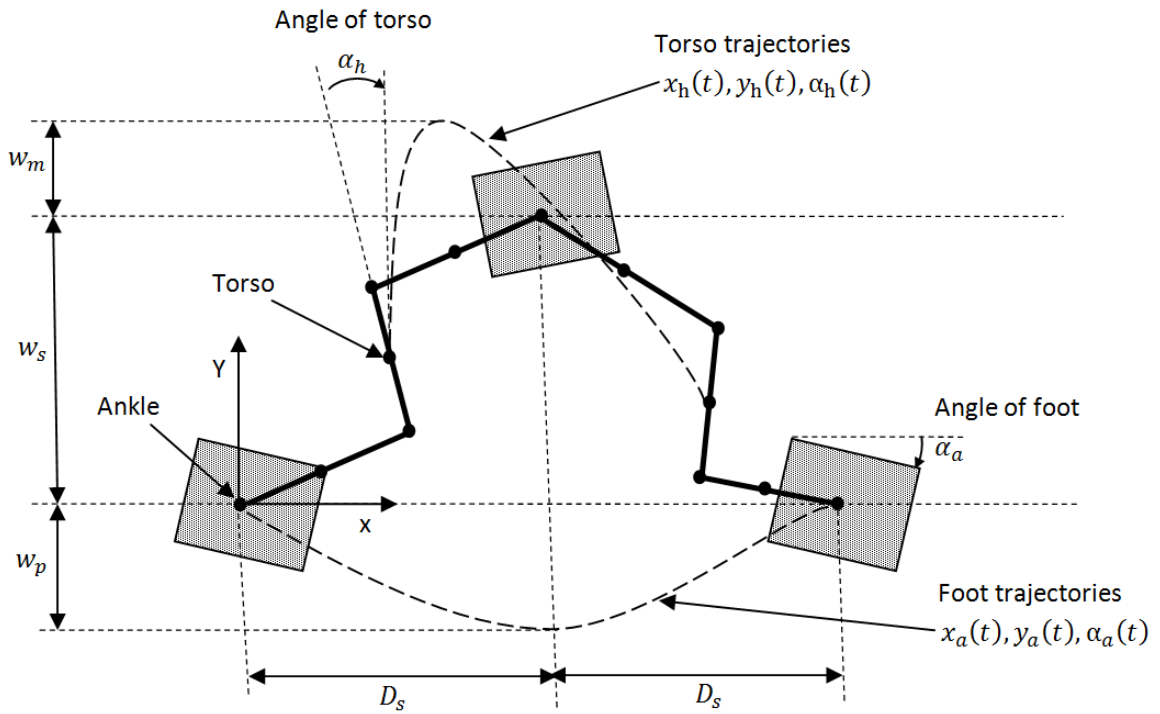


Fig. 4.8 Walking parameters in the transversal plane (top view)

It is worth to mention that the right foot is a single support at $t = T_c + T_d$, thus it is level with the ground. Similarly, For left leg we have:

$$\begin{cases} \theta_a(0) = -q_f \\ \theta_a(T_d) = 0 \\ \theta_a(T_c) = 0 \\ \theta_a(T_c + T_d) = q_b \\ \theta_a(2T_c) = -q_f \end{cases} \quad 4-2$$

Without loss of generality, the trajectory of $\alpha_a(t)$ and $\beta_a(t)$ are assumed to be constant during walking. In other case, the constraints like θ_a can be introduced for varying trajectory of $\alpha_a(t)$ and $\beta_a(t)$ during walking.

Accordingly, constraints due to the contact of the right foot with the ground for $x_a(t)$, $z_a(t)$ and $y_a(t)$ can be defined as follows:

$$\left\{ \begin{array}{l} x_a(0) = 0 \\ x_a(T_d) = l_{an} \sin q_b + l_{af}(1 - \cos q_b) \\ x_a(T_o) = l_{ao} \\ x_a(T_c) = 2D_s - l_{an} \sin q_f - l_{ab}(1 - \cos q_f) \\ x_a(T_c + T_d) = 2D_s \\ x_a(2T_c) = 2D_s \end{array} \right. \quad 4-3$$

$$\left\{ \begin{array}{l} z_a(0) = l_{an} \\ z_a(T_d) = l_{af} \sin q_b + l_{an} \cos q_b \\ z_a(T_o) = H_{ao} \\ z_a(T_c) = l_{ab} \sin q_f + l_{an} \cos q_f \\ z_a(T_c + T_d) = l_{an} \\ z_a(2T_c) = l_{an} \end{array} \right. \quad 4-4$$

$$\left\{ \begin{array}{l} y_a(0) = 0 \\ y_a(T_d) = 0 \\ y_a\left(\frac{T_c+T_d}{2}\right) = -w_p \\ y_a(T_c) = 0 \\ y_a(T_c + T_d) = 0 \\ y_a(2T_c) = 0 \end{array} \right. \quad 4-5$$

And for the left foot we have:

$$\left\{ \begin{array}{l} x_a(0) = D_s - l_{an} \sin q_f - l_{ab}(1 - \cos q_f) \\ x_a(T_d) = D_s \\ x_a(T_c) = D_s \\ x_a(T_c + T_d) = D_s + l_{an} \sin q_b + l_{af}(1 - \cos q_b) \\ x_a(T_c + T_o) = D_s + l_{ao} \\ x_a(2T_c) = 3D_s - l_{an} \sin q_f - l_{ab}(1 - \cos q_f) \end{array} \right. \quad 4-6$$

$$\left\{ \begin{array}{l} z_a(0) = l_{ab} \sin q_f + l_{an} \cos q_f \\ z_a(T_d) = l_{an} \\ z_a(T_c) = l_{an} \\ z_a(T_c + T_d) = l_{af} \sin q_b + l_{an} \cos q_b \\ z_a(T_c + T_o) = H_{ao} \\ z_a(2T_c) = l_{ab} \sin q_f + l_{an} \cos q_f \end{array} \right. \quad 4-7$$

$$\left\{ \begin{array}{l} y_a(0) = w_s \\ y_a(T_d) = w_s \\ y_a(T_c) = w_s \\ y_a(T_c + T_d) = w_s \\ y_a\left(T_c + \frac{T_c + T_d}{2}\right) = w_s + w_p \\ y_a(2T_c) = w_s \end{array} \right. \quad 4-8$$

where (H_{ao}, l_{ao}) is the position of the highest point of swing foot to pass the obstacle (Fig. 4.7), D_s is the length of one step, T_o and $T_c + T_o$ are the time when right and left foot are at their highest point, respectively, l_{an} is the height of the foot, l_{af} is the length from the ankle joint to the toe, l_{ab} is the length from the ankle joint to the heel (Fig. 4.5), w_s is the step width, w_p is the maximum side movement of the swing foot which occurs at the middle of the swing phase, i.e. $t = (T_c + T_d)/2$ (Fig. 4.8).

When the entire sole of foot is in contact with the ground, the derivative of position and angle trajectory should be zero. So for the right foot following constraints should be satisfied:

$$\left\{ \begin{array}{l} \dot{\theta}_a(0) = 0 \\ \dot{\theta}_a(T_c + T_d) = 0 \\ \dot{\theta}_a(2T_c) = -q_f \\ \dot{x}_a(0) = 0 \\ \dot{x}_a(T_c + T_d) = 0 \\ \dot{x}_a(2T_c) = 0 \\ \dot{z}_a(0) = 0 \\ \dot{z}_a(T_c + T_d) = 0 \\ \dot{z}_a(2T_c) = 0 \end{array} \right. \quad 4-9$$

Similarly for the left foot:

$$\left\{ \begin{array}{l} \dot{\theta}_a(T_d) = 0 \\ \dot{\theta}_a(T_c) = 0 \\ \dot{x}_a(T_d) = 0 \\ \dot{x}_a(T_c) = 0 \\ \dot{z}_a(T_d) = 0 \\ \dot{z}_a(T_c) = 0 \end{array} \right. \quad 4-10$$

To derive a smooth trajectory, it is vital that the first derivative ,i.e. velocity, terms \dot{x}_a, \dot{z}_a , and $\dot{\theta}_a$ be differentiable and the second derivative ,i.e. acceleration, terms \ddot{x}_a, \ddot{z}_a , and $\ddot{\theta}_a$ be continuous at all t, including all breakpoints.

There are two important methods to construct trajectories that satisfy constraints and the continuity conditions of the first derivative and the second derivative; polynomial and cubic spline interpolation. The drawback of the polynomial interpolation for such a problem is that the order of the polynomial will be too high and its computation time is long. Thus cubic spline interpolation is used to obtain smooth trajectories. In this method, $x_a(t), z_a(t), y_a(t)$ and θ_a are characterized by third-order polynomial expressions, and the second derivative \ddot{x}_a, \ddot{z}_a and $\ddot{\theta}_a$ are always continuous. In order to control more on the shape of trajectories between two arbitrary breakpoints, other constraints can be added in this interval. In this case, the order of the polynomial in this interval are increased by the number of the added constraints.

The most common piecewise-polynomial approximation uses cubic polynomials between each successive pair of nodes and is called cubic spline interpolation. A general cubic polynomial involves four constants, so there is sufficient flexibility in the cubic spline procedure to ensure that the interpolant is not only continuously differentiable on the interval, but also has a continuous second derivative (Burden & Faires, 2005).

After constructing cubic spline derived from constraints, both foot trajectories are formulated based on the walking parameters $D_s, q_b, q_f, w_s, w_p, l_{ao}$ and H_{ao} and the time parameters T_c, T_d and T_o . Therefore, by varying the values of these parameters, different foot trajectories can be easily obtained.

4.2.3 Torso trajectories

Up to now, the foot trajectories are derived with defined constraints. In this section, the torso trajectories are formulated based on the physical constraints. Usually $\theta_h(t)$ that denotes the angle of the torso in sagittal plane is selected as constant in walking to maintain the stability of robot. The common value is $\theta_h(t) = 90$ degrees. Similarly, $\alpha_h(t)$ and $\beta_h(t)$ which are the angles of the torso in transversal and frontal plane, respectively, are assumed to be constant and in special case equal to zero during walking.

Motion of the torso in the vertical direction z_h does not affect directly on the stability of the robot. So in order to match with human walking, a trajectory is specified that varies within a fixed range. Assuming that the hip is at its highest position H_{max} at time T_{max} that is usually selected at the middle of single-support phase, and at its lowest position H_{min} at time T_{min} that is usually selected at the middle of double-support phase. To control more on the shape of trajectory, it is desirable to specify position of H_0 for $t = 0$ and $t = T_c$. So z_h has the following constraints:

$$\left\{ \begin{array}{l} z_h(0) = H_0 \\ z_h(T_{min}) = H_{min} \\ z_h(T_{max}) = H_{max} \\ z_h(T_c) = H_0 \\ z_h(T_c + T_{min}) = H_{min} \\ z_h(T_c + T_{max}) = H_{max} \\ z_h(2T_c) = H_0 \end{array} \right. \quad 4-11$$

where $0 < T_{min} < T_{max} < T_c$. Using cubic spline interpolation, the trajectory of z_h is constructed such that satisfy above constraints and the second derivative continuity.

The most important factor that affects the stability of biped robot walking in sagittal plane is x_h . To define constraints for x_h , two important parameters are defined; x_{sd} and x_{ed} that represent distance along the x-axis from the hip to the ankle of the support foot at the start and end of the single-support phase, respectively (see Fig. 4.6). so we have:

$$\left\{ \begin{array}{l} x_h(0) = x_{ed} \\ x_h(T_d) = D_s - x_{sd} \\ x_h(T_c) = D_s + x_{ed} \\ x_h(T_c + T_d) = 2D_s - x_{sd} \\ x_h(2T_c) = 2D_s + x_{ed} \end{array} \right. \quad 4-12$$

To construct a smooth trajectory for x_h , the formulas presented in (Huang et al., 2001) are used. This formulas were derived using the third-order periodic spline interpolation.

So we have

$$x_h = \frac{D_s - x_{ed} - x_{sd}}{T_d^2(T_c - T_d)} [(T_d - t)^3 - t^3 - T_d^2(T_d - t) + T_d^2 \cdot t] + \frac{x_{ed}}{T_d}(T_d - t) + \frac{D_s - x_{sd}}{T_d} \cdot t \quad 0 < t < T_d$$

4-13

$$x_h = \frac{D_s - x_{ed} - x_{sd}}{T_d(T_c - T_d)^2} [(t - T_d)^3 - (T_c - t)^3 + (T_c - T_d)^2 \cdot (T_c - t) - (T_c - T_d)^2 \cdot (t - T_d)] \\ + \frac{D_s - x_{sd}}{T_c - T_d} \cdot (T_c - t) + \frac{D_s + x_{ed}}{T_c - T_d} \cdot (t - T_d) \quad T_d < t < T_c$$

4-14

$$x_h = D_s + \frac{D_s - x_{ed} - x_{sd}}{T_d^2(T_c - T_d)} [(T_d + T_c - t)^3 - (t - T_c)^3 - T_d^2(T_d + T_c - t) + T_d^2(t - T_c)] \\ + \frac{x_{ed}}{T_d} (T_d + T_c - t) + \frac{D_s - x_{sd}}{T_d} (t - T_c) \quad T_c < t < T_c + T_d$$

4-15

$$x_h = D_s + \frac{D_s - x_{ed} - x_{sd}}{T_d(T_c - T_d)^2} [(t - T_c - T_d)^3 - (2T_c - t)^3 + (T_c - T_d)^2 \cdot (2T_c - t) \\ - (T_c - T_d)^2 \cdot (t - T_c - T_d)] + \frac{D_s - x_{sd}}{T_c - T_d} \cdot (2T_c - t) + \frac{D_s + x_{ed}}{T_c - T_d} \cdot (t \\ - T_c - T_d) \quad T_c + T_d < t < 2T_c$$

4-16

The side movement of torso y_h is very important for the stability of biped robot. In order to maintain stability of robot, the robot's center of gravity, in both case of static and dynamic walking, must be transferred from the rear foot to the front foot during the short double-support phase. This ability is added to our robot Archie by additional ankle joint that moves in frontal plane (Dezfouli, 2013). Hence we have the following constraints for y_h :

$$\begin{cases} y_h(0) = -w_m \\ y_h(T_d) = w_s + w_m \\ y_h(T_c) = w_s + w_m \\ y_h(T_c + T_d) = -w_m \\ y_h(2T_c) = -w_m \end{cases} \quad 4-17$$

where w_m is the maximum distance that the torso should pass the support foot to maintain the stability of the robot during the single-support phase, see Fig. 4.8. Based on these constraints, the torso moves in y direction only in double-support phase and then during single-support phase has no side movement. In this manner, with proper selecting the parameter w_m , the moment of the center of gravity cancel the moment of the swing leg around the support area to maintain the stability of the robot. The trajectory of y_h is constructed using cubic spline interpolation to meet the requirements of constraints.

Up to know the position and orientation trajectories of the feet and the torso are designed with respect to the base frame. Therefore, the trajectories of R_0^B , P_0^B , R_6^B and P_6^B are specified for a walking cycle. But as stated in chapter 3, in order to find the inverse kinematics of each leg, the position and orientation of the frame attached to the torso is needed with respect to the frame attached to the foot, i.e. P_6^0 and R_6^0 . Hence in the following, the needed position vector and rotation matrix are derived according to the planned trajectories using geometric relations.

In order to find the rotation matrix R_6^0 , we know from definition of the rotation matrix that

$$R_6^0 = R_B^0 \cdot R_6^B \quad 4-18$$

where

$$R_B^0 = (R_0^B)^T \quad 4-19$$

So R_B^0 can be found as

$$R_6^0 = (R_0^B)^T \cdot R_6^B \quad 4-20$$

Therefore finding the rotation matrix of frame $\{0\}$, attached to the foot, with respect to the base frame, i.e. R_0^B , and also the rotation matrix of frame $\{6\}$, attached to the torso, with respect to the base frame, i.e. R_6^B , leads to find the rotation matrix of the frame $\{6\}$ with respect to the frame $\{0\}$.

Fig. 4.9 shows the coordinates $\{0\}$ and $\{6\}$ which are attached to the left foot and the torso based on the D-H convention presented in chapter 3, respectively. The base frame $\{B\}$ attached to the ground as defined in Fig. 4.5 is also shown in Fig. 4.9. This frame was used to define the trajectories of the feet and the torso during walking.

According to the Fig. 4.9 the frame $\{B\}$ can be rotated as follows to construct the frame $\{0\}$ for the left leg:

First $\pi/2$ deg rotation around the Y_B axis followed by π deg rotation around the new Z axis of the rotated coordinate. Then a rotation of θ_a around the new Y axis. Then we have two rotation of β_a and α_a around the previous fixed Z and X axis, respectively. Now the rotation matrix can be found as:

$$R_0^B = R_{X,\alpha_a} R_{Z,\beta_a} R_{Y,\pi/2} R_{Z,\pi} R_{Y,\theta_a} \quad 4-21$$

It is noted that the rotation matrix of a rotation around a new axis is post-multiplied and conversely the rotation matrix of a rotation around a fixed axis is pre-multiplied in above formula (Spong et al., 2006).

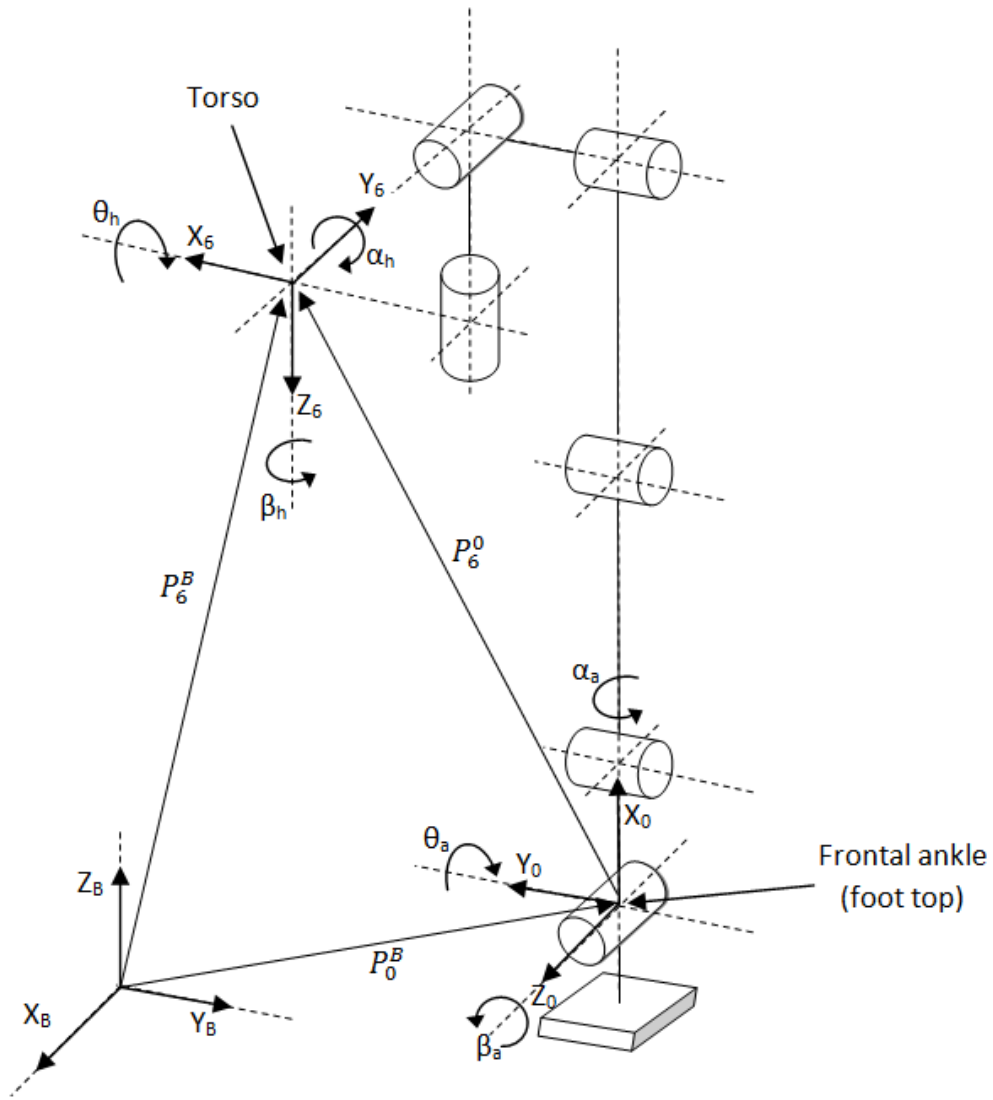


Fig. 4.9 Position and orientation of the frames attached to the left foot and the torso with respect to the base frame

The rotation matrices around principal axes X , Y and Z can be found as:

$$R_{X,\theta} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix} \quad 4-22$$

$$R_{Y,\theta} = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix} \quad 4-23$$

$$R_{Z,\theta} = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad 4-24$$

Substituting into the formula for R_0^B we have:

$$R_0^B = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha_a & -\sin \alpha_a \\ 0 & \sin \alpha_a & \cos \alpha_a \end{bmatrix} \begin{bmatrix} \cos \beta_a & -\sin \beta_a & 0 \\ \sin \beta_a & \cos \beta_a & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix} \\ \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \theta_a & 0 & \sin \theta_a \\ 0 & 1 & 0 \\ -\sin \theta_a & 0 & \cos \theta_a \end{bmatrix}$$

4-25

In special case when $\alpha_a = \beta_a = 0$ then

$$R_0^B = \begin{bmatrix} -\sin \theta_a & 0 & \cos \theta_a \\ 0 & -1 & 0 \\ \cos \theta_a & 0 & \sin \theta_a \end{bmatrix}$$

4-26

Similarly, the frame attached to the torso $\{6\}$ can be constructed from the base frame $\{B\}$ by following rotation matrix:

$$R_6^B = R_{Y,\alpha_h} R_{Z,\beta_h} R_{Z,\frac{\pi}{2}} R_{Y,\pi} R_{X,\theta_h}$$

4-27

$$R_6^B = \begin{bmatrix} \cos \alpha_h & 0 & \sin \alpha_h \\ 0 & 1 & 0 \\ -\sin \alpha_h & 0 & \cos \alpha_h \end{bmatrix} \begin{bmatrix} \cos \beta_h & -\sin \beta_h & 0 \\ \sin \beta_h & \cos \beta_h & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta_h & -\sin \theta_h \\ 0 & \sin \theta_h & \cos \theta_h \end{bmatrix}$$

4-28

For the case when $\theta_h = \alpha_h = \beta_h = 0$ (as selected in planning trajectory) we have

$$R_6^B = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

4-29

Now position of the frame $\{6\}$ stated in the frame $\{0\}$, i.e. P_6^0 , can be found according to Fig. 4.9 for the left leg as

$$P_6^0 = (R_0^B)^T (P_6^B - P_0^B)$$

4-30

where $P_0^B = \begin{bmatrix} x_a(t) \\ y_a(t) \\ z_a(t) \end{bmatrix}$ and $P_6^B = \begin{bmatrix} x_h(t) \\ y_h(t) \\ z_h(t) \end{bmatrix}$ which are the trajectories of the left foot and the torso planned for biped walking.

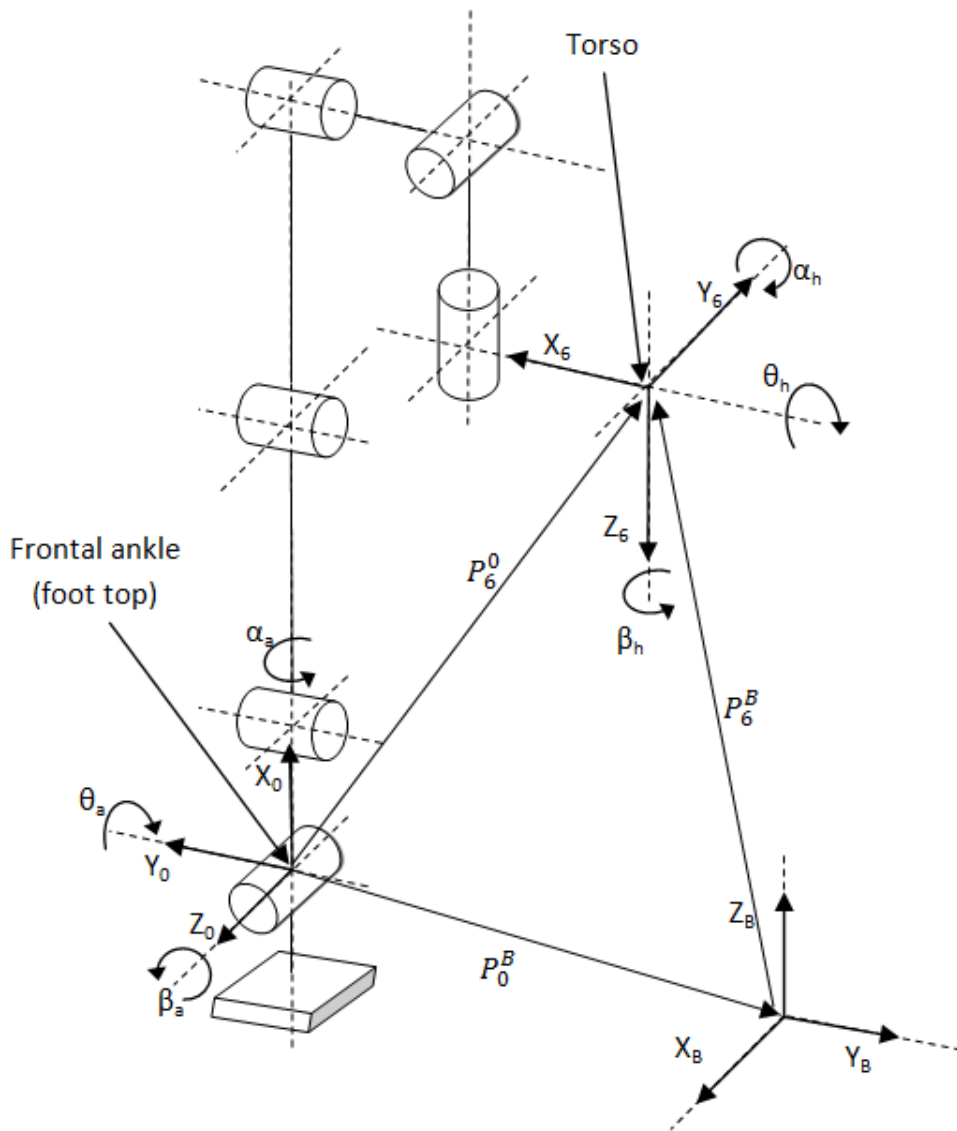


Fig. 4.10 Position and orientation of the frames attached to the right foot and the torso with respect to the base frame

Similarly, according to Fig. 4.10, for the right foot we have the same formula for R_0^B and P_6^0 . The only difference is that the trajectories specified for the right foot should be utilized in the formulas.

The trajectory of angle θ_a for both feet that is specified in previous section, are used in above formula by multiplying by -1 due to different rotation definition.

4.3 Walking trajectories

In this section, the trajectories of the feet and the torso derived in the preceding section are depicted according to the selected walking parameter. Let time parameters are selected as

$T_c = 2 \text{ sec}$, $T_d = 0.2 T_c = 0.4 \text{ sec}$, $T_o = 1.2 \text{ sec}$, $T_{min} = 0.08 \text{ sec}$, $T_{max} = (T_c + T_d)/2 = 1.2 \text{ sec}$. The foot dimension for Archie are $l_{an} = 5.6 \text{ cm}$, $l_{af} = 12.5 \text{ cm}$ and $l_{ab} = 6.5 \text{ cm}$. Let $q_b = 0.2 \text{ rad}$, $q_f = 0.3 \text{ rad}$, $D_s = 20 \text{ cm}$, $l_{ao} = 20 \text{ cm}$, $H_{ao} = 12 \text{ cm}$, $w_s = 17 \text{ cm}$, $w_m = 2 \text{ cm}$, $w_p = 5 \text{ cm}$, $H_{min} = 54 \text{ cm}$, $H_{max} = 57 \text{ cm}$ and $H_0 = 54.2 \text{ cm}$, $x_{ed} = 7 \text{ cm}$, $x_{sd} = 7 \text{ cm}$. The following figures show the trajectories of the feet and the torso according to the mentioned walking parameters.

Fig. 4.11 shows the both foot angle trajectories according to the leaving and landing angles of $q_b = 0.2 \text{ rad}$, $q_f = 0.3 \text{ rad}$, respectively. In order to control the shape of trajectories in specific intervals, other constraints are introduced as follows:

For right foot:

$$\begin{cases} \theta_a(T_1) = 0 \\ \theta_a(T_2) = -q_f \\ \theta_a(T_3) = 0 \end{cases} \quad 4-31$$

And for left foot:

$$\begin{cases} \theta_a(T_1 + T_4) = 0 \\ \theta_a(T_c + T_1) = 0 \\ \theta_a(T_c + T_2) = -q_f \end{cases} \quad 4-32$$

where $T_1 = 0.08 \text{ sec}$, $T_2 = 1.9 \text{ sec}$, $T_3 = 2.28 \text{ sec}$ and $T_4 = 0.2 \text{ sec}$.

The forward movement trajectories of both foot x_a and torso x_h are plotted in Fig. 4.12. As it is evident, all breakpoints are continuous and the curve at both side of breakpoints has the same speed (first derivative). This smooth trajectory guarantee the continuous trajectory of acceleration (second derivative). Therefore, motor is not commanded to produce different acceleration at the short time. This acceleration continuousness guarantees the smooth movement of motor without any impact or shaking.

The average speed of the biped robot walking can be calculated as

$$V = (D_s/T_c) \times 0.036 \text{ km/h} \quad 4-33$$

In the case of mentioned parameters, the average velocity of robot is 10 cm/sec or 0.36 km/h . By change of T_c speed of the biped robot changes without changing in trajectory shape of forward movement. Another way to change the speed is to change step length D_s . It is worth to mention that step length of the robot is limited due to the kinematic constraints of the robot structure.

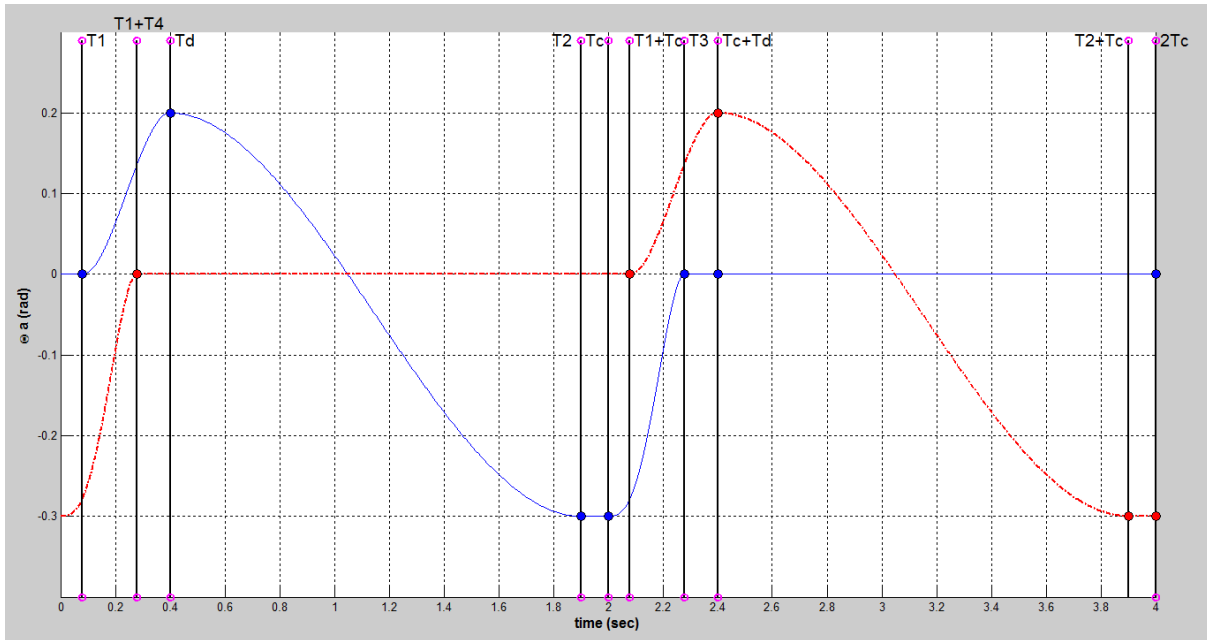


Fig. 4.11 Trajectories of θ_a for right foot (blue) and left foot (red)

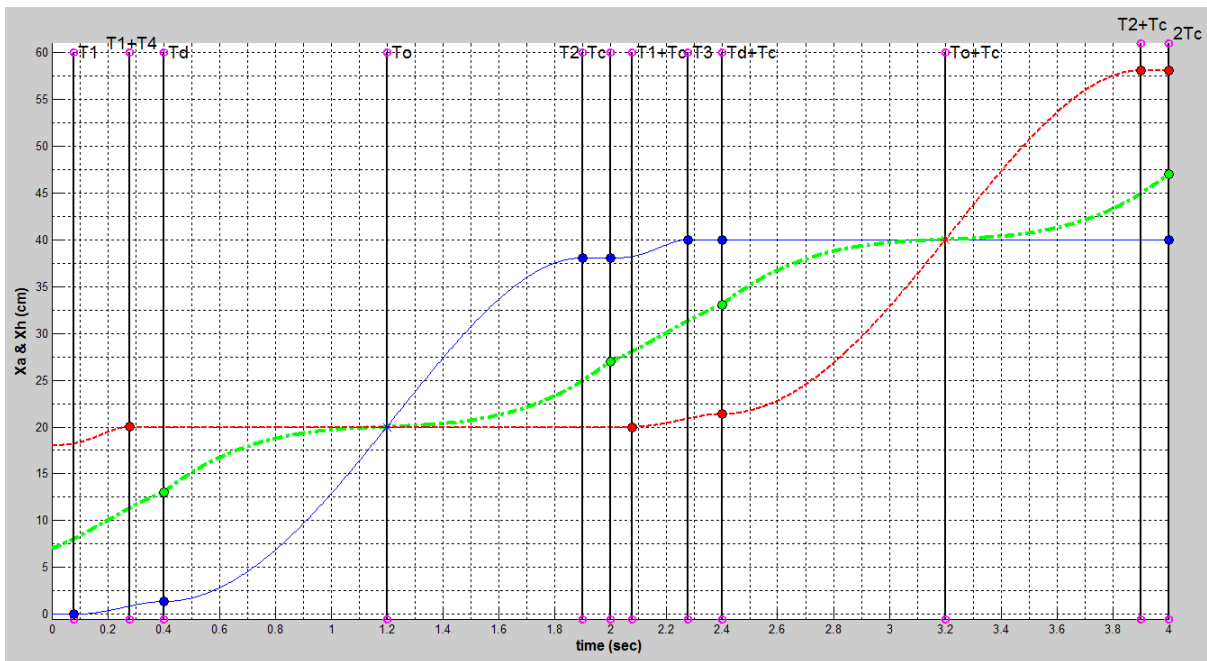


Fig. 4.12 Right foot (blue), left foot (red) and torso (green) trajectories in x direction

Value of x_{ed} and x_{sd} has a great impact on the shape of torso trajectory. In fact, these value determine the position of center of gravity that is located near the torso before and after the single support phase. Thus they play a key role in the stability of the biped walking robot.

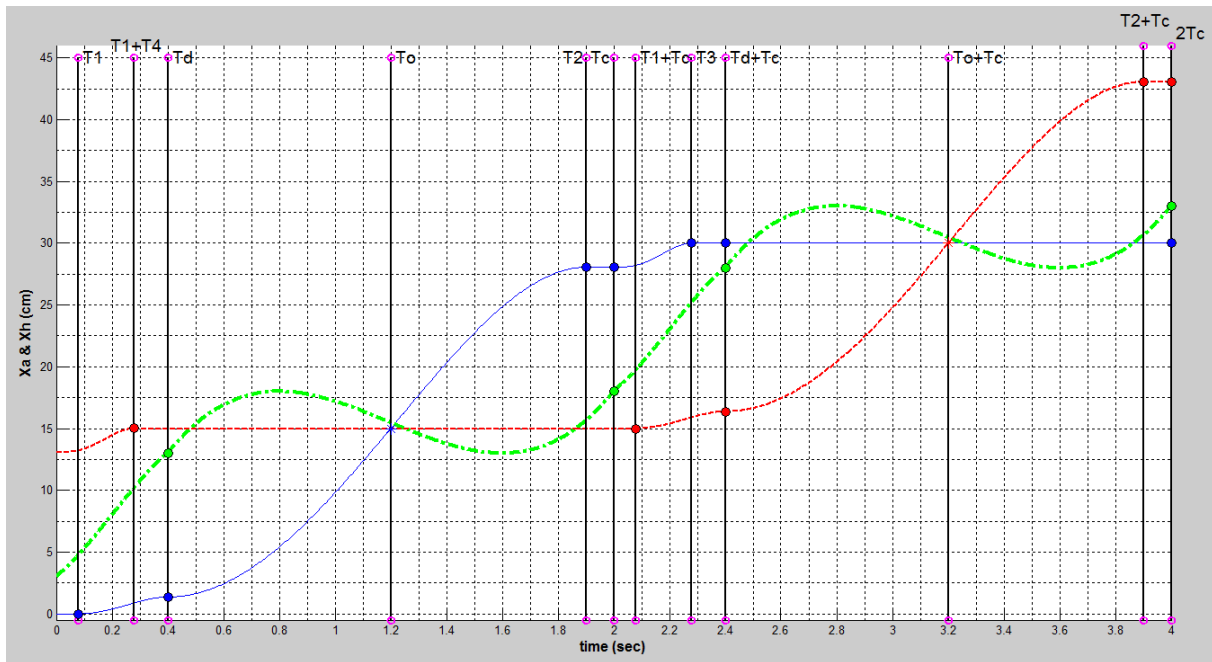


Fig. 4.13 Right foot (blue), left foot(red) and torso (green) trajectories in x direction

As shown in Fig. 4.12, with selected value for $x_{ed} = 7 \text{ cm}$, $x_{sd} = 7 \text{ cm}$ and using the formula proposed in (Huang et al., 2001), the trajectory of x_h is continuous and correct. But for instance, if these values are selected as $x_{ed} = 3 \text{ cm}$, $x_{sd} = 2 \text{ cm}$, then the torso trajectory resulted from such formula has a undesirable overshoot and undershoot in the intervals of $T_d < t < T_c$ and $T_c + T_d < t < 2T_c$, as shown in Fig. 4.13. If this trajectory is commanded to the walking robot, the torso will go forward and backward during the single-support phase that is not acceptable for appropriate walking. In order to overcome this problem, we can use linear interpolation in single-support phase intervals, as seen in Fig. 4.14. Although the acceleration , i.e. second derivative of the trajectory, is no longer continuous, the change of acceleration is negligible for these values. The optimum value of x_{ed} and x_{sd} for stable walking is determined by experimental test as will be discussed in following chapters.

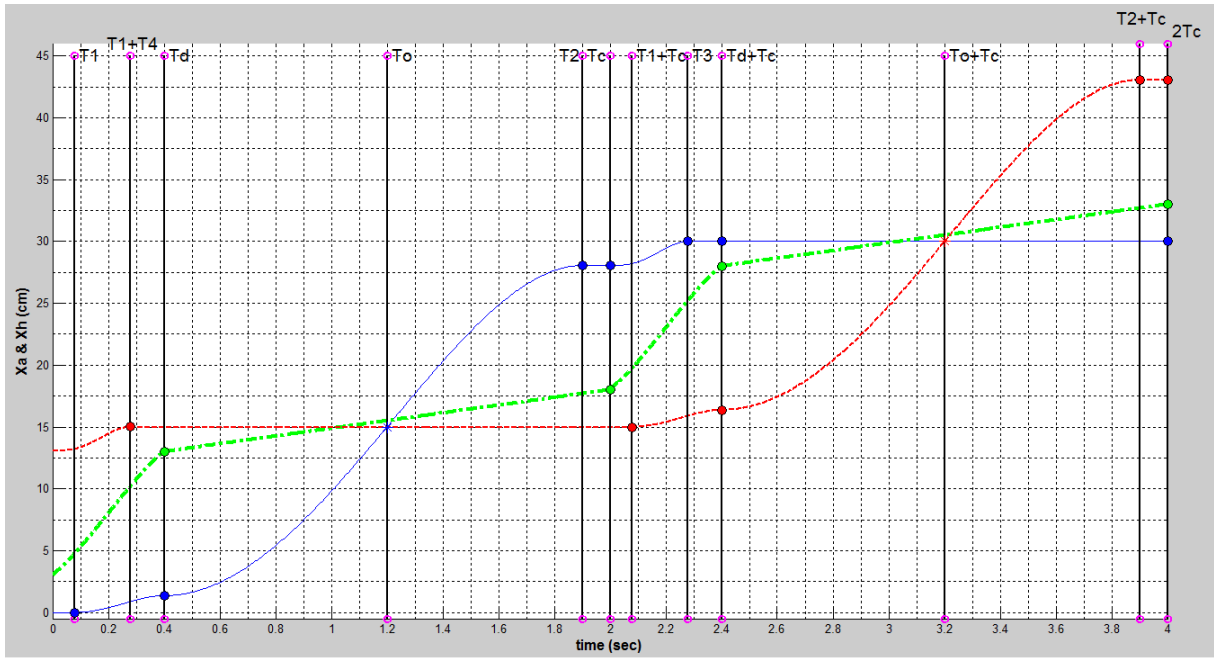


Fig. 4.14 Modified trajectory of torso in comparison with Fig. 4.13

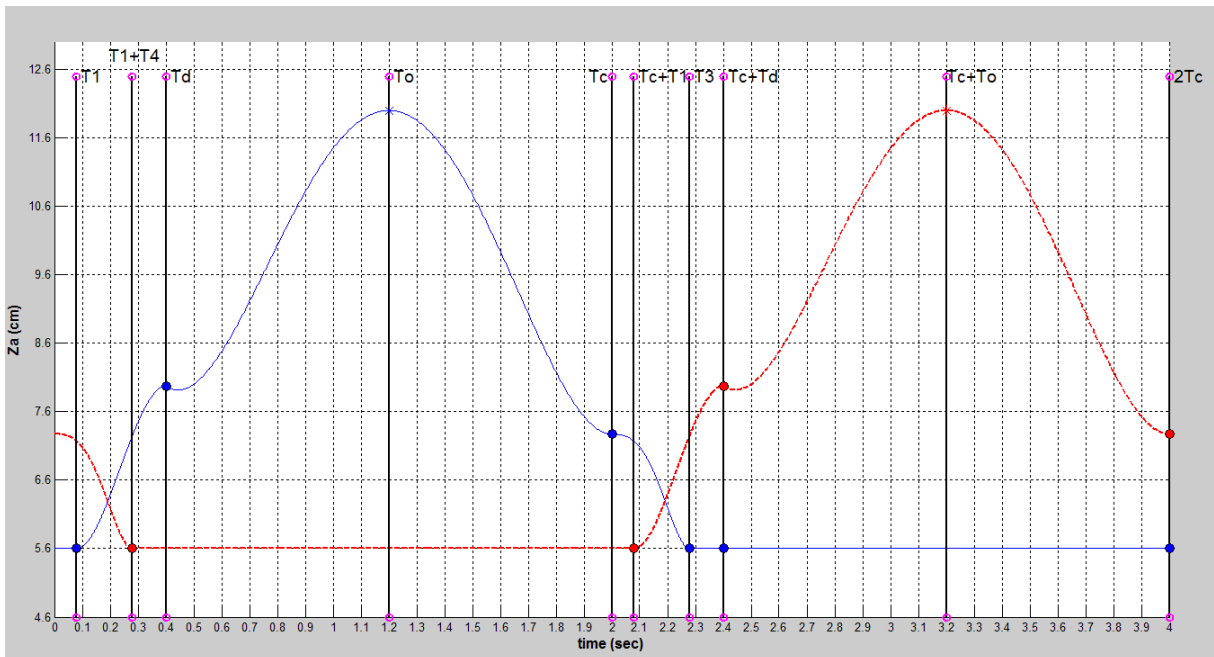


Fig. 4.15 Right (blue) and left (red) foot trajectory for upward movement

Fig. 4.15 shows the upward trajectory of feet z_a during walking. The minimum value for the upward trajectory is $l_{an} = 5.6 \text{ cm}$ which is the distance from frontal ankle joint to the sole of the foot. The maximum value of the trajectory is specified with $H_{ao} = 12 \text{ cm}$ that is selected according to the obstacle height. H_{ao} should be selected large enough that robot able to pass the obstacle.

Fig. 4.16 shows the trajectory of the torso z_h constrained by $H_{min} = 54 \text{ cm}$, $H_{max} = 57 \text{ cm}$ which are the minimum and maximum height of the torso during walking, respectively. These values determine the height of the torso from the ground that is important in case of the stability. By positioning the torso toward the ground the stability of the robot will be improved. But their values are limited due to the physical limitation of the joint movements. In addition, the height of torso affect selecting other parameters such as step length. Because if the height of the torso is selected too high, the step length will be limited to short distance due to the kinematic singularity of the robot. This case can be improved with changing other parameters like x_{ed} and x_{sd} to control the forward movement of the robot to prevent the robot to be in singular position.

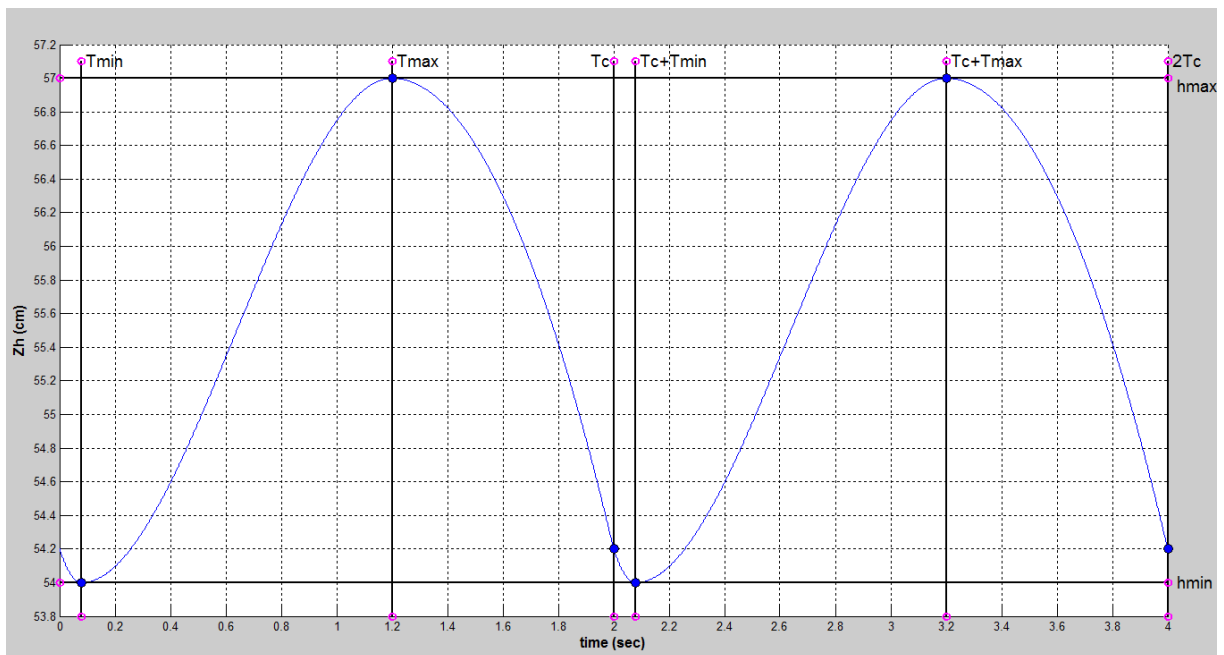


Fig. 4.16 Trajectory of upward movement of torso

The side movement of the feet and the torso are depicted in Fig. 4.17. The right foot (blue trajectory) starts to move to the right side while leaving the ground at T_d and reaches the maximum value represented by $w_p = 5 \text{ cm}$ at time $T_9 = \frac{T_c + T_d}{2}$. The parameter w_p is selected such that it prevent of collision of the feet when they are passing beside each other. Similarly the left foot (red trajectory) move in the same trajectory but on the other side.

The torso also moves from left leg to right leg during the double-support phase to compensate the swing foot's weight during the single-support phase. The torso passes the left leg as $w_m = 2 \text{ cm}$ in y direction. The value of w_m should be selected such that the balance of weight about the support foot is maintained during the single-support phase. To find the optimum value, an experimental test is used that will be explained in next chapters.

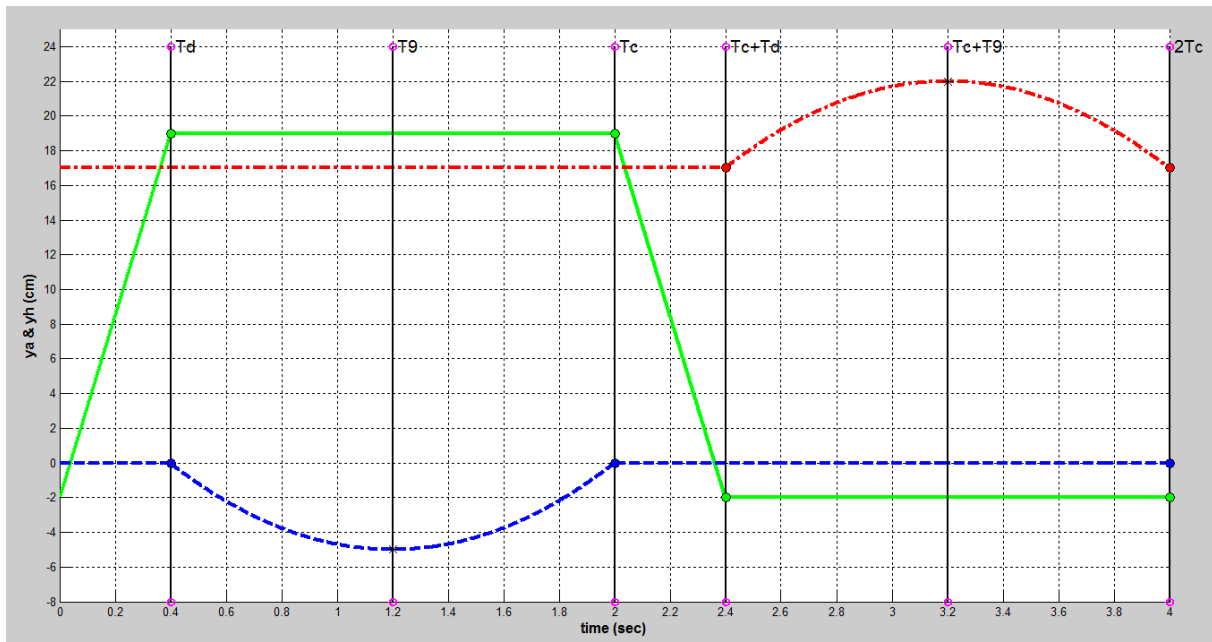


Fig. 4.17 Trajectory of left foot (red), right foot (blue) and torso (green) in y direction

In previous figures, we analyzed the trajectories of the feet and torso plotted versus time to evaluate the utilized formula. Now, the traversed trajectories of the feet and torso in each plane and also in 3D space can be taken into account. In this manner, both axes show the position of the feet or the torso in *cm* and in the same scale. Therefore the following figures show the real trajectories that the feet and the torso will pass through according to the selected parameters mentioned in the first of this subsection.

First of all, Fig. 4.18 shows the trajectory passed by the feet in the z-x plane. According to our assumption (see Fig. 4.7), it is noted that this is the trajectory of the point on the top of the feet that is connected to the ankle joint. As we expected the shape of the trajectory is like a half ellipse for the swing phase. But in the double-support phase the position of each foot is determined based on the constraints imposed by the ground. This part of the curve is shaped with the leaving and landing angle of the feet with the ground, i.e. q_b , q_f . The reader should notice that the figure only shows the position of the feet during walking not versus the time. That means for example during the single-support phase the blue trajectory starts from around $x_a = 2$ cm, $z_a = 8$ cm and finishes at around $x_a = 28$ cm, $z_a = 8$ cm (the right foot swing on the air), while the red trajectory stays in one point $x_a = 15$ cm, $z_a = 5.6$ cm (the left foot stay on the ground).

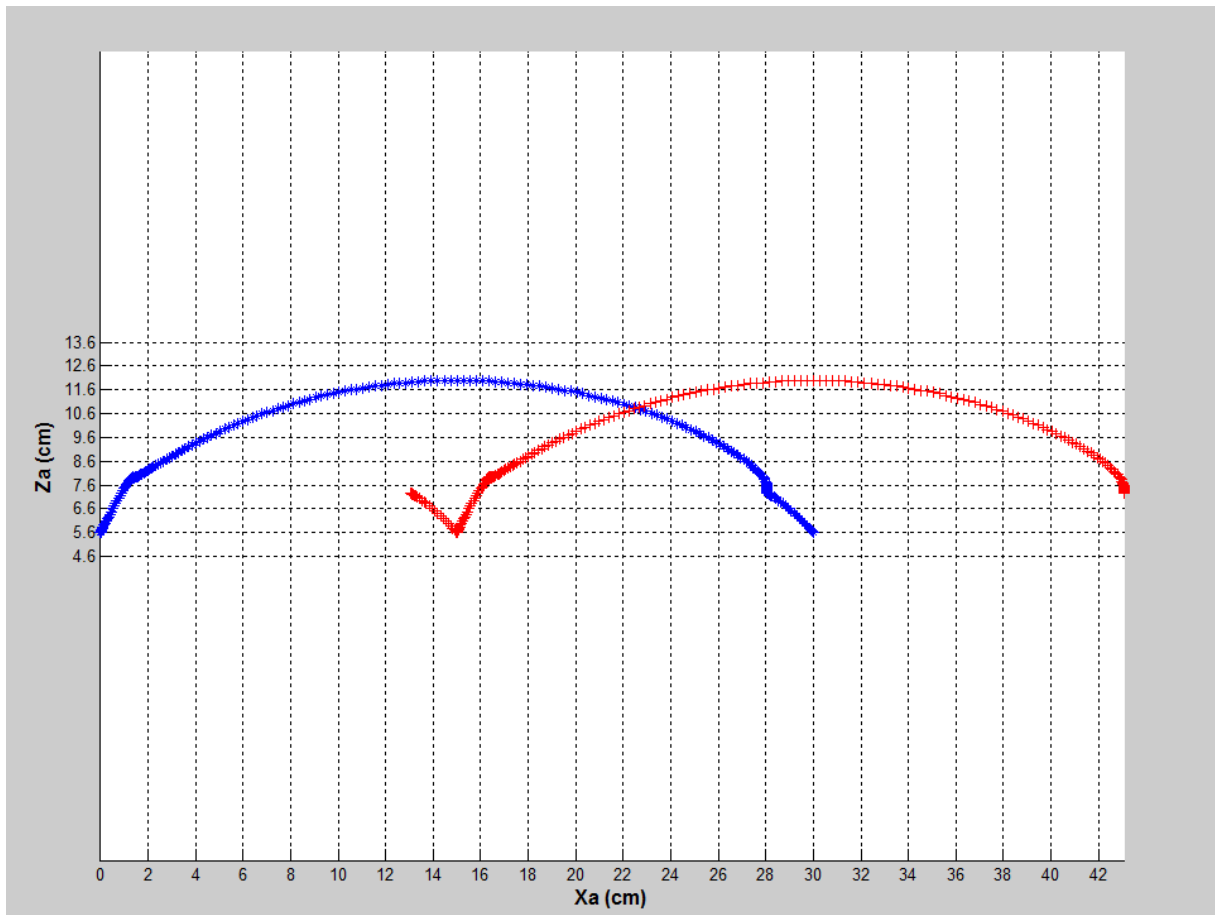


Fig. 4.18 Traversed trajectory of the right foot (blue) and the left foot (red) in the z-x plane

Similarly, Fig. 4.19 shows the trajectory of the torso during walking. The red points show the position of the torso during the double-support phase while the green points show the position of the torso during the single-support phase. It is clear that in the double-support phase the movement in x direction dominate the movement in z direction since the torso should move from back leg to front leg in short period of the double-support phase time while in the single-support phase, the z direction movement dominate the movement in the x direction.

As explained before, the torso movement in z direction restricted between 54 and 57 cm which is determined with parameters H_{min} and H_{max} . In this case we assume that the torso is in the highest position in the middle of single-support phase which is compatible with natural human walking pattern and is in its lowest position in the middle of the double-support phase.

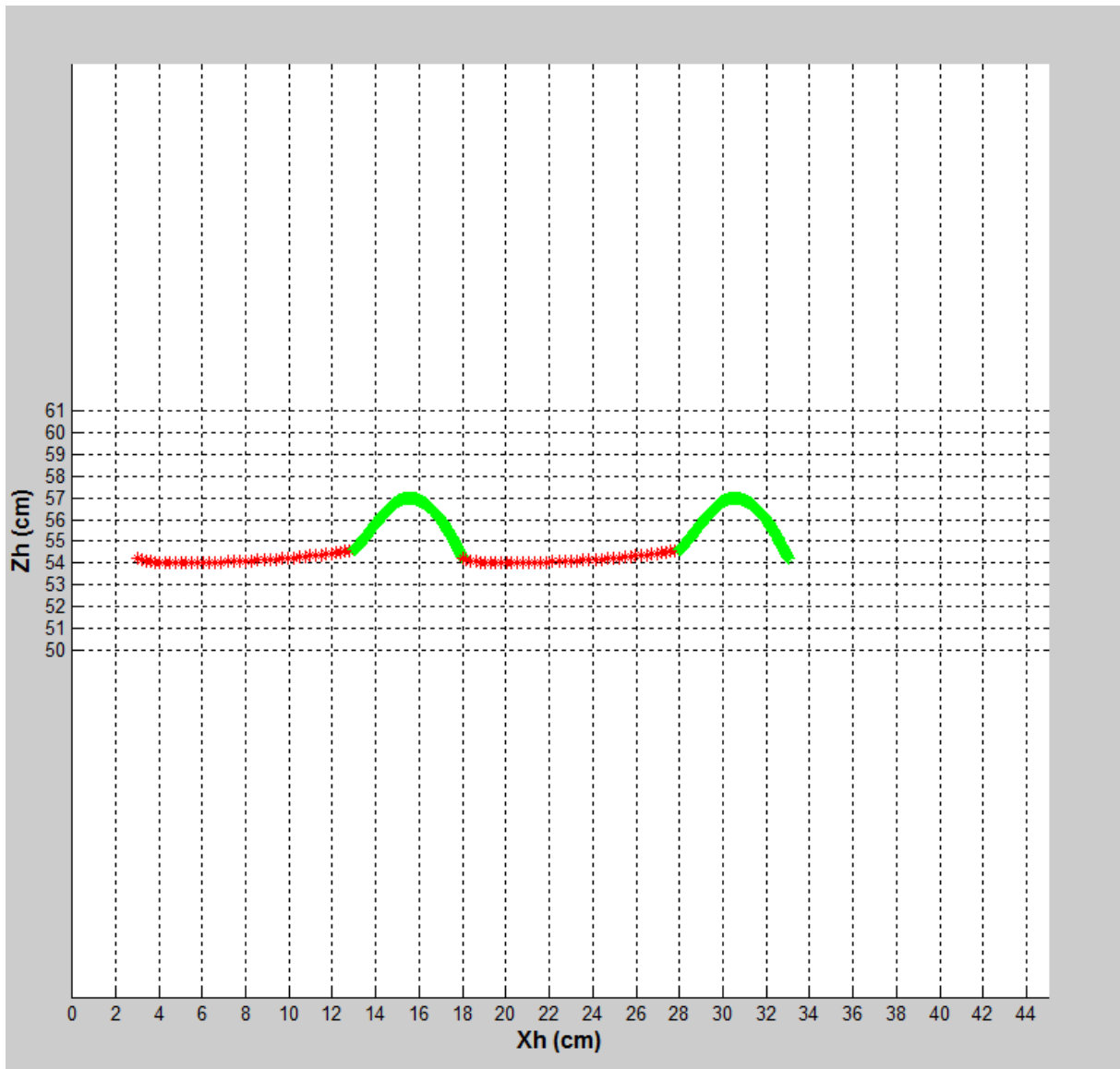


Fig. 4.19 Traversed trajectory of the torso in the z-x plane during the double-support phase (red) and during the single-support phase (green)

The most important advantage of the position trajectory, is that anybody can realize the final trajectory of the movement of the feet and torso in all plane and then decide to change the associated walking parameters to shape the desired path. For example to change the trajectory of Fig. 4.19, in x direction x_{ed} and x_{sd} have important role while in z direction H_{min} and H_{max} and their times are important.

For instance if the time of maximum position of torso is selected as $T_{max} = T_d$ instead of $T_{max} = (T_c + T_d)/2$ the trajectory of the torso is changed as shown in Fig. 4.20.

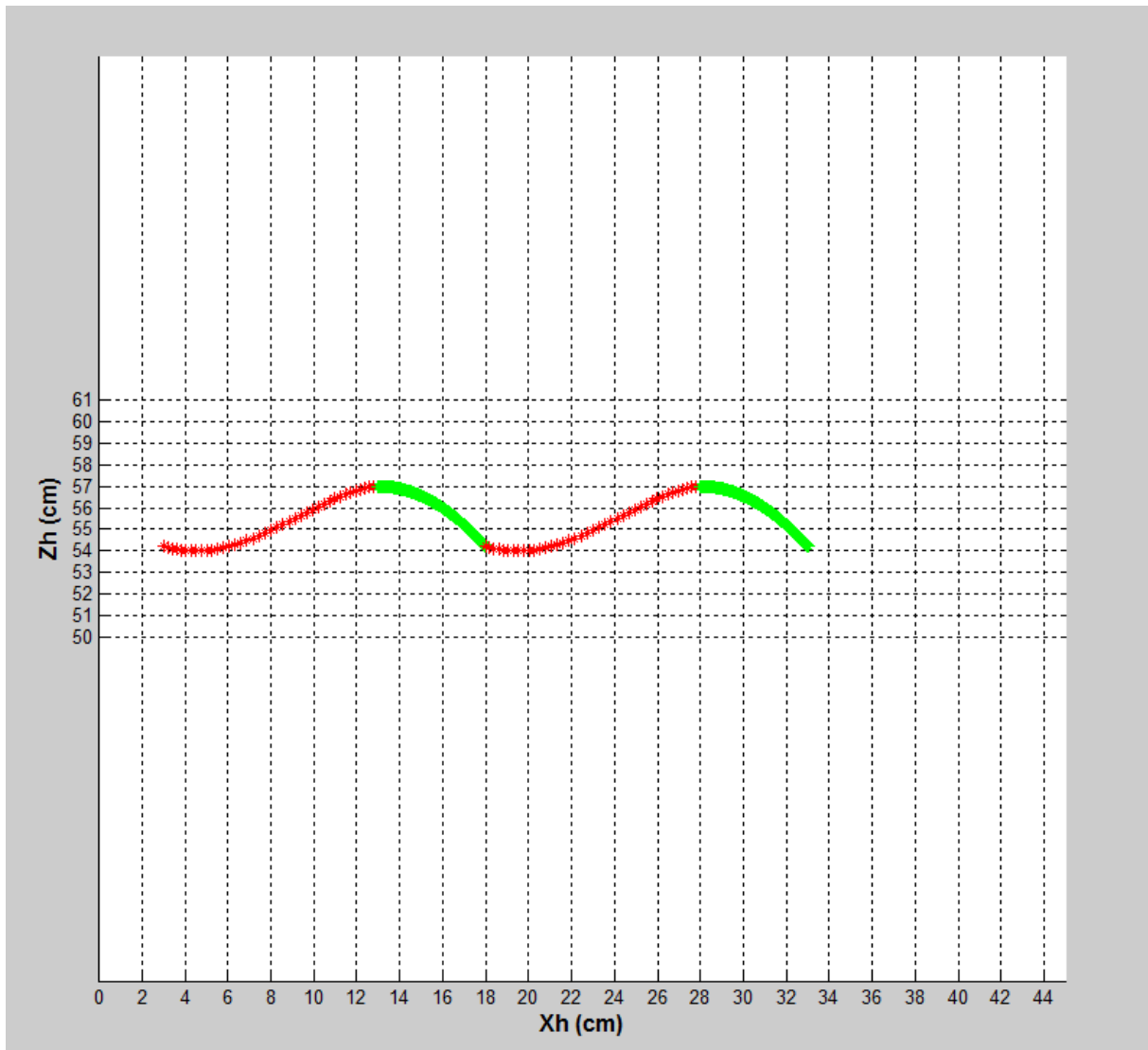


Fig. 4.20 Traversed trajectory of the torso in the z-x plane during the double-support phase (red) and during the single-support phase (green)

The trajectories of the feet and the torso in transversal (y-x) plane are depicted in Fig. 4.21. The feet has no side movement during the double-support phase and has only forward movement according to the angle of leaving and landing. Again it is noted that the point of foot that its trajectory is drawn is located at the top of the foot where connected to the ankle joint not any point on the sole.

The torso also has only forward movement during the double-support phase while in the single-support phase has both forward and side movement as depicted as a green trajectory in Fig. 4.21. The most important factor to plan a stable walking trajectory is the position of the torso during the single-support phase. As shown in Fig. 4.21, the position of the torso is designed to be close to the support foot. This allows us to keep the center of gravity inside the support area , i.e. sole of the support foot, and finally stabilize the robot during single-support phase.

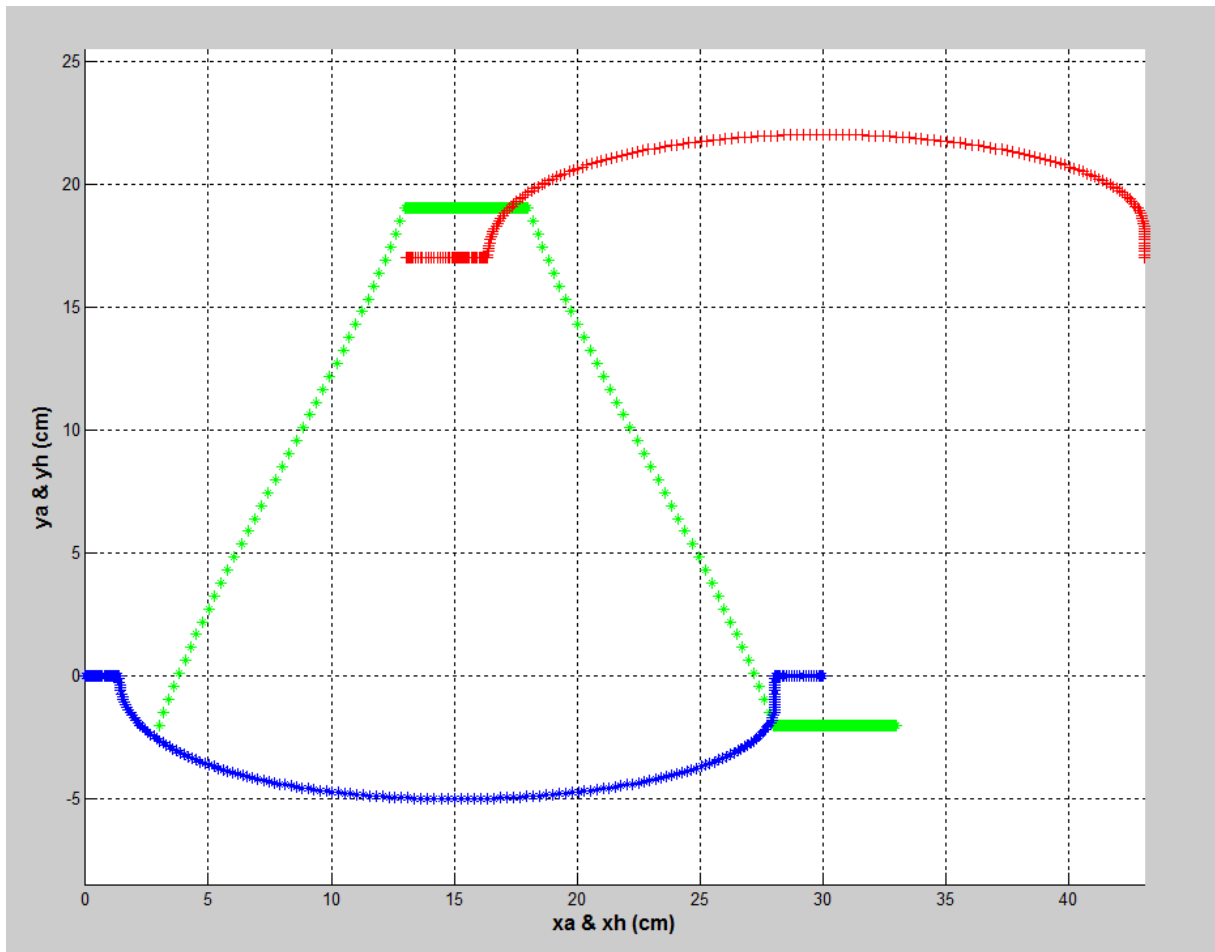


Fig. 4.21 Position trajectory of the right foot (blue) and the left foot (red) and the torso (green) in the y-x plane

Fig. 4.22 shows the position trajectory of the feet and the torso in the frontal (z-y) plane. Since during the double-support phase, the feet has no movement along y direction, the trajectory is vertical in this interval. While the side movement of the feet in the double-support phase changes the shape of the curve as depicted in blue and red trajectories in this interval. The maximum side movement of the feet is $w_p = 5 \text{ cm}$ and the maximum height of the feet is $H_{ao} = 12 \text{ cm}$. It is noted because this trajectory is for the top of the foot that has distance $l_{an} = 5.6 \text{ cm}$ from the sole, the distance between the ground and the sole is shorter that depends on the angle of the foot. This note should be mentioned in the parameter selection for passing an obstacle.

The trajectory of the torso during the first double-support phase is depicted in yellow and during the second double-support phase is shown by black. The green trajectory demonstrate the position of the torso during the both single-support phase. The torso has a little change in the elevation during the both double-support phase compared to the trajectory related to the single-support phase.

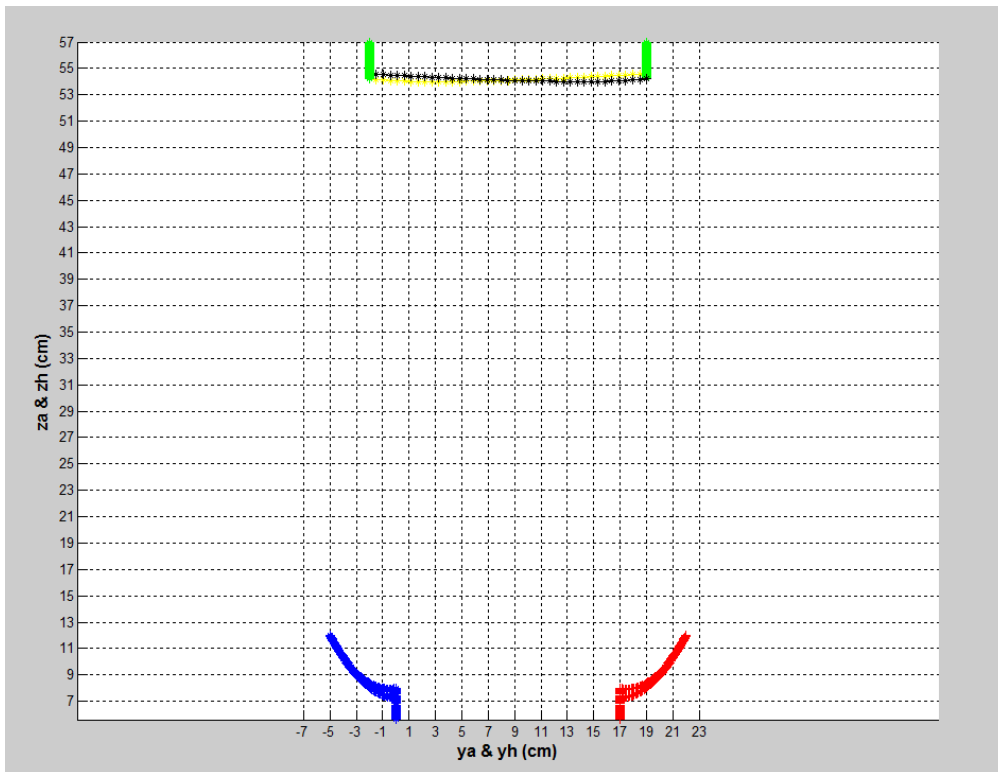


Fig. 4.22 Position trajectory of the right foot (blue) and the left foot (red) and the torso (green-black-yellow) in the z-y plane

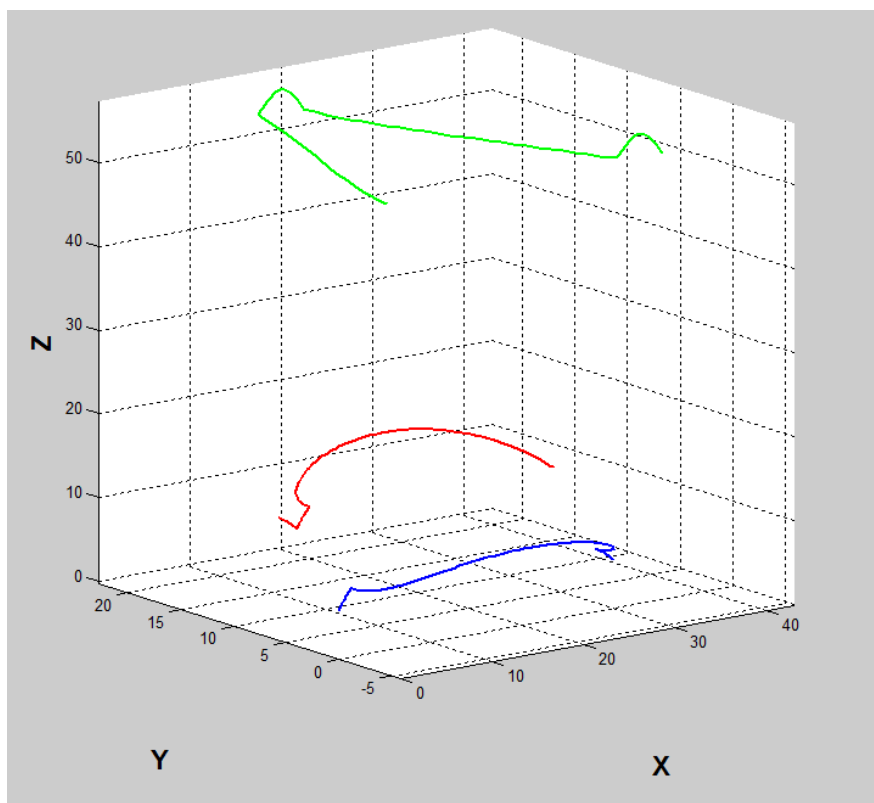


Fig. 4.23 Position trajectory of the right foot (blue), left foot (red) and the torso (green) in three dimension

Fig. 4.23 also shows the three dimensional trajectory of the feet and the torso during one complete cycle of walking according to the selected walking parameters.

Up to here, the trajectory of the both feet and the torso are constructed according to the walking parameters. Now, one can find the joint angle trajectories based on the inverse kinematic algorithm presented in chapter 3 and given the trajectories of the feet and the torso. These joint angles will be used as a reference command for control of each joint. Since our control is a digital control, the trajectories should be discretized. So a new parameters are defined as number of desired point for discretization of the trajectories :

$$N = \text{number of discrete points}$$

Therefore, the time required for one complete cycle of walking, i.e. $0 < t < 2T_c$, can be discretized to N points as:

$$t = \left\{ 0, \frac{2T_c}{N-1}, 2 \times \frac{2T_c}{N-1}, 3 \times \frac{2T_c}{N-1}, \dots, 2T_c \right\} \quad 4-34$$

Then using inverse kinematic formulas, the discrete joint angles for $N = 300$ points are derived. Fig. 4.24 shows the trajectories of discrete values of the frontal ankle joint angles. As expected, in double-support phase both joint rotates in a same direction to move the torso from top of the right leg to the top of the left leg. In the single-support phase the left frontal ankle stay approximately unchanged to keep the torso without side movement while the right frontal ankle start to rotate to level the swing foot (the right foot) with the ground. It is noted that the sign of rotation angle is based on the D-H convention proposed in the chapter 3.

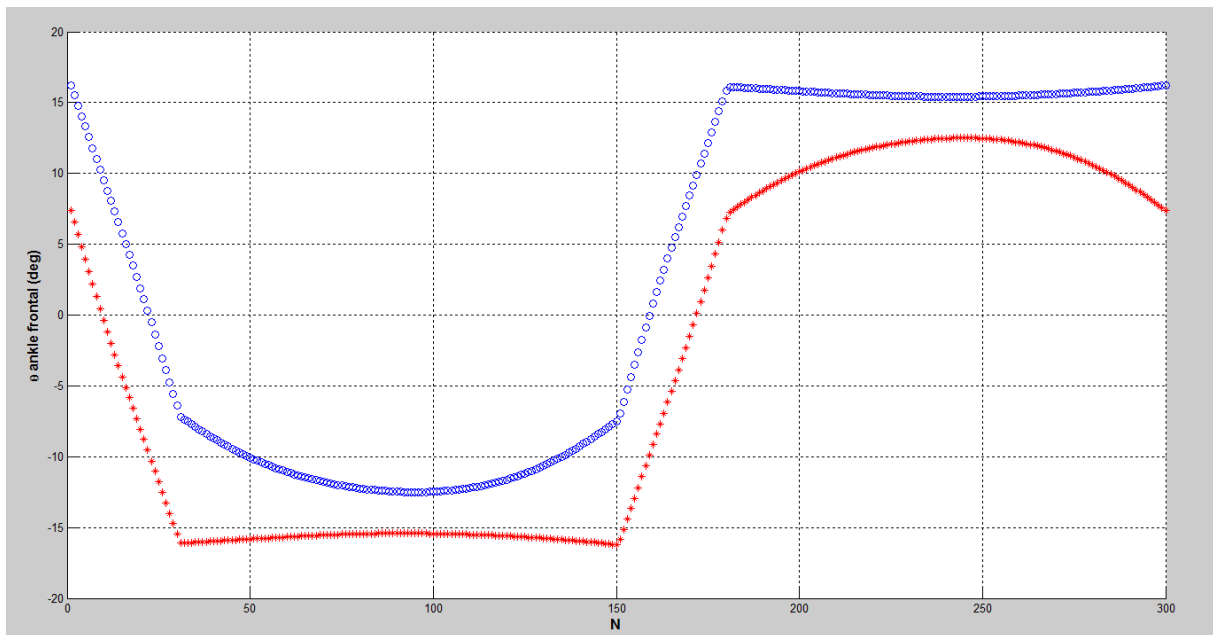


Fig. 4.24 Frontal ankle joint angles for right leg (blue) and left leg (red)

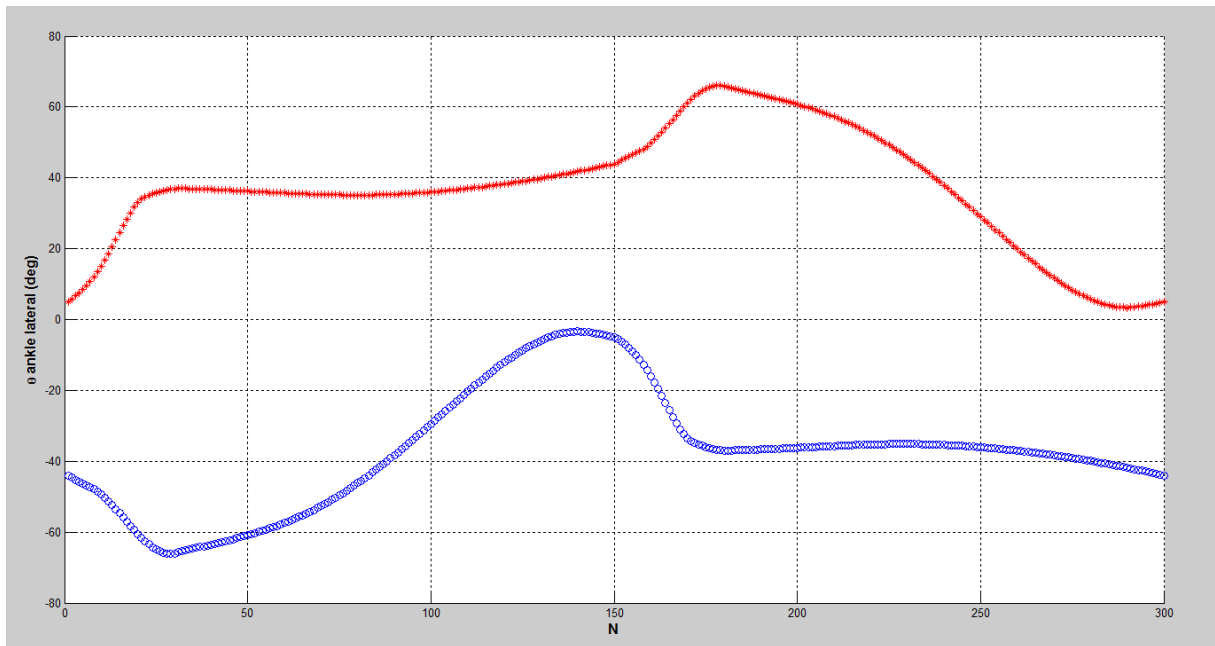


Fig. 4.25 Lateral ankle joint angles for right leg (blue) and left leg (red)

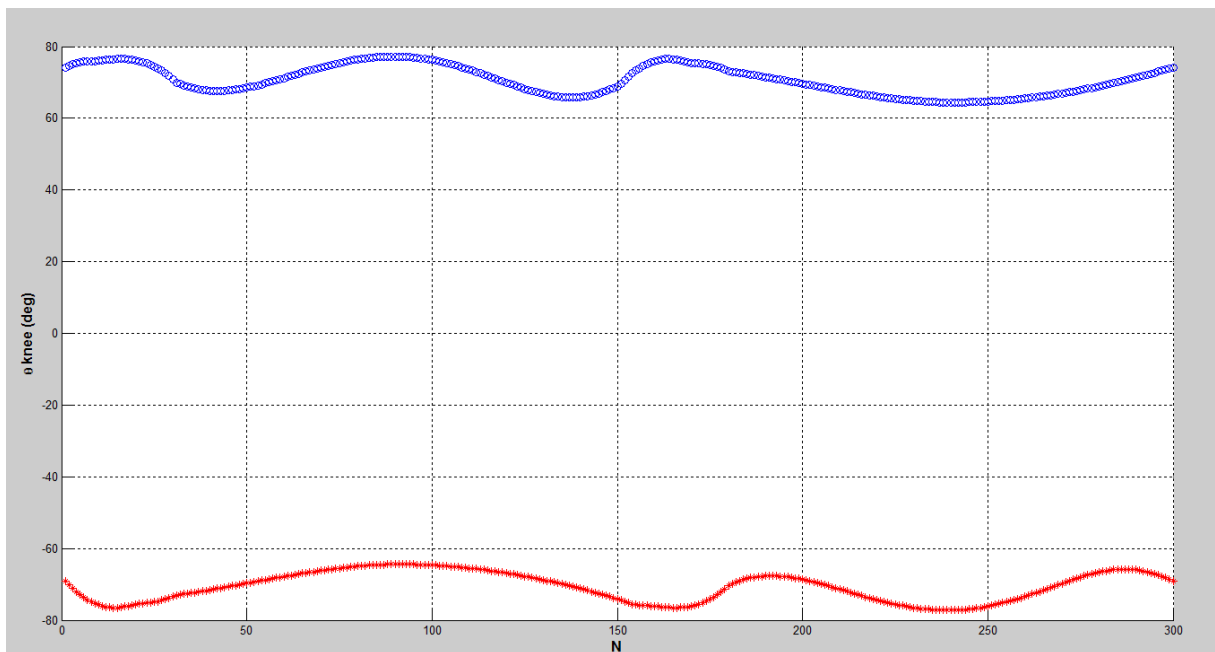


Fig. 4.26 Knee joint angles for right leg (blue) and left leg (red)

Fig. 4.25 shows the trajectory of the lateral ankle joint which enable the robot moving in the sagittal (lateral) plane. The noticeable point is that after $N = 150$ the angle values of the right and left lateral ankle are exchanged with a negative sign. This is due to the change of support foot after one step and starting a new step with the new swing foot. Fig. 4.26 also shows the knee joint angle trajectory for the both feet.

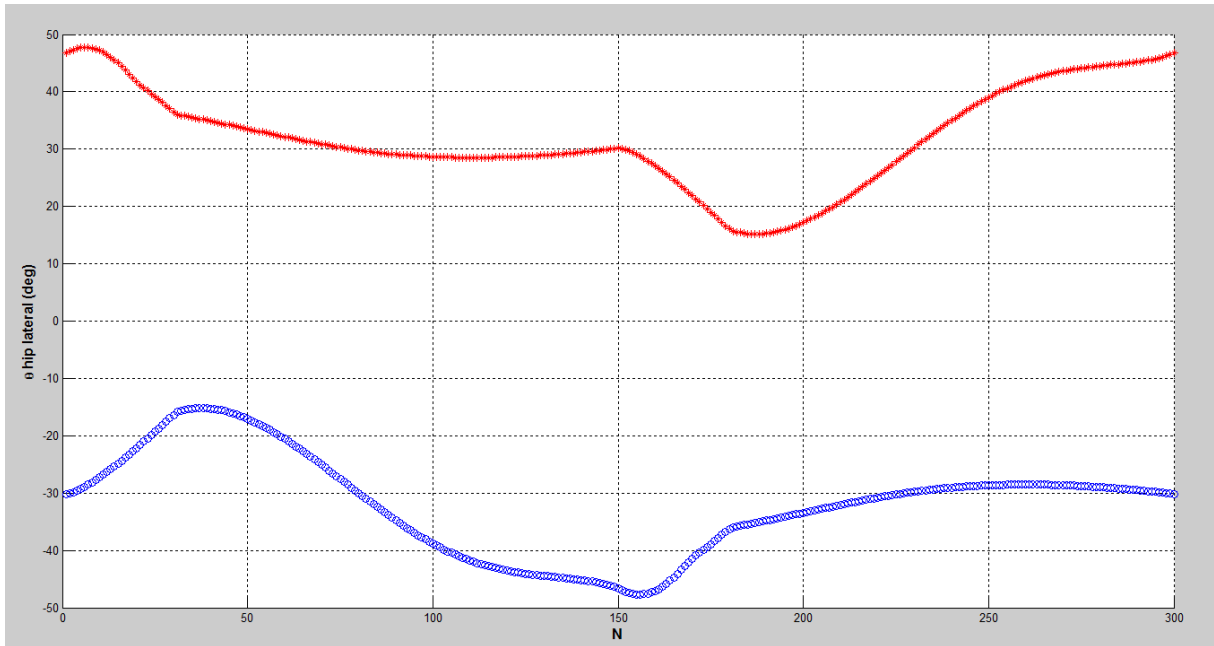


Fig. 4.27 Lateral hip joint angles for right leg (blue) and left leg (red)

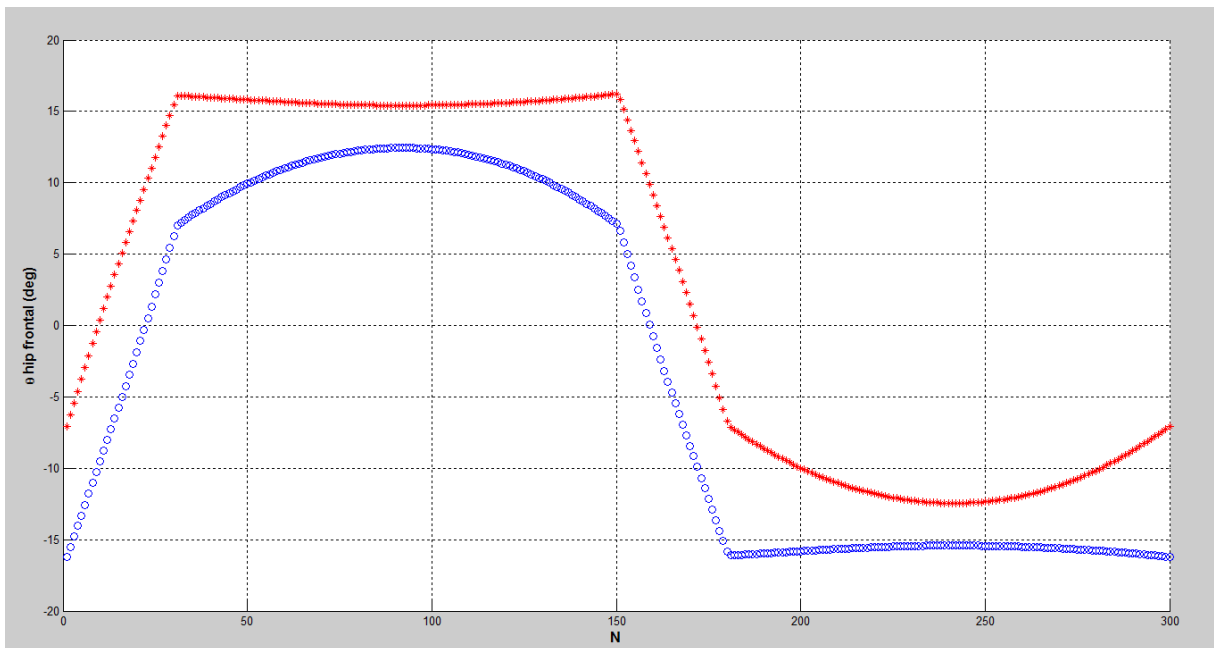


Fig. 4.28 Frontal hip joint angles for right leg (blue) and left leg (red)

Fig. 4.27 and 4.28 show the angle trajectory of the lateral and frontal joints during walking. As expected, angle trajectory of the frontal hip joint is a mirror of the angle trajectory of the frontal ankle joint.

On the other hand, Fig. 4.29 shows the angle trajectory of the transversal hip joint. The movement of this joint is limited to 3 and -3 degrees.

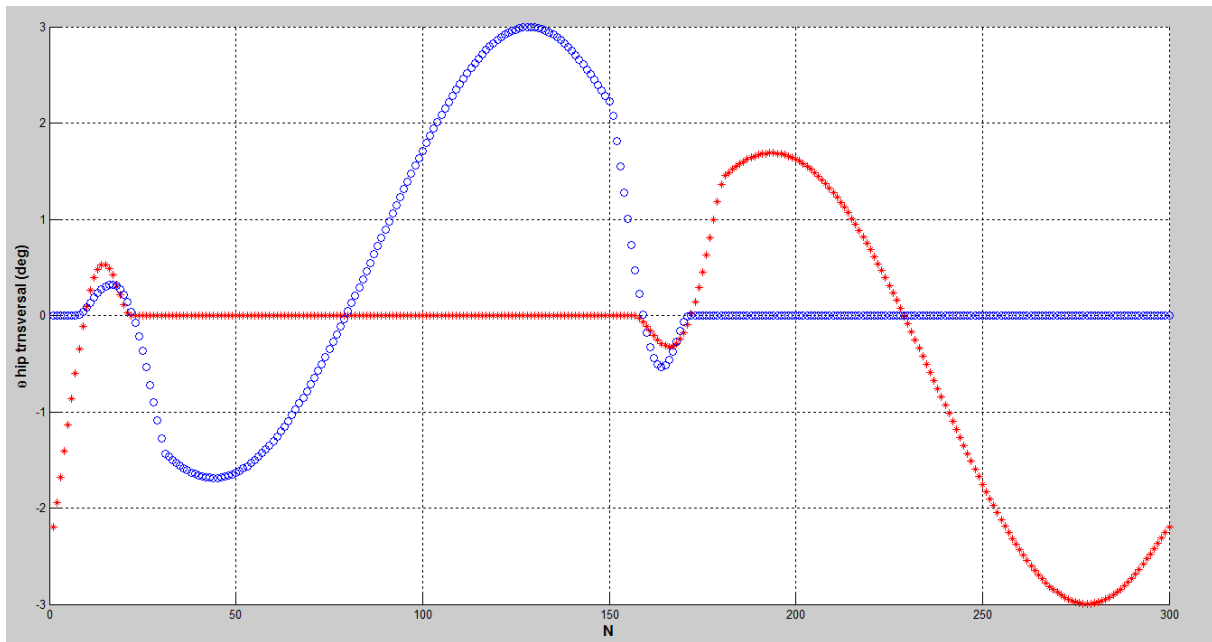


Fig. 4.29 Transversal hip joint angles for right leg (blue) and left leg (red)

4.4 Kinematic simulation

In previous section, the joint angles trajectories for biped walking of Archie are obtained according to the desired walking parameters and using the inverse kinematics. Now the walking of the biped robot can be simulated based on the obtained joint angles and using the forward kinematics established in chapter 3. The schematic diagram of the simulation program are depicted in Fig. 4.30. A Matlab program creates the simulation of the biped robot based on the desired walking parameters mentioned before such as step length, step height and so on. This simulation is a powerful tool that enable us to examine the walking parameters before implementing to the real robot. By this simulation the general position of the robot are predicted and we can analyze the effect of each walking parameter before real implementation into robot. It is noted this simulation only shows the kinematic behavior of the robot not the robot dynamics. So in contrary to the real robot no falling is occurred during simulation. Generally speaking , this is the simulation of robot without mass. Actually, To find the effect of the walking parameters on stability of the robot, they should be implemented into the robot.

Fig. 4.31 to 4.39 show the simulation of the robot position at some important points, i.e. start, middle and end of the double and single-support phase. All figures contain four parts; one for 3D simulation and the others are for 3 principal planes. To be more clear, the right foot is shown with blue lines while the left leg is shown with red lines. The position of the joints are depicted with the hollow black circle. It is noted also all dimensions are in *cm* in these figures.

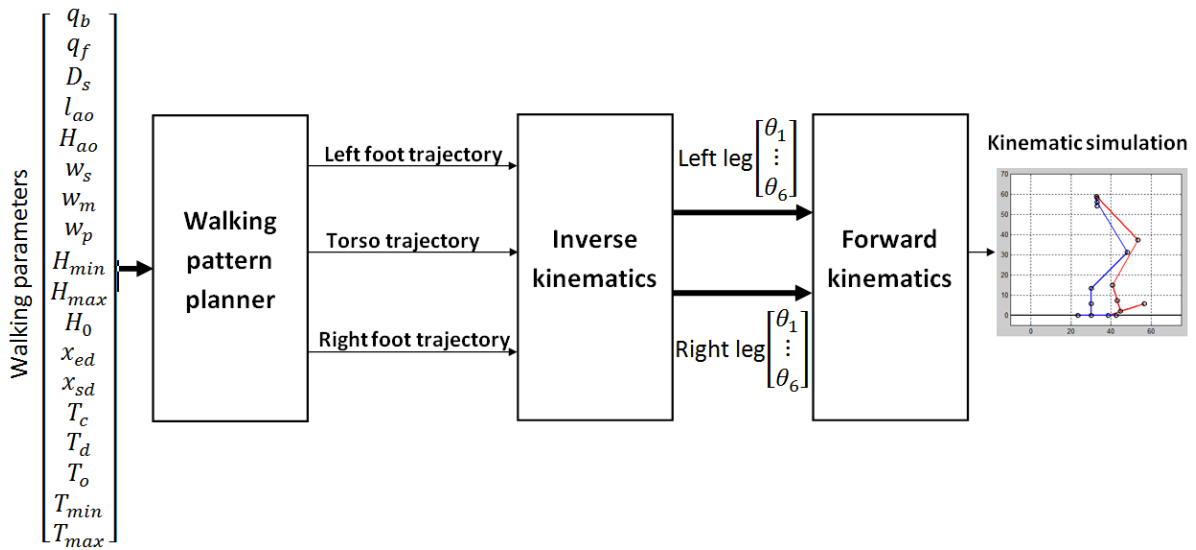


Fig. 4.30 Schematic diagram of the proposed program for kinematic simulation of the biped walking

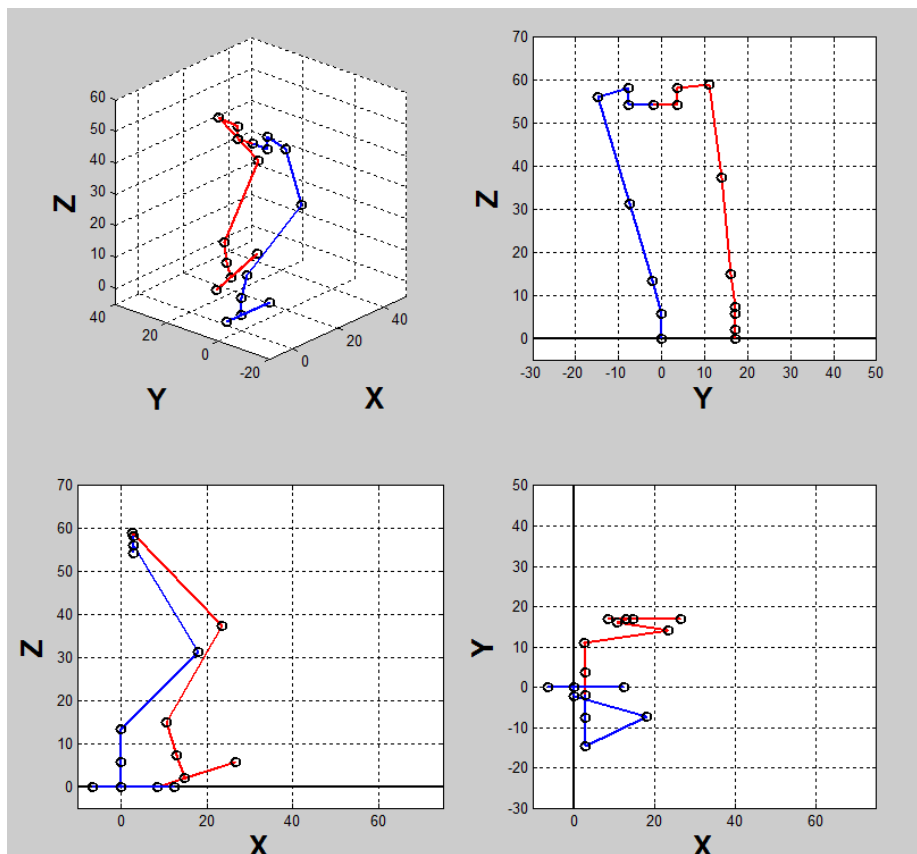


Fig. 4.31 Robot simulation at time $t = 0$

Fig. 4.31 shows the robot configuration at the start of walking cycle where the heel of the left foot is in contact with the ground and the torso is in above the right foot. Fig. 4.31 and

4.32 show the robot configuration at the middle and the end of the first double-support phase, respectively.

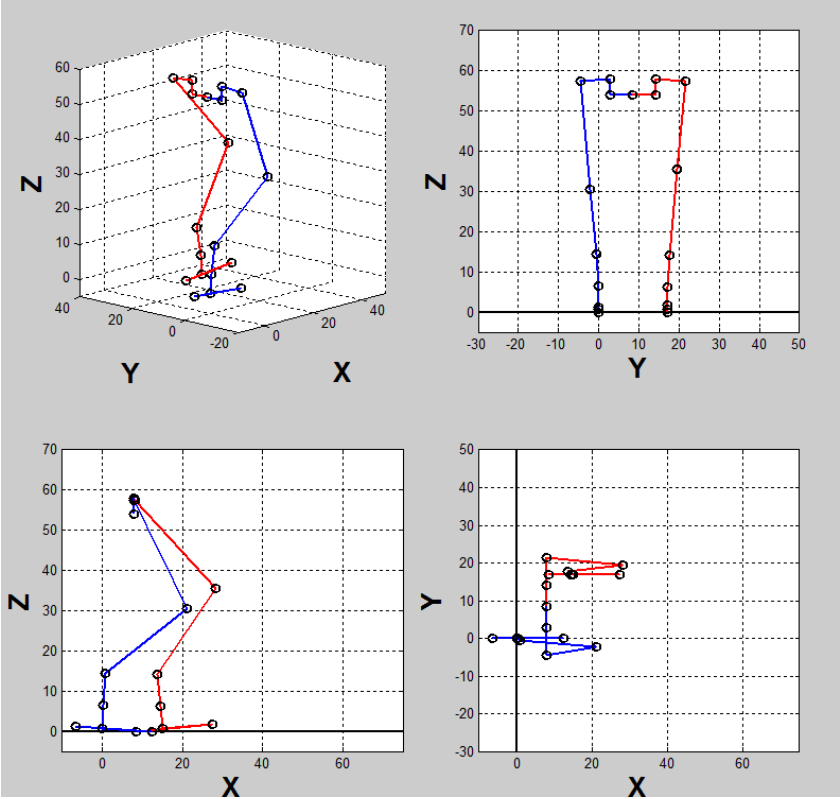


Fig. 4.32 Robot simulation at time $t = T_d/2$

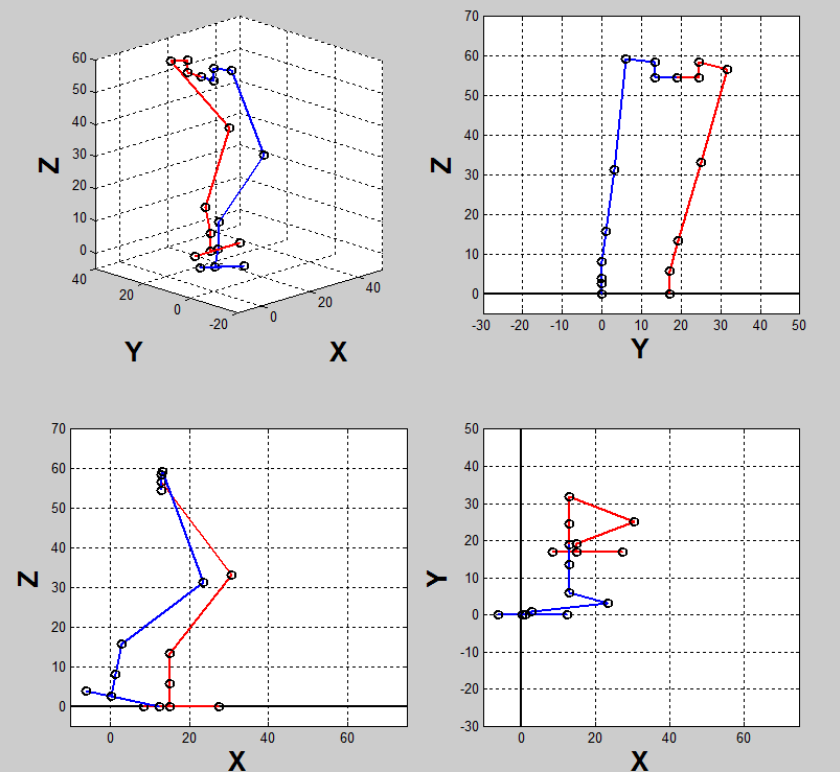


Fig. 4.33 Robot simulation at the time $t = T_d$

Fig. 4.34 and 4.35 show the robot configuration at the middle and the end of the first single-support phase (the left leg is the stance leg), respectively.

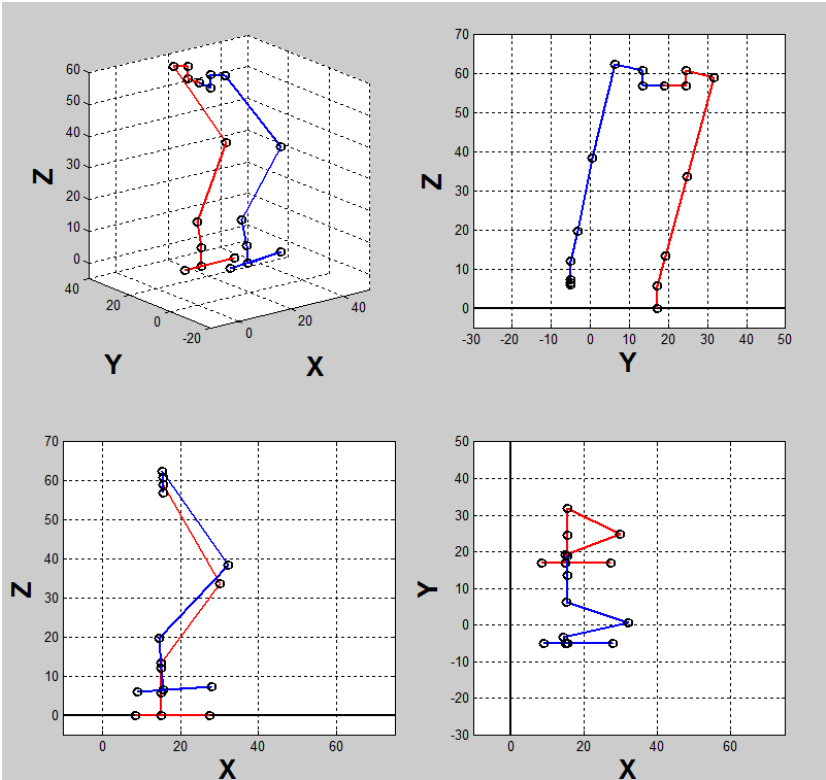


Fig. 4.34 Robot simulation at the time $t = T_{max} = (T_c + T_d)/2$

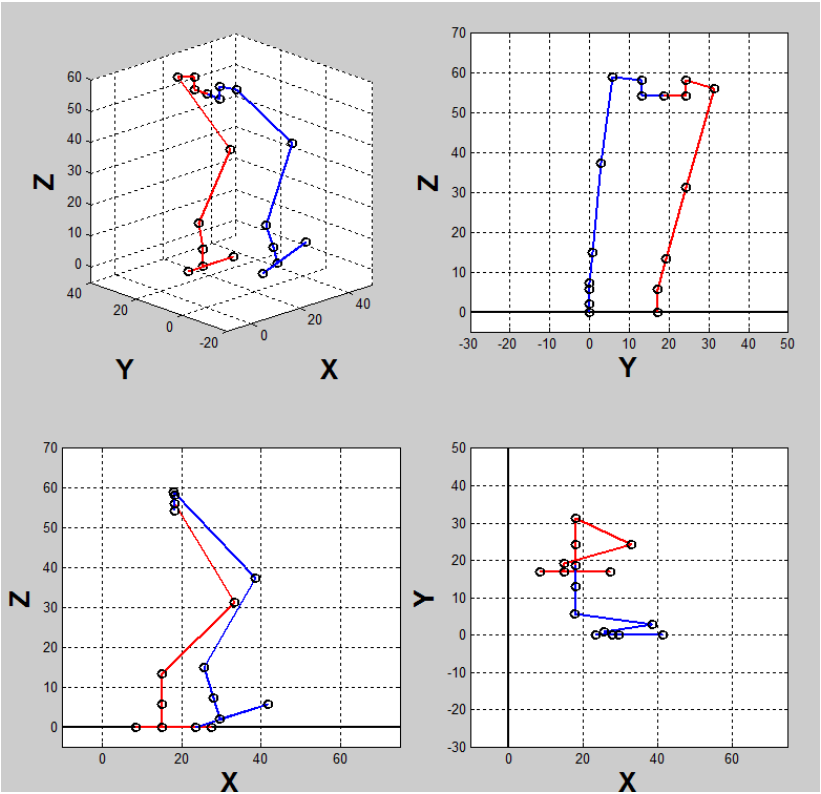


Fig. 4.35 Robot simulation at the time $t = T_c$

Similarly, Fig. 4.36 and 4.37 show the robot configuration at the middle and the end of the second double-support phase, respectively. In this phase the torso moves from the above of the left foot to above of the right foot.

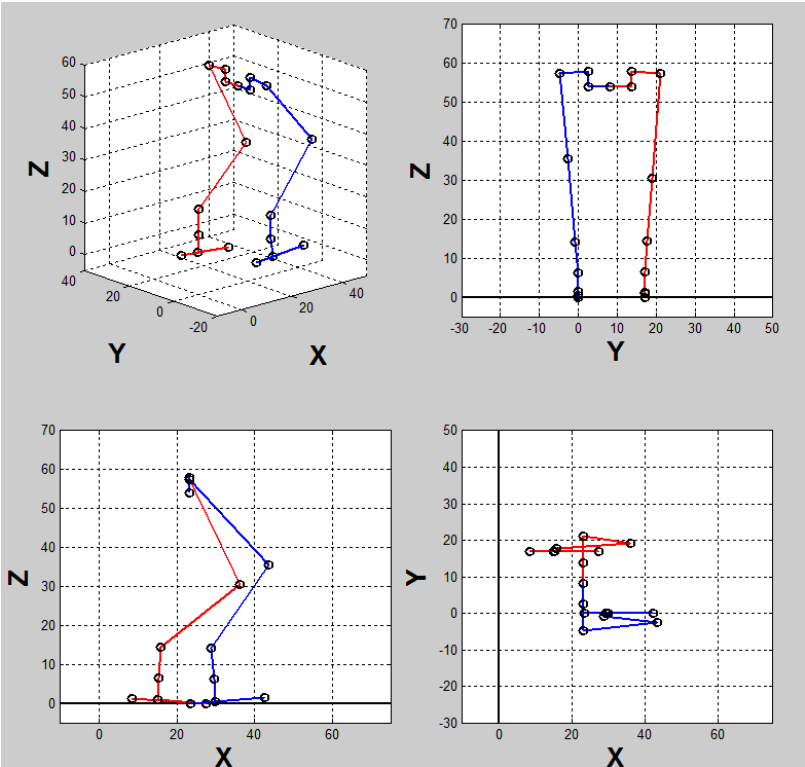


Fig. 4.36 Robot simulation at the time $t = T_c + (T_d/2)$

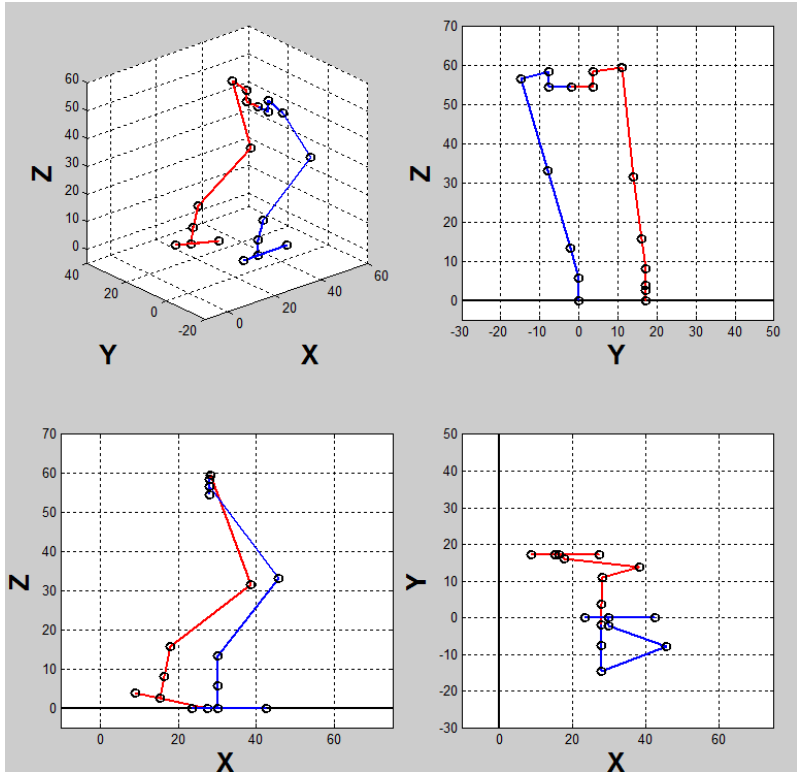


Fig. 4.37 Robot simulation at the time $t = T_c + T_d$

Fig. 4.38 and 4.39 show the robot configuration at the middle and the end of the second single-support phase. During this phase the left foot swing on the air while the right foot is in contact with the ground.

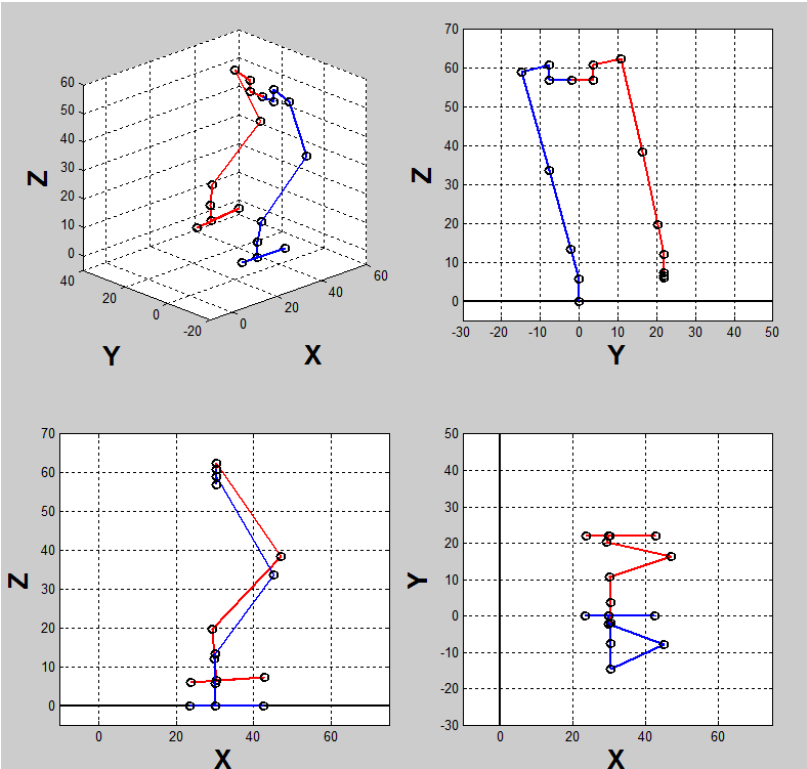


Fig. 4.38 Robot simulation at the time $t = T_c + (T_c + T_a)/2$

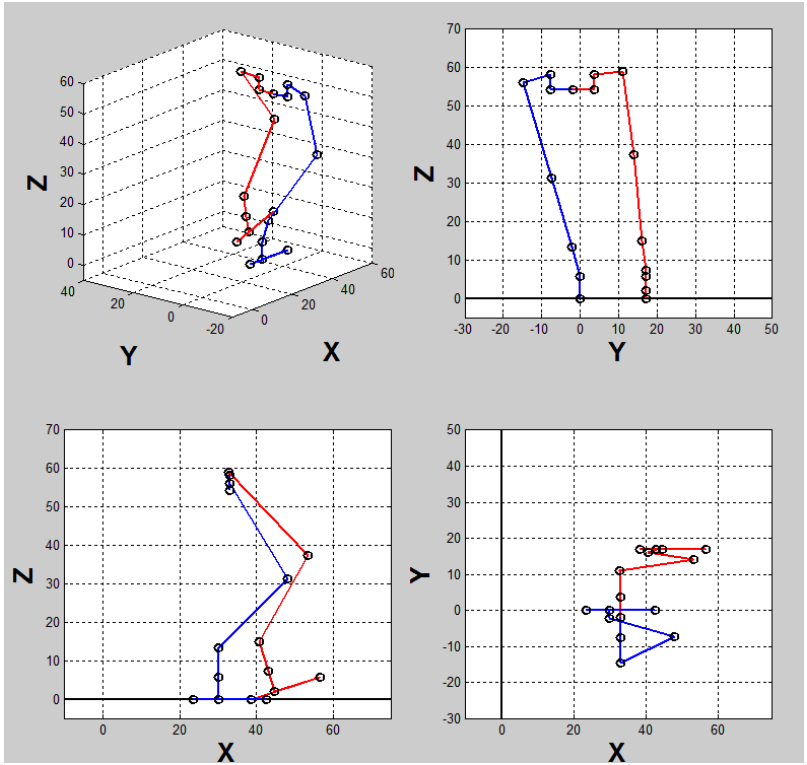


Fig. 4.39 Robot simulation at the time $t = 2 T_c$

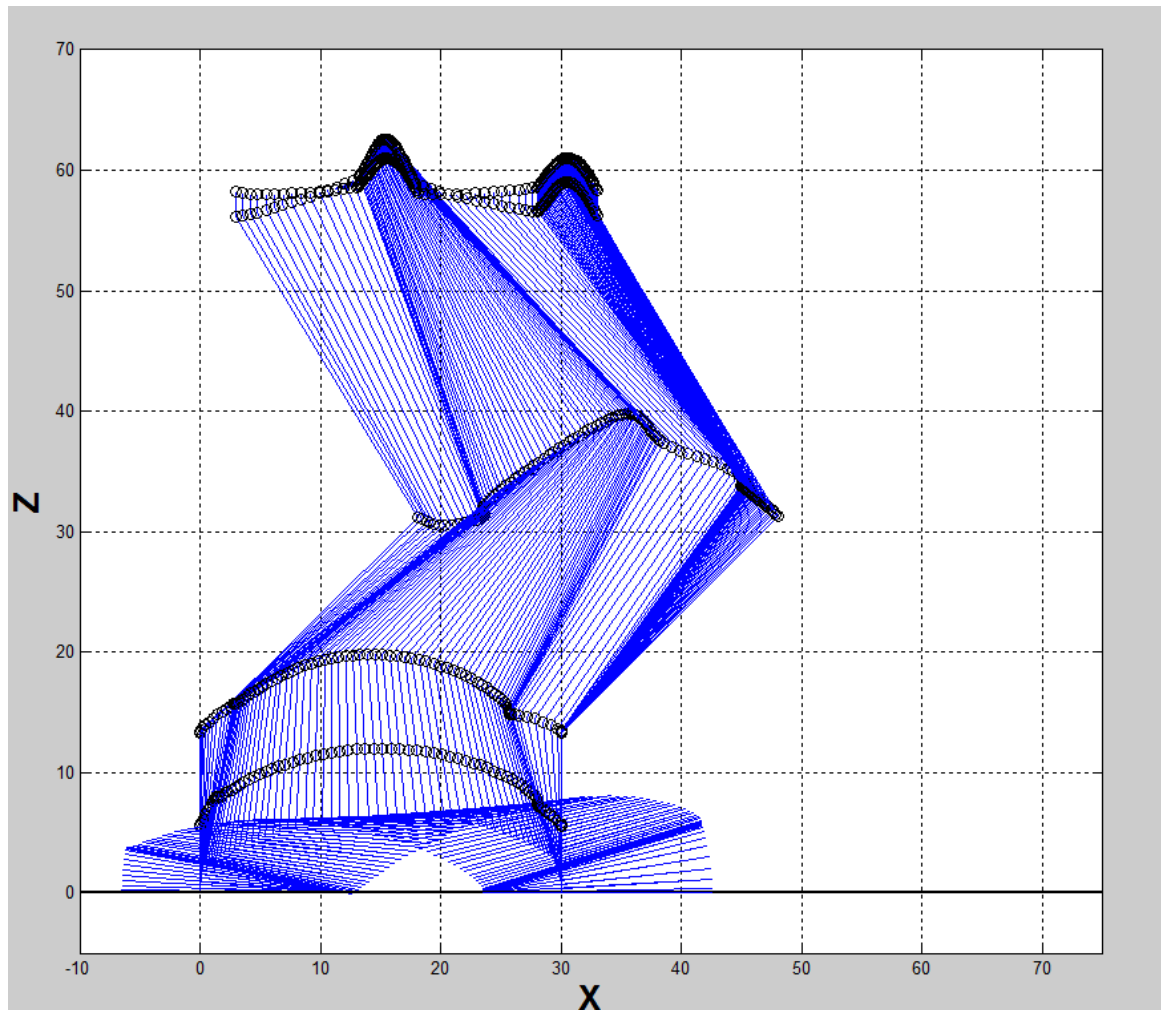


Fig. 4.40 The right leg simulation during one stride walking

Fig. 4.40 and 4.41 show the position of the robot links and joints during one complete cycle of walking in X-Z plane for the right and the left leg, respectively (both axis dimension are *cm*). The trajectories of the frontal ankle joint (foot top) and the torso are depicted in this figures which are the same trajectories planned according to the desired walking parameters. In addition, the trajectories for the lateral ankle joint, the knee joint and the hip joint can be seen in this figure. These trajectories are obtained based on the inverse kinematics and given the desired foot and the torso trajectories.

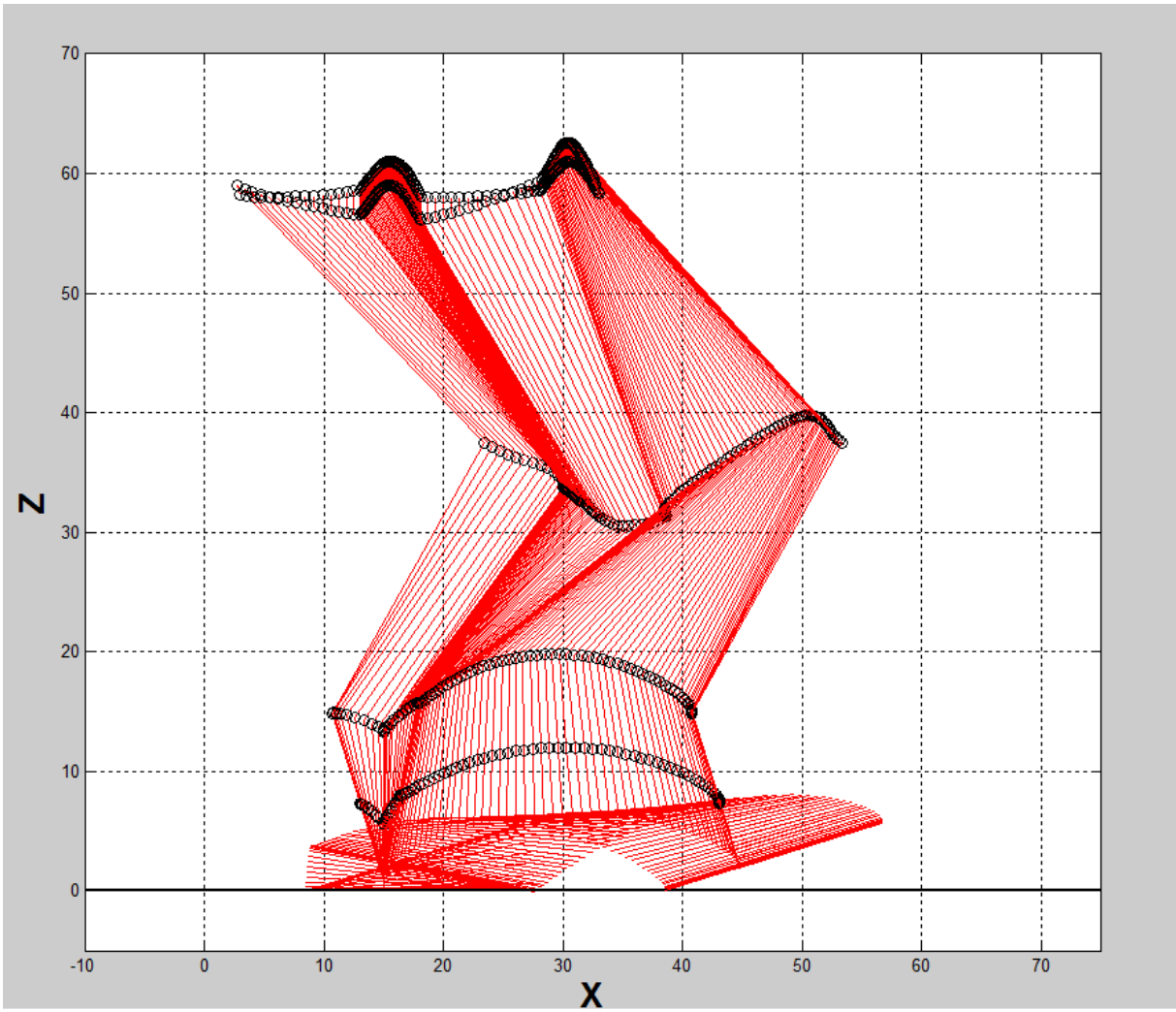


Fig. 4.41 The left leg simulation during one stride walking

Chapter 5 Control design

In the chapter 4, trajectory planning for human-like walking have been presented that allows the generation of the reference inputs to the motion control system. The problem of controlling a manipulator can be expressed as to specify the time history of the generalized torques to be produced by the joint actuators, i.e. in our case electric motors, so as to guarantee execution of the commanded task while satisfying given transient and steady-state requirements.

In spite of robot manipulator type, desired task (end-effector motion) is regularly done in the operational space, while control actions (joint actuator torques) are carried out in the joint space. This fact leads to considering two kinds of general control schemes, namely, a joint space control scheme and an operational space control scheme. In both methods, the closed loop control is used to utilize the good characteristics provided by feedback, i.e., robustness to modeling uncertainties and reduction of disturbance effects. The difference between these schemes are depicted in Fig. 5.1 and 5.2. In both control system the input is the desired trajectory of the end-effector in the Cartesian (operational) space, i.e. X_d , and the output is the actual trajectory of the end effector in the operational space , i.e. X_e . The main difference is related to the output measurement.

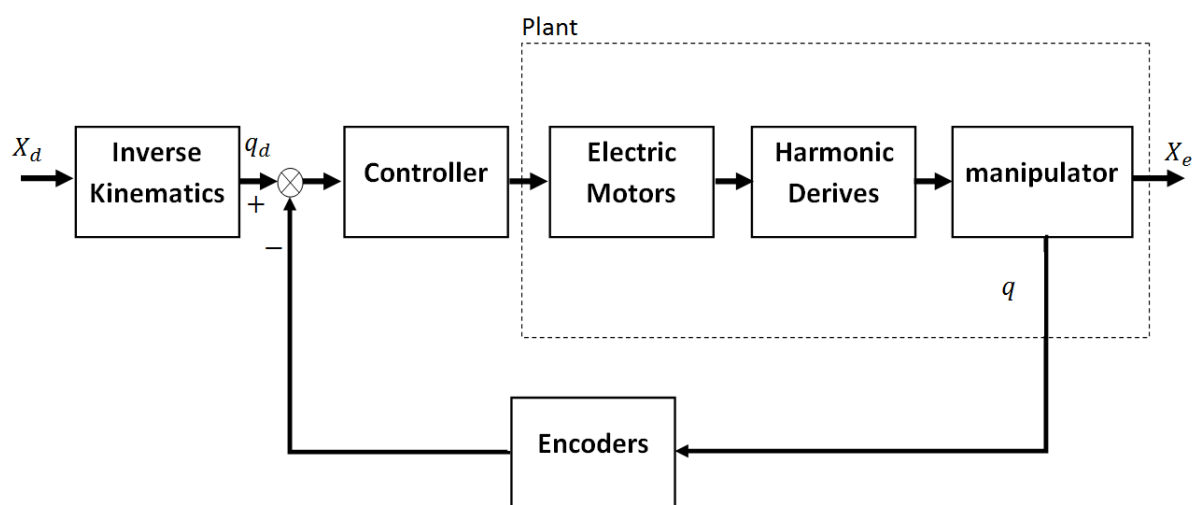


Fig. 5.1 General schematic diagram of joint space control scheme

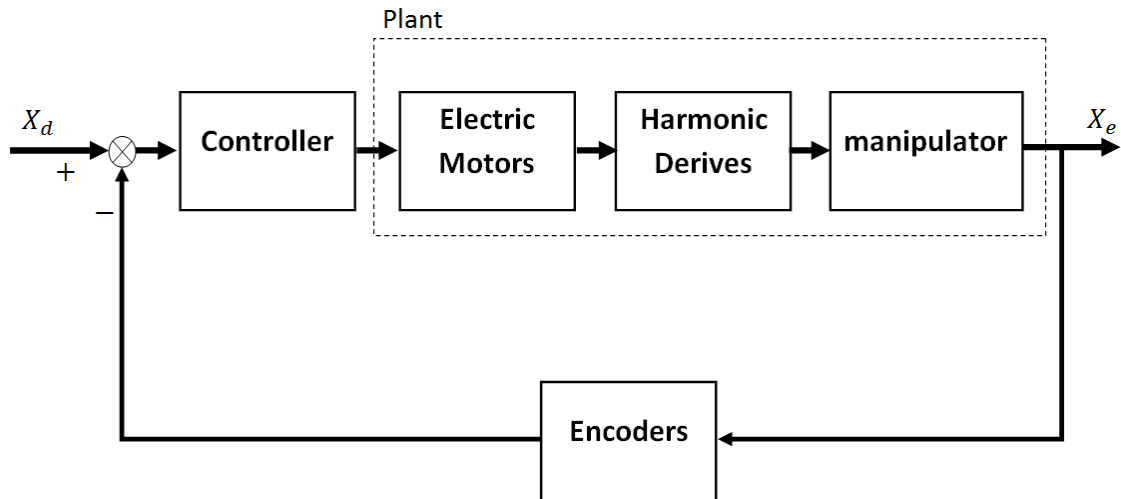


Fig. 5.2 General schematic diagram of operational space control scheme

In the joint space control problem the desired trajectory X_d in the Cartesian space is transformed to the corresponding trajectory q_d in the joint space via inverse kinematics. Then, a joint space control scheme is designed that allows the actual motion q to track the reference inputs. In this case, the encoders measure the actual joint angles to create feedback, see Fig. 5.1.

On the other hand, the operational space control method uses measured trajectory of the end-effector in the operational space as feedback. Then using an operational control design to track the desired trajectory of the end-effector, see Fig. 5.2. Although it seems this method act directly on operational space variables, this is somewhat only a potential advantage. Since measurement of operational space variables is often performed not directly, but through the evaluation of direct kinematics functions starting from measured joint space variables. It is worth to considering that the operational space control problem follows a global approach that requires a greater algorithmic complexity; notice that inverse kinematics is now embedded into the feedback control loop.

Therefore the joint space control method is more suitable for motion control while the operational space control is often used for the case that control of interaction with environment is important. Here the joint space control scheme is used to control our robot walking.

From another point of view, the motion control method are generally divided into two main groups; decentralized and centralized method. In the decentralized method, each joint is controlled independently while in centralized controller the joints are controlled considering the dynamic effect of joint in each other.

Choosing the suitable control strategy is the very important that results in different performance. The driving system of the joints has an important effect on the type of control

strategy used. If a manipulator is actuated by electric motors with reduction gears of high ratios (as in the case of our biped robot), the presence of gears tends to linearize system dynamics, and thus to decouple the joints in view of the reduction of nonlinearity effects. This issue will be explained in detail in the next subsection.

5.1 Decentralized control

In order to explain the decentralized control method, It is worth to briefly analyze the dynamic behavior of the biped robot. The general equation of motion of a manipulator can be described as follows: (Siciliano et al., 2009)

$$\mathbf{B}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{F}_v\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} \quad 5-1$$

where \mathbf{q} is a vector of a set of variables $q_i, i = 1, \dots, n$ termed generalized coordinates which are chosen such that completely describe the link position of n -DOF manipulator, $\boldsymbol{\tau}$ is a vector of actuator torques, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}$ is a vector of the Coriolis and centrifugal forces, $\mathbf{g}(\mathbf{q})$ is a vector represents gravity forces, $\mathbf{B}(\mathbf{q})$ is the inertia matrix and $\mathbf{F}_v\dot{\mathbf{q}}$ is a vector of viscous friction torques. As it is clear from equation of motion, the system generally consist of the multi inputs, i.e. the joint torques ($\boldsymbol{\tau}$), and the multi outputs, i.e. the generalized coordinates (\mathbf{q}).

The objective of the joint space control scheme is to find proper joint torque vector such that the generalized coordinates vector \mathbf{q} tracks the vector of desired joint trajectory \mathbf{q}_d . The joint torques are produced by actuators thorough transmission. In our biped robot, harmonic drives for brushless DC motors and planetary gears for brushed DC motors transmit torques from motors to joints. In order to embed the transmission equations into the equation of motion, let \mathbf{q}_m be the vector of joint actuator angle. Assuming a rigid transmission without backlash we have

$$\mathbf{K}_r\mathbf{q} = \mathbf{q}_m \quad 5-2$$

where \mathbf{K}_r is an $(n \times n)$ diagonal matrix, whose elements are gear ratios corresponding to each joint's harmonic drive or planetary gear. As mentioned in the chapter 2, the gear ratios of harmonic drives and planetary gears used in Archie are 160 and 415, respectively. Since each joint has its own actuator and transmission system, the kinematic relations of transmission are not coupled and the matrix \mathbf{K}_r is diagonal.

On the other hand, if $\boldsymbol{\tau}_m$ is a vector of the actuator torques, the following relation can be derived

$$\boldsymbol{\tau}_m = \mathbf{K}_r^{-1}\boldsymbol{\tau} \quad 5-3$$

Now using above formulas, one can derive the equation of motion of manipulator equipped with mechanical drives at the motor side by the following equation:

$$\mathbf{K}_r^{-1}\mathbf{B}(\mathbf{q})\mathbf{K}_r^{-1}\ddot{\mathbf{q}}_m + \mathbf{K}_r^{-1}\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\mathbf{K}_r^{-1}\dot{\mathbf{q}}_m + \mathbf{K}_r^{-1}\mathbf{F}_v\mathbf{K}_r^{-1}\dot{\mathbf{q}}_m + \mathbf{K}_r^{-1}\mathbf{g}(\mathbf{q}) = \boldsymbol{\tau}_m \quad 5-4$$

The diagonal elements of the inertia matrix $\mathbf{B}(\mathbf{q})$ are made with two part; One is constant and the other is configuration dependent, i.e. trigonometric function of joint angles. Thus it can be divided as follows

$$\mathbf{B}(\mathbf{q}) = \bar{\mathbf{B}} + \Delta\mathbf{B}(\mathbf{q}) \quad 5-5$$

where $\bar{\mathbf{B}}$ is the average inertia matrix whose diagonal elements are constant and the others are zero. Substituting this equation in the general equation of motion gives following equation

$$\mathbf{K}_r^{-1}\bar{\mathbf{B}}\mathbf{K}_r^{-1}\ddot{\mathbf{q}}_m + \mathbf{K}_r^{-1}\mathbf{F}_v\mathbf{K}_r^{-1}\dot{\mathbf{q}}_m + \mathbf{D} = \boldsymbol{\tau}_m \quad 5-6$$

where

$$\mathbf{D} = \mathbf{K}_r^{-1}(\Delta\mathbf{B}(\mathbf{q})\mathbf{K}_r^{-1}\ddot{\mathbf{q}}_m + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\mathbf{K}_r^{-1}\dot{\mathbf{q}}_m + \mathbf{g}(\mathbf{q})) \quad 5-7$$

denotes the terms varying corresponding to different configurations.

Therefore $\ddot{\mathbf{q}}_m$ can be found by

$$\ddot{\mathbf{q}}_m = \mathbf{K}_r\bar{\mathbf{B}}^{-1}\mathbf{K}_r(\boldsymbol{\tau}_m - \mathbf{K}_r^{-1}\mathbf{F}_v\mathbf{K}_r^{-1}\dot{\mathbf{q}}_m - \mathbf{D}) \quad 5-8$$

As shown in Fig. 5.3, the dynamic model of the manipulator system with drives can be divided into two subsystem; The first system has two inputs which are actuator torques $\boldsymbol{\tau}_m$ and the disturbance vector \mathbf{D} and one output that is joint actuator angle. The second subsystem has three inputs as \mathbf{q}_m , $\dot{\mathbf{q}}_m$ and $\ddot{\mathbf{q}}_m$ and one output which is \mathbf{D} . It is obvious from formulas that with changing on each element of the input vectors of the first system, i.e. $\boldsymbol{\tau}_m$ and \mathbf{D} , the corresponding elements of the output \mathbf{q}_m will only be affected. In addition the relationship between the inputs and the output is linear for each joint. Therefore the first subsystem is linear and decoupled.

Conversely, in the second subsystem, change of input elements not only affects the corresponding element of the output, but also affects the other elements of the output. Hence, the second subsystem is coupled. Additionally, In this system the output is a nonlinear function of the inputs as described in the formula for \mathbf{D} .

In order to control such a system depicted in Fig. 5.3, many control strategy have been proposed. One method is to view the problem as a multi-input multi-output system, i.e. considering the whole system as a one block, that leads to a centralized control strategy. In this manner, the complete dynamic model of the system is needed. Therefore any uncertainty and imperfectness on the dynamic model of the system leads to error in tracking the desired trajectory. This method requires complex and time consuming computation to solve the inverse dynamic problem that is not suitable for real time implementation. But this drawback can be overcome using high speed processors.

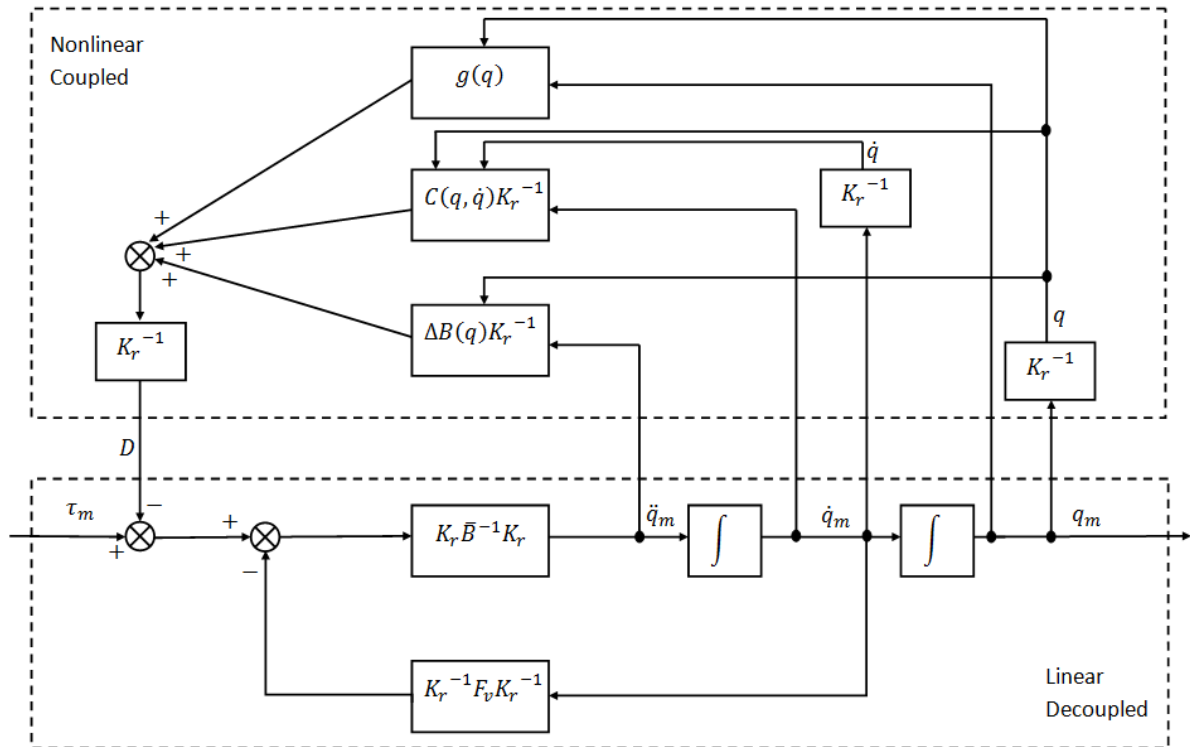


Fig. 5.3 Block diagram for equation of motion of manipulator and drive(Siciliano et al., 2009)

On the other hand, it is clear that the second input to this subsystem, i.e. \mathbf{D} , is divided by the gear ratio (inverse of diagonal matrix \mathbf{K}_r is a diagonal matrix whose elements are inverse of gear ratios of corresponding joint). Consequently, in the case of high gear ratio, the second input is small compared to the first input to this subsystem. In this case, the system can be regarded as a single-input single-output system with a disturbance of d_i (the i th element of the disturbance matrix \mathbf{D} corresponding to the i th joint). Since each joint is controlled independently, the system is called decentralized controlled system. The decentralized control method is used for Archie due to the presence of the gear transmission with high gear ratio that convert system to multiple single-input single-output system.

Based on this method, the goal is to design a controller for each joint such that the output tracks any desired reference trajectory. In addition to the control signal, disturbance is also an input to the system. This uncontrolled input has its own effect on the behavior of the output. So the controller should be designed such that it can reduce the effect of the disturbance. Tracking and disturbance rejection are two important goals of the controller design in single-input single-output system.

In order to control the motor, the mathematical model of DC motor should be investigated. In this manner, next section present model the motor based on two different control views; Velocity control and torque control. In the next section, these two models are explained and compared.

5.2 Velocity versus torque control

As mentioned before, Archie is equipped with a combination of brushed (permanent-magnet) DC and brushless DC motors. These motors can be described with a same mathematical model. The armature (rotor) current can be described as the following differential equation:

$$L \frac{dI_a}{dt} + RI_a = V_a - V_g \quad 5-9$$

where L is the armature inductance, R is the armature resistance, V_a and I_a are voltage and current of the armature, respectively and V_g is the back emf (electromotive force) voltage generated in the armature of the motor which is proportional to the angular speed of the motor ω_m :

$$V_g = k_v \omega_m \quad 5-10$$

where k_v is the voltage constant that depends on the characteristic of the motor and the magnetic flux of the motor coil.

The torque produced by the motor τ_m is also proportional to the armature current I_a through torque constant k_t :

$$\tau_m = k_t I_a \quad 5-11$$

It can be noted that in the SI unit, the numerical values of k_t and k_v are the same (de silva, 2007).

Fig. 5.4 shows a set of torque-speed curves versus various applied voltage that is useful to determine the torque constant. The rated voltage V_r is corresponding to the blocked rotor (stall) torque τ_0 when motor is stalled.

The mechanical equation of motion of the rotor can be described as:

$$\tau_m = I_m \frac{d\omega_m}{dt} + F_m \omega_m + \tau_l \quad 5-12$$

where τ_l is the reaction torque exerted from manipulator, I_m and F_m represent the moment of inertia and the viscous friction coefficient at the motor shaft, respectively. By substituting and transforming to the Laplace domain we have the following motor equation:

$$k_t I_a = (sI_m + F_m) \omega_m + \tau_l \quad 5-13$$

$$V_a = (sL + R) I_a + k_v \omega_m \quad 5-14$$

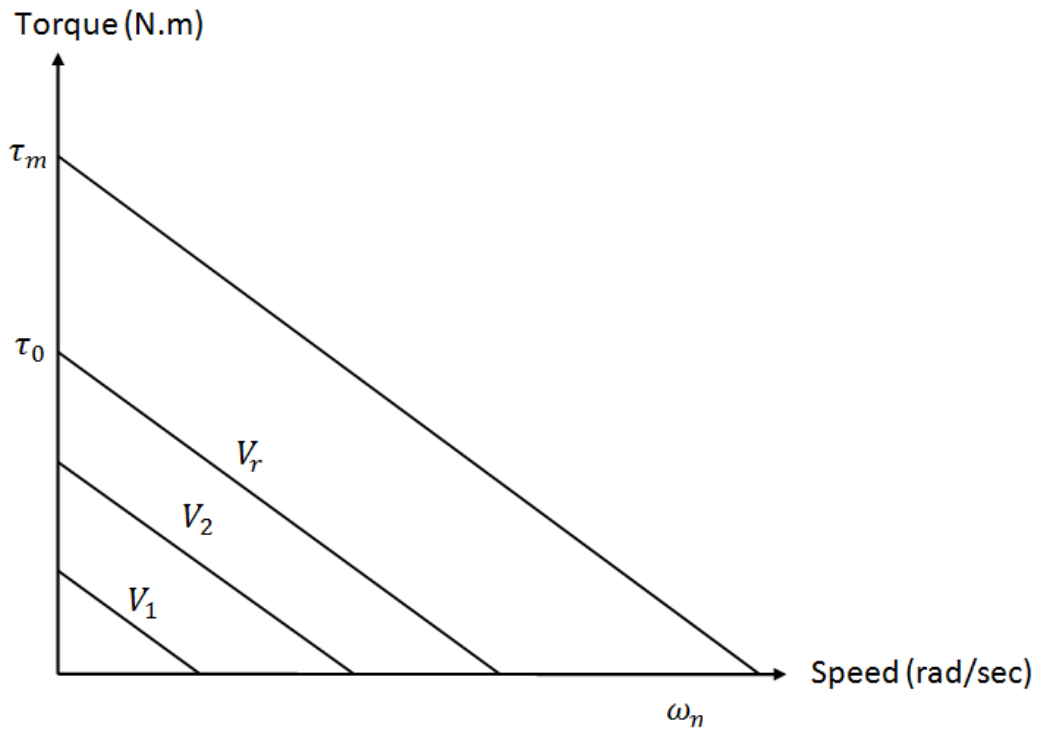


Fig. 5.4 General torque-speed curves of DC motor

Fig. 5.5 shows the block diagram of the above mentioned equation of DC motor in Laplace domain. Considering that the value of the electric time constant L/R is very small in comparison with mechanical time constant I_m/F_m leads to neglect the electric time constant in motor model. This is common assumption that is correct for the most of DC motors.

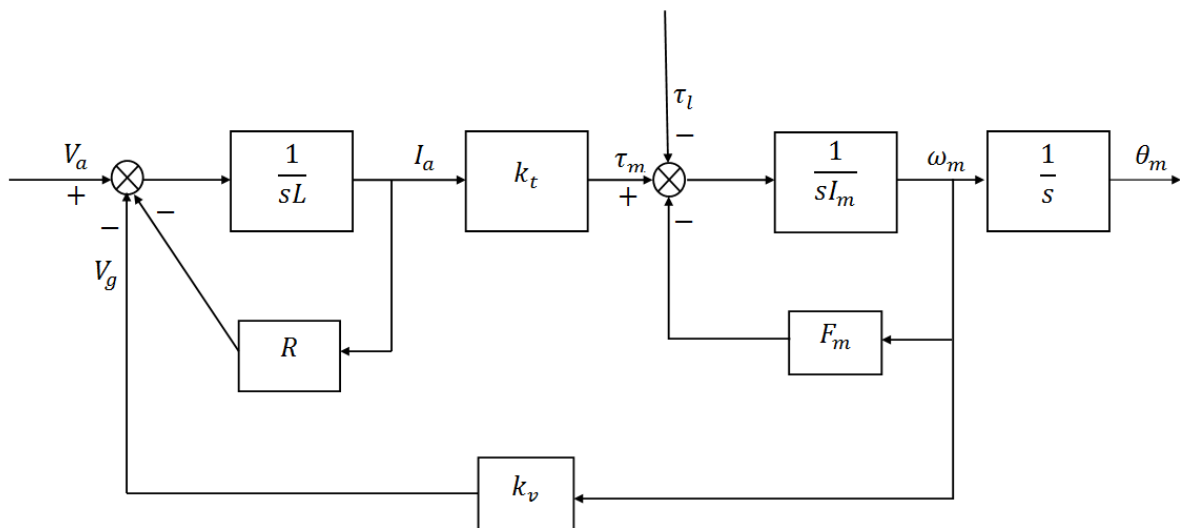


Fig. 5.5 Block diagram of DC motor

Now , the reduced order model of motor can be described in matrix form for whole manipulator by:

$$\mathbf{K}_r^{-1}\boldsymbol{\tau} = \mathbf{K}_t\mathbf{i}_a \quad 5-15$$

$$\mathbf{v}_a = \mathbf{R}\mathbf{i}_a + \mathbf{K}_v\dot{\mathbf{q}}_m \quad 5-16$$

where \mathbf{i}_a is a vector of the armature current, \mathbf{K}_t is the diagonal matrix whose elements are torque constants, \mathbf{v}_a is a vector of the armature voltage, \mathbf{K}_v is the diagonal matrix of the voltage constants, \mathbf{R} is the diagonal matrix of the armature resistances of the n motors.

Furthermore, any electric motor needs power amplifier (electric drive) to generate required voltage for the motor by amplifying the power source voltage. Then the relation between the vector of armature voltage \mathbf{v}_a and the vector of control voltage \mathbf{v}_c of the n servomotor can be written as

$$\mathbf{v}_a = \mathbf{G}_v\mathbf{v}_c \quad 5-17$$

where \mathbf{G}_v is the diagonal matrix of gains of the n amplifiers. Substituting the last three equations on each other leads to find the vector of joint torques as:

$$\boldsymbol{\tau} = \mathbf{K}_r\mathbf{K}_t\mathbf{R}^{-1}(\mathbf{G}_v\mathbf{v}_c - \mathbf{K}_v\mathbf{K}_r\dot{\mathbf{q}}) \quad 5-18$$

The block diagram of the system are depicted in Fig. 5.6.

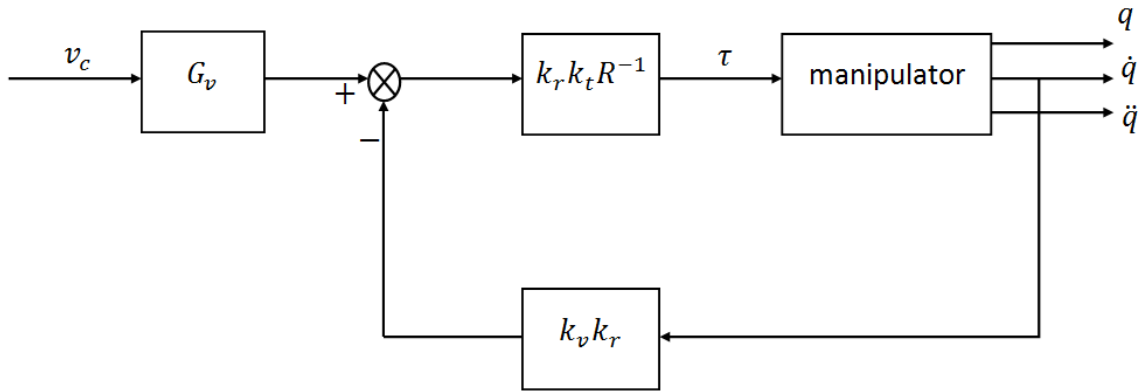


Fig. 5.6 Block diagram of the manipulator and motor system as voltage-controlled system(Siciliano et al., 2009)

Now, one can specify the required control voltage corresponding to \mathbf{v}_c corresponding to the desired joint velocity vector $\dot{\mathbf{q}}_d$ and desired vector of joint torques $\boldsymbol{\tau}_d$ based on the following relation:

$$\mathbf{v}_c = \mathbf{G}_v^{-1}\mathbf{K}_r^{-1}\mathbf{K}_t^{-1}\mathbf{R}\boldsymbol{\tau}_d + \mathbf{G}_v^{-1}\mathbf{K}_v\mathbf{K}_r\dot{\mathbf{q}}_d \quad 5-19$$

To control the above system two cases can be taken into account:

1. The first case is when the gear ratio of drives, i.e. diagonal elements of the matrix \mathbf{K}_r are large and the values of motor resistance, i.e. elements of the matrix \mathbf{R} are very small and assuming the values of the joint torques required for performing the desired motion are not very large. In this case the first term of the control voltage $\mathbf{G}_v^{-1}\mathbf{K}_r^{-1}\mathbf{K}_t^{-1}\mathbf{R}\boldsymbol{\tau}_d$ is negligible. Then control voltage is reduced to

$$\mathbf{v}_c \approx \mathbf{G}_v^{-1}\mathbf{K}_v\mathbf{K}_r\dot{\mathbf{q}}_d \quad 5-20$$

This scheme is called velocity or voltage control method since the control voltage is achieved according to the desired velocity. It is noted that the relationship between voltage \mathbf{v}_c and speed $\dot{\mathbf{q}}_d$ is independent of manipulator parameters. Therefore the control system is robust with respect to the parameter variation of manipulator model. The greater the value of the gear ratios, the more robust to the parameter variation.

Furthermore, due to the fact that the matrix $\mathbf{G}_v^{-1}\mathbf{K}_v\mathbf{K}_r$ is diagonal, the control voltage of each joint only depends on the speed of the same joint and not depends on the speed of the other joints. Hence a decentralized control method can be used for position control of the joint in this case, since each joint can be controlled independently.

2. In the second case, on the other hand, the required joint torques for desired motion control are large or the system is direct-drive ($\mathbf{K}_r = \mathbf{I}$). In this case, the first term of control voltage is no longer negligible. Therefore it is needed to specify the required joint torques $\boldsymbol{\tau}_d(t)$ to track any desired motion in terms of the joint accelerations $\ddot{\mathbf{q}}(t)$, velocities $\dot{\mathbf{q}}(t)$ and positions $\mathbf{q}(t)$. This can be done using inverse dynamics technique that requires the accurate knowledge of the manipulator dynamic model. In this manner, the control system is needed to be a centralized control method, because computing the torque history at each joint requires to know the time evolution of the motion of all the joints.

According to the above mentioned relation, the control voltage in this case is determined based on the desired torque values and desired joint velocities:

$$\mathbf{v}_c = \mathbf{G}_v^{-1}\mathbf{K}_r^{-1}\mathbf{K}_t^{-1}\mathbf{R}\boldsymbol{\tau}_d + \mathbf{G}_v^{-1}\mathbf{K}_v\mathbf{K}_r\dot{\mathbf{q}}_d \quad 5-21$$

Since the matrices \mathbf{K}_t^{-1} , \mathbf{K}_v and \mathbf{R} are related to the characteristic of the motor and changing according to the different operation conditions of the motors, the motor control can be described as a current control instead of voltage control. In this case, the control system is less sensitive to the parameter variations of the motors. The equation of the actuator that perform as a torque-controlled generator can be specified as:

$$\mathbf{i}_a = \mathbf{G}_i\mathbf{v}_c \quad 5-22$$

where \mathbf{G}_i is the constant diagonal matrix relates the armature currents \mathbf{i}_a and the control voltages \mathbf{v}_c . Therefore the joint torques are derived as

$$\boldsymbol{\tau} = \mathbf{K}_r \mathbf{K}_t \mathbf{G}_i \mathbf{v}_c \quad 5-23$$

As a consequence, the vector of the voltage control for a torque-controlled method is obtained by

$$\mathbf{v}_c = \mathbf{G}_i^{-1} \mathbf{K}_r^{-1} \mathbf{K}_t^{-1} \boldsymbol{\tau}_d \quad 5-24$$

where desired joint torques $\boldsymbol{\tau}_d$ are obtained by computation of the inverse dynamics for a desired motion. Although, the centralized control method seems to be a feedforward system, the use of error between the actual and desired trajectory is necessary. Because the dynamic model of the system, even though a very complicated one, is anyhow idealization of reality which does not include dimension tolerances, friction and gear backlash and also uses simplified assumption such as link rigidity.

With comparing two cases, The first case assumptions are more compatible with Archie characteristic, since the values of gear ratios used in Archie are large, i.e. 160 for brushless and 450 for brushed DC motor, and the resistance of utilized DC motors are very small and also the required joint torques needed for walking are not very large. Thus velocity control (voltage control) method is used in each joint of Archie that is compatible with decentralized control strategy. So the next section explains the independent joint control strategies used for decentralized control.

5.3 Independent joint control

The objective of this section, is to design a compensator for a single-input single-output (SISO) system such that the actual trajectory can track the desired one while reject the effects of the disturbance. Fig. 5.7 shows the general diagram of SISO feedback control system.

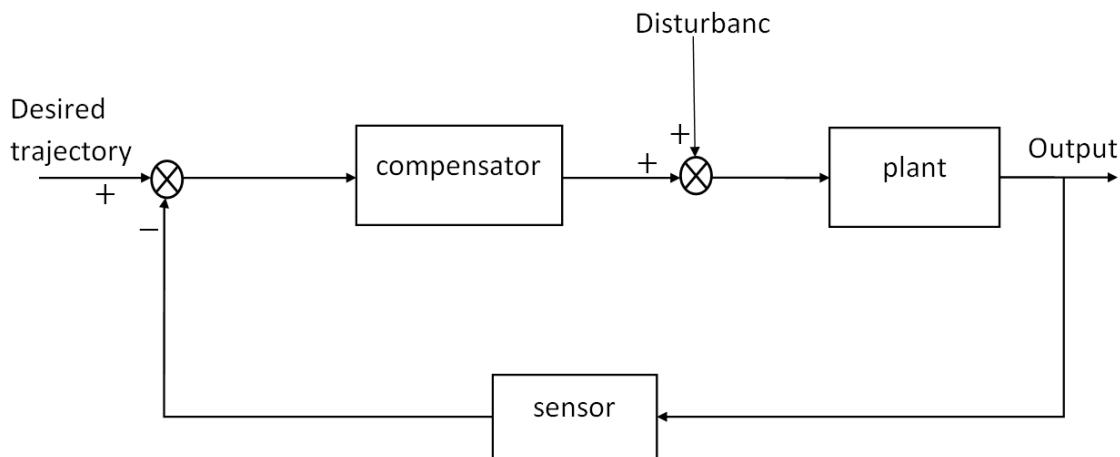


Fig. 5.7 General block diagram of feedback control system for SISO system

It is worth to mention that all parameters in this section are scalar rather than vector, since each joint is controlled independently. Therefore, the reference trajectory is desired joint angle θ_{md} and the output trajectory is actual joint angle of each motor θ_m . It is noted that θ_m and θ_{md} are the actual and desired rotation angle of the motor shaft that are defined as follows

$$\theta_m = k_r \theta \quad 5-25$$

$$\theta_{md} = k_r \theta_d \quad 5-26$$

The disturbance also is a scalar value denoted by d , which is the effect of the dynamic behavior of the links of manipulator.

In order to reach twin goal of the control system for rejecting the disturbance and tracking desired trajectory, two basic considerations should be taken into account; First, the amplifier gain should be a large value acting before the point of entering of the disturbance. Second, in order to vanish the steady state of the error generated due to the gravitational effect, the controller should be designed to have integral action. These requirements leads to use the simplest and most effective control method that is proportional-integral (PI) controller. The PI controller transfer function is

$$C(s) = K_p + \frac{K_I}{s} \quad 5-27$$

where K_p is the proportional gain and K_I is the Integral gain.

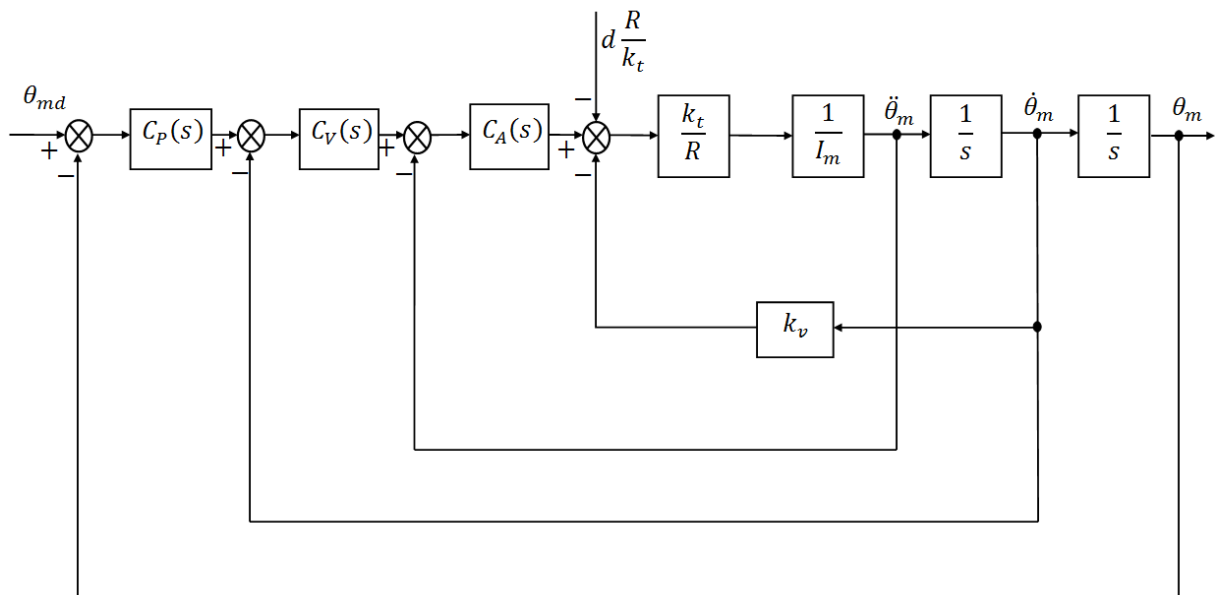


Fig. 5.8 General block diagram of cascaded control method for independent joint control

The performance of the controlled system can be improved using more local feedback loops closed around the disturbance (Siciliano et al., 2009). This method is called cascaded control

loop method. In addition to the position feedback loop, the velocity and the acceleration closed-loop can be used as inner cascaded loops. The block diagram of cascaded control method which uses all the three possibilities for closed-loop as position, velocity and acceleration is shown in Fig. 5.8. In this figure, $C_P(s)$, $C_V(s)$ and $C_A(s)$ denote the transfer functions for position, velocity and acceleration controller, respectively. It is noted that in this scheme, the inmost controller should be PI controller to cancel the effect of constant disturbances at steady state condition. As shown in Fig. 5.8, the term R/k_t is multiplied by disturbance torque d to transform it into voltage rather than torque.

In the case that the desired signal and disturbance are time varying, as in the case of the walking control of biped robot, feedforward compensation is required. In this manner, the tracking error will be reduced. Fig. 5.9 shows the feedforward compensation when the transfer function of cascaded feedback controllers are as follows

$$C_P(s) = K_P \tag{5-28}$$

$$C_V(s) = K_V \tag{5-29}$$

$$C_A(s) = K_A \frac{1+sT_A}{s} \tag{5-30}$$

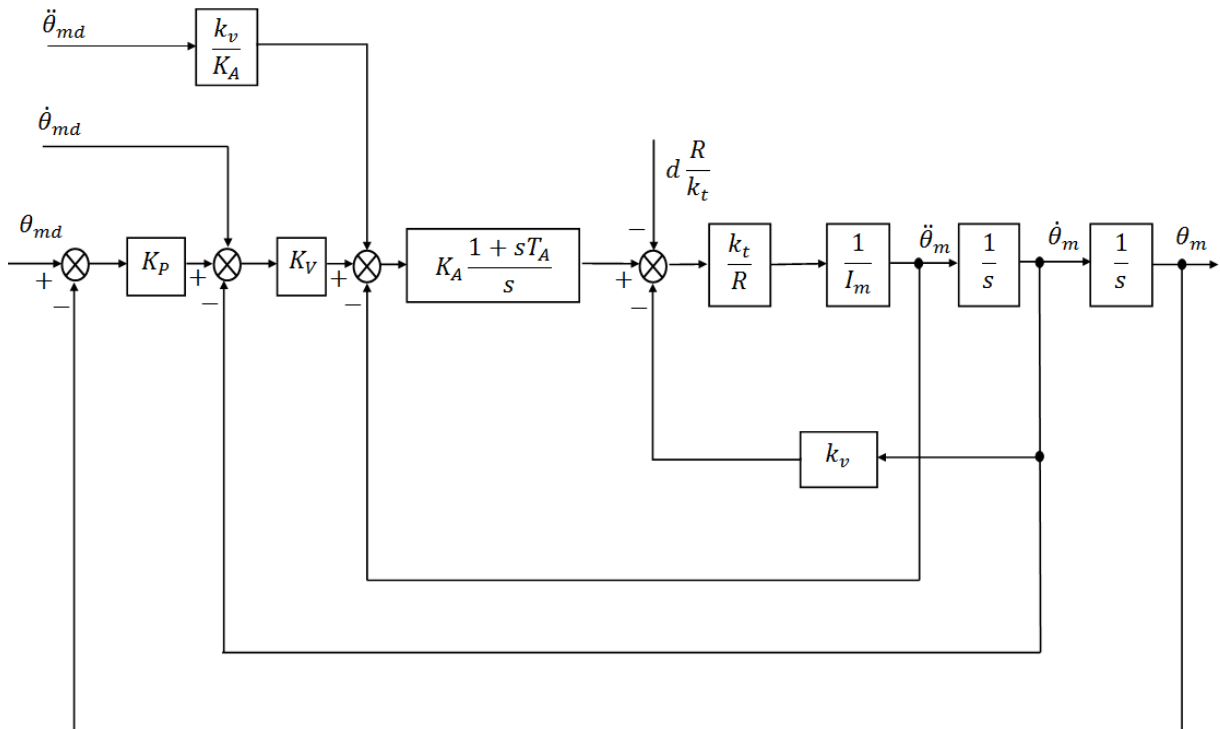


Fig. 5.9 Block diagram of feedforward compensation added to position, velocity and acceleration feedback controller

5.4 Biped walking control system

The overall decentralized control system is illustrated in Fig. 5.10. In this scheme, the walking pattern planner proposed in the chapter 4 is used to find the trajectories for the left and the right foot and also the torso corresponding to the desired walking parameters. Then the joints angle trajectories are found using the inverse kinematics derived in the chapter 3. In fact, the Inverse kinematics transformed the desired walking trajectories from Cartesian to joint space. Therefore, the outputs of the inverse kinematics are desired joint angles θ_{d1} to θ_{d6} for both legs. The reference trajectories for the independent joint controller of the biped robot are obtained by multiplying desired joint angle θ_{di} by the corresponding gear ratio k_{ri} . The reference trajectories are actually the rotation angles of the motors θ_{mdi} .

After that, the independent joint controller is used to control each joint such that the output tracks the desired motor joint angle. As shown in Fig. 5.11, The proposed controller uses the cascaded feedback loop of position and velocity to produce the appropriate control voltage v_{ci} according to the desired motor angle θ_{mdi} . Although, the acceleration feedback loop also can be used to improve the controller performance, the direct or indirect measurement of acceleration is not available in this project. Then a position feedback is obtained in each joint via incremental encoder. The velocity feedback is constructed by a speed estimator algorithm provided in the Elmo motion driver.

Furthermore, feedforward compensation obtained by first and second derivative of the desired trajectory of motor angle is utilized to decrease the tracking error for the varying reference trajectory. As mentioned before, the inmost controller, here velocity controller, should be a PI controller. Therefore, in this scheme, the position loop has a P controller while the velocity loop contains a PI controller. Due to the limitation of physical parameters such as voltage or velocity, the saturation is integrated into control system to avoid unpredicted problem.

The disturbance is due to the Coriolis , centrifugal and gravity forces are derived based on the formula mentioned in section 1 of this chapter. In Fig. 5.10, the block denoted by transfer function \mathbf{N} , is nonlinear and coupled part of the manipulator relations shown in upper part of Fig. 5.3. The disturbance \mathbf{D} is obtained as the nonlinear function of the motor angle vector \mathbf{q}_m , the motor velocity vector $\dot{\mathbf{q}}_m$ and the motor acceleration vector $\ddot{\mathbf{q}}_m$. Therefore, the disturbance torque exerted on the i th joint d_i by other links is due to the dynamic behavior of the manipulator and in general depends on the joint angles of all other joint. In order to transform disturbance torque to disturbance voltage, it is multiplied by R_i/k_{ti} , see Fig. 5.10.

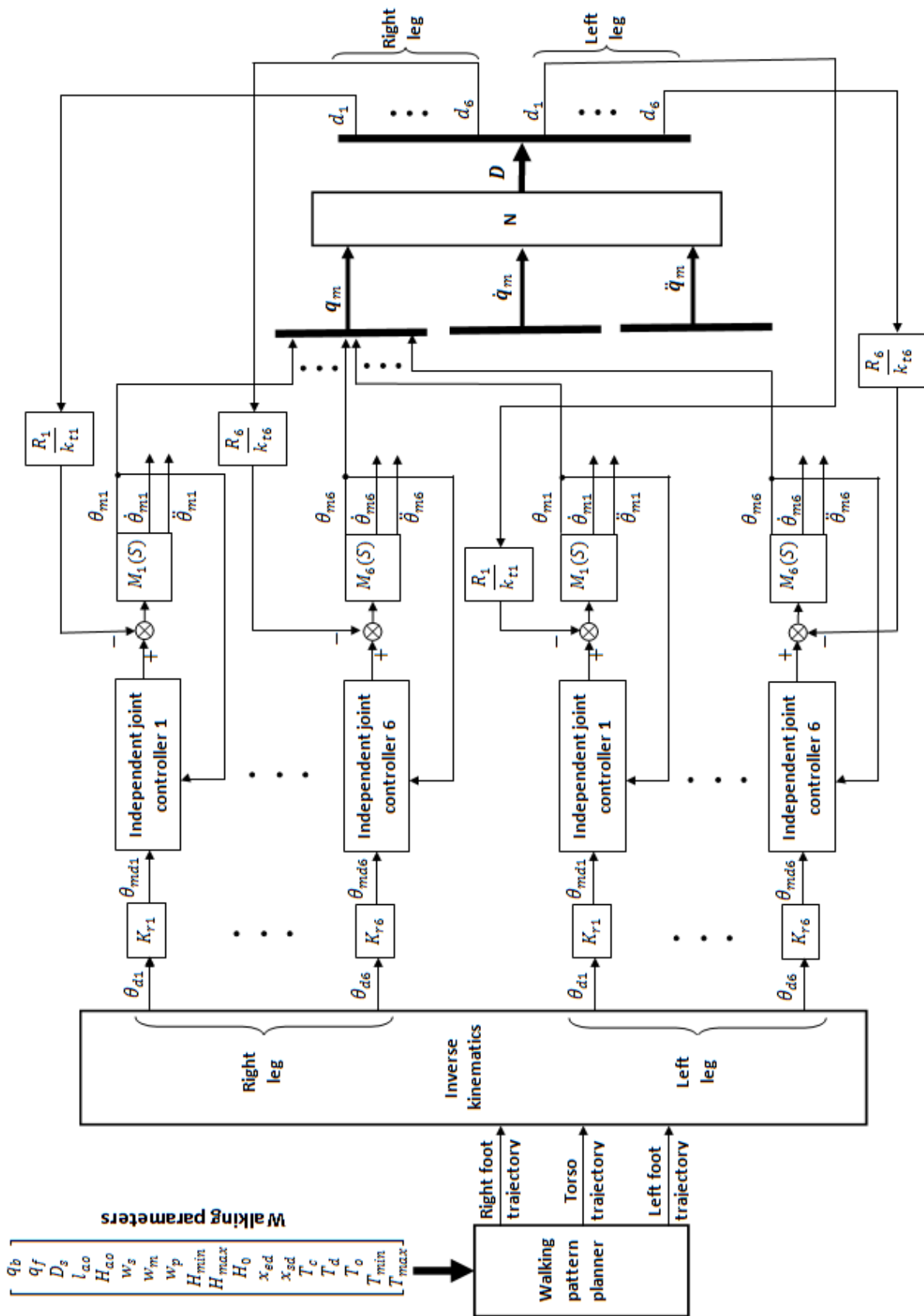


Fig. 5.10 Overall control system of the biped robot

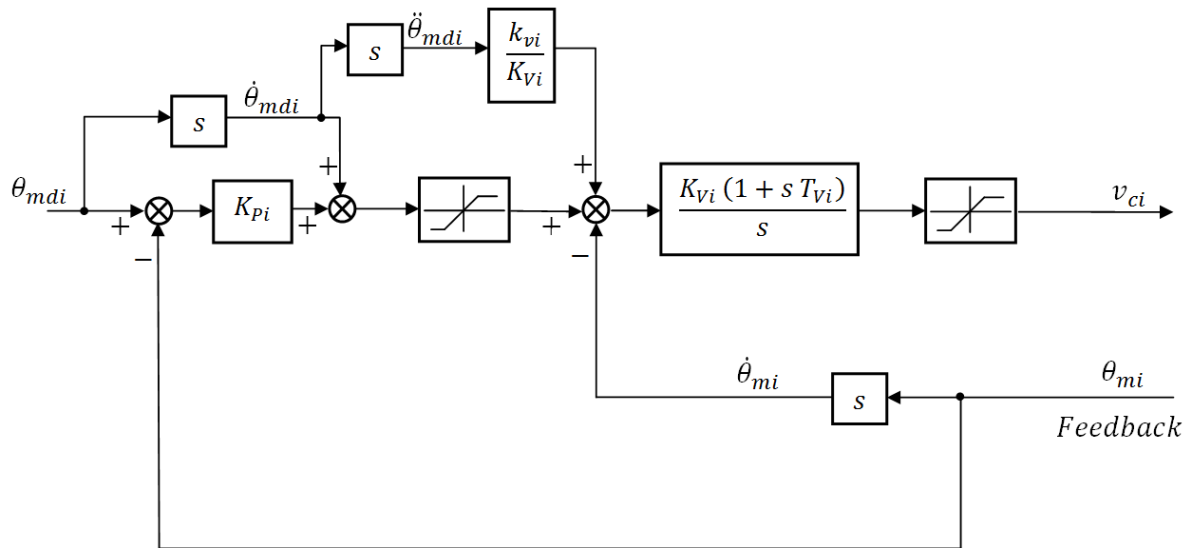


Fig. 5.11 Block diagram of independent joint control for the i th joint

Now the control voltage minus the disturbance voltage is applied to the motor. The transfer function of electric motor and transmission and dynamic of link, i.e. its linear and decoupled part, that is denoted in Fig. 5.10 by $M_i(s)$ for the i th joint is shown in detail in Fig. 5.12.

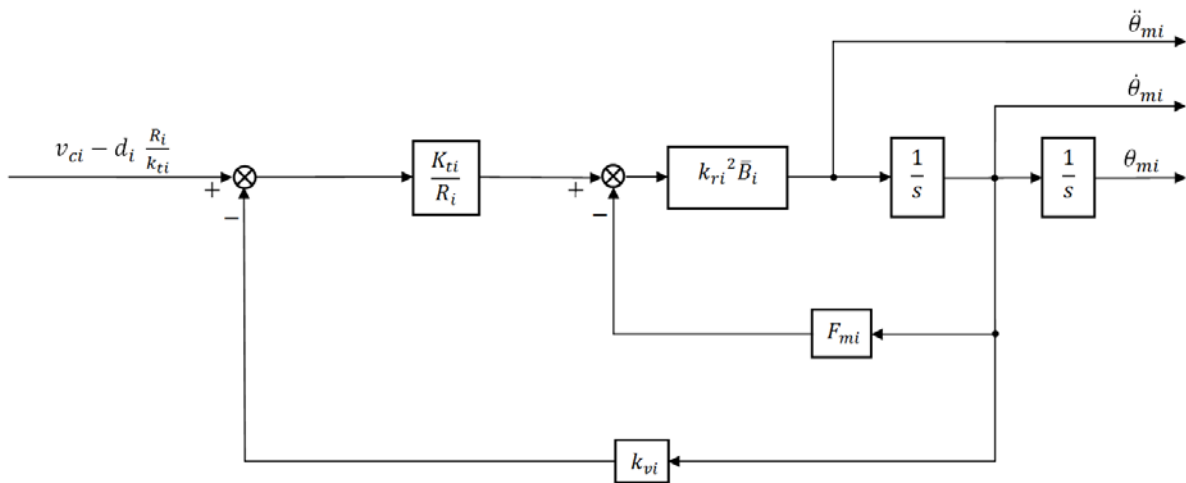


Fig. 5.12 Block diagram of the motor and linear part of manipulat

Chapter 6 Implementation and results

In the previous chapters the relations required for the proposed control method for walking of the biped robot were completely explained. This chapter aims to explain the implementation procedure of the presented controller for the biped robot, Archie, which its structure as well as its electric parts were described in the chapter 2. Furthermore, Archie walking realization results are presented.

6.1 Implementation

In order to implement the proposed controller for Archie, a C++ program was developed in Linux. As shown in Fig. 6.1, The program consists of the interface that enables the user to set the walking parameters, i.e. the parameters introduced in chapter 4 for walking pattern generation. In addition, user can select the favorite joints to move in this interface. For more details of interface structure and performance see (Dezfouli, 2013).

Additionally, the developed program contains of a calculation part. This part is responsible for calculating the joint angles corresponding to the selected walking parameters using the relations derived for the inverse kinematics and pattern generation. It is noted in this scheme, first the trajectories of the both feet and the torso are planned according to the desired parameters in the Cartesian space. Thereafter, the desired joint angles are calculated in joint space based on the closed-form solution of inverse kinematics presented in the chapter 3.

In order to use the desired joint trajectories as reference values for the digital independent joint controllers, the joint angle trajectories should be discretized. The discretization procedure was explained in the chapter 3.

After that, the calculated discrete values of desired joint angles should be converted into the format to use as the reference trajectories for the independent joint controllers. An Industrial controller named Elmo motion controller is used as the joint controller to control each joint movement independently according to the reference values at the specified time. Therefore, A position-time (PT) Table is used based on the predefined subroutines presented by Elmo motion controller factory (www.elmomc.com, Last visit October 2013). This table will be explained in the following section.

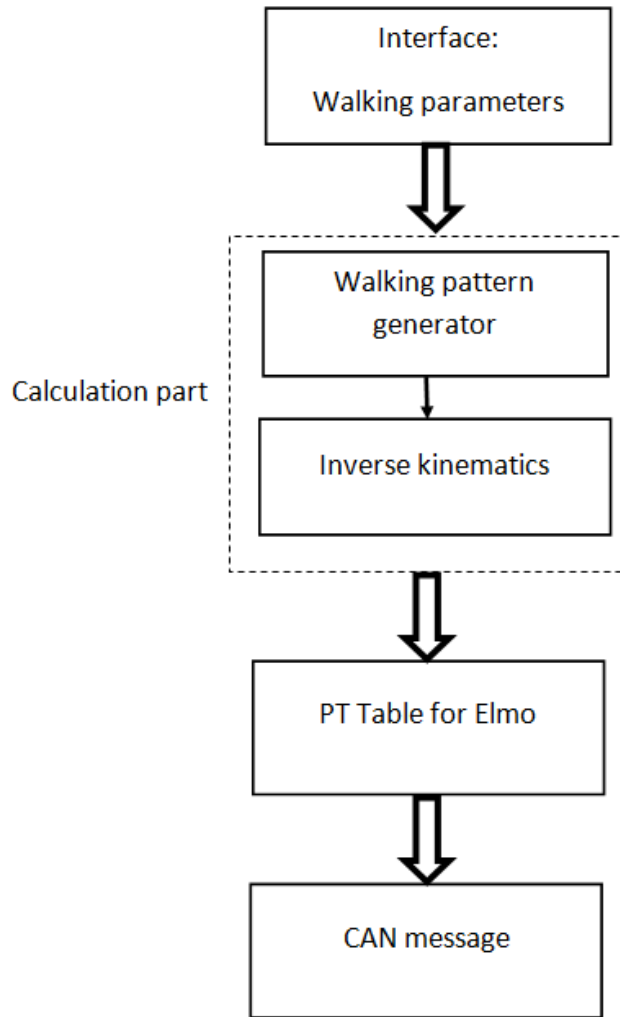


Fig. 6.1 Control program algorithm

Beside the format of the utilized controller, the desired joint angles should meet the required binary format of the utilized communication network in Archie, i.e. CAN communication network. The details of CAN message format is described in the reference: (Dezfouli, 2013).

6.1.1 PT table:

In order to control synchronously motion of all joints, PT motion mode of the Elmo motion controller is used. In this mode, a sequences of reference position with equal time spaces to be visited by the motor can be specified. The sample time used in PT motion should be selected as an integer value multiplied with the position control loop sample time. The Elmo drive interpolate smooth trajectory between the user specified positions.

In fact, in PT motion, a third order polynominal interpolate the values for the points between positions provide by the user. The sample time T of the discrete PT trajectory is defined as

$$T = m T_s \quad 6-1$$

where m is selected by user as an integer value ranged between 1 to 256 and T_s is the sample time of position controller in Elmo drive. This value depends on the parameters of the position controller which are fixed after tuning. The value of position controller sampling time is calculated by Elmo drive. So with changing the values of m , one can change the sample time.

In addition to the sample time, the vector QP defines the position of N points, where N is the number of sampled point in the reference joint angle trajectory. Since the joints angle trajectories are derived for a complete cycle in a duration of $2T_c$ then the following relation should satisfy

$$N T = 2T_c \quad 6-2$$

Then number of discrete points N and the integer value m will be found. It is worth to mention that the greater value selected for N , the more precise trajectory of desired joint angle. On the other hand the sending time of the vector QP to each drive via CAN communication will be increased.

Fig. 6.2 shows the flowchart of decision making in PT mode to construct the desired joint angle values for position controller. In this scheme, a read pointer n is defined. This pointer specify that which element of the vector QP is read. Since the sample time of position controller T_s is less than the sampling time of desired trajectory T , in the case $m > 1$, the $m - 1$ points should be found using interpolation. For example when the read pointer $n = 5$, the present motion segment starts at position $QP[5]$ and ends at $QP[6]$. After m control sample time, i.e. $m \times T_s$, the drive increments the read pointer to $n + 1$ and reads $QP[n + 2]$ to calculate the parameters of the next motion segment. Using cyclic motion ability, after N sample time, i.e. $N \times T$, read pointer n starts again from 1 and repeat the procedure. This leads to cyclic motion of walking that repeat the predefined planned trajectories over and over.

6.1.2 Controller tuning:

As stated before, the decentralized strategy is used to control walking of Archie. In this manner, the independent joint controller is responsible to control tracking the reference joint angle in each joint. The Elmo motion controller which is an industrial drive is utilized to produce the proper voltage for each motor based on the cascaded control method. In this scheme, the inner loop is a velocity close loop with a PI compensator while the outer loop is a position closed loop with a P compensator. These controller gains should be tuned according to the motor and link parameters. The Elmo motion controller provides both automatic and manual tuning. The proposed method for tuning controller gains in Elmo is a heuristic method that is based on compromising between the rise time, the settling time and the overshoot resulted from the step response.

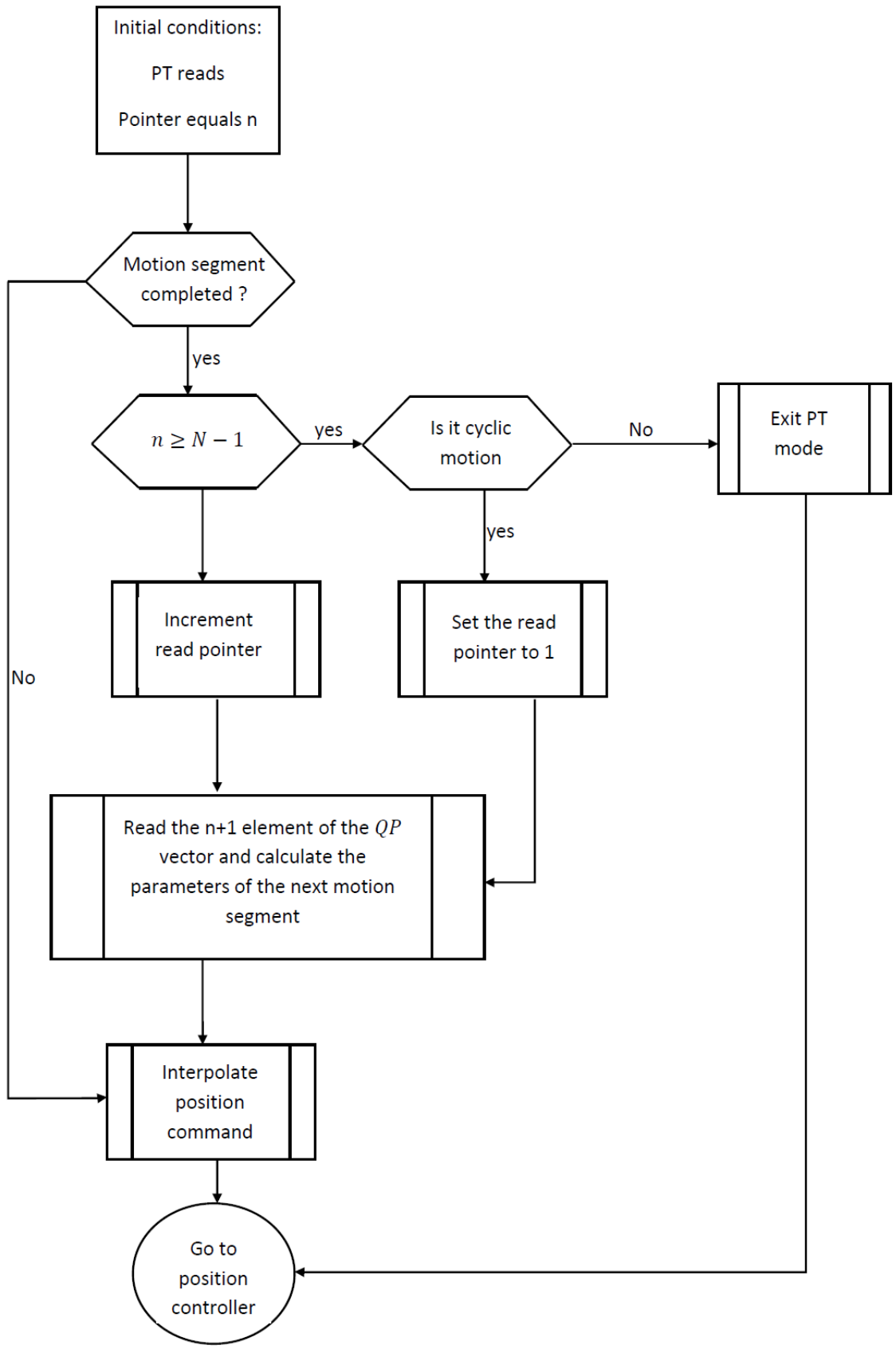


Fig. 6.2 PT motion decisions flowchart

The overshoot level, as well as the ratio between the rise time and the settling time, reflect the gain and the phase margins. Gain or phase margin results that are too low may result in a high step response overshoot (more than 40%) followed by an undershoot and a long settling time. If the phase margin is too high, the settling time is too long.

In order to tune the parameters of the cascaded controller, first the proportional K_p and integral K_I gains of the inner closed loop, i.e. the velocity loop, should specify and then the proportional gain K_p of the outer loop, i.e. the position closed loop, is determined.

Finding the best values of K_p and K_I for the velocity controller is based on an iterative process. In this process, the K_p and K_I are chosen and then the closed loop step response with selected gains is evaluated. Based on this evaluation, reselect the PI controller gains to improve the response of the closed loop system.

Now one example of joint controller tuning for our motor will be presented as follows:

First we start from the very low gain $K_p = 1 \text{ V}\cdot\text{sec}/\text{rad}$ and $K_I = 3 \text{ V}/\text{rad}$ to check the behavior of the velocity closed loop system. On the other hand, step reference command is selected by following parameters:

The record time is set long first, since with low gains the response is very slow. The maximum record time accepted by the Elmo drive is 0.48 seconds. The reference velocity is chosen 1200 rpm which is selected low for the first trial and after more confident of the motor response, it can be increased. Since the encoder used in Archie for this motor is 360 counts /revolution, the velocity is transformed to 72000 counts / second.

Fig. 6.3 shows the velocity step response and the motor current corresponding to the above mentioned PI controller parameters and reference. As expected, the response with these low gains is to sluggish. Form this figure, one can easily observe that the measured output velocity does not reach the reference step and the motor current is far from saturation.

For next trails, the K_p and K_I are increased simultaneously by 50% till the step response shows one of the following feature; the overshoot of about 20% occurs in the step response or the resonant oscillation observed in step response.

Fig. 6.4 shows the step response of the velocity closed loop corresponding to four different PI controller parameters. With choosing the PI controller parameters as $K_p = 40 \text{ V}\cdot\text{sec}/\text{rad}$ and $K_I = 120 \text{ V}/\text{rad}$, the step response has a overshoot about 20%. But the steady state error exists in the step response.

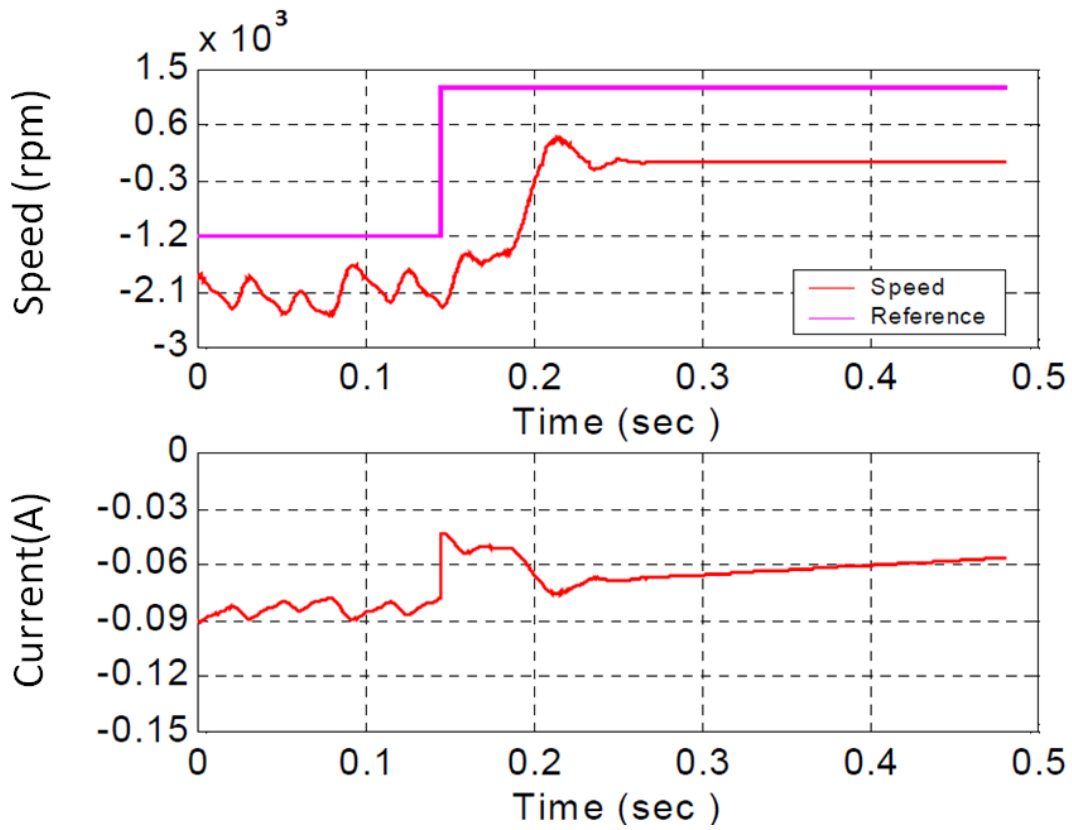


Fig. 6.3 Step response for first selected K_P and K_I

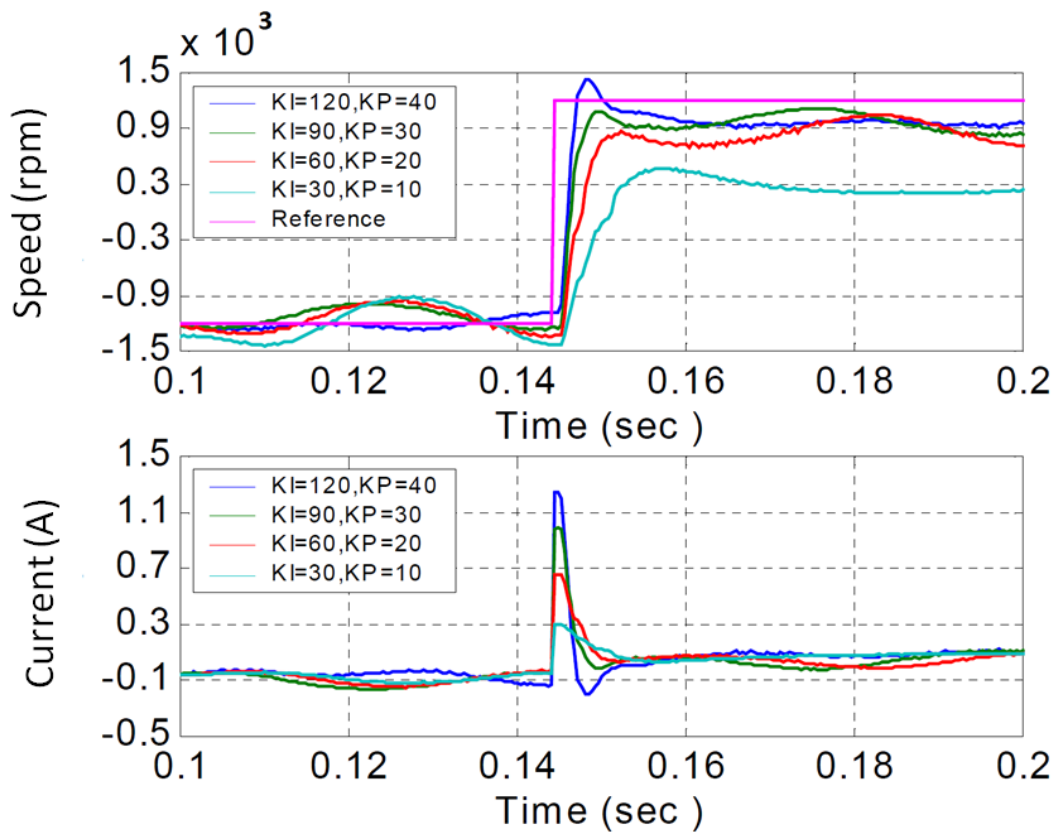


Fig. 6.4 Velocity step response for different values of K_P ($V \cdot sec/rad$) and K_I (V/rad)

In the next step, the new value for K_p is set by dividing the value of the last K_p over 3, i.e. $K_p = 30 \text{ V}\cdot\text{sec}/\text{rad}$. Then the test is performed with increasing the K_I by a factor of 1.3 at each trail. So in the first trail the Integral gain is selected as $K_I = 39 \text{ V}/\text{rad}$ and increasing in the next trails. It is noted, that the integral gain is responsible to cancel steady state error in the closed loop system. The increasing continues till the system exhibits an overshoot of about 30%. Fig. 6.5 shows the step response for increasing the K_I while K_p is fixed. The final value for the case that overshoot is about 30% is $K_I = 8000 \text{ V}/\text{rad}$.

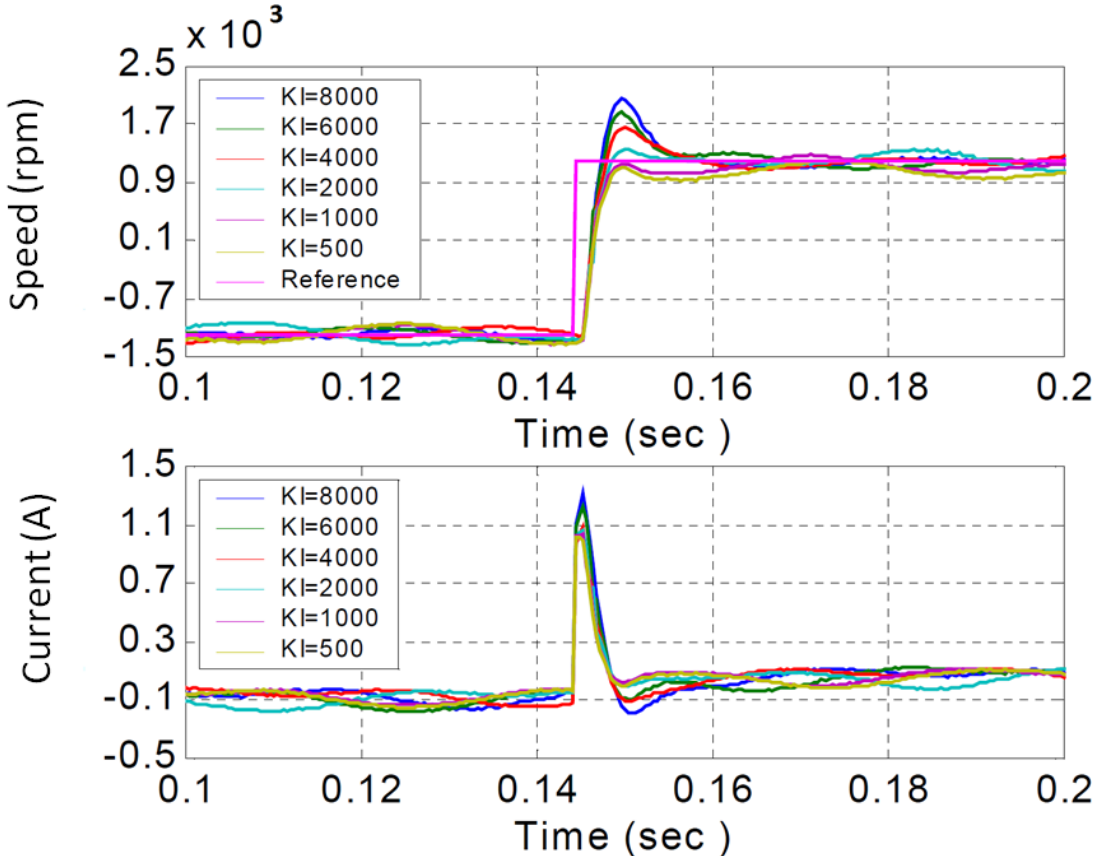


Fig. 6.5 Step response of velocity closed loop when $K_p = 30 \text{ V}\cdot\text{sec}/\text{rad}$ and $K_I(\text{V}/\text{rad})$ is varying

In the final step, the PI controller parameters can be tuned accurately to find the best step response of the velocity closed loop system. In this manner, K_p and K_I is changed only 10% in each trail and evaluating the step response to find the best controller parameters. It is worth to mention that although increasing the K_I is a good approach to find the best performance, too much increases leads to create unacceptable oscillations.

After fine tuning of the PI parameters, the optimal values for K_p and K_I are found as:

$$K_p = 45 \text{ V}\cdot\text{sec}/\text{rad} \quad \text{and} \quad K_I = 6800 \text{ V}/\text{rad}$$

The step response corresponding to these gains are depicted in Fig. 6.6. This response has about 25% overshoot and no undershoot as well as non-saturating current command.

After tuning the inner velocity controller loop gains, now it is desired to find the proportional gain K_p of the outer position controller. To do this, the heuristic method is used to tune the Position controller gain. Since the gain of the outer loop K_p decrease the phase margin, the integral gain of the outer loop K_I should be decreased from selected previous value to leave some extra phase margin for K_p of the outer loop. On the other hand, the K_p of the inner loop mostly specify the gain margin of the system.

Therefore the K_I of the inner loop is selected as the half of the value designed in previous stage but the K_p of the inner loop remains unchanged. Based on the estimate of the system's bandwidth, the first value for K_p of the outer loop is found by the following formula:

$$K_p = 0.5/dT \quad 6-3$$

where dT is the rise time for the step response of the velocity loop. It is noted that rise time is the time takes that the measured output reaches the 90% of the reference step. The rise time for the velocity loop of our case is about $dT = 0.0033 \text{ sec}$, see Fig. 6.8. Thus, the first estimation of K_p of the outer loop is $K_p = 150 \text{ sec}^{-1}$.

Then, the fine tuning of K_p of the outer loop and K_I of the inner loop can be done by trial and error. The iteration range for the K_I of the inner loop should not exceed 50% of its original value, and the range of iteration for the K_p of the outer loop should not exceed 100% of its original value.

Fig. 6.9 shows the position response for the total cascaded closed loop system with the optimal values of inner PI controller and outer P controller gains.

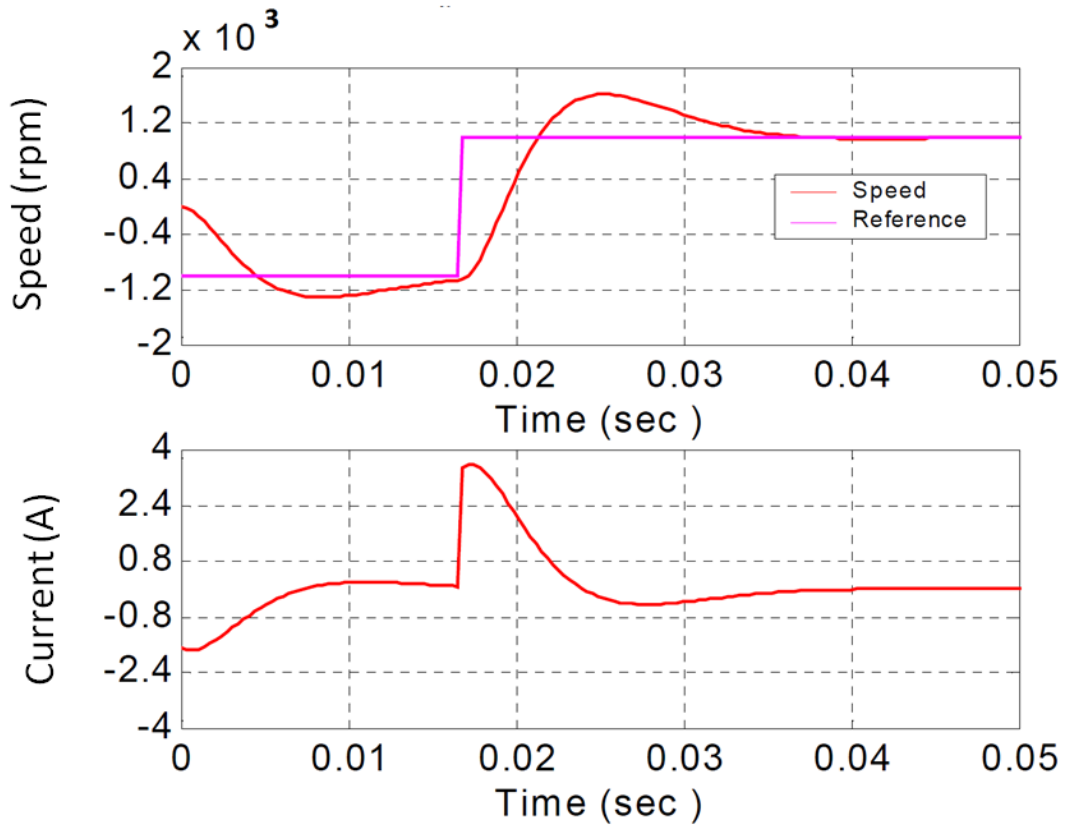


Fig. 6.6 Best step response for optimal values of K_P and K_I

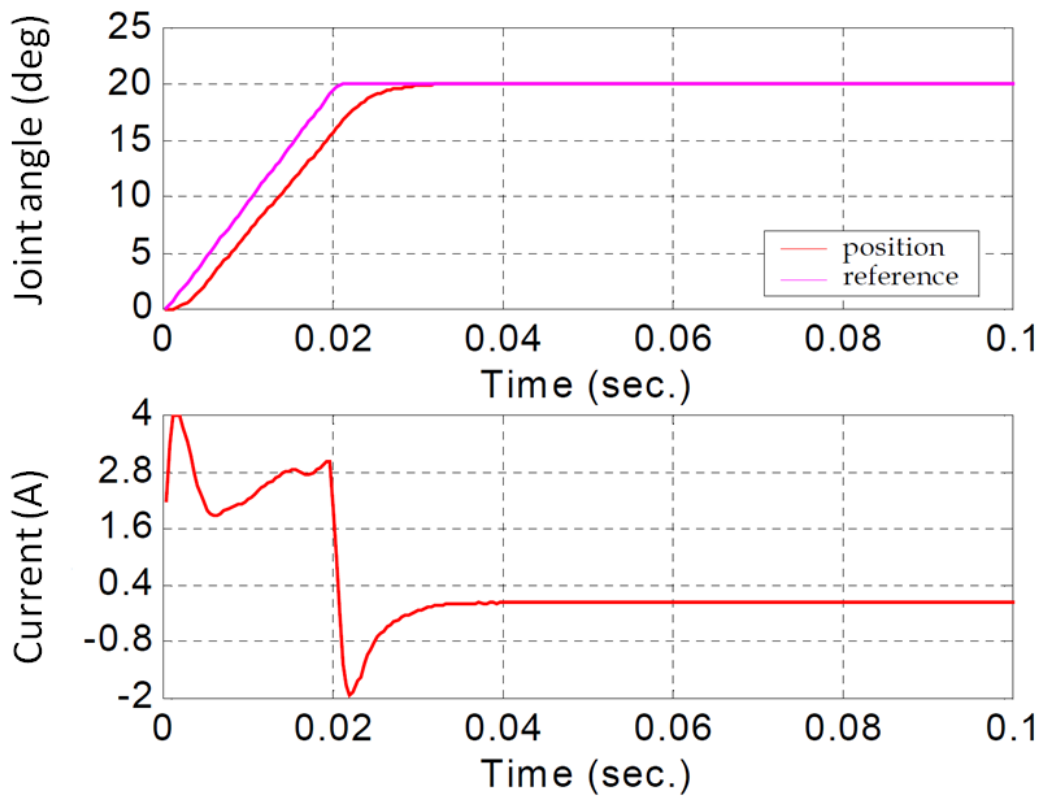


Fig. 6.7 Position step response of the cascaded close loop system

6.2 Walking realization

In the preceding section, the implementation procedure of the designed control system for the biped walking was explained. The proposed software enables us to command the actuators to track the desired trajectories. These desired trajectories are resulted from the pattern generator according to the desired walking parameters defined in the chapter 4.

Regarding stability of the biped walking robot, two types of walking are defined: static and dynamic walking. In static walking, the center of mass of the robot should be always located above the support foot area. If the robot moves slowly enough (thus the name static), by locating the center of mass above the support foot area during swing phase, the robot will be stable in walking. In this case, the torso should be transferred from rear to front leg during the double-support phase to ensure that the center of mass is located above the support foot area.

On the other hand, In dynamic walking, Zero Moment Point (ZMP) (Vukobratovic & Borovac, 2004), a point on the ground where the inertial and gravity moments cancel out resulting in a net zero moment, should be located strictly in the support foot area to maintain the stability of the robot. In order to control the position of ZMP in a closed-loop system, additional sensors such as the Inertia Measurement Unit (IMU) to measure the acceleration of the center of mass and the force sensor located in the sole of both feet to find center of pressure are required. Furthermore, the comprehensive dynamic modeling is needed to predict the varying location of the center of mass in different configurations of the robot as well as the inertial forces exerted to the center of mass.

Generally speaking, the static walking is a special case of the dynamic walking such that the inertial forces are small compared to the gravitational forces due to slow motion of the robot. Therefore, In static walking, the gravitational force specify the ZMP. In the static walking, the only way to maintain the ZMP inside the support foot area is to locate the center of mass above the support area. In this case, the resultant moment of forces about a point (ZMP) where located in the support area is zero. Thus, the robot will not fall or tip over during walking. While in the dynamic walking, the inertial forces can be controlled such that cancel the moment of gravitational forces at a point inside the support area, i.e. controlling the position of the ZMP by control of the inertial forces.

In order to find the optimal values of the walking parameters for stable static walking, we have done many walking experiments. Based on these experiments, the most important parameters for the stability of the biped walking are : T_c , T_d , D_s , w_m , w_s , x_{ed} and x_{sd} . The other parameters such as the feet and the torso height affect more on the shape of the walking and their effects on the walking stability are small.

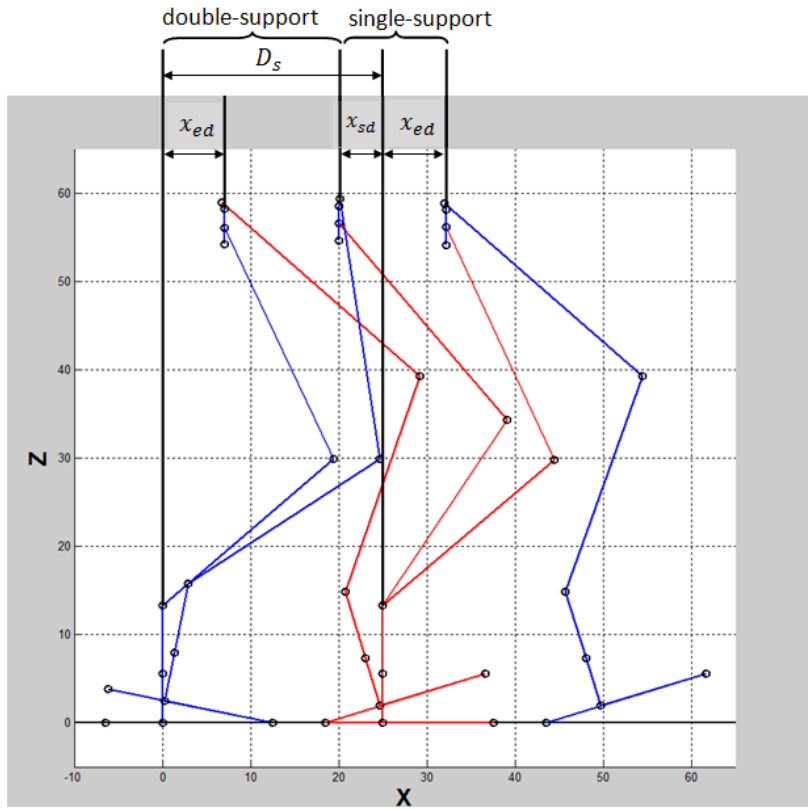


Fig. 6.8 Configurations of the robot in z - x plane at times $t = 0, T_d, T_c$

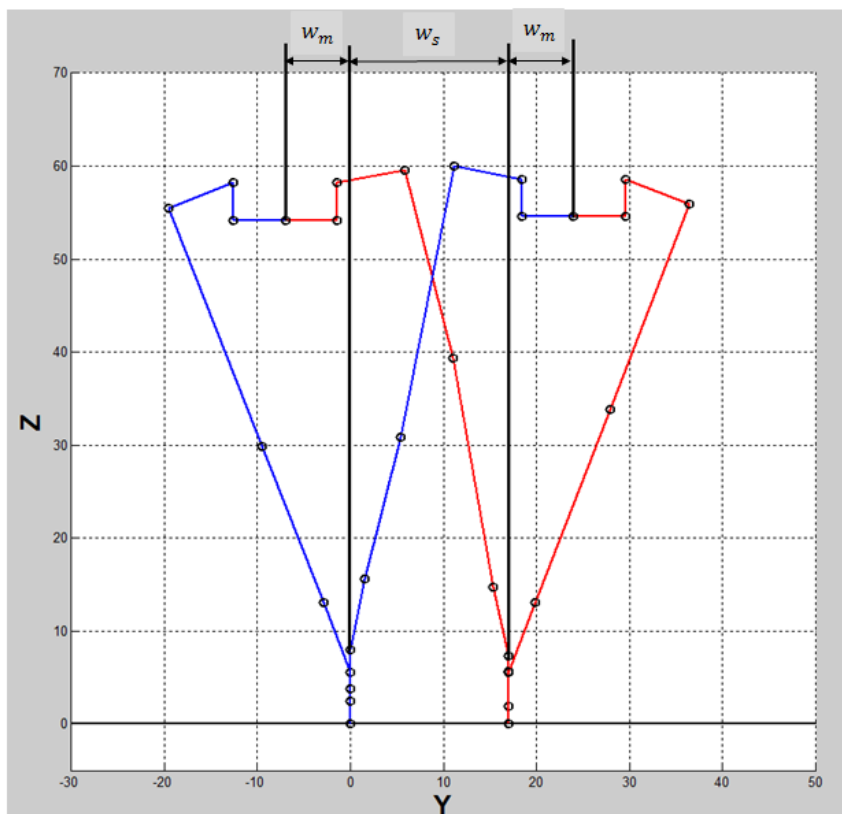


Fig. 6.9 Configurations of the robot in z - y plane at times $t = 0, T_d$

The general strategy to maintain the stability of the robot in the sense of static walking is to move slowly the torso from rear leg to front leg in the double-support phase to locate the center of mass above the support foot area. So the torso position at the start and the end of the double-support phase where the single-support phase ends and starts, respectively, are very important for the stability. In addition the velocity of the torso should be specified in the double-support phase such that the deceleration applied at the end of this phase be small.

As explained in the chapter 4, the forward position of the torso at the start of the first double-support phase is at x_{ed} and at the end of this phase is at $D_s - x_{sd}$. As a result, The forward distance that the torso moves in the double-support phase is $D_s - (x_{sd} + x_{ed})$, see Fig. 6.8.

On the other hand, the side distance that the torso moves in the double-support phase is specified by $w_s + 2w_m$ where w_m is the distance that the torso moves further from the top of the support foot to compensate the mass of the swing leg that pushes the position of the center of mass toward itself, see Fig. 6.9.

The optimal value of w_m is found using experimental tests. As shown in Fig. 6.10, in these tests, the robot joints are actuated such that the robot stands in the left leg, while the right leg is in the air and the torso is moved to locate at the top of the left foot, i.e. $w_m = 0$. As expected in this case, the robot tips over to the right side. Then the value of w_m is increased to reach the stability region. In fact, there is a range of w_m that robot is stable. This range is obtained for Archie as $1 \leq w_m \leq 5 \text{ cm}$. But the optimal value of w_m for stable walking is obtained by various walking experiments as $w_m = 2$.

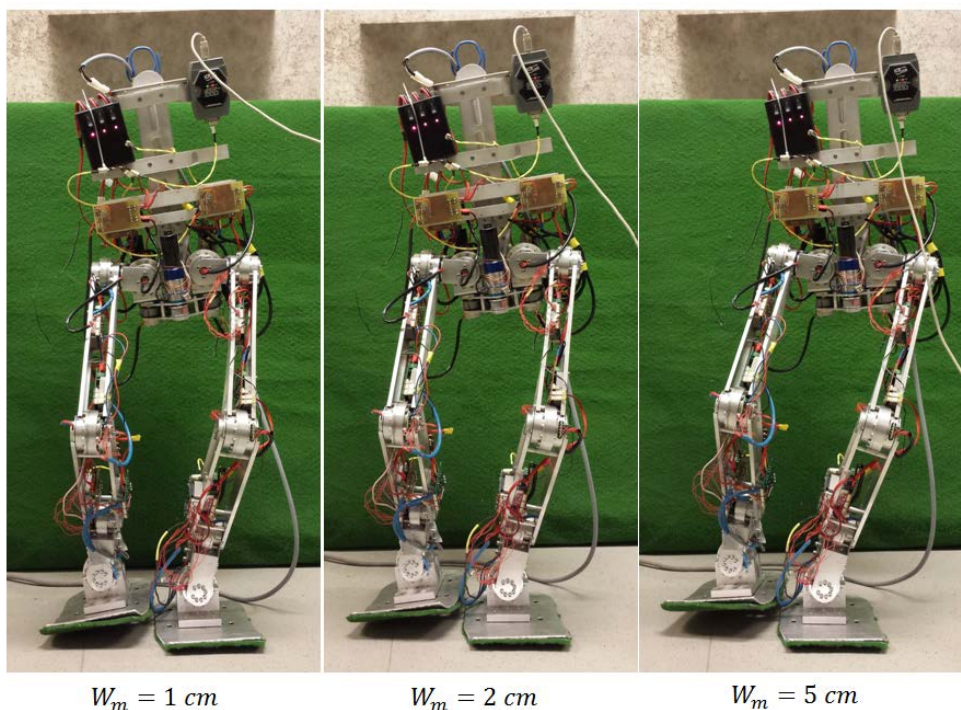


Fig. 6.10 Experimental tests to find the optimal value of w_m

The step width w_s when all joint angles are zero (according to the joint angle definition presented in the chapter 3) is equal to 26 cm . In order to decrease the speed of the torso required for the stability of the robot the step width should be selected as small as possible. In this case, the side distance traversed by torso, i.e. $w_s + 2w_m$, will be minimum. This is also compatible with the natural walking of human. But the minimum value of the step width is limited due to the collision of the right and the left foot. Due to the foot dimension, the optimal value of the step width is selected as $w_s = 17\text{ cm}$.

Finding appropriate values for x_{sd} and x_{ed} for stable walking are more complicated compared to the other parameters. This is due to the high sensitivity of the robot stability to these parameters. The optimal values of x_{sd} and x_{ed} are obtained for Archie by trial and error. So the optimal values that experimentally maintain the stability of Archie during walking were obtained as $x_{sd} = -3.5\text{ cm}$ and $x_{ed} = 6.5\text{ cm}$. It is worth to mention that the torso that its trajectory is planned based on the walking parameters and using the pattern generator presented in the chapter 4, is located at the center of the frame $\{6\}$ where depicted in Fig. 3.1 and 3.2. But the position of the center of mass is varying during walking depending on the different configuration of the robot. Therefore the minus sign of optimal value of x_{sd} can be obvious since the center of mass of the robot is located behind the center of the frame $\{6\}$ due to the robot configuration at the end of the double-support phase. With this value, the center of mass of the robot is located above the support foot area at the start of the single-support phase. During single support phase, the movement of the torso should be minimized since the center of mass should be maintained over the support area during this phase. The traversed distance by the robot during the single-support phase is $x_{sd} + x_{ed}$ which in the optimal case is 3 cm . This short movement ensure that ZMP remains inside the support area in the worst case of the single-support phase.

So by selecting parameters w_m , w_s , x_{ed} and x_{sd} , the forward and side movement constraints for planning the torso trajectory is determined. The height of the torso is specified with choosing the desired parameters of H_{min} and H_{max} . Generally when the height of the robot torso is short, the momentum distance of the inertial forces about support foot is short and thus the effects of the inertial forces to fall over the robot during walking become less. But the more reducing the height of the torso, the more increasing of the knee joints angles. The knee joint angle is limited like other joint due to the physical structure of the joint. On the other hand, by specifying the torso position during walking, the maximum of the step length D_s is limited due to the links constant length. The values selected for walking realization of Archie for the torso height and the step length are

$$H_{min} = 54\text{ cm} , H_{max} = 57\text{ cm} , D_s = 25\text{ cm}$$

After finding the parameters related to the stability of walking, the other parameters will be selected to shape the walking. w_p is selected such that no collision occurs between the swing foot and the stance leg when the swing leg is passing beside the stance leg. This parameter selection is depends highly on the step width w_s . For the optimal step width $w_s = 17\text{ cm}$, it is found from experiments that $w_p = 5\text{ cm}$.

The Step height H_{ao} should be selected such that ensures that no collision will occur with obstacle. In usual walking, it is assumed no obstacle exists in front of the robot. Therefore, the step height is selected as a value that let the foot detach the ground and have no contact with it during the swing phase. In this case, the step height is selected as $H_{ao} = 12 \text{ cm}$. It is noted that H_{ao} is the distance between the origin of frame $\{0\}$ (the center of the frontal ankle joint) and the ground. With $H_{ao} = 12 \text{ cm}$, the distance between the center of the sole of the swing foot and the ground is 6.4 cm when the swing foot is level with the ground. This is due to the fact that the distance between the frame $\{0\}$ and the center of the sole is $l_{an} = 5.6 \text{ cm}$. Furthermore, the position of the maximum step height is located in the middle of the swing phase, i.e. $l_{ao} = D_S$.

Although the leaving and landing angles of the foot q_b and q_f help more smooth walking in the natural walking (dynamic walking), the walking experiments on Archie shows that nonzero values of them leads to instability of the robot in landing and leaving time. Hence, these values for static walking of Archie are selected as zero.

As mentioned in the chapter 4, in natural human walking which is a dynamic walking the time of the double-support phase is about 20% of the time of one stride, i.e. $T_d = 0.2 T_c$. But in the case of the static walking this time is too short which leads to increase the torso velocity to cover its trajectory in the double-support phase. This high speed cause instability of the robot. Based on the walking experimental results of Archie, the optimal value for the double-support time is $T_d = 0.4 T_c$. That means the double-support time is increased as twice to decrease the torso speed and consequently decrease the inertial forces to maintain the stability during walking.

By selecting $T_d = 0.4 T_c$, the other time parameters can be expressed as a function of T_c :

$$T_o = T_{max} = T_9 = \frac{(T_c + T_d)}{2} = 0.7 T_c \quad 6-4$$

$$T_{min} = 0.2 T_d = 0.08 T_c \quad 6-5$$

Selecting the above walking parameters and based on the pattern generator developed in the chapter 4, the both feet and the torso trajectories are obtained. It is noted that these trajectories are planned as the function of T_c . Fig. 6.11 to 6.14 shows the position component trajectories of the right and the left foot as well as the torso for the optimal values of the walking parameters. It is noted that these trajectories are planned as the function of T_c .

In addition, Fig. 6.15 to 6.19 demonstrate the traversed trajectories of the both feet and the torso in three planes generated by pattern generation corresponding to the optimal values resulted from experimental tests.

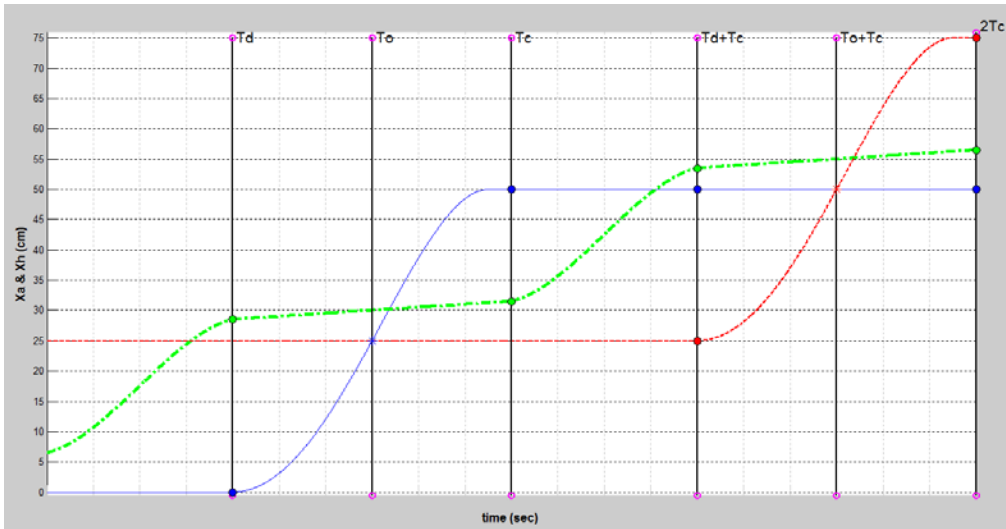


Fig. 6.11 Right foot (blue), left foot (red) and torso (green) trajectories in x direction

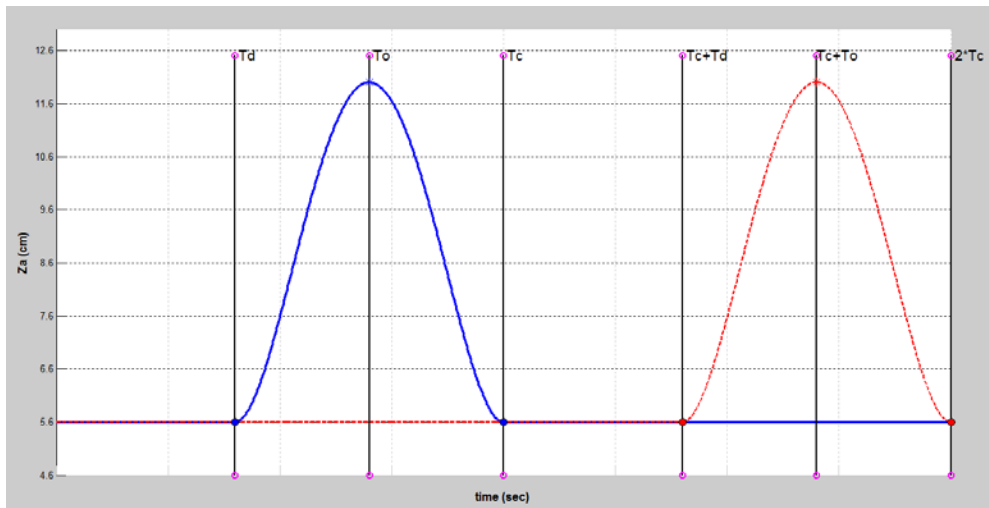


Fig. 6.12 Right (blue) and left (red) foot trajectory for upward movement z_a

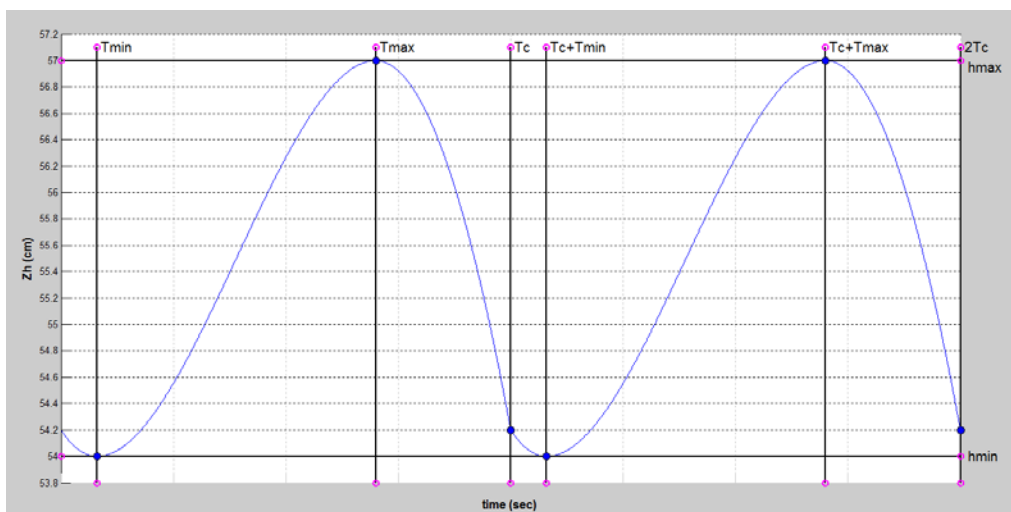


Fig. 6.13 Trajectory of upward movement of torso z_h

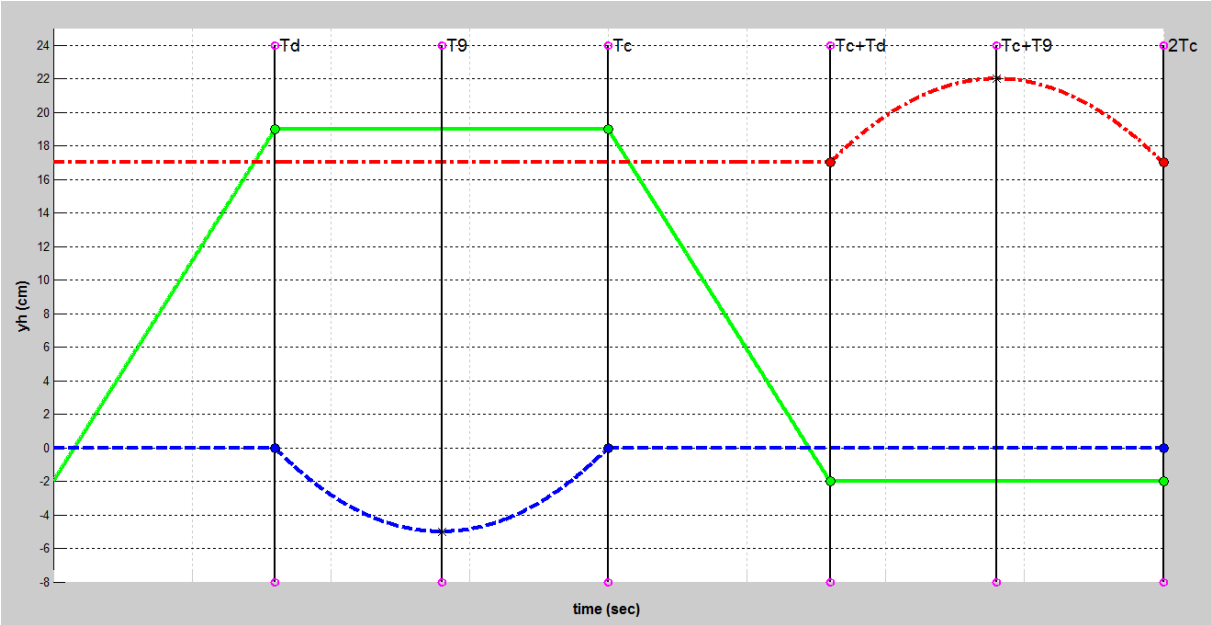


Fig. 6.14 Trajectory of left foot (red), right foot (blue) and torso (green) in y direction

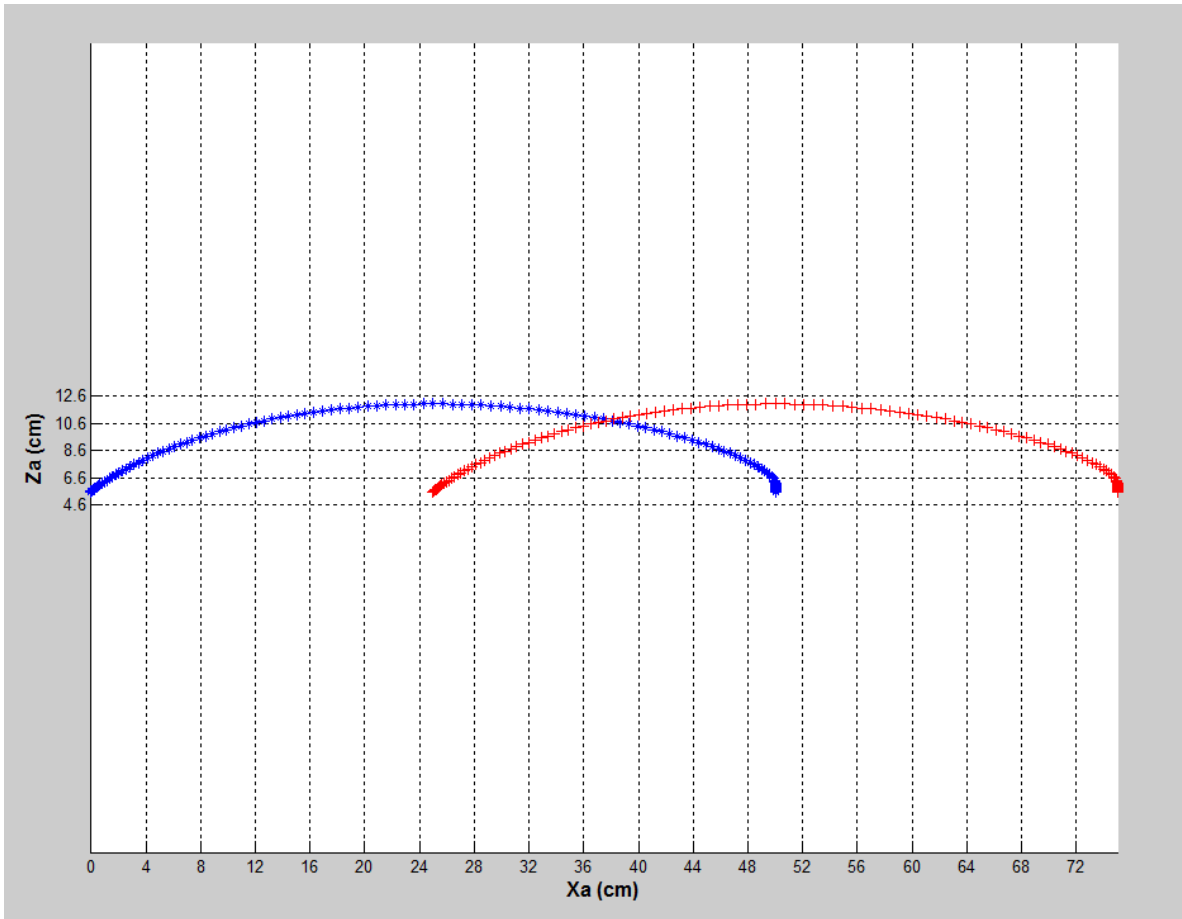


Fig. 6.15 Traversed trajectory of the right foot (blue) and the left foot (red) in the z-x plane

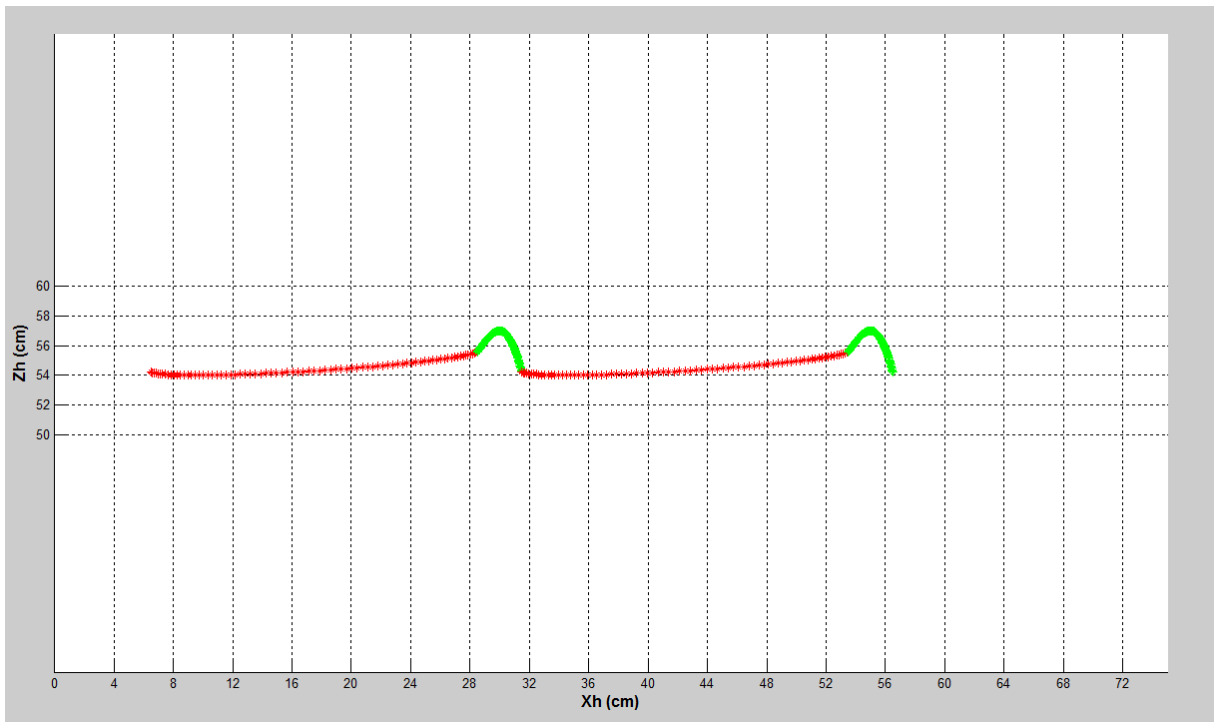


Fig. 6.16 Traversed trajectory of the torso in the z-x plane during the double-support phase (red) and during the single-support phase (green)

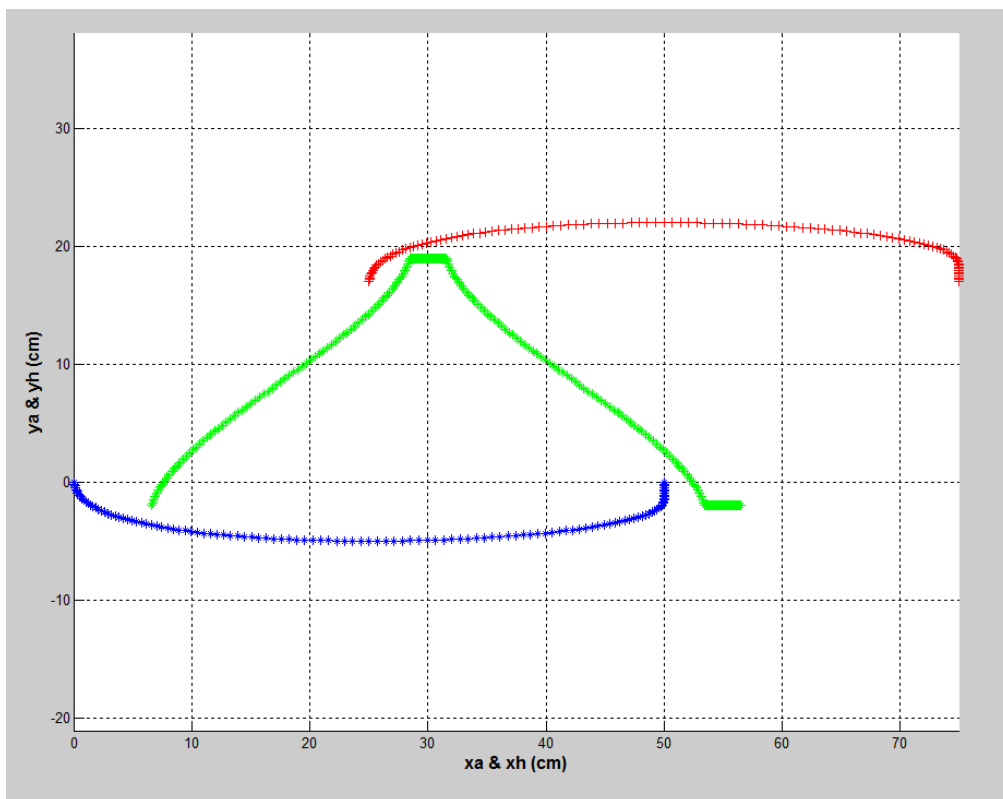


Fig. 6.17 Position trajectory of the right foot (blue) and the left foot (red) and the torso (green) in the y-x plane

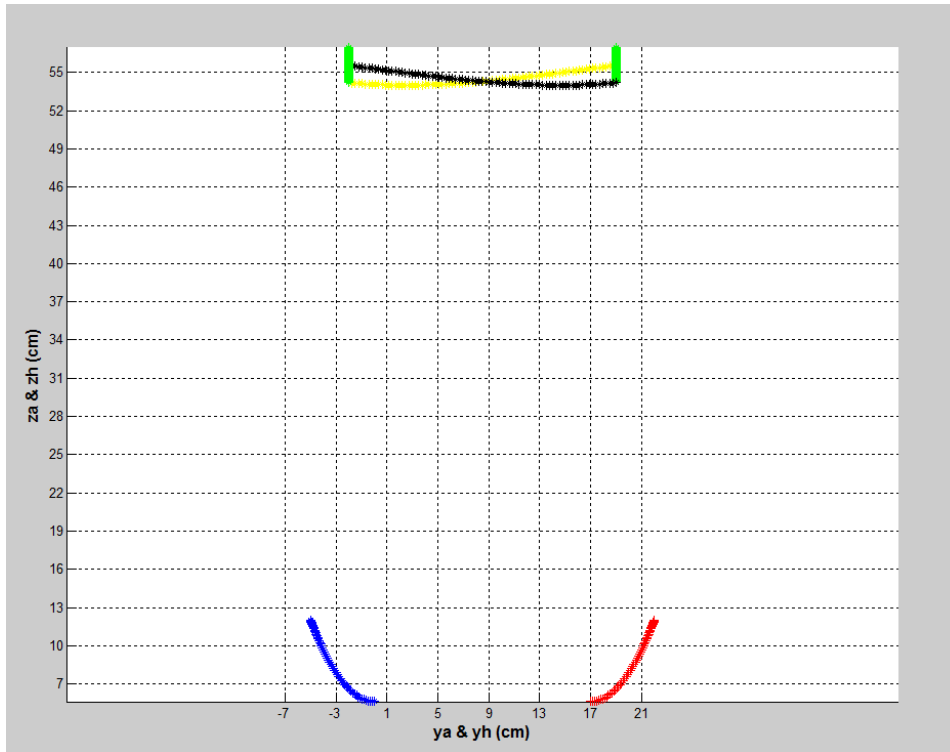


Fig. 6.18 Position trajectory of the right foot (blue) and the left foot (red) and the torso (yellow: first double-support, black: second double-support, green: both two single-support phases) in the z-y plane

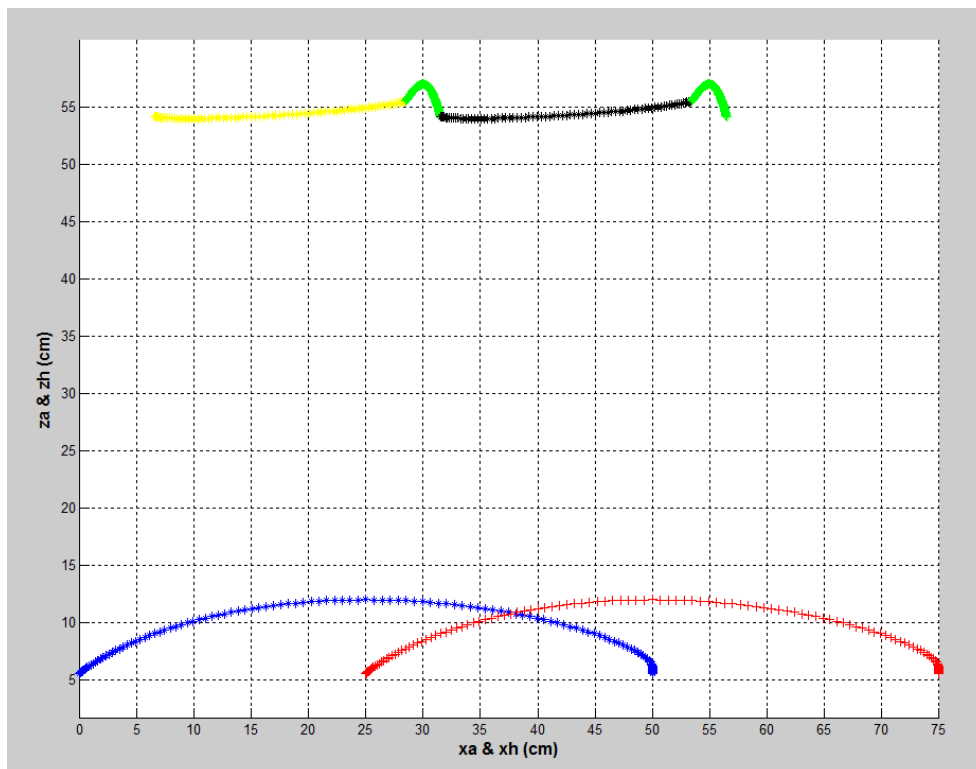


Fig. 6.19 Position trajectory of the right foot (blue) and the left foot (red) and the torso (yellow: first double-support, black: second double-support, green: both two single-support phases) in the z-x plane

After planning the desired trajectories, the desired joint angle trajectories are calculated using inverse kinematics. Since these trajectories should be discrete to command as inputs to the digital controller, the actual value of time for one complete cycle of walking will be specified with choosing two parameters: N and m by following formula

$$T_c = \frac{1}{2} N m T_s \quad 6-6$$

where T_s is the sample time of position controller in Elmo drive. This sample time is a fixed value specified based on the position closed-loop system parameters. With the tuning mentioned before in this chapter, the sample time for position control closed-loop system is $T_s = 320 \mu\text{sec}$. That means the output of the position controller is generated at each $320 \mu\text{sec}$ to produce the desired command for inner velocity loop which its sample time is less than $320 \mu\text{sec}$. It is noted that this value is computed by Elmo drive and it is used in the program using the command explained in the Elmo manual.

The number of points used to digitize the trajectories to construct the PT table is selected as $N = 300$. With these numbers of sampled data, there is no jump (high acceleration) between any two consecutive positions.

Knowing the values of N and T_s , by choosing the value for m one can easily specifies the time of walking cycle T_c and relatively the robot speed $V = (D_s/T_c)$. The lower the values of m , the more speed of the robot. The maximum speed of walking was realized by various experiments is $V = 0.076 \text{ km/h} = 2.1 \text{ cm/sec}$. The value selected for m in this case was $m = 250$. Therefore, the sample time between each pair of consecutive points was $T = m T_s = 0.08 \text{ sec}$. In fact in this case, the position controller needs one desired position at each $T_s = 320 \mu\text{sec}$ but the pattern generator specified one position at each $T = 0.08 \text{ sec}$. Thus 250 intermediate positions are required to produce between each two consecutive points generated by the pattern generation. The intermediate positions are calculated by Elmo drive using third-order polynomial interpolation. The time of walking cycle for this speed can be calculated as $T_c = 12 \text{ sec}$.

Fig. 6.20 to 6.25 shows the discrete values of desired angles for all joints of Archie. These joint angles corresponding to the above mentioned optimal walking parameters. After that, these values are tabulated in PT table by the developed program to send for each joint independent controller as the reference values. Each joint controller produce enough voltage related to the error between the desired and actual joint angles. The relation between the joint angle error and the generated voltage is determined based on the tuned parameters of cascaded controller as explained in the chapter 5.

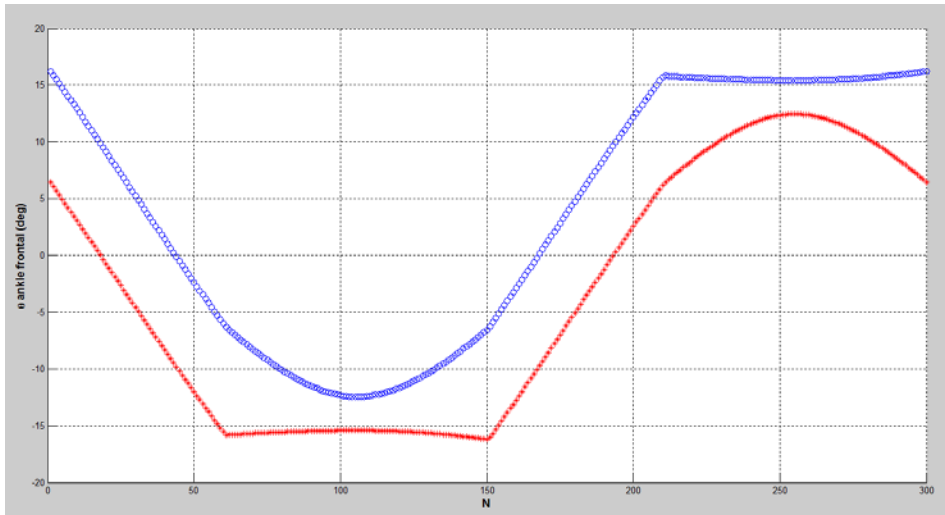


Fig. 6.20 Frontal ankle joint angles for the right leg (blue) and the left leg (red)

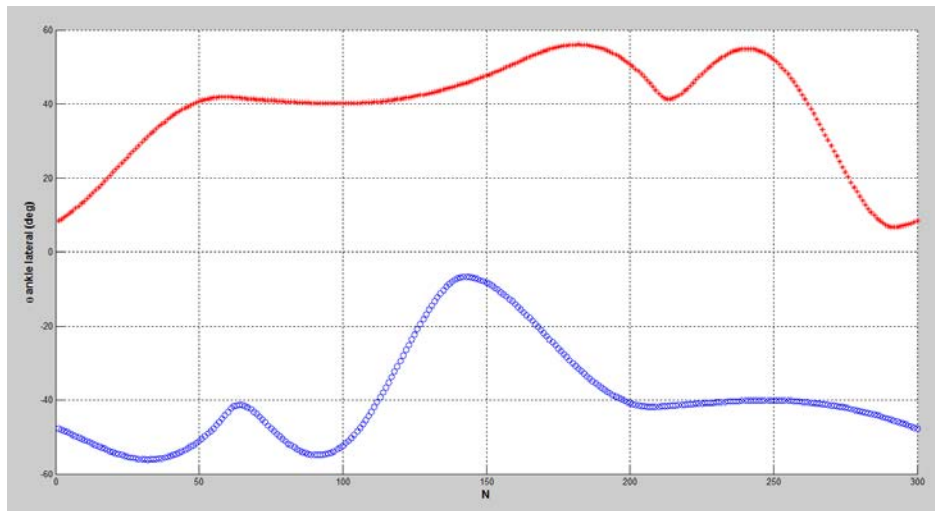


Fig. 6.21 Lateral ankle joint angles for the right leg (blue) and the left leg (red)

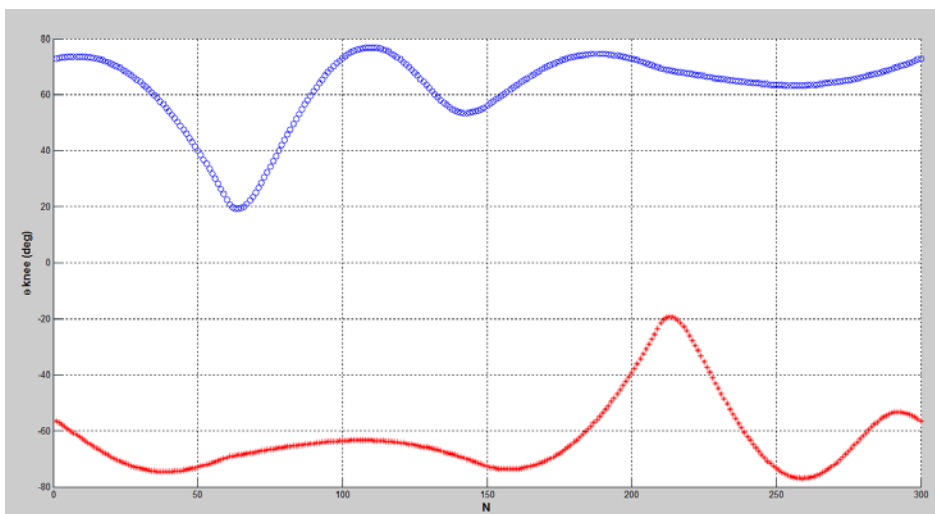


Fig. 6.22 Knee joint angles for the right leg (blue) and the left leg (red)

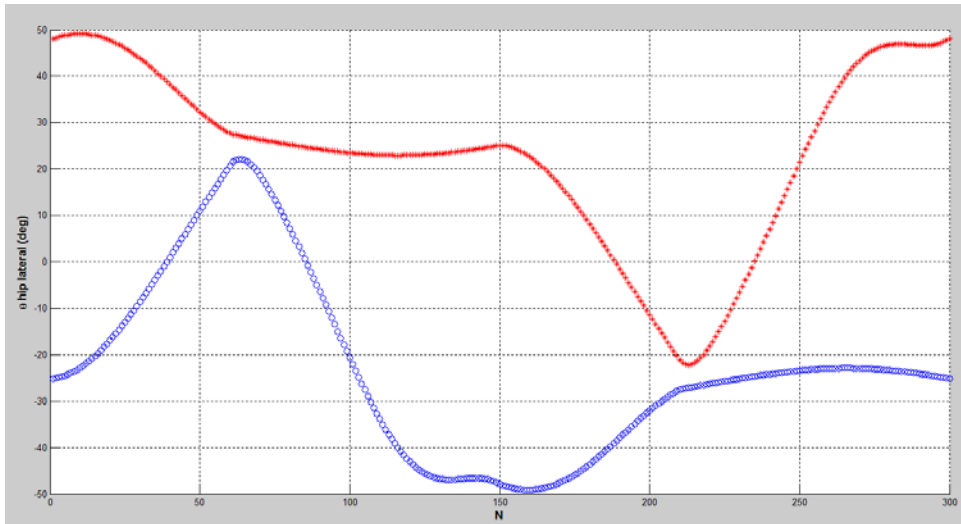


Fig. 6.23 Lateral hip joint angles for the right leg (blue) and the left leg (red)

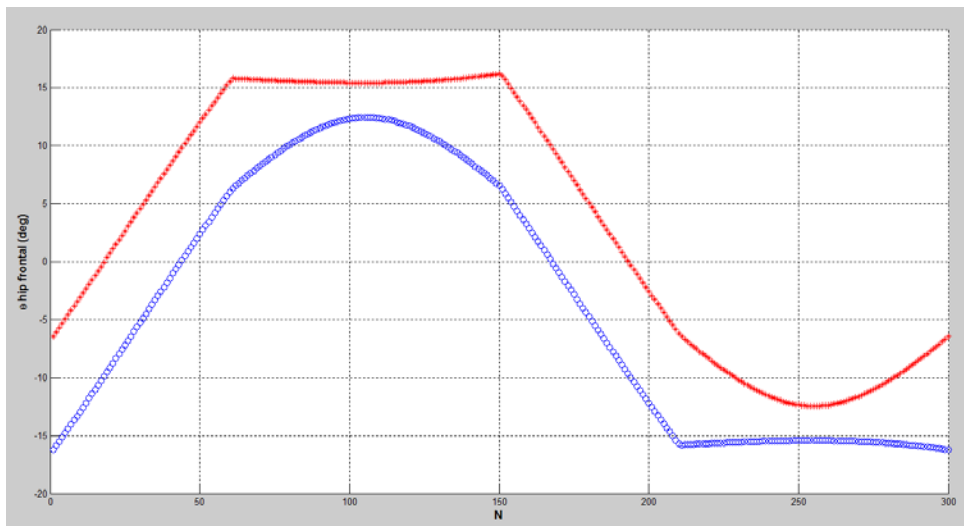


Fig. 6.24 Frontal hip joint angles for the right leg (blue) and the left leg (red)

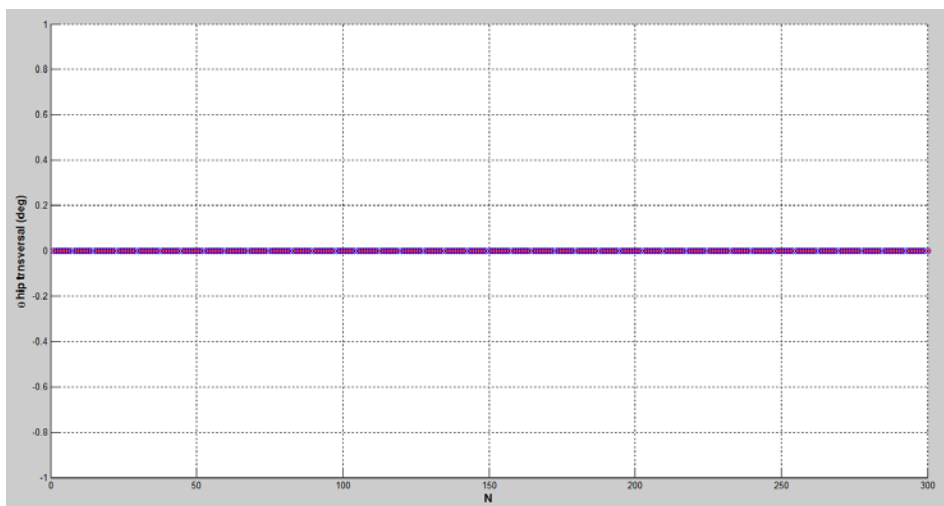


Fig. 6.25 Transversal hip joint angles for the right leg (blue) and the left leg (red)

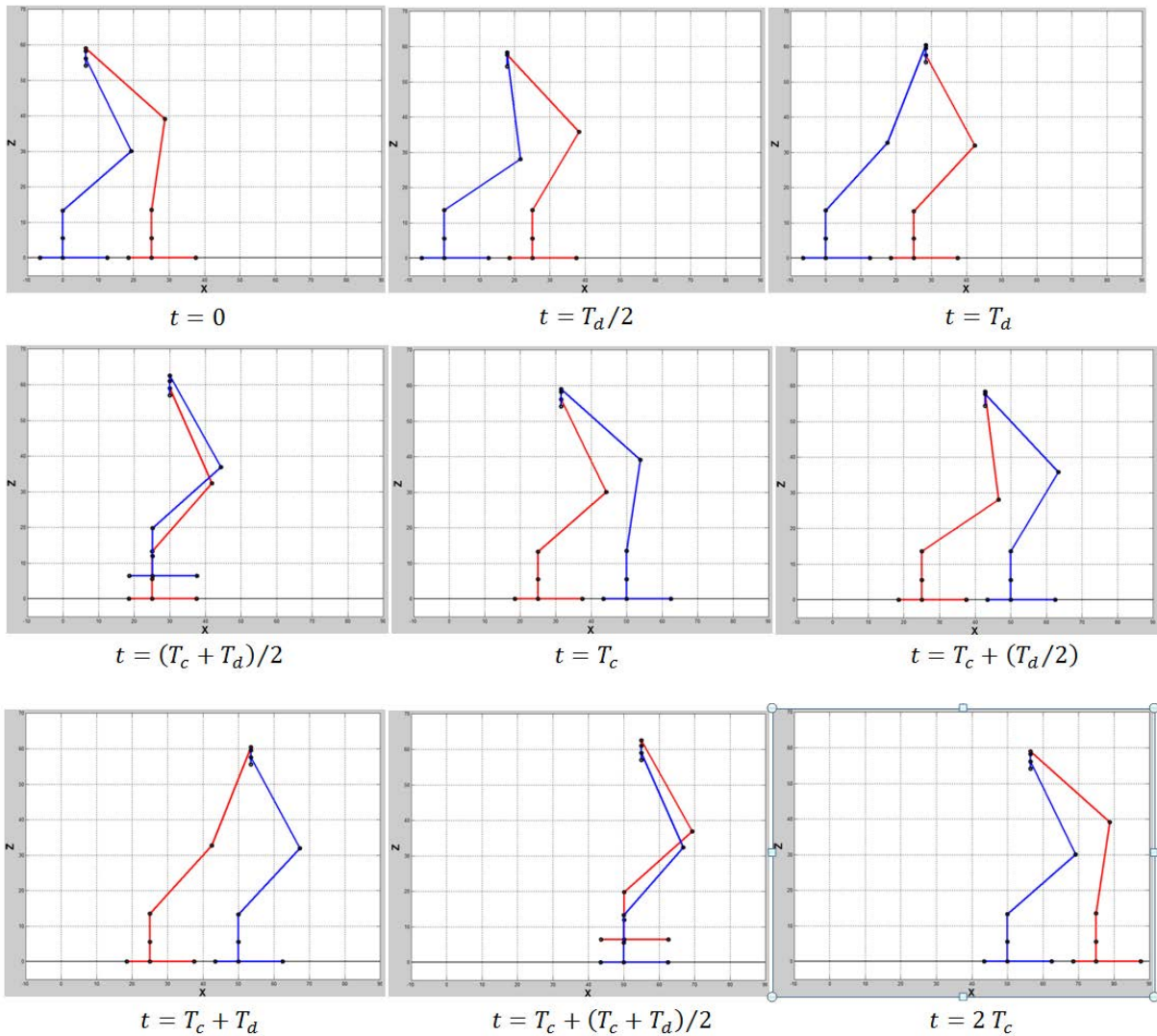


Fig. 6.26 Archie configurations in z-x plane according to the resulted joint angles for optimal walking parameters (all dimensions are *cm*)

Fig. 6.26 and 6.27 show the configuration of the robot kinematically simulated at some important times during walking in z-x plane and y-z plane, respectively. These times are at the start, the end and the middle of the single and the double-support phases. On the other hand, Fig. 6.28 shows the picture of the real robot at the same times. It is clear that the robot weight changes the configuration of the real robot compared to the corresponding kinematic simulation of the robot. But these changes does not lead to instability of the robot during walking. Therefore, with the mentioned optimal walking parameters, the robot stable walking was realized several times.

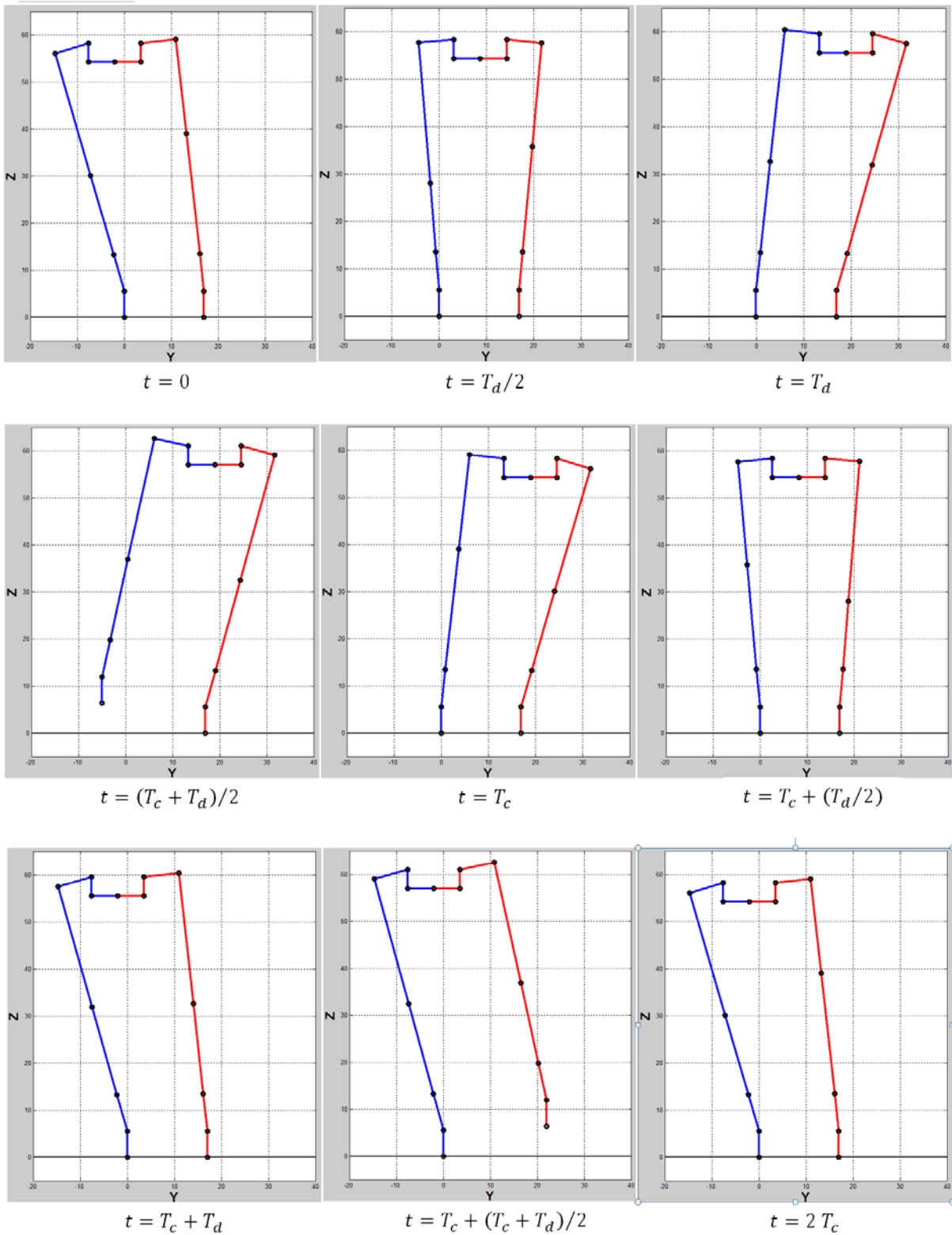
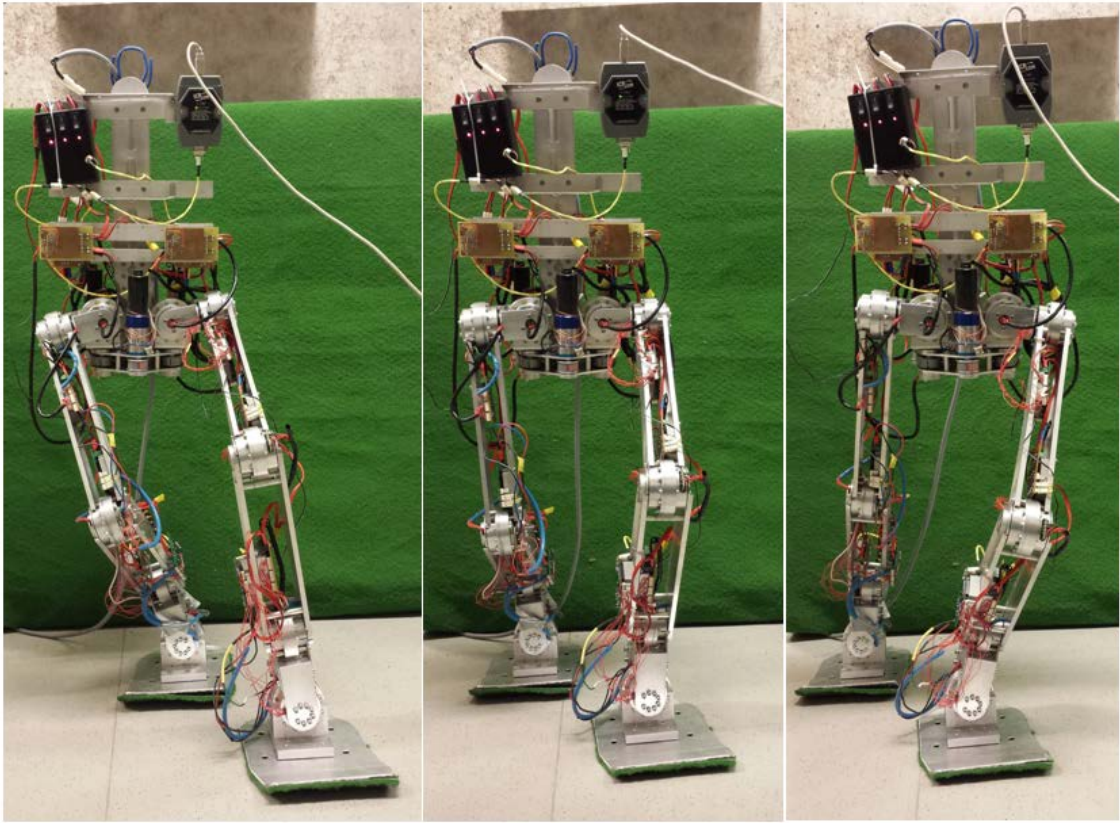


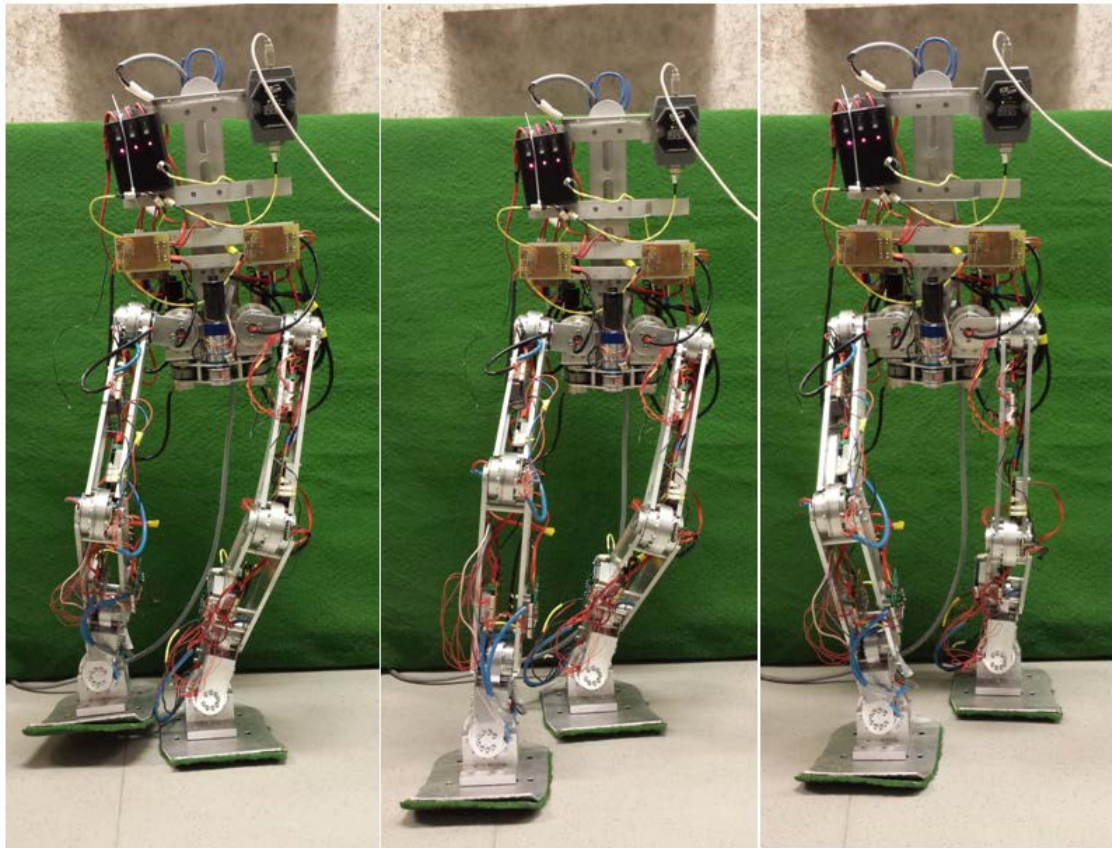
Fig. 6.27 Archie configurations in z-y plane according to the resulted joint angles for optimal walking parameters (all dimensions are *cm*)



$t = 0$

$t = T_d/2$

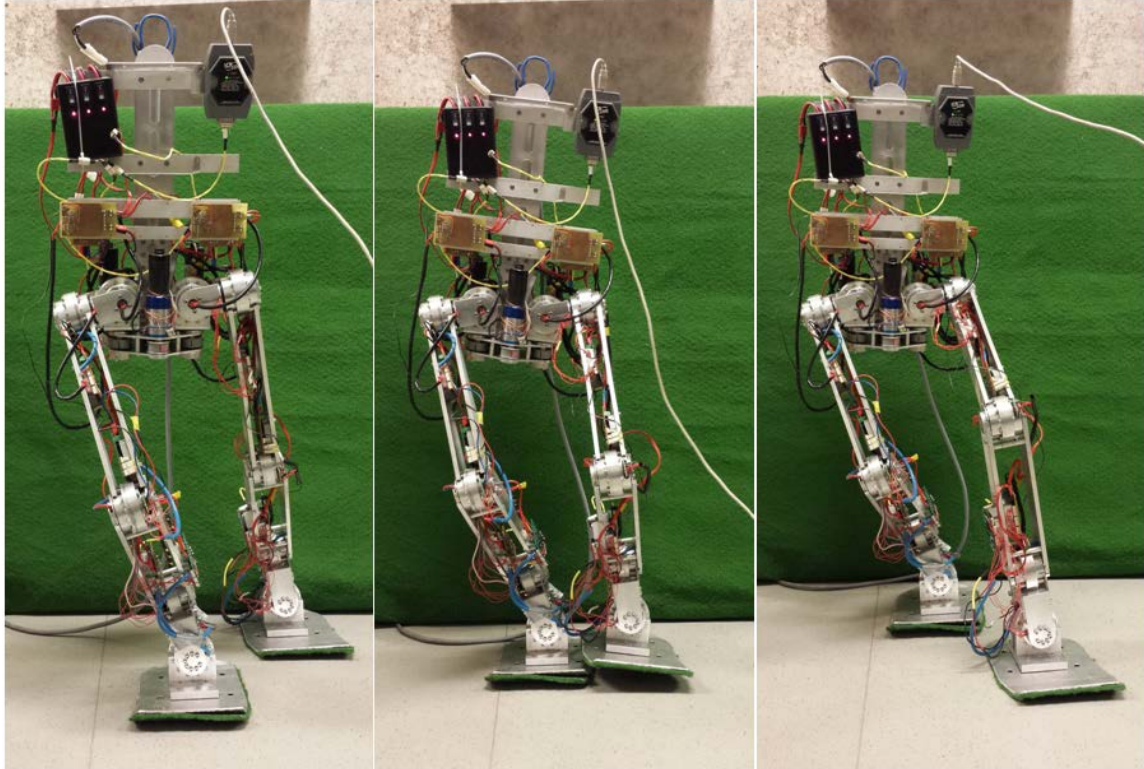
$t = T_d$



$t = (T_c + T_d)/2$

$t = T_c$

$t = T_c + (T_d/2)$



$$t = T_c + T_d$$

$$t = T_c + (T_c + T_d)/2$$

$$t = 2 T_c$$

Fig. 6.28 Complete cycle of walking depicted at nine important specified time

Chapter 7 Conclusions and future works

In this project, walking of the biped robot, Archie, realized. The designed control system was implemented to control stable walking of the biped robot. The proposed controller in this thesis is based on the decentralized control method. In this method, each joint angle is controlled to track the reference angle independent of the other joints of manipulator. In order to reduce the effect of nonlinear disturbances, the velocity closed-loop with PI controller are cascaded with the position closed-loop with P controller. The controller gains were tuned based on the heuristic method presented in this thesis for independent joint control. The step response of the joint's tuned controller shows that the actual joint angle tracks the reference signal fast and without oscillation.

On the other hand, walking pattern includes trajectories of the feet and the torso are generated corresponding to the walking parameters defined by user. These trajectories are calculated using cubic spline interpolation. The trajectory constraints are developed based on the walking analysis of human. In order to find the optimal values of walking parameters that ensure stable walking, several experiments have been done. In fact, the walking parameters were selected such that the center of gravity of the robot is moved from the rear foot to the front foot in the double-support phase. In addition, these parameters ensure that the center of gravity always stays on the above of the support foot area to maintain the ZMP inside the support area and consequently stabilize the walking robot.

Another most important part of the controller is the conversion of the Cartesian space trajectories generated by the pattern generator to the joint space trajectories which are required as reference signals for each independent joint controller. Therefore, first, forward kinematic relations based on the D-H convention were developed and then the closed-form solution of the inverse kinematics was derived. This was done by decoupling the inverse kinematics problem into two problem: inverse position and inverse orientation. The closed-form solution has less computation compared to the iterative methods. Thus it is more suitable in the case of real time implementation.

The reference trajectories of joint angles are digitized to use as reference values for digital controller of each joint. In order to synchronize motion of all joints to create smooth walking, these data are tabulated in PT table. These data are converted to CAN message format and then sent to each joint controller. The independent joint controller generate enough voltage to actuate the motor to reach the desired value at the specified time. The

developed C++ program is responsible to calculate, convert and send the values to the joint controller via CAN communication (Dezfouli & Daniali, 2012).

In conclusion, the proposed control system is capable to stabilize the biped robot during walking up to speed of 2.1 cm/sec . In fact, the control system can be used only in the case of static stability of the biped walking. The following statements are resulted from several walking experiments:

- The walking of Archie is very sensitive to zero reference position of joints (zero position is specified at standing configuration of the robot). Since the robot system uses the incremental encoder, the zero reference position should be redefined always after turning on the robot. In Archie this is done either automatic (using hall sensor) or manually. Although the control system uses feedback in joint space, the control system in operational space (the Cartesian space) is feedforward. Therefore, the difference between zero reference position of the actual robot with the zero reference position defined in kinematic model leads to error in tracking the desired walking shape. In addition, other kinematic model uncertainty such as measurement inaccuracy and joint and link flexibility are not compensated in the proposed feedforward system. Therefore, the configuration of the robot is different from the planned configuration by the controller at specified time during walking. In this case, the robot's center of mass is moved from preplanned position and consequently leads to static instability of the robot at this moment.
- When the ground is not flat (even a little bumpy) , the control system cannot maintain the stability of the robot. This is due to the fact that the controller has no information about the environment to update the predefined walking trajectory planned by the pattern generator. Therefore the fixed preplanned trajectory is unable to retrieve the stability of the robot when the configuration of the robot changes due to the interaction force between the foot and the ground.
- In the kinematic model it is assumed that the support foot is completely attached to the ground but in reality this assumption is not true. Because the support foot can partially detach the ground due to the effect of robot's weight. In this case, the swing foot will contact the ground before the specified time of starting the double-support phase and thus leads to instability of the robot. It is noted that the controller has no information about the early contact of the swing foot to update the preplanned trajectory.

As much as the speed of walking increases, the above mentioned problem will be sever. Therefore the maximum speed that can be achieved with the proposed control system is obtained in experimental test as about 2.1 cm/sec . In order to increase the walking speed while maintaining the stability (either static or dynamic), the walking trajectory should be updated based on the information obtained by sensors from environment. In this manner, the operational space also use feedback in addition to existing joint space closed loop

system. Therefore, for improvement of the proposed control system, information from environment is vital.

Based on the results of this project and other biped walking control project presented in the literature, the following future works can be suggested:

- Force sensor: for dynamic walking, the interaction force between the feet and ground should be controlled. The walking motion is constrained by limitations of contact situation. Exceeding force and torque limitations of the feet leads to slippage or tilt of the foot (Löffler et al., 2003). Therefore it is important to measure and control the ground reaction forces and torques. The force sensor also is useful to specify the walking phase (single-support or double-support phase) to change the walking trajectory corresponding to the situation.
- Orientation sensor: for stable motion especially in high speed walking or in uneven ground it is necessary to measure the orientation of the robot with respect to the gravity vector. Since in dynamic walking the robot motion is associated with rather high acceleration, the use of inclination sensor or accelerometers leads to poor results (Löffler et al., 2003). Therefore, the combination of accelerometers and gyroscopic sensor is employed. Inertial Measurement Unit (IMU) is an industrial orientation sensor that use this combination.
- ZMP estimation: based on information from orientation and force sensor, the real ZMP can be estimated. The difference between the desired ZMP and estimated actual ZMP generate the error. Then a controller should designed such that minimize this error by changing the torso trajectory by reselecting the corresponding walking parameters. It is noted in this case the walking trajectories are updated online during walking to stabilize the biped walking.

References

- Aikawa, H., Shimomura, K., Kondo, H., Morishima, A., & Takanishi, A. (2006). Development of a New Humanoid Robot WABIAN-2. In *IEEE International Conference on Robotics and Automation* (pp. 76–81). Orlando, Florida.
- Baltes, J., Byagowi, A., Anderson, J., & Kopacek, P. (2009). Teen Sized Humanoid Robot: Archie. In *Progress in Robotics* (pp. 34–41). Incheon, Korea.
- Burden, R. L., & Faires, J. Douglas. (2005). *Numerical analysis* (9th ed.). Cengage Learning.
- Byagowi, A. (2010). *Control System for a Humanoid Robot*. Vienna University of Technology.
- Chestnutt, J., Lau, M., & Cheung, G. (2005). Footstep Planning for the Honda ASIMO Humanoid. In *IEEE International Conference on Robotics and Automation* (pp. 629–634). Barcelona, Spain.
- De silva, c w. (2007). *Sensors and actuators: control systems instrumentation*. CRC Press.
- Dezfouli, S. (2013). *Motion Creating System Dseign and Impelmentation for a Humanoid Robot*. Vienna University of Technology.
- Dezfouli, S., & Daniali, M. M. (2012). Motion Controller Design for a Biped Humanoid Robot. In *ASME International Design Engineering Technical Conferences & Computers and Information in Engineering Conference (IDETC/CIE)* (pp. 1–7). Chicago, IL, USA.
- Dezfouli, S., Kopacek, P., & Daniali, M. M. (2011). Cost Oriented Humanoid Robot Archie. *International Journal Automation Austria*, 19(2), 62–70.
- Duindam, V., & Stramigioli, S. (2006). *Modeling and Control for Efficient Bipedal Walking Robots - a Port-Base Approach*. Springer.
- Hirai, K., Hirose, M., & Haikawa, Y. (1998). The Development of Honda Humanoid Robot. In *IEEE International Conference on Robotics and Automation* (pp. 1321–1326). Leuven, Belgium.
- Huang, Q., Yokoi, K., & Kajita, S. (2001). Planning Walking Patterns for a Biped Robot. *IEEE Transactions on Robotics and Automation*, 17(3), 280–289.

- Kajita, S., & Tani, K. (1996). Experimental Study of Biped Dynamic Walking. *IEEE Control Systems Magazine*, 16(1), 9–13.
- Kajita, S., Yamaura, T., & Kobayashi, A. (1992). Dynamic Walking Control of Biped Robot along a Potential Energy Conserving Orbit. *IEEE Transactions on Robotics and Automation*, 8(4), 431–437.
- Kaneko, K., Harada, K., & Kanehiro, F. (2008). Humanoid Robot HRP-3. In *IEEE/RSJ international conferece on intelligent robots and systems* (pp. 2471–2478). Nice, France.
- Kaneko, K., Kanehiro, F., Morisawa, M., Akachi, K., Miyomori, G., Hayashi, A., & Kanehira, N. (2011). Humanoid Robot HRP-4 -Humanoid Robotics Platform with Lightweight and Slim Body-. In *IEEE/RSJ international conferece on intelligent robots and systems* (pp. 4400–4407). San Fransisco, CA, USA.
- Kaneko, K., Kanehiro, F., Morisawa, M., Miura, K., Nakaoka, S., & Kajita, S. (2009). Cybernetic Human HRP-4C. In *IEEE-RAS International Conference on Humanoid Robots* (pp. 7–14). Paris, France.
- Kato, I., & Tsuiki, H. (1972). The hydraulically powered biped walking with a high carrying capacity. In *The forth international syposium on external control of human extremities* (pp. 410–421). Dubrovnik, Croatia.
- Löffler, K., Gienger, M., & Pfeiffer, F. (2003). Sensors and Control Concept of Walking “Johnnie.” *The International Journal of Robotics Research*, 22(3-4), 229–239.
- Lohmeier, S., & Buschmann, T. (2009a). Humanoid robot LOLA. In *IEEE International Conference on Robotics and Automation* (pp. 775–780). Kobe, Japan.
- Lohmeier, S., & Buschmann, T. (2009b). Humanoid Robot LOLA—Research Platform for High-SpeedWalking. In *Motion and Vibration control* (pp. 221–230). Springer.
- Pat Muuray, M., Bernard Drought, A., & Kory, R. (1964). Walking Pattern of a Normal Men. *The Journal of Bone & Joint Surgery*, 46(2), 35–360.
- Perry, J. (1992). *Gait Analysis Normal and Pathological Function*. SLACK Incorporated.
- Pratt, J., Chee, M. C., Torres, A., & Pratt, G. A. (2001). Virtual Model Control: An Intuitive Approach for Bipedal Locomotion. *The International Journal of Robotics Research*, 20(2), 129–143.
- Pratt, J. E. (2000). *Exploiting Inherent Robustness and Natural Dynamics in the Control of Bipedal Walking Robots*. Massachusetts Institute of Technology.
- Sakagami, Y., Watanabe, R., Aoyama, C., Matsunaga, S., Higaki, N., & Fujimura, K. (2002). The Intelligent ASIMO: System Overview and Integration. In *IEEE/RSJ International Conferece on Intelligent Robots and Systems*. Lausanne, Switzerland.

- Siciliano, B., Sciavicco, L., Villani, L., & Oroilo, G. (2009). *Robotics: Modelling, Planning and Control*. Springer.
- Spong, M., Hutchinson, S., & Vidyasagar, M. (2006). *Robot Modeling and Control*. Wiley.
- Takanishi, A., Ishida, M., Yamazaki, Y., & Kato, I. (1985). The Realization of Dynamic Walking by the Biped Walking Robot WL-10RD. In *The International Conference on Advanced Robotics* (pp. 459–566).
- Vaughan, C., Davis, B., & O'connor, J. (1992). *Dynamics of Human Gait* (2nd ed.). Kiboho.
- Vukobratovic, M., & Borovac, B. (2004). Zero-Moment Point — Thirty Five Years of Its Life. *International Journal of Humanoid Robotics*, 1(1), 157–173.
- Westervelt, E., Grizzle, J., Chevallereau, C., Choi, jun ho, & Morris, B. (2007). *Feedback Control of Dynamic Bipedal Robot Locomotion*. CRC Press.
- www.asimo.honda.com, (last visit October 2013).
- www.elmomc.com (last visit October 2013).
- www.faulhaber.com (last visit October 2013).
- www.maxonmotor.de (last visit October 2013).
- Yokoi, K., Kanehiro, F., Kaneko, K., Fujiwara, K., Kajita, S., & Hirukawa, H. (2003). Experimental Study of Biped Locomotion of Humanoid Robot HRP-1S. In *Experimental robotics VIII* (pp. 75–84). Springer.

Curriculum Vitae

Mohsen Mohamadi Daniali

Email: e1128821@student.tuwien.ac.at , daniali61@yahoo.com

Nationality: Iranian

Date of Birth: 7th August 1982

Education

- **PHD in Mechanical Engineering, Mechatronics (2011-2013)**
 - Institute of Mechanics and Mechatronics, Vienna University of Technology, Vienna, Austria.
- **M. Sc. in Mechanical Engineering, Mechatronics (2006 – 2009)**
 - Sharif University of Technology, International Campus, Kish Island, Iran.
- **B. Sc. in Mechanical Engineering, Heat and Fluid (2000 – 2005)**
 - Babol Noshirvani University of Technology, Babol, Iran.

Publications

- Daniali, M., Vossoughi, G., **Intelligent active vibration control of constrained manipulator in robotic deburring**, International conference on Industrial Mechatronics and Automation, Chengdu, China, 2009, pp. 76-80.
- Dezfouli, S., Kopacek, P., Daniali M., M., **Cost oriented humanoid robot Archie**, International Journal Automation Austria, 2011, 19(2), pp. 62-70.
- Dezfouli, S., Daniali, M., **Motion controller design for a biped humanoid robot**, Proceedings of the ASME 2012 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference (IDETC/CIE), Aug 12-15, 2012, Chicago, IL, USA.

Research Interests

- Robotics
- Humanoid robot
- Biped walking
- Intelligent control
- Neural network
- Fuzzy system
- Active vibration control
- Design and optimization of automated robotic manufacturing

Graduated Courses

Robotics kinematics and dynamics
Control system design (hydraulic, pneumatic & PLC)
Fuzzy control
Advanced engineering mathematics
Advanced automatic control
Advanced dynamics
Nonlinear control

Mechatronics
Technical writing
Computer structure and language
Electronics
Digital logic circuit
Adaptive and predictive control
Autonomous and mobile robot

Research Experiences

- Research in parallel manipulator design and control methods.
- Implementing neuro-fuzzy controller to double inverted pendulum.
- Simulation of cutting in milling machine.
- Research in various types of intelligent controllers.
- Controller design and implementation for walking control of humanoid robot

Engineering Software Proficiency

- MATLAB and Simulink
- Solid works
- Pro engineer
- C++

Language skill

- Persian (native language)
- English (fluent)

Reference

Em. o.Univ.Prof. Dr.techn.Dr.hc.mult. Peter Kopacek
Vienna University of Technology, IHRT
Favoritenstrasse 9-11/E325A6
A - 1040 Wien
Tel: ++43 1 58801 31800
FAX: ++43 1 58801 31899
e-mail: peter.kopacek@tuwien.ac.at