DEVELOPMENT OF REINFORCEMENT LEARNING ALGORITHM FOR MOTION CONTROL OF AN AUTONOMOUS HUMANOID

A thesis submitted to the
University of Petroleum and Energy Studies

For the Award of
Doctor of Philosophy
in
Mechanical Engineering

By
Deepak Bharadwaj
(SAP ID 500020494)

August 2019

Supervisor

Dr. Manish Prateek

Department of Mechanical Engineering
School of Engineering
University of Petroleum & Energy Studies
Dehradun-248007: Uttarakhand

DEVELOPMENT OF REINFORCEMENT LEARNING ALGORITHM FOR MOTION CONTROL OF AN AUTONOMOUS HUMANOID

A thesis submitted to the
University of Petroleum and Energy Studies

For the Award of
Doctor of Philosophy
in
Mechanical Engineering

By
Deepak Bharadwaj
(SAP ID 500020494)

August 2019

Supervisor

Dr. Manish Prateek
Professor & Dean
School of Computer Science
University of Petroleum & Energy Studies

Department of Mechanical Engineering
School of Engineering
University of Petroleum & Energy Studies
Dehradun-248007: Uttarakhand

August 2019

DECLARATION

I declare that the thesis entitled “Development Of Reinforcement Learning Algorithm For Motion Control Of an Autonomous Humanoid”, has been prepared by me under the guidance of Dr. Manish Prateek, Professor & Dean of computer Science, University of Petroleum & Energy Studies. No part of this thesis has formed the basis for the award of any degree or fellowship previously.

Deepak Bharadwaj

CERTIFICATE

I certify that Deepak Bharadwaj has prepared his thesis entitled “Development Of Reinforcement Learning Algorithm For Motion Control Of an Autonomous Humanoid”, for the award of PhD degree of the University of Petroleum & Energy Studies, under my guidance. He has carried out the work at the Department of Mechanical Engineering, University of Petroleum & Energy Studies.

Internal Supervisor

Dr. Manish Prateek
Professor & Dean
School of Computer Science
University of Petroleum & Energy Studies
Dehradun

Date:

ABSTRACT

The research work provides a framework for reinforcement learning algorithm of the humanoid robot. The pre-programmed humnaoid robots performs task in known environment.But, with the existence of obscure environment, due to lack of awareness of the environment,humanoid robot fails to perform random task. In such cases the preprogrammed robot need to be reprogrammed to enable it to perform in the changed environment. The reinforcement algorithm helps to do the task in such type of unstructured and unknown environment. With the existing kinematics configuration of humanoid robot, fast movement and sudden turn is not possible, and stability of humanoid robot becomes an issue. This research work was carried out for the modified kinematic configuration of the lower body of the humanoid robot. The thesis also identifies the state of art technologies and the challenges faced.

Each leg has five degree of freedom hip, knee and ankle leg motion of the humanoid robot is proposed to assist the motion of leg for the unstructured and unknown environment. The proposed model is developed in SolidWorks and MATLAB multibody toolbox. Mathematical expression was derived for the joint space trajectories and joint torques. Singularities posture of the robot was analyzed for smooth motion.

The computed torque control system has been implemented for the joint control. The control parameters were selected based on the stiffness of the material properties.

Partial Markov Decision Process was adopted for implementation of the reinforcement algorithm for a vertical position of robot torso in a continuous state space. Basic reinforcement learning problems are usually formalized as Markov Decision Processes (MDP). The goal of RL agent is to minimize its immediate and expected costs. When the system interacts with the MDP, an RL agent passes through a sequence of states that are coupled to one another by transition probabilities and action agents take, and the agents experience a sequence of immediate costs incurred. Reinforcement Learning and teaching approach like Queue Learning (Q-Learning) is implemented for humanoid robot for navigation and exploration. The Q-learning expresses the expected costs to go of a state action pair defined, which is meant to express the expected costs arising after having taken action in state following policy. The transition probabilities of the controller depend on the randomness of the controller. The random values of the controller decide the action. The Q-learning implemented in the MATLAB and interfacing is done with the robot.

Several simulations were carried out for finding the randomness of the system. The simulation results show the switching of humanoid robot from vertical upright position to down position.

ACKNOWLEDGEMENTS

I, first and foremost, would like to express my deepest gratitude to Dr. Manish Prateek, my supervisor for encouraging me to take this research problem on Development of Reinforcement Learning Algorithm for Motion Control of an Autonomous Humanoid. My supervisor has been a source of inspiration and monument of motivation to me throughout this work. Without them, this work would never have been completed.

I am thankful to Chancellor Dr.S.J. Chopra, Vice-Chancellor Dr. Deependra Kumar Jha, CEO Mr Sharad Mehra ,Dean SOE Dr. Kamal Bansal, Associate Dean R&D Dr. Jitendar Pandey. I would like to mention a special thanks to team R&D UPES for providing constant guidance and support.

I am extremely grateful to my doctoral members Dr. Deepak Kumar,Dr Girish Chandra ,Dr.Adesh Kumar, Dr. Mukul Gupta, Dr. Sushobhan Choudhury for their constant encouragement and fruitful discussion during the work.

I should mention my gratitude to HoD, Mechanical Engineering Department Dr. Ajay Kumar who has always motivated me throughout the journey.

I would like to thank Mr. Amneesh Singla,Mr. Natraj Mishra, Dr Rajnish Garg, Mr. V Senthil, Dr. Roushan Kumar,Mr.Caneon Kurien, Mr. S R V Siva Prasanna, Ms Varnita, Ms. Meera C.S and Mr .Akash Gupta to help me while implementing the software and theoretical framework of this research.

Also, I am thankful to each and every faculty members of Mechanical Engineering, Electronics Engineering and Computer Science Engineering for their support and cooperation.

I would like to thank Ms. Rakhi Ruhal, Mr. Sony Sandeep Farmer and other staff members for their help and numerous occasions.

Finally, I must express my very profound gratitude to my father Mr. Bachneshwar Pathak, mother Mrs. Ranju Devi, father-in-law Mr. Ashok Kumar Pathak, mother-in-law Mrs. Rambha Devi, brother Mr. Alok Bharadwaj, sister Mrs. Priyanka Jha for their support and numerous blessings.

Most importantly, I thank my wife Mrs. Guriya Bharadwaj and son Divyansh Bharadwaj, Daksh Bharadwaj for always being there with me and being a persistent source of encouragement.

Thank You

Deepak Bharadwaj

UPES Dehradun

August 2019

TABLE OF CONTENTS

LIST OF SYMBOLS

θ_1: $theta1$

θ_2: $theta2$

θ_3: $theta3$

θ_4: $theta4$

θ_5: $theta5$

l_1: $link\ length\ 1$

l_2: $link\ length\ 2$

l_3: $link\ length\ 3$

l_4: $link\ length\ 4$

l_5: $link\ length\ 5$

${}^{0}_{1}T$: *transformation of link 1 with respect to the link 0*

${}^{1}_{2}T$: *transformation of link 2 with respect to the link 1*

${}^{2}_{3}T$: *transformation of link 3 with respect to the link 2*

${}^{3}_{4}T$: *transformation of link 4 with respect to the link 3*

${}^{4}_{5}T$: *transformation of link 5 with respect to the link 4*

$C\Theta_1$:*cos(theta1)*

$S\Theta_1$:*sin(theta1)*

$C\Theta_{12}$:*cos(theta1+theta2)*

$S\Theta_{12}$:*Sin(theta1+theta2)*

$C\Theta_{34}$:*cos(theta3+theta4)*

$S\Theta_{34}$:*Sin(theta3+theta4)*

p_x *:position of foot in x-direction*

p_y *:position of foot in y-direction*

p_z *:position of foot in z-direction*

ξ*:damping factor*

k_p: *proportional gain*

k_d: *derivative gain*

J: jacobian

ω_n:*natural frequency of link*

ω_{res}:*resonance frequency of link*

LIST OF ABBREVIATIONS

ASIMO	Advanced step in innovative mobility
iCub	Cub standing for cognitive universal body
PETMAN	Protection Ensemble Test Mannequim
PD	Proportional derivative
PPD	Partitioned proportional derivative
CTC	Computed torque control
MDP	Markov Decision Processes
RL	Reinforcement learning
B^3IA	Behavior based behavior intervention architecture

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1 INTRODUCTION

Humanoid robot, robot with an anthropomorphic body defined as a programmable walking machine which can do the action similar to humans. Humanoid robot can acquire information from its surroundings and ability to carry the predefine task. From last four decades, research community worldwide is working on the issues like locomotion, manipulation, perception, interaction, adaption and self-learning tasks associated with humanoid robots. These efforts from the research community are motivated to create a new kind of humanoid robot that works in the environment that deigned to suit the society needs. The highly specialized robot is implemented in the industry for mass production. Due to human like shape, the humanoid robot is the best suited for acting in our everyday environment like stairs climbing, door closing and opening handles, tools and human -centered designs. Humanoid robot lacks the ability to perform the task in unstructured and unknown environment. The humanoid robot must learn from its mistakes and will adapt accordingly without any help from the other guidance. Force sensing and compliance at each humanoid robot joint can allow the robot to safety act in the unknown environment. Implementation of humanoid robot to the community is always an economic and social issues. The principle motive behind the designing of the humanoid robot is to ease human efforts and do the jobs on behalf of them. The development of advanced and humanoid robotics has certain impact on industrial growth and social impact. The futuristic humanoid robots will be able to do industrial and no-industrial work. Humans are eager to project emotion into machines and replicate themselves into mechanical form.

1.1 The History of Humanoid robot

In the year 250B.C Liezi describe the word automation. In his philosophy, automation is a self-operating machine, designed to follow a predetermined task automatically. In the year 50A.D, Greek mathematician described a machine that serves the wine automatically for party guest.

In the year 1206, Al-Jazri developed a band made up of humanoid automation, which performed the body and facial expression on the beat of music. In 1495, Leonardo da Vinci, designed and construct the Mechanical knight, was a humanoid automation. The Mechanical knight was operated by pulleys and cable. The Mechanical knight could sit, stand, and raise its visor. It could independently move its arms and had and automatically correct jaw. In the year 1738, a French inventor, Jacques de Vaucanson developed a figure of a shepherd that could play songs on the flute. He also developed the Tambourine Player that played a flute and a drum. In the year 1774, Pierre Jacquet-Droz developed animated dolls, to help his firm to sell the watch. Afterward Pierre Jacquet-Droz and his son Henri -Louis created a figure of boy like the Draughtsman, that could type the messages. In 1927, the humanoid robot Herbert Televox developed by Ron Wensley. The humanoid robot could lift the receiver to answer a call and correspondingly controlling the task with the help of switch. This robot did not have any ability to speak. The sonic signal controlling the humanoid robot. It was similar to humanoid in appearance. In 1941, Sir Isaac Asimov purposed the laws of robotics for safety restriction in a robot. The story was recompiled in 'I' Robot movie in the year 2004.In the year 2000, Honda developed ASIMO humanoid robot. It had 11 robotic predecessors to control the bipedal walking robot. Since 2000 it has been continually developed. ASIMO robot is capable of jumping, running and can climb on the stairs. The latest robot Hanson Robotics release Sophia in the year 2016. Sophia with silicone skin can interact with the people and display facial expressions[1].

1.2 Characteristics of Humanoid robot.

1.2.1 Manipulation tasks

The manipulating ability of humanoid robotic mechanisms in positioning and orienting end-effectors. The humanoid robots to enter human-centered environments. Due to large number of degree of freedom, a Human can manipulate the objects of many shapes, sizes, weights, and materials [2]. Due to more number of joints and links, singularity posture of humanoid robot reduces the dexterity of the humanoid robot[3]. Due to lack of learning ability and limited manipulation of the humanoid robot, it is unable to pick the new object in unknown environment.

1.2.2 Vision system

Visual perception is essential to humanoid robotics systems working in human environments .The vision system of humanoid robot extract the information of joint position and manages its own body[4]. Vision system manipulates the tool's movement through its own body and handles the target object. A humanoid robot is required to manage the tools and object. Vision system extract the information of shape and size of the target from the external environment. An overhead camera employed to humanoid robot for computing the desired goal location and screens to interpret the images[5]. The image processing technology provide the direct and indirect estimate of the target in the environment to the humanoid robot. In principle, machine vision includes acquiring an image, process it using digital image processing and analysis techniques and take decision based on extracted information[6]. The functions are involved in the real -time 3D vision are the generation of 3D depth map, 3D depth flow generation and Plane segmentation finder. The Realtime depth map generation system and target finder run on humanoid robot body computer. The other complex vision as face recognizer and plane finder, run on the network computers.

1.2.3 Sensing behavior

The sensing behavior of humanoid robot is the mapping of sensory input to the different joint motor actions. This behavior is the appropriateness of the humanoid robotic response to a given task and environment. The reactive behavior provides the humanoid robot to interact to the dynamic and unknown conditions without planning. These behaviors of the humanoid robot deal with targets independently and coordinating the different joint in a desired way. Based on the environment interaction several types of sensors are used in the humanoid robot.

Force and tactile sensors are elementary sensor for humanoid robot when interacting with environment[7]. Humanoid robots entirely depend on the fusion of multiple sensors to provide them with information about their surroundings. The sensory input to the humanoid robot helps to understand the environment and navigation.

1.2.4 Mobile platform

Humanoid robots are moving towards applications beyond the structured environment[8]. The current generation of humanoid robot have mobile platform. Due to mobile platform, humanoid robot is entering the everyday world that people inhabit. Wheel based humanoid robot is working in the static environment. The introduction of legged mobile platform into humanoid robot, will assist the humanoid to perform the task in unstructured and dynamic environment.

1.3 Application of Humanoid Robot

1.3.1 Home Management Services

Humanoid robot to observe the house in the absence of people. Humanoid robots can be controlled remotely by the people with the help of a simple mobile terminal. Humanoid robot can perform the operation of household activity in the absence of people. They can check the apparatus condition inside the house. [9].

1.3.2 Healthcare

The verbal and gesture interaction of the Humanoid Robots can serve the patient in the hospital environment [10] .

Humanoid robot assisting the nursing staff to take care of the patient. It can provide the support while physically handling the patient in efficient way.

1.3.3 Humanoid robot for aging people

The older age people would like a humanoid robot for assisting to daily routine task. With the recent advancement in technology of humanoid robot, it can prevent old age people from falling, medication alerts and managing the location of old age people[11]. These tasks need to maintain independence and dignity. The old age people are no longer able to care themselves and their lives. It is also disrespectful to older age people using a device that look like a toy. The appearance of humanoid robot influencing the old age people. Humanoid Robots are appreciated for their ability to offer improved communication between old age people and healthcare professionals.

1.3.4 Industrial Application

The collaboration of Human and humanoid robot contributes the sustainable growth of factories. A place in which humanoid robot and people can work together to reach the goal [12]. Safety is the most critical aspects in industry. A humanoid robot to look after the safety of worker inside the industry while execution the task. Some tasks are complex to be performed by human. They require engineering special tools. An effective bidirectional human–robot communication contributes growth and safety to industrial development

1.3.5 Space exploration

With the advancement in satellite technology , satellites have been deployed into space for exploration and colonization of other planets [13] . The aim of this efforts to establish the possibility of life on other planets. Such plans require to create living environments on the planet. Humanoid robots can work and assist in space to establish the living environment. Humanoid robot can interact with unknown environment of space can interact with astronaut.

Application of humanoid robot is not limited to these areas only. It can be used in different field of industrial and non-industrial application.

1.4 Motivation

The child suffering from the autism spectrum disorder disease deficit in social interaction and communication to real world. Similarly, the old age people unable to do the household activity in the home alone environment. In such type of scenario, a walking machine humanoid robot plays an important role to the social life of old age people and child suffering from the autism disorder disease.

The behavioral based interaction and self-decision-making humanoid robot recognize the eye's glaze and behavior of the old age people and child. After recognizing these parameters, humanoid robot executes the task for the people suffering from physical and mental disorder disease. Till today, partial human thinking behavior implemented in the humanoid robot. The motivation of this research work is to implement the human thinking in the humanoid robot, so that it can serve for the social service to the society.

1.5 Research Contribution

- The major contribution of this thesis is development of model free based reinforcement learning control algorithm of an autonomous humanoid robot.
- The other contribution of thesis is modified the existing humanoid robot skeleton, so that it can move fast, run and take sudden turn.

1.6 Thesis Outline

- Chapter 2 concludes the literature review about the history of humanoid robot, current generation of humanoid robot, bipedal robot motion, mechanical design of humanoid robot, control architecture of humanoid robot and development of model free based reinforcement
- Chapter 3 concludes the biomechanics of lower body of humanoid robot. Biomechanics deals with the motion and orientation of joint position. The design input parameter considered from the standard human measurement and decided the link length and joint trajectory.

- Chapter 4 introduce the mathematical modelling of the humanoid robot. The kinematic and dynamics of the lower body evaluated for the selection of the joint motor. The gait trajectory and smooth motion evaluated for the different conditions. The kinematic and dynamic model implemented in the MATLAB platform and validated with the Multibody Toolbox of MATLAB
- Chapter 5 introduce the simple control architecture of the humanoid robot for joint control. The controller gain parameter selected on the joint stiffness and system design for critical damped. The computed torque control law implemented in MATLAB platform and interfacing is done with the proposed model.
- Chapter 6 describe the method to develop the model free based reinforcement control algorithm. The positive reward awarded to the system to execute the task. The optimal policy decided for the Q-learning algorithm and average randomness for controlled has been evaluated to tune the controller. The reinforcement algorithm implemented in the MATALAB and simulation of experiment carried out.
- Chapter7 include the result for the different condition of the environment. Simulation were carried out for the different values of learning parameters. The optimal control policy was decided for the tuning of controller.
- Chapter8 concluding and summarizes the research work on the development of reinforcement of reinforcement learning algorithm for motion control of an autonomous humanoid robot along with the suggestions for future work regarding the development of hardware parts.

CHAPTER 2 LITERATURE REVIEW

Development of humanoid robots can do useful work in the real world. Research community and companies have been working on bipedal locomotion and humanoid robots for decades and trying to find real world home for the humanoid robot. In the robotics research, humanoid robotic is a challenging and emerging field and received significant attention during the past year. The major problems is the implementation of self-awareness information processing in the humanoid. After implementing the self-awareness processing, humanoid can deal with the real world. Recent advancement in technology and research, contributing the development of humanoid robot and it is visible to the world. This section of Literature review carried out for the latest humanoid robot existing in the world.

2.1 Current Humanoids

2.1.1 Sophia

Sophia is a humanoid robot with an attractive female face. The main technological quality of Sophia is her ability to learn human behaviors through her interaction with people. She built so far only to its waist. She has an electronic synthetic voice system that allows her to speak and gesticulate as she makes her speech. One of the goals of its creators, is that Sophia manages not only to maintain an intelligent conversation with human beings on any topic, and on her own initiative, but that her dialogues are always accompanied by the emotional charge that characterizes the normal conversation between two people[14]. The first concept work behind the Sophia robot is measurement of consciousness of a cognitive system while reading and conversing with the help of Tononi Phi values[15]. The Phi values measured for the short-term importance depends on the size of the probability distribution vectors.

The second concept work behind the Sophia robot is realistic humanlike robots for treatment[16] . Real robots can give a profoundly precise reproduction of a human-to-human social experience, with exceptional advantages of controlled repeatability, vigorous reiteration, and nonappearance of advisor dissatisfaction or other uncontrolled negative effect towards a patient or customer. The third idea work behind the Sophia is the manner by which individuals normally portray robot behavior [17] .The target of this examination is to build up a metaphysics appropriate for end clients, that can be utilized to portray robot conduct dependent on an investigation of social insurance robots. The objective is to make an end client robot programming condition, at first for human services experts who don't have programming knowledge, and in the long run for end clients in other robot spaces.

2.1.2 Atlas

Atlas is the world's most dynamic humanoid robot. Atlas is able to balance the whole-body motion to achieve the task with the help of both -handed mobile manipulation. It consists of twenty-eight joint and 1.5 meter high. It can carry the payload of 11kg. Total weigh of Atlas humanoid robot is 75 kg. The Atlas hardware developed with the help of 3D printing technology. Atlas is compact robot with high strength to weight ratio and have a large space. Atlas humanoid robot ability to manipulate the object in its environment. It can travel on rough terrain. The control system of Atlas robot implemented on robot operating System packages[17]. Robot operating system automatically generate and implement the software of a state machine that instant .The motion plan will be synthesized by grounding it to the robotic system's capabilities .A dynamics-model based method adopted in Atlas humanoid robot ,which calculates the required torques for the given motion that reduced the tracking feedback gains and make the Atlas more compliant. The controller layer determines the cartesian trajectories and track them by PD controller [17].

2.1.3 Manav

Manav is the first 3D-printed humanoid robot weighing 2kg and 2 feet tall. It consists of 21 sensors, two mikes, and two cameras on the head and eye socket.

The in-built processor and pre-programmed allow the Manav to perform the task on the response of human voice. Till today Manav is capable of walking, talking and dancing without the help of an input and in future it will lift the load. The open-source code helps to be taught to learn and respond like a human child. The rechargeable lithium polymer battery use for powered the Manav humanoid robot and it can work for an hour of a single charge. has a that can work for an hour with a single full charge. It can communicate with Wi-Fi and Bluetooth connectivity.

2.1.4 ASIMO

ASIMO, is a humanoid robot designed and developed by Honda. ASIMO has 34 degree of freedom, each leg has 6 degree of freedom and total 12 degree of freedom provided in the leg. Each arm has 7degree of freedom and total 14 degree of freedom in the arm. The hand of ASIMO has 2 degree of freedom for 4 fingers to grasp the object and total 4 degree of freedom to the hand. The head of the ASIMO consists of three degree of freedom and one degree of freedom is provided in the hip. ASIMO is 130 cm tall in height and weighing 54 kg and can carry pay load of 1kg[18].

2.1.5 iCub

The iCub humanoid robot is 105 cm tall and weigh 20.3 kg and total 53 degree of freedom. . The iCub robotic platform adopted by 20 laboratories worldwide for research and academic development of the robotic project. It is an open cognitive robotic plantroom. iCub can see and hear. iCub humanoid attached 53 motors to control the movement of head, legs, waist, arms and hands. It has the sensing capability of body configuration and movement. The iCub robotic platform has the two goal. First creating an open and freely available humanoid platform iCub and secondly is advancing of cognitive system by exploiting this platform in the study of cognitive development. The goal can be achieved by plan to construct an embodied system able to learn the interaction between environment and complex task. Further the communication being stablished between motor for manipulate the task[18].

iCub humanoid platform is a result of RobotCub project. To compute the physical interaction of rigid body to the environment and object, iCub simulator uses the open dynamic engine. Open dynamics engine is a reliable physics engine , computing the physical interaction between the object and environment [19].

2.1.6 POPPY

Poppy is a robust and accessible and 3D printed Humanoid robot. The open-source software and open hardware of Poppy humanoid robot allows the programming and experimentation of various robotics morphologies. Poppy is 83 cm tall and weigh 3.5 kg. It has 25 smart actuators. One LCD screen and two wide cameras are attached on the head of Poppy. The behaviors of Poppy were partially dependent on their bodies and controlled using pre-wired electronics circuit. The controlled using pre-wired electronic circuits. The Python programming control the Poppy hardware. The joint motor of Poppy control the human -robot physical interaction. The design of Poppy is modular and easily modify and adapted it to particular needs [20].

2.1.7 Romeo

The main aim of the development of Romeo humanoid robot is to assist a person suffering from loss of autonomy. The Romeo humanoid robot interact with the human, roaming around human's physical environment and helping their needs. Romeo humanoid robot has a close relationship between human. Romeo is capable able to extract a realistic perception and interact in natural way with individuals. During the development of the robot , a special attention given on the facial movements, voice and gestures , so that it will increase the effectiveness of the human-robot interaction[21]. Different layers of perception been analyzed in Romeo from sense to interact. Sensor and their fusion build 3D point cloud world. A relevant information gathered from the 3D point cloud help to learn Romeo to categorize emotions and instructions[22].

2.1.8 PETMAN

PETMAN is an anthropomorphic robot designed to test the protection against chemical warfare agents. The embedded chemical sensor inside the skin of the PETMAN humanoid robot measure and detect the chemical in the suit under controlled temperature and wind conditions. It maintains the dynamic balancing when pushed moderately from the side. PETMAN weigh 80 kg and 140 cm tall and can carry a payload of 23 kg. PETMAN is a free-standing biped robot powered by hydraulic power separately. Onboard control systems sensing, computing and control the movement. PETMAN has twenty-nine joints integrated with sensors for measuring position and force. One passive degree of freedom in each wrist and foot provide compliant interaction with the environment[23].

2.1.9 NAO

An approachable designed of NAO humanoid robot had been kept purposely. The height of NAO is .57 meters and weigh 4.5 kg. The appearance of NAO humanoid robot is similar to a human toddler. Out of 11 degree of freedom provided to lower limb and rest 14 degree of freedom provided to upper part of the body. Due to large number of degrees of freedom, NAO offers great mobility. The open loop engine is controlling the walking of NAO. Absence of feedback arise the instability in NAO movement. The open loop stabilizer controlling the motion straight walk and follow arc without falling on the flat and hard ground[24].The special pelvis kinematics design of NAO requires one motor. This especial pelvic design gives NAO to bend forward without movement of the legs. The MaxonTM coreless brush DC motors use for actuation the joint in NAO robot. Gyro meters and accelerometers gives the real time real time signal to the NAO open loop engine controller. Capacitive sensors are located in three segment of the body for touching the environment. [25].

2.1.10 RoboThespian

RoboThespian, the ultimate most iconic robot designed for human interaction in a public environment. The elegant design balanced the range of movement wherever it travels. In public environment RoboThespian can interact with the people and can communicate. RoboThespian has 24 degree of freedom. The body of RoboThespian is made from the aluminum. The muscle of RoboThespian is pneumatically actuated. The dc motor and passive spring elements simulate the body motion. The control of RoboThespian robot is done in two modes. First mode is the execution of preprogrammed scripts and secondly teleoperation using an interface. Incorporating all the features, RoboThespian, the ultimate most iconic robot designed for human interaction in a public environment.

The elegant design balanced the range of movement wherever it travels. In public environment RoboThespian can interact with the people and can communicate. RoboThespian has 24 degree of freedom. The body of RoboThespian is made from the aluminum. The muscle of RoboThespian is pneumatically actuated. The dc motor and passive spring elements simulate the body motion. The control of RoboThespian robot is done in two modes. First mode is the execution of preprogrammed scripts and secondly teleoperation using an interface. Incorporating all the features, RoboThespian can perform autonomously. The operator can command and control the movement and interaction in the teleoperation mode, the artificial muscles and dc motor are capable to create full-arm and abdominal movements. RoboThespian observes its surroundings with the help of head movement and through a high camera mounted on its forehead. [26].

2.1.11 Actroid-SIT

Actroid humanoid robot able to makes eye contact and gestures in the direction of a person trying to speak to her.The Actroid has a women -like figure and possesses 42 degree of freedoms . Out of 42 degree of freedom ,29 degree of freedom provided for the body guesture control and rest 13 degree of freedom provided for the facial expression control.

Since the Actroid joint is control by pneumatic actuators, it is very less chance to damaged damage caused by an accidental collision comparatively motor-driven rigid robots. The flexible gestures in real time can be generate due to a reconfigurable motion database. The reconfigurable motion database of Actroid has the two main features: motion interruptivity and motion parameterization. Whenever there is a failure due to speaker interrupts happens in Actroid robot's current reaction, it is terminating the social task and then switching to the next reaction. A human-like motion sequence obtained from the motion capture system. The reconfigurable motion database can extract similar sequence regions from all of the sequences by the function of the Match Web. Then, given a target parameter like 3D position of hand and the angle of face against the gestures, the target motion is obtained by adding all the similar motions with a given weight [27].

2.2 Bipedal robot motion

Bipedal walking in humanoid robot is a complicated task. Due to several degree of freedom coupled with nonlinear dynamics; locomotion problem arises. The human body control the leg movement with the help of biological rhythmic behavior. The biological rhythm is called central pattern generator. Central pattern oscillator controls the bipedal movement and whole humanoid body motion. In unstructured and unknown environment, humanoid robot is unable to control the leg movement. This section of literature review [28-48] carried out for the bipedal movement.

Lim and Yeap[28] has describe the movement steps in bipedal movement considering the human characteristics. Six servo motors, acceleration sensor, Lithium Polymer battery pack and remote control used in the bipedal robot project. The wireless remote controller is used to transmit the signal to the servomotor for the joint movement. Six push button switches controlling the movement of feet. Several experiments carried out on the hardware to get the correct posture of the locomotion and same value implemented in the software to control the movement.

J. Park[29] adopted pattern-based walking planner for the zero-moment point control. A pattern-based walking planner generates the trajectory of a set of pattern variables like velocity and direction. The pattern controller controlling the center of mass and horizontal angular motion to generate the desired motion of pattern. The inverse kinematics coupled with the joint servo motor. Pattern controllers is keeping track of the desired trajectory of bipedal robot.

Ken'ichiro[30] described an evaluative genetic algorithm to neural network controller. Camera captured the real time visual and parallelly it generates the guided swing motion for the robot. Neural network controller has four layers. One layer consists 2 neurons, second layer consists 4 neurons, third layer consist of 4 neurons and fourth layer consists of 1 neuron respectively. The synaptic weights and thresholds value are optimized by the genetic algorithm, which are real numbers. Simulation is carried out on the virtual environment considering the noise as disturbances. Execution of the program done on a parallel computer. The robot performs the task by decoding the gene of the best individual into neural network.

Yang, Chew, and Poo [31] describe the generating the gait for different type of terrain based on a truncated Fourier series formulation. The tuning will be done by the genetic algorithm. In this approach the feet of the robot always maintain parallel to the ground by adjusting the ankle-pitch and knee-pitch angles of both feet. For avoiding the topple, the zero-moment point criterion is used for evaluating generated gait given in joint coordinates. The genetic algorithm maintaining the zero-moment point to be within the footprint.

S. C. Y. Kim and Hutchinson [32] proposed a hierarchical planner based on the workspace decomposition. The workspace decomposition consists of the passage map, gradient map, obstacle map, navigation map and local map. Greedy Hierarchical Planner algorithm used to motion plan. The workspace decomposition and connectivity graph store in a data structure. The hierarchical algorithm consisting of local plans, global plans and sub goal.

The decomposition of three-dimensional maps encoded into two-dimensional workspace. The navigation map computes the passage map and the obstacle map. The robot is moving only to the free space of the target environment.

Niiyama, Nishikawa, and Kuniyoshi [33]investigate musculoskeletal movement in bipedal running. The patterns of kinetic data and muscle activity data measured for the leg from the human movement. The bi-articular muscles supply torques at hip and knee joints simultaneously. The activation of hip joint and knee joint motor command use the sparse coding of activation method. The required muscle force determined from the desired force. The human electromyographic data extracted for the muscle activation and the muscle activation pattern used for the activation of the athlete robot.

Inada [34] introduced the Matsuoka neuron model in central pattern generator and investigated for the bipedal movement. The neurons output generates the target angles of individual joints. The neuron creates the rhythmic oscillation. The central pattern generator involves in the rhythmic activities like locomotion. The trajectories of each joint are captured from the human movement. The genetic algorithm adjusted the central pattern generator parameter to generate the control output close to the desired trajectories. The central pattern generator adapts to extremal signals from the unpredictable environment. if the extremal signals are removed, the outputs of CPG return to the natural oscillation.

Thuilot, Goswami, and Espiau [35] analyzed the behavior of the compass robot gait model of the simplest biped robot. The two identical legs are jointed at the hip and the mass is concentrated the hip and two legs. Compass has no knee joint. In place of knee joint, the prismatic joint attached with the lower leg. All the joint in compass are passive joint, required not external power source. The compass gait consists of the swing and transition stages. In swing stage, the compass hip fixed nearby the point of support of leg on the ground. The other leg swings forward. In transition stage, the support is transferred from one leg to another. Compass robot is making stable walking on the inclined surface.

Nishino and Takanishi [36] discussed the algorithm to control the motion and added the computation module to compute to the algorithm to improve the generality. They discuss control method of the driven joints. The dynamic walking of humanoid robot is controlled by the nonlinear spring mechanism. The spring mechanism consist the wire and motor. The rotary encoder gives the angle feedback to control the tension in the driving wire. Coordination of the motor on both sides, allow the humanoid robot to take step forward. One motor control the spring tension of one side and other motor is controlling the joint movement of other side.

Inoue and Takanishi [37] develop a control method for whole body of the humanoid robot for dynamic walking. The control system consists of 15 AC servomotors and 16DC servomotors are controlled through boards. The trunk generates three- axis moment. The yaw-axis actuator generates the moment along the yaw-axis which is attached near by the neck of the humanoid robot. Swinging of the upper limbs generates the moment along pitch and roll axis. Algorithm computes the motion of trunk, upper limb and lower limb arbitrarily. The control system compensates the ZMP and yaw -axis moment.

Komatsu [38] proposed modified central pattern generator method to control the motion and force between the leg and ground for unknown environment conditions. The control architecture of the hybrid central pattern generator control method consists three layers. The first layer is creating the rhythm motion for the legs. Second layer control the force indirectly between the ground and foot and the third layer control the attitude of the hip. The rhythm generators consist of the neurons model and each neural model consists of four unit of oscillator. Oscillator generates the torque for individual joint.

Reil and Husbands [39] describe an evolutionary algorithm that control the stable bipedal movement in a straight line. To achieve this task no proprioceptive information is required. The recurrent dynamical neural network-based controllers implemented to achieve the bipedal motion. The population parameters are deciding the fitness and weigh in the recurrent dynamical neural network.

The weight of the joint movement and time span remains constant throughout the motion. The fitness function depends on minimum distance from origin and does not allow the hip to a certain height to avoid the bend in bipedal robot.

Kuffner [40] present a heuristic safe navigation algorithm for biped robots in obstacle environments. The approach is to build a set of feasible locations of footstep by computing footstep placements to an obstacle cluttered environment. Planner generates a sequence of footstep placement by taking the input from the environment which is collision free. The polygon-polygon intersection method used to avoid the collision for safe navigation.

Tlalolini, Chevallereau, and Aoustin [41] proposed an optimal walking motion with flat-foot and foot-rotation. The Newtonian-Euler's approach adopted for calculating the different joint torque. The optimal trajectory depends on the reasonable parameters. A cubic spline trajectory and constrains have been added to find the optimal trajectory. Simulation carried out for the bipedal robot walk with foot rotation and without foot rotation.

Tlalolini, Chevallereau, and Aoustin [41] established the localization technique using stereo vision for a humanoid robot. They found that the stereo vision creates the jerky motion while walking due to noise present in the environment. The feature-based approaches use depth maps for localization. The stereo vision system mounted on the top of the humanoid robot and the obstacle present to the environment has been captured. Motion captures system crates the elevation map and it provides the localization to the humanoid robot on the ground.

Rostami and Bessonnet [42] considers the instability during bipedal movement for single support phase. The approach minimizing the joint actuating torque, ensuring stability and controlled walking achieved by impact less and non-sliding heel touch. The Pontryagin maximum principle is applied for the computing optimal motion synthesis for joint trajectory. During the swing, the biped robot is very unstable and to overcome this problem, smooth motion and less energy consumption have been computed.

F. R. Copyright, F. S. Company [43] proposes the integrated motion control for the walking, jumping and running. The integrated motion control generates real time-based motion pattern based on the dynamics involve in the humanoid robot. The adaptive motion control method implemented for the controlling of zero moment point and leg ground contact. The zero-moment point has no acceleration and the height of center of mass remains constant. The constraint had been applied on vertical zero moment point trajectory and angular momentum. The simulation is carried out on the dynamic simulation software of QRIO humanoid robot and testing being done for the walking, jumping and running.

Caldwel and Bowler [44] explore the structure of pneumatic muscle to reduce the energy consumption. The internal structure of the pneumatic muscle actuator is different from the conventional pneumatic actuator. The inner layer of pneumatic muscle cylinder made from rubber tubing and the ends of the tube seal by the two aluminum plugs. The mechanism of bipedal design consists of free motion at the hips and knees. The limbs of the bipedal are constructed from steel and aluminum. The actuation is provided to the leg by the antagonistic pneumatic muscle actuators.

Grizzle et al. [45] introduces MABEL platform for the study of walking and running locomotion in bipedal robots. The first purpose of this platform is the development of new feedback control system for running and walking on the rough terrain. Second purpose is to create motivation for building robot for technology outreach. The mechanical design of the MABEL bipedal robot is a planar robot. It consists of five links. Two legs with knee are assembled on the torso and the legs are terminated in point feet. A real time computing and data acquisition system acquire the data from sensors. The software framework switching controller module and controlling the he robot state.

Y. Kuroki et al.[46] developed a SDR-4X small biped entertainment robot. This software creates whole body motion. The SDR software consists motion creating system and foot trajectory generator system. The upper body motion edits and created by the motion editor creator and the lower body trajectory creates by the

foot trajectory generator. The upper body notion creates by the motion designer by loading the music into the system. The dance steps create with the help of foot trajectory generator.

Huang et al.[47] obtained adjustable step length and velocity during dynamic bipedal walking. The compliance of the joints is controlling the passive walking to obtain the natural motion robot experiments. A kinematic coupling used to keep the upper body centered between the two legs. The bipedal walker travels forward on level ground similar to the real robot. By changing the joint compliance, walking pattern is observed. The step length and velocity control the natural dynamics of walker.

Garofalo, Ott, and Albu-Schäffer [48] suggested a periodic walking motion for spring loaded inverted version of bipedal robot. The controller architecture of SLIP bipedal robot consists upper layer and lower layer control. The upper layer control is directly connected to the SLIP and ensuring a periodic walking pattern. The lower layer controller controls the force. The main function of the controller is established interface between the dynamics of the real robot.

2.3 Mechanical design of humanoid robot

Human body consists of bones, muscles, cartilages and joint. Push and pull of muscles control the movement of body. It is difficult to develop the muscular skeletal system in humanoid robot by mechanical components. The goal of mechanical design is to develop the humanoid robot similar to the human. This section of literature review [49-68] carried out for the mechanical design of humanoid robot.

Yu et al.[49]presents the mechanical design of the humanoid robot. They also proposed the control architecture of control system based on the multichannel communication system. The mechanical design of both leg of humanoid robot consists six degree of freedom. The length of the thigh is 0.3 meter and the length of the shank is also 0.3 meter.

Mass of leg include gear, motor and coupling is 16.5 kg. Each arm consists of 7 degree of freedom and total mass of the arm is 4.5 kg. The length of upper arm is 0.30 meter and fore arm is 0.22 meter in length. The design principle consists of symmetry of the body like human and high stiffness and light weight. Link and joint were fabricated by mechanical casting. The distributed control architecture implemented to control the joint motor movement.

Borst et al. [50] develop a research platform for the manipulation of two-handed dexterous arm of the humanoid robot. While designing the arm of the robot a 2meter of workspace is considered. The robotic arm has anthropomorphic kinematic configuration that can easily grasp the object from both the hand. The mechanical design consists of torso, arms, head, neck and fingers. Whole the assembly is seated on the table. The dexterous arm has 14 degree of freedom and hand has 24 degree of freedom.

Kanehira et al.[51] developed an advance leg module for rough terrains and prevent tippling over to protect the damage of the robot body. The design of the advanced leg consists six degree of freedom. Three degree of freedom provided in the torso and one degree of freedom in knee and two degree of freedom in ankle. The length of the upper leg is 300mm and the length of lower leg is 300mm. The ankle length is 91mm. The length of the torso is 120mm.Total weight of both legs is 17.2 kg and dummy weight considered on the torso is 22.6 kg. To prevent the tipping over, the design follows the cantilever type structure of the hip joint. The design mechanism for walking on rough terrain consist six-axis force sensor and rubber bushes.

Oh et al. [52] developed android Albert HUBO humanoid robot. It is137cm in height and 57Kg in weight. The actuator of the Albert HUBO humanoid consists of gear and DC motor. Planetary gear used for finger joint to reduce the backlash and small error. Finger and wrist movement does not affect the stability of whole-body motion. Harmonic gear used for the arm movement and leg movement, so that the system stability and joint position stability repeatability maintain.

The weight distribution van be done by keeping all the power source battery and controller on the torso.

Iwata and Sugano [53] propose an anthropomorphic design of TWENDY-ONE human robots that provide supports to elder women while securing contact safety.

The passive impedance mechanism used in the TWENDY-ONE robot. The upper body is attached on the base of omni wheel. The concept design of TWENDY-ONE humanoid robot started from the setting of task scene in daily life. The anthropomorphic design of upper limb consists of ram and trunk. The arm has seven degree of freedom with one redundancy and the four-finger hand has thirteen degree of freedom.

Fukaya and Toyama [54] propose the novel design of humanoid hand of TUAT /Karlsruhe humanoid arm. The hand of the TUAT /Karlsruhe consists of 20 degree of freedom. The first four fingers of the humanoid arm are identical and each one consists of three joints of four degree of freedom. The palm consists of two degree of freedom and the little finger is moving freely. The joint of finger is driven by one actuator. A special mechanism of link rod pulls the link plate and the finger moves. one actuator drive all the joints of four fingers which placed into or around the hand. When the proximal joint touches the object, the finger curls around the object and middle proximal part moved by the link.

Ogura et al.[55] describes the development of a humanoid simulator and design of robot WABIAN-2 humanoid robot. Each leg of this humanoid robot consists 7 degree of freedom and 2 degree of freedom to waist. The focus given on the development of lower body of humanoid robot. Aluminum alloy used for the fabrication of WABIAN-2 humanoid robot, so that body have low weigh and more stiffness. The 3D -CAD model used for the design of the Humanoid robot.

Akachi et al.[56] presented the mechanical and electrical features of the HRP-3P humanoid robot. The mechanical and structural features of the HRP-3P protecting the body of humanoid robot against the water and dust.

The mechanical features include the height, width and depth of the humanoid robot. The height of HRP-3P humanoid robot is 1600mm, width is 664 mm and depth are 363 mm. Total degree of freedom of this structure is thirty-six. One of the unique structures of HRP-3P is cantilever type hip joint and it allow the robot to move cross leg.

Ha et al.[57] presents the design method of open humanoid platform DARwIn-OP. The mechanical design of DARwIn-OP consists 20 degree of freedom. The center of mass lying between the center of hip. The frames of this humanoid robot are hollow in structures, so that anyone can put the sensor between the gap of frame. The height of the robot is 0.455meter and weight is 2.8 kg.

Endo et al.[58]present the design and evaluation of the limited degree of freedom of head of the robot for facial expression of the WABIAN-2humanoid robot. To provide the emotion expression in the humanoid robot, the ankle joint movement along yaw axis and the trunk joint along roll axis removed are removed from the humanoid robot. A new head design proposed f balancing the head. The light weight and downsizing of the head are considered to mount on the body. The wires and torsion spring attached on the yaw of eye. The lip is of spindle type and actuated by wires made by using springs.

Lohmeier, Buschmann, and Ulbrich [59] developed the fast walking humanoid LOLA robot. The emphasis given on the improving weight to achieve good dynamic performances. The height of the LOLA robot is 180cm and 55 kg in weight. The redundant kinematic structure of the leg is lightweight and allow the natural and flexible gait. The active toe joint occurs momentarily before the swing leg coming in contact of ground reduce the joint loading. The additional actively driven link allows the swing leg to be in extend more. The toe contact with the ground stabilizes the robot and allow forward.

I. W. Park et al.[60] developed the mechanical design of KHR-3 humanoid robot without hand. Pulley belt mechanism drive the joint of KHR-3 humanoid robot. Belt tension maintained by changing the motor position.

Hip joint of KHR-3 robot designed as tube type crossing structure. The internal part of the tube is almost hollow except gear actuation assembly. All the frames of the KHR-3 humanoid robot is 2D in shape. The closed kinematic configuration of the robot's provides more support when it is supporting on the ground. If some position error in the feet, the motor position error is more as compare to other joint motor error.

Ambrose et al.[61] developed space humanoid Robonaut robot work in the space environment to assist the astronaut. They focus on the upper body design of the Robonaut. The arm and finger of Robonaut are offering dexterity and sensing. The hand of Robonaut's will fit into astronaut's hand gloved. The hand has total 14 degree of freedom consist of forearm, wrist with 2 degree of freedom and the five fingers with 12 degree of freedom. The design of Robonaut 's arm is similar to the human arm. The arm has five degree of freedom and matches with the 14 degree of freedom hand. The head consist two color cameras provide virtual reality and depth perception.

Yamasaki et al.[62] proposed the basic architecture and design of PINO humanoid robot. The PINO has twenty-six degree of freedom. PINO robot's each leg consists 6 degree of freedom, each arm consists 5 degree of freedom, neck consists 2 degree of freedom and trunk consist of 2 degree of freedom. All the joints are actuated by the DC motor and consists 26 DC motor. The metal gears reinforcement against the high torque.

Tellez et al.[63] introduces a robotic platform Reem-B in the field of service robot. The cognitive abilities of Reema-B humanoid robot able to perform dynamic walking and interaction with the people. The physical structure of Reema-B humanoid robot consists of leg and torso. A tradeoff between stiffness and weight has been decided based on the finite element analysis, so that stiffness should be high, and weight be less. The kinematic structure is similar to another humanoid robot. The Reema-B humanoid robot has one hand with four fingers only. The fingers have 11 degree of freedom.

Tsagarakis et al.[64] developed the lower body of cCub child humanoid robot that is the advancement of the iCub humanoid robot. The new leg movement mechanism considered in the cCub humanoid robot. The kinematic layout of the cCub humanoid robot follows the six degree of freedom in each leg, three degree of freedom at the hip, one degree of freedom at the knee and two degree of freedom at the ankle. The leg hag andromorphic kinematic form consist of hip, thigh with knee joint, calf with ankle joint. The leg assembly desired passive compliance provided by the actuation unit. The hip joint is supported on single side supported cantilever structure.

Gienger, Loffler, and Pfeiffer [65] deals with the design and control architecture of the humanoid robot. For human walking motion, the pelvic rotation, pelvic roll, knee and ankle interaction and displacement of pelvis are major determinant. The joint torques depends on the movement of the pitch motion profile. The ankle joint actuated by two linear ball screw drives. Permissible speed of the ball- screws is controlled by the toothed belt drives.

Mohamed and Capi [66] developed a mobile humanoid robot to assist the elderly people working in home and hospital. The visual sensor recognizes the object. The laser finder sensor attached in the lower body of mobile platform. Robot start moving towards the target object after determining the target object. The upper part consists of arms, gripper and head. Whole upper body is attached on the wheel. The mechanical design done using Solid works program. It allows the arm to be checked and visualized at the same time.

Wyeth et al.[67] describe the design of humanoid robot. The complete CAD model and specific motor and transmission selected for developing the mechanical design and it is under development phase. The mechanical dimension of the body is taken the biomechanical property of human body scaled to a height of 1200mm.The autonomous humanoid robot consist 23 degree of freedom. The leg and abdominal consists of 15 degree of freedom and remaining 8 degree of freedom consist with the upper body parts.

J. Kim et al.[68] proposed the development of novel biped walking of the humanoid robot Robray. The focus is on development of lower body considering two design of humanoid robot. The first design considers the experimental platform to measure the torque and force in walking. The second design consider the compliant and elastic leg mechanism to reduce high frequency. The kinematic configuration follows the six degree of freedom in each leg, one degree of freedom in waist and two degree of freedom in ankle. The range of movement of hip joint increased by providing the offset between the axis of hip joint.

2.4 Control architecture of Humanoid robot

The controller architecture of humanoid robot has extremely complicated due to exact model including the inertia and inverse calculation. The second order response noisy. A robust control system needs to control the bipedal motion and whole-body motion of the humanoid robot. Several degrees of freedom in the humanoid system makes the system unstable and unable to do the desired task in the unstructured and unknown environment.

This section of literature review [69-88] carried out for the controller architecture for the humanoid body motion control.

Burghart et al.[69] presents the cognitive architecture of humanoid robot. Cognitive architecture is a mixture of three-layered hierarchical form and behavior specific module. The control system of humanoid robot is separated into different module. Each part has its own software and hardware module. The first layer of cognitive control architecture consists of sensor and motor and its need not required any access form the system knowledge database. The data coming from joint position sensor, force sensor and tactile artificial sensitive skin passed to the middle layer and task execution. The middle layer of cognitive control architecture recognizes components of the system. These recognize components access the database, where information is stored. The active model of humanoid robot serves for the short time memory and provide environmental information about the object.

The third layer within perception are organizing all the components information in a single modality. The highest level of perception interprets actions by the user.

Kanda et al.[70] propose a constructive approach for implementing a behavior in Humanoid robot to interact with people. The knowledge obtained from cognitive experiments used for designing the behavior of humanoid robot and guide to communicate with people. situated module and episode rule are the basic component of the constructive approach. To govern the execution, order the robot system executes situated modules and episode rules sequentially. The interaction between humanoid and people based on the situated module. Episode rules helps the humanoid robot of interaction with people. The current state of the humanoid robot is compared by the all episode's rule and execution history to determine the next situated module to execute next.

Rosenblatt and Payton [71] present fine-grained layered control system for robot control. The fine-grained control architecture provides intelligent control mechanism through composition of simple layers. The behaviors are divided into a set of decision making. Each unit of network model receives inputs from other units and from external sources. After receiving the input, the network process an activation level and generate a single output. There are no interconnect homogeneous units in structure network and from a distributed knowledge. The distributed behavior expressed by the output values among several units. unit. A behavior directly indicates that which choice ids desirable.

Brooks [72] has adopted a layered approach control system for autonomous mobile robots. The outline of this approach is decomposing the problem into the parallel task. The robot control system starts with achieve lowest level task and never alter the system. This is zeroth level control system. Another layer control system builds and examine the data from low level system to achieve new task. The low level continuously unaware of the upper layer processing. The same processing is repeated to achieve new task. Each processor controls the specific task and runs asynchronously.

Ly, Regenstein, and Asfour [73] presented a distributed control architecture for achieving the goal task for a humanoid robot. The control architecture organizes the three levels and mapped the functional features int the hardware and software module. The control architecture consists of task planning, task coordinating and the task execution layer. In task level planning, the task description receive form the human operator or autonomously and allocate the subtask for multiple subsystem of the robot. The subsystem controller was selected. After selecting the subsystem controller, actions for the execution level generates by the task level coordination to achieve the given task. The specified sensory motor control command executed by the task execution level.

Wang and Butner [74] describe a computer control architecture to optimize the computational processing of the robot control. The proposed control architecture decomposes the task parallelly into the system. The processor receives the information from the sensor and pass the trajectory information to the robotic processor. The trajectory will be interpolated by the robotic processors. The trajectory point set by the servo controller. The interfacing between control system and manipulator actuator being established by interface card. The host computer is not involved in the real time processing. Decomposition of the task simultaneously, the system response takes very less time to execute.

Yoshihiro Kuroki et al [75] developed entertainment applications in SDR-4X humanoid robot. The real time sensor based adaptive control applied for the SDR-4X humanoid robot to control the body motion on rough and unlevel terrain. The stabilizing motion control establish the stabilization of whole-body motion in real time and generate obstacle avoidance motion for the upper body motion. The robot inclination and body posture calculated with the help of accelerometer and force sensor. The adaptive control system realizes the deviation and control the body posture and prevent to falling down.

Simmons and Apfelbaum [76] developed task description language to control the robot. The structure of TDL is the extension of the C++ language. The task control architecture consists of three layered i.e. planning layer, executing layer and the behavior layer. Tasks are defined in TDL language in the similar manner like C++ language. The class identifier like goal, command, monitor and exception followed by its argument. The TDL do not have a return value so that control is not returned until the next command have been handled.

Khatib [77] proposed a framework for the controlling of end effector force on the constrained environment. The two-level control architecture enhances the performance of the position and active force control of robot manipulator. The real time position and force control implemented in the operational space with obstacle programming system. The end effector equation established in dynamic decoupling. For the stabilization of the redundant manipulator, the joint force and dynamic behavior identified.

Mansard et al. [78] present a framework for the implementation of stack of task for the humanoid robot to work in the collaborative environment. The stack of task consists of the task definition and handling of the set of tasks. The software framework started with entities and graph of entities. At one time control iteration has to be performed. For each task system computes the error related to a task. The scripting of the software framework interface to handling the task.

Posadas et al [79] design a portable and modular control architecture for controlling the mobile robot. A distributed blackboard communication established between the mobile software agent. The propose architecture reduce the temporal problem by separating the elapse communication time from the execution time. First the architecture establishes the offline communication to move the robot by real time soft bus and then process the code. When offline communication terminated, the mobile robot obtained the agents. The agent executed the task without any external communication.

Erhart, Sieber, and Hirche [80] developed impedance control scheme for robot cooperative manipulation. The cooperative manipulation reduces the internal stresses on the joint. The kinematic uncertainties arise due to improper grasping of the object. The impedance control scheme computed the end effector trajectory and remove the kinematic uncertainties. The evaluated trajectory is compatible with object motion. During manipulation task, the kinematic coordination achieves by the closed loop manipulator dynamics.

Asfour et al.[81] propose a hierarchically control architecture of humanoid robot for household activity. The hierarchical control architecture has three layer and structured into task planning, coordination and actuator-sensor level. In the task planning level, the task will recognize from user and specifies the subtask for the robot. The corresponding actions generates by the task coordination level. The sensory -motor level execute the task to achieve the goal by the humanoid robot.

Feil-Seifer and Matarić [82] describes a novel control architecture of B^3IA in autonomous robot system for the behavior intervention of autism spectrum disorder children. The control architecture of the behavior-based behavior intervention Architecture (B3IA) consists of the sensor and the interpreter module. This module helps the robot to observes and control the behavior of humans and objects in the environment. The operational decision is made by the robot in the task module. The operation of hardware is controlled by the effector module. The human actions and the robot actions are stored in the activity history module. The analysis and interaction of robot with human evaluated in the evaluation module, which can be used as a robot's parameter.

Galindo, Gonzalez, and Fernández-Madrigal [83] implemented the human–robot-integration control architecture into a robotic wheelchair. This control architecture permits a person to deliberate the activity. The human -robot control architecture scheme made up of a number of elements. Module are grouped into three layers. The hierarchical and symbolic representation of environment maintain in the deliberative layer.

The internal world model used to produce plans and establish the communication human-robot communication. In the execution layer, the sequences of the task executed and supervise the information receives from the robot's sensor. The functional layer controls the navigation and manipulation between two point and provide the guidance.

Naumann et al.[84] deals with the control architecture of process-oriented programming for robot cells that enables in the production environment. The interconnector module of the software takes as input descriptions and processes. The process-oriented programming used to trigger the machining operation. To understand the process command, an ontology introduced. Ontology considered relevant classes of the robotic domain.

Liu, Iberall, and Bekey [85] proposed the adaptive neural network for robot hand control to grasp the object. The control architecture based on the prehensile function of human. The object analyzer module establishes the relation between object shape and the grasping modes. The object analyzer module generates suitable grasp mode for the robotic hand. The neural network generates the eight generic grasp mode for the robotic hand. This approach reduced the building of device dependent grassing mode.

J. Y. Kim et al.[86] used distributed control architecture to control the joint of humanoid robot to reduce the computation time of main controller. The main controller is attached on the back of humanoid robot and communicates with the sub controller in real time through controller network area. Master and slave feature of the controllers connected with CAN bus line. The controller can receive and send the data to sub controller at same time. The sub controller design separately for the joint motor control and inertia sensor.

Yokohama and Takashima [87] developed open hardware platform for the humanoid robotics. The virtual humanoid robot platform is compatible with real humanoid robot as it is. The OpenHRP has the dynamics computation, contact and collision computation and unification of the controller features.

The OpenHRP implemented on the common object request broker architecture, which support the C++ ang java programming. The simulation of OpenHRP controlled by a CORBA client on integrated simulation environment.

Rohmer, Singh, and Freese [88] introduces a virtual robot simulation platform for the integration of actuator, sensor and control. The distributed control technique used in the versatile and scalable robot simulation platform. Three technique used to achieve the simulation. The first technique is execution of the control code on another machine. so that computation time for simulation is very less. The second technique is execution of control code on the same machine for other process than the simulation. The third technique is execution of control code on the same machine other than the simulation loop.

2.5Reinforcement learning control algorithm

The humanoid robot can understand the unstructured and unknown environment to perform the desired task. They must learn from the environment and execute the task without any human programmer or user. Reinforcement algorithm helps the humanoid robot to take the decision in the critical condition. This section of literature review [89-108] carried out for the reinforcement learning in the humanoid robot. This is the current trends is going on the developing the humanoid platform for the real environment.

Velentzas et al.[89] applied the reinforcement algorithm in assistive robot for educational application. The child's gaze provides the information to the robot. The reinforcement algorithm has a set of state. State is the dimensional features. The action has finite discrete set of action and generate set of actions for the different state. The Q-learning rule helps to choose the action depending upon the task. After choosing the action, the transition takes place and a reward associated to the action and learns from the past history. The reinforcement algorithm decomposes the task into set of discrete action, so that it can be easily understood by children-robot interaction.

Katic and Vukobratovic [90] present the hybrid control of intelligent control system for humanoid robot. The reinforcement learning accumulated the knowledge from the dynamic balance of the humanoid robot and improving the gait during walking. The control architecture of gait synthesizer has three components. The neural network trains the action selection network using the error signal received from the external reinforcement. For the desired state, the action evaluation maps a state and a failure into a scalar score. The stochastic action modifier use the recommendation action and reinforcement to produce a dynamic walking.

Kober and Peters [91] work on episodic reinforcement learning to control the motor primitives in dynamic situation. The policy gradient method used in reinforcement algorithm. The motor primitives described as two coupled differential equations i.e. a canonical system with movement phase and possible external coupling. For the desired control of motor both the dynamics of system are choose for stable condition. The deterministic mean policy depends on the joint position and the basis function. The basis function is the motor primitive parameters.

Kartoun, Stern, and Edan [92] proposed a collaborative interaction between human and robot based on reinforcement learning. The learning is based on collaborative Q-learning approach and provides the robot to self-awareness and autonomy. In collaborative Q learning algorithm, there is two levels of collaboration between human and robot. In the first level, the robot decides the action and update its state-action values. In second level of collaboration, robot takes the request from human advisors. Robot is switching from the autonomous mode to semi-autonomous mode based on the polices.

Wawrzyński [93] demonstrates the reinforcement learning algorithm for humanoid gait optimization. The actor-critic learning applied for the experience replay and fixed-point method to determine the step size. The Markov decision process provides the solution for the reinforcement algorithm to control the humanoid robot gait.

The control process of actor-critic works in discrete time to select the state and select the proper action. The transition between current state to next state happens and a reward assigned to state and action. The stochastic control and value function update the learning parameter based on the data collected.

Kati and Vukobratovi [94] proposed the fuzzy reinforcement hybrid control algorithm for the bipedal robot locomotion. The controller has two feedback loops around the zero-moment point. The centralized dynamic controller keeps tracking of the robot's normal trajectory and a fuzzy reinforcement feedback compensate the dynamic reactions of the ground around the zero-moment point. The fuzzy reinforcement control algorithm structure based on the actor critic temporal difference method. The policy represents the set of control parameters.

Frank et al.[95] proposed the reinforcement learning which could control the iCub humanoid robot. ICub learns a world model from experience and controlling the actual hardware in real time with some restrictions. Reinforcement learning discretize the real configuration of the robot in configuration space. The modular behavior environment of iCub humanoid robot generate the action and robot try to go in the transition state.The Markov model develop the path planner and connect the state to the near state.

Christen and Stev[96] proposed the deep reinforcement learning algorithm to train the control policies for the humanoid robot interactions. The control problem is formalized from the Markov decision process. The input to the control policy is, joint position, velocity and sensor reading of the hand. The motion capture system captures the position of the hand. The output of the control policy actuates the humanoid arm. The reward is provided to correct end configuration of the humanoid arm.

Lober et al.[97] improve the reinforcement learning using Bayesian optimization for whole body motion control. The Bayesian optimization is a nonlinear and nonconvex optimization technique. It evaluates the cost function in the robotics and optimize the set of parameters.

To ensure smooth trajectory, the whole-body control guided by the task in a series of waypoints. Three components of costar are evaluated for the execution of task. The optimization variables selected from the trajectory waypoint.

Hester, Quinlan, and Stone [98] describe a model-based reinforcement algorithm with decision tree to train the humanoid robot to kick goals. The model-based reinforcement algorithm, learning takes place aggressively during model learning. The Q-learning approach adopted for the model free reinforcement learning. The Q-learning update the state -action for every state -action pair. The reinforcement learning with decision tree take the action with a highest value and entering into a new state. After entering into a new state, award will be received in the new state. Observing new experience through the model, the algorithm updates the parameter through the model.

Riedmiller et al.[99] work on the application of batch reinforcement learning in challenging and crucial domain. Reinforcement learning help the robot to gain the ideas form the repetitive interaction from the environment. The batch reinforcement control algorithm consists of sampling experience, training and batch supervised learning. The training pattern set estimates the value function. The batch supervised generates a new estimate for the value function form the training set pattern. The behavior-based approach used to implement the reinforcement algorithm to take the decision.

Iida [100] proposed adaptive allocation method for reinforcement control algorithm for humanoid motion control. The actor critic learning adopted for the reinforcement learning. This method has a separate memory to represent the policy i.e. independent from the value function. The actor calculates the action value for the humanoid robot when it observes the state in the environment. The critic receives the reward and provide the temporal difference. The learning is simulated on the virtual body of the humanoid robot to stand up from a chair. The humanoid observes the wait, knee, ankle and pitch angle of body.

The humanoid robot learns to fall down backward. Afterwards it falls down forward. Finally, it stands up and control its body.

Sko [101] proposed the dynamic control approach for the humanoid bipedal walking. The controller involves two feedback loops. The computational torque controller receive the input from impact force controller and reinforcement controller. The reinforcement controller maintains the torso movement with the help fuzzy feedback. The policy gradient reinforcement learning control the trajectory of dynamic walking of the humanoid robot.

Danel [102] proposed actor critic neural network architecture for continuous action policy of reinforcement learning. The deep deterministic policy gradient method controls the humanoid body control task. The deterministic policy developed by the actor network. The action value generates by the critic network. The temporal difference minimized by the training of critic network. The immediate reward received upon the action and update the learning parameter to the database of control architecture.

Peters, Vijayakumar, and Schaal [103] discussed the different approach for reinforcement learning algorithm for humanoid robot. The natural actor critic learning control the motor of humanoid robot. The movement plan has set of joint position and joint velocity of the humanoid robot. The system has point to point continuous movement i.e. the episodic task of the reinforcement algorithm. The evaluation of basis function for the value is done by the actor critic network.

Guenter et al.[104] developed algorithm for programming robot by demonstration. When unexpected perturbations occur, robot is unable to perform the task. The reach of constrained task to the robot by a learned speed trajectory. The natural actor critic network evaluates the policy by approximating the state action values. The simulation is carried out for the cubic box and obstacle. Using the reinforcement learning, the system takes 330 trials to achieve the goal.

Lowrey et al. [105] developed the control policies in simulation that can transfer to dynamical physical system. The policy gradient learning method used in reinforcement algorithm to optimize the parameters. The natural policy gradient algorithm pushing the task to learn. The training of the policy determines the action to take and gain a good reward. The structure of training informs the policy behavior with the time required to execute the task. The reward function reduces the gap between the robot and the target.

Kober, Oztop, and Peters [106] develop the reinforcement algorithm which maps the circumstances to meta parameters. The motor primitives used for the meta parameters learning. The dynamical movement of the motor represented in the first order differential equation for the critical damped. The goal parameter is the function of the amplitude parameter represents the complex movement. All degree of freedom of the system synchronize in the dynamical equation in the canonical form

I. J. Silva et al. [107] proposed a reinforcement learning algorithm to optimize the parameter values for the generation of gait pattern in humanoid robot. Locomotion control achieve by the central pattern generator. The three-oscillator attached in the foot of the humanoid robot. Each oscillator has six sub oscillators related to the axis and configured to the parameters. The parameters are divided into three groups of offset parameters, oscillation parameter and feedback parameters. Two parameters have selected for the optimize the gait.

S. K. Kim et al.[108] demonstrated the intrinsic interactive reinforcement learning algorithm for human robot interaction based on the gesture posture. The human electroencephalogram generated feedback used for the reward. The leap motion controller recognizes the human gesture to learn the robot and parallelly the robot maps the gesture for action. The contextual bandit approach used to enable the robot's action provided by the human gestures.

2.6 Research gap

After a detailed literature review, it has been found that there still exist problems in the mechanical design, locomotion and control of the humanoid robots.

- Stability of humanoid robots is not achieved fully in dynamic environment.
- Bipedal locomotion of humanoid robots is not stable in rough terrain.
- Fast walking and running of the humanoid robot are challenging. Sustained and stable sudden turns of the robotic structures remain unsolved problem for the robotics research community.
- Controlling of robotic manipulators with several degrees of freedom in humanoid robots is a complicated task.
- Self-awareness in humanoid robots while taking the decisions in an unknown environment is challenging.

2.7 Research Objective

The robotics research community is trying to implement the human thinking ability and behavior in the humanoid robots in such a way that it can understand its environment and take decisions in critical condition. The reinforcement learning algorithm allows the humanoid robot to learn, think and perform the action in unknown environment. Development of reinforcement algorithm for several degree of freedom is a very challenging task. Therefore, this research work aims to develop reinforcement learning algorithm for motion control in an autonomous humanoid robot.

2.8 Summary

This chapter introduced the background information of history and current generation of the humanoid robots. The requirements, mechanical design, bipedal locomotion, control and self-awareness performance evaluation methods are explained. Most of the difficulty arises due to the gait trajectory. Artificial Intelligence programming adopted for the smooth gait trajectory and whole humanoid body control. In the literature review, issues such as locomotion, mechanical design, control architecture and self-awareness were identified.

CHAPTER 3 PROPOSED 3 D MODEL

The proposed model considers the lower parts of the humanoid robot. The model consists of ten degree of freedom with each leg having five degree of freedom. Both the ends of the legs are connected to the torso. The Torso is the rigid body. A 3-D CAD model of humanoid robot is made on the Solid works software. For the purpose of designing and developing the lower body humanoid robot, the anatomy of human lower limb is taken into consideration [109].Table 3.1 shows the lower body parameter of the humanoid robot.

Table 3.1:Lower body parameter

Parameters	Dimension in mm
Foot length	240
Foot width	90
Foot height	100
Lower leg height	380
Lower leg diameter	370
Upper leg height	380
Upper leg diameter	480
Torso length	330
Torso width	150

The human body is designed in a shape which supports the skeletal system. The bones of human body are rigid while the cartilages components are flexible. The axial skeleton consists the skull, verbal column and ribcage

The appendicular skeleton of the human body arms, legs and supporting structure in the shoulder and pelvis. The pelvic forms the supportive framework for the lower body. The biomechanical factor considered for the design of lower parts. The lower body system of humanoid robot consists of left leg, right leg and the torso.

The kinematic configuration of the joint degree of freedom, joint range motion and link length relates the motion of lower body system to move together at any given time without the restricting each other's motion[110].Table 3.2 shows the degree of freedom of different joint and possible joint range of motion[111].

Table 3.2:Joint range degree of freedom and motion

Joint		Standard Human leg (in degree)	Proposed Humanoid leg (in degree)
Torso	Pitch	-15 to 130	-15 to 100
	Yaw	-45 to 50	-45 to 45
	Roll	-30 to 45	0 to 45
Knee	Pitch	-10 to 155	0 to 120
Ankle	Pitch	-20 to 50	-20 to 40

3.1: 3- D CAD Model

While creating the different parts of the humanoid robot, plane consideration had been taken. The torso and the hip joint created in frontal plane of sketch command. The rest of the part created in the sagittal plane of the sketch. In actual model diameter remains constant throughout the height of the lower leg and upper leg. The torso is kept fixed for assembling all the parts. A proper mate had been taken for the joining of the two parts. The concentric and coincidence mate had been taken for the assembly. The alloy of aluminum 1060 has selected for the proposed model. The aluminum alloy has high stiffness and low weight. While developing the assembly, the actuator part is not taken in consideration. Direct coupling of two links are taken into consideration for simplicity in the model. The details of each part shown in Appendix A.

Figure 3.1 shows the complete 3 D model of lower parts of humanoid robot created in the Solid work software without ground by taking the scale ratio of 1:4.

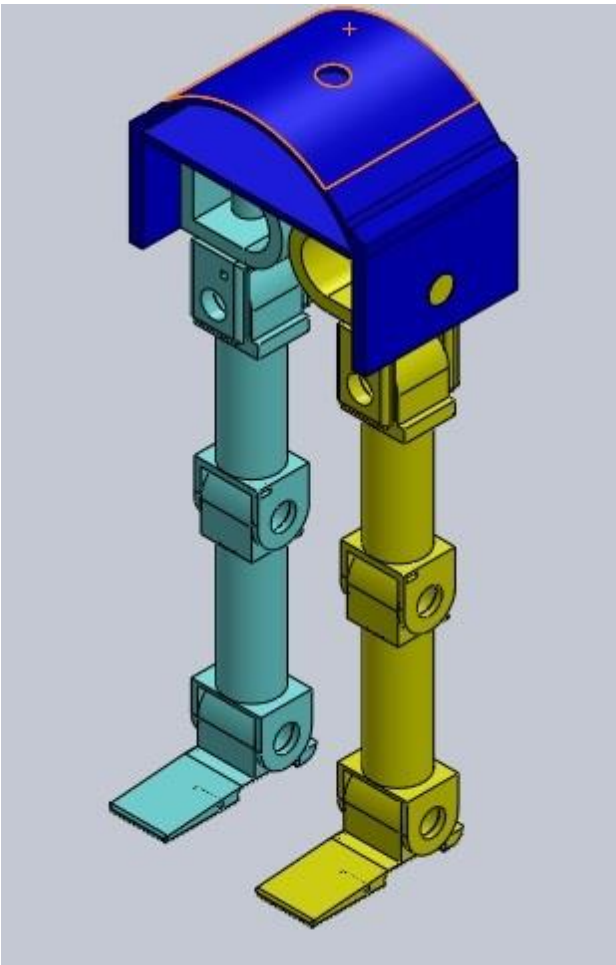


Figure 3.1:Proposed 3-D model of lower body

3.2: Simulink Model

The Solid Works 3-D model exported the MATLAB with the help of smlink module. The smlink module generates the ‘xml’ file of the system. Further these ‘xml’ file imported into the Multibody dynamic toolbox. The Multibody toolbox converted the ‘xml’ file of the system of block diagram. Ground of the proposed model is created in the MATLAB Simscape multibody toolbox. The Simscape multibody toolbox provides the simulation environment for the mechanical system. It integrates the different physical system to the model.

The proposed model develops in the Solid work platform and model exported to MATALB multibody toolbox. Figure 3.2 shows the MATLAB Simulink model of the lower body of humanoid robot without the ground.

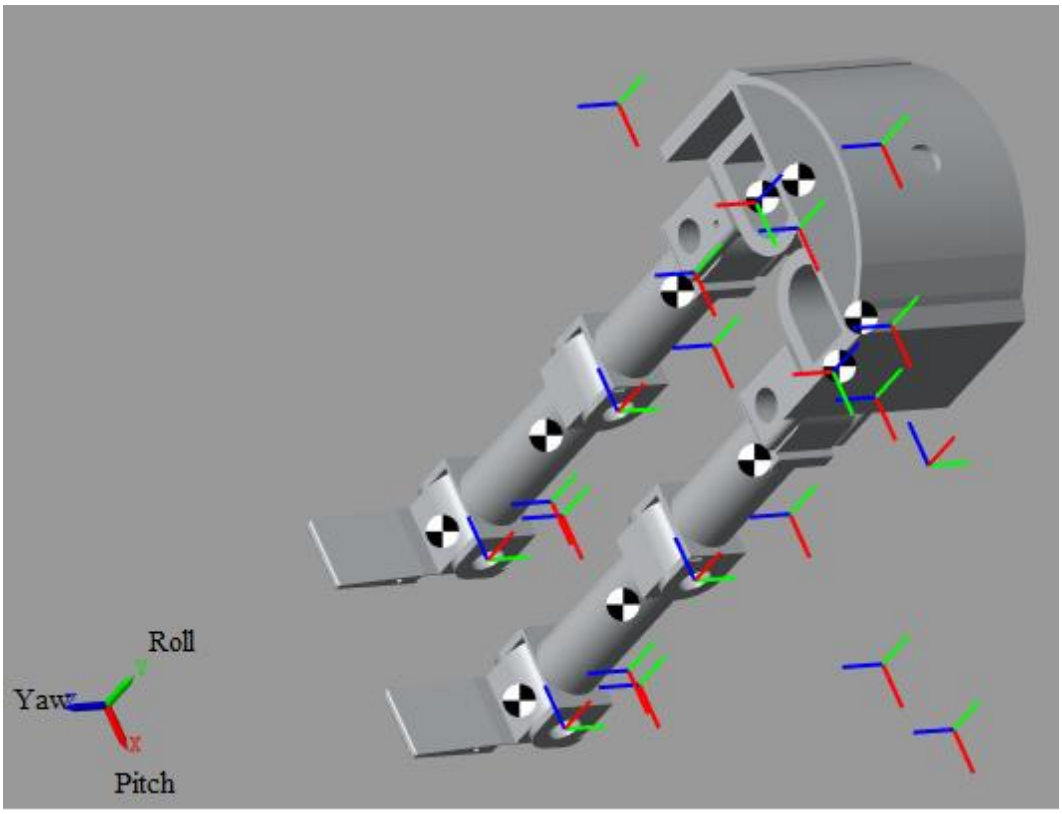


Figure 3.2:Simulink model without ground

Figure3.3-3.5 shows the joint, frame, and rigid body MATLAB block representation of the lower body of the humanoid robot.

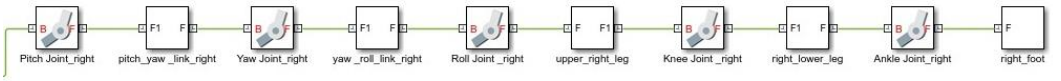


Figure 3.3:Matlab Simulink block of right leg

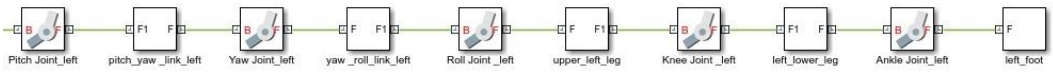


Figure 3.4:Matlab Simulink block of left leg

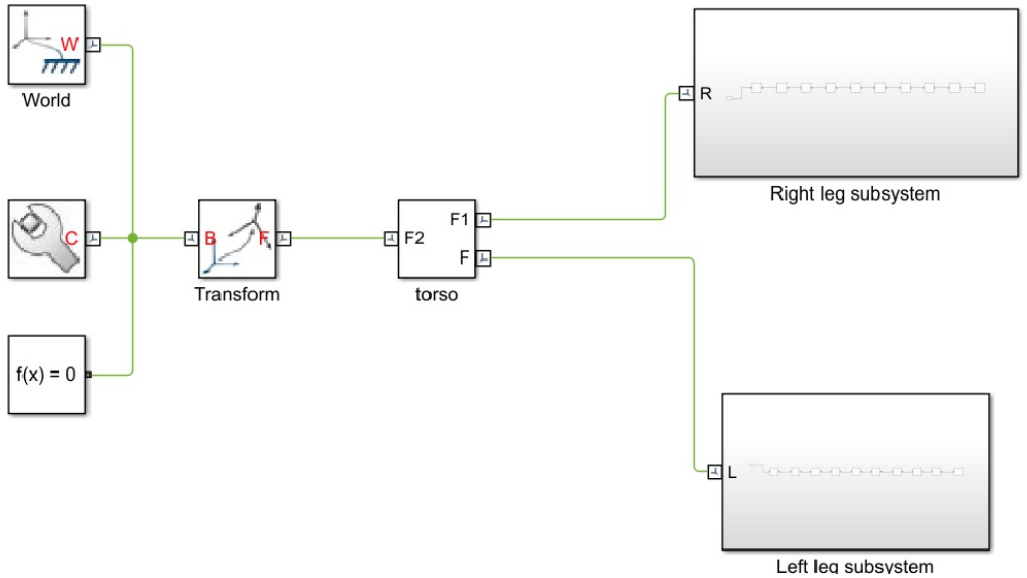


Figure 3.5:Simulink model of lower body

3.2.1 Simulink model with ground and contact forces

The feet of the lower body of the humanoid robot applying the forces on the ground. The friction force between the ground and the feet helps to walk. For creating the ground in MATLAB multibody toolbox, a brick element of body block is chosen. For each body block two rigid transform block are required[112].The physical properties of the ground like length, width and the height considered to create the feet. The height of ground selected along to the original height of proposed model which is developed in the Solid Works. The tangential force enables the robot to move forward on the ground. The normal force ensures that the humanoid robot will always be above the ground during simulation. Figure 3.6 and Figure 3.7 shows the Simulink block for the ground and contact forces of the foot interaction to the ground[113].

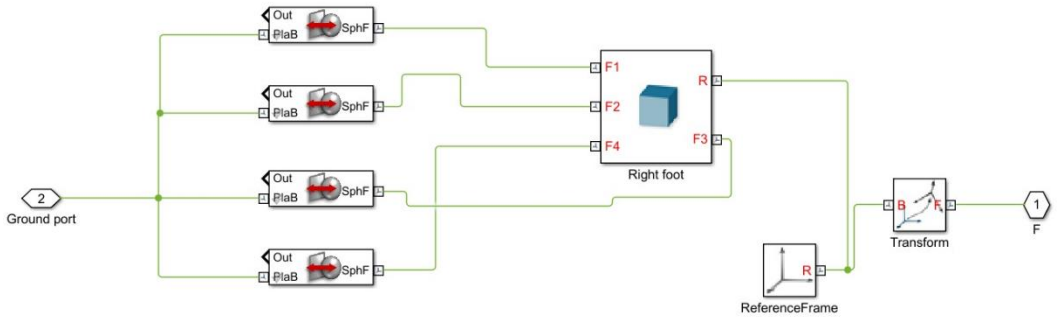


Figure 3.6:Simulink block diagram of contact forces of right leg with ground

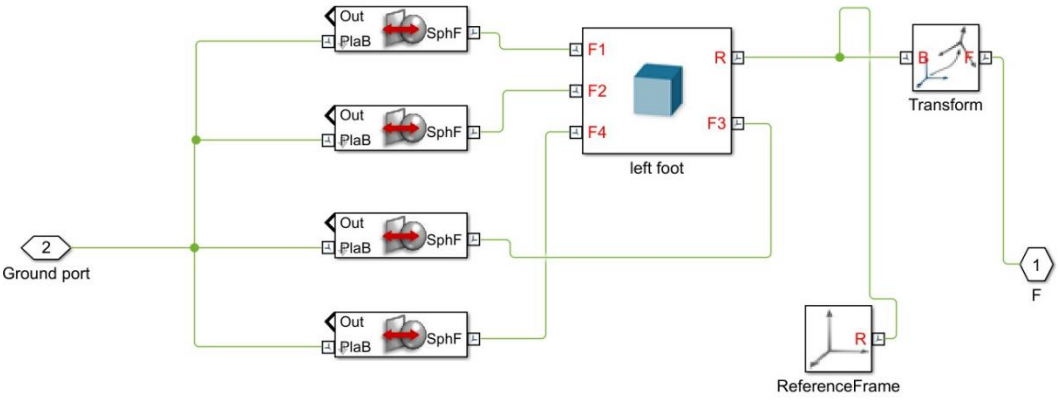


Figure 3.7:Simulink block diagram of contact forces of left leg with ground

After calculating the different forces, the Simulink model of the ground is made with the help of different Simulink blocks. The detailed parameter of the ground block, rigid transform and humanoid parameters are given in Appendix B. Figure 3.8 shows the Simulink model of the lower body of humanoid robot with ground and contact forces. The static friction and kinematic friction between the foot and the ground has been considered for a regular walk and sudden stable turn.

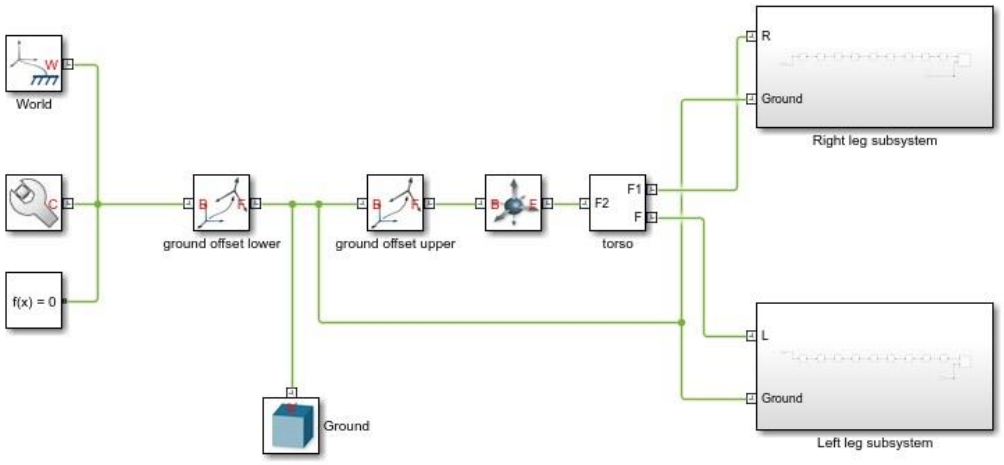


Figure 3.8:Simulink block diagram of lower body of humanoid robot with ground

Figure 3.9 shows the Simulink model of the lower body of humanoid robot generated by the mechanics explorer of the MATLAB.

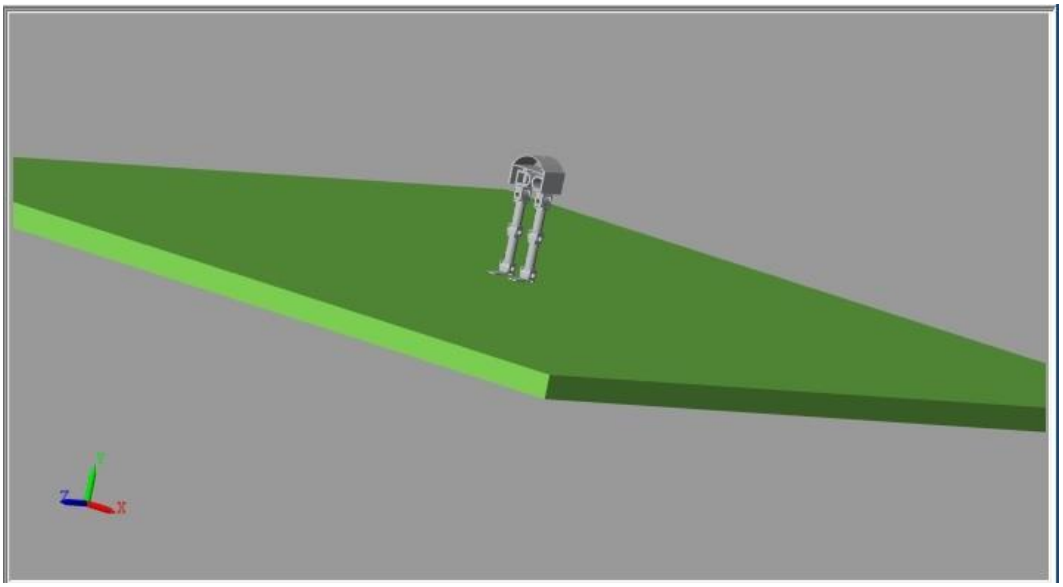


Figure 3.9:Simulink model of lower body of humanoid robot with ground

CHAPTER 4 MATHEMATICAL MODELING

The set of mathematical expression of the humanoid robot describe the joint torque values and joint position with respect to the time. The kinematic configuration of the proposed model has ten degrees of freedom. Each leg has five degree of freedom. Nearby the torso of the humanoid robot, three degrees of freedom are provided for the fast movement and sudden turn. Knee has one degree of freedom and ankle has one degree of freedom. The design principle of the developing the humanoid robot is similar to the human body. In the proposed model, it is assumed that the workspace created by the right leg is similar to the workspace created by the left leg. The upper body weight is acting on the both foot and balancing of the body will be supported by the both feet. Humanoid body is symmetric about the sagittal plane. Based on this concept, the mathematical expression is calculated for the joint torque, joint position of the right leg is equal to the left joint position and torque.

4.1 Forward kinematics

Kinematic model describes the spatial position of joints and links, also the position and orientation of the end effector. The derivatives of the kinematic model deals with the mechanics of motion without considering the forces that cause it. Kinematic model is the analytical description of the spatial geometry of motion of the manipulator with respect to a fixed reference frame, as a function to time. The definition of a manipulator with four joint –link parameters for each link and a systematic procedure for assigning right-handed orthonormal coordinate frames, one to each link in an open kinematic chain, was proposed by Denavit-Hartenberg notation[114].

Denavit-Hartenberg method is used to calculate the workspace of right leg. Each leg can be modelled as a kinematic chain with five link connected by five revolute joints. All the joints and links are identified and labeled from 1 to 5 respectively. The complete frame assignment is shown in figure4.1 from which all the joint-link parameters are determined and tabulated in table 4.1. Applying the algorithm of Denavit-Hartenberg step by step, the transformation matrix of each joint can be calculated[115].

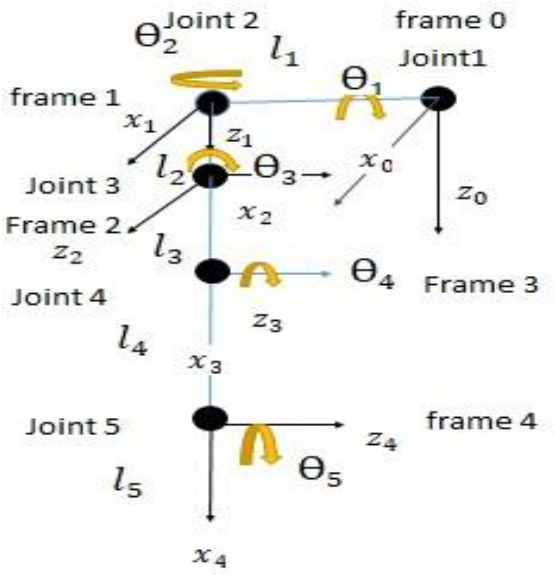


Figure 4.1:Link frame assignment of lower body of right leg

Table 4.1:DH parameters for right leg

DH Parameters		**Link**			
	1	**2**	**3**	**4**	**5**
$\boldsymbol{\theta_i}$	$\boldsymbol{\theta_1}$	$\boldsymbol{\theta_2 + 90}$	$\boldsymbol{\theta_3 - 90}$	$\boldsymbol{\theta_4}$	$\boldsymbol{\theta_5}$
$\boldsymbol{d_i}$	$\boldsymbol{l_1}$	$\boldsymbol{l_2}$	0	0	0
$\boldsymbol{a_i}$	0	0	$\boldsymbol{l_3}$	$\boldsymbol{l_4}$	$\boldsymbol{l_5}$
$\boldsymbol{\alpha_i}$	90	-90	90	0	0

Now, the individual transformation matrices that specify the relationship between adjacent links as below**;**

$${}^{0}_{1}T = Rot(z,\theta_1) * Trans(0,0,l_1) * Trans(0,0,0) * Rot(x,90) \quad (4.1)$$

$${}^{1}_{2}T = Rot(z,\theta_2 + 90) * Trans(0,0,l_2) * Trans(0,0,0) * Rot(x,-90) \quad (4.2)$$

$${}^{2}_{3}T = Rot(z,\theta_3 - 90) * Trans(0,0,0) * Trans(0,0,l_3) * Rot(x,90) \quad (4.3)$$

$${}^{3}_{4}T = Rot(z,\theta_4) * Trans(0,0,0) * Trans(0,0,l_4) * Rot(x,0) \quad (4.4)$$

$${}^{4}_{5}T = Rot(z,\theta_5) * Trans(0,0,0) * Trans(0,0,l_5) * Rot(x,0) \quad (4.5)$$

Where ${}^{i-1}_{i}T$ is a general link transformation matrix relating the *i*-th coordinate frame to the(*i-1)th* coordinate frame. After tabulating the value of individual transformation matrices are as follows

$${}^{0}_{1}T = A_1 = \begin{bmatrix} C\Theta_1 & 0 & S\Theta_1 & 0 \\ S\Theta_1 & 0 & -C\Theta_1 & 0 \\ 0 & 0 & 0 & l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^{1}_{2}T = A_2 = \begin{bmatrix} -S\Theta_2 & 0 & -C\Theta_2 & 0 \\ C\Theta_2 & 0 & -S\Theta_2 & 0 \\ 0 & -1 & 0 & l_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^{2}_{3}T = A_3 = \begin{bmatrix} S\Theta_3 & 0 & -C\Theta_3 & l_3 * S\Theta_3 \\ -C\Theta_3 & 0 & -S\Theta_3 & -l_3 * C\Theta_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^{3}_{4}T = A_4 = \begin{bmatrix} C\Theta_4 & -S\Theta_4 & 0 & l_4 * C\Theta_4 \\ S\Theta_4 & C\Theta_4 & 0 & l_4 * \Theta_4 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$ {}^{4}_{5}T = A_5 = \begin{bmatrix} C\Theta_5 & -S\Theta_5 & 0 & l_5 * C\Theta_5 \\ S\Theta_5 & C\Theta_5 & 0 & l_5 * S\Theta_5 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

The workspace created by the right foot obtained as follows;

$$ {}^{0}_{5}T = {}^{0}_{1}T * {}^{1}_{2}T * {}^{2}_{3}T * {}^{3}_{4}T * {}^{4}_{5}T * = \begin{bmatrix} r_{11} & r_{12} & r_{13} & \boldsymbol{r_{14}} \\ r_{21} & r_{22} & r_{23} & \boldsymbol{r_{24}} \\ r_{31} & \boldsymbol{r_{32}} & \boldsymbol{r_{33}} & \boldsymbol{r_{34}} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.6) $$

Where

$$ r_{11} = C\Theta_5(C\Theta_4(C\Theta_3\, S\Theta_1 - C\Theta_1 S\Theta_2\, S\Theta_3) - C\Theta_1\, C\Theta_2\, S\Theta_4) - $$

$$ S\Theta_5(S\Theta_4(C\Theta_3\, S\Theta_1 - C\Theta_1 S\Theta_2\, S\Theta_3) + C\Theta_1\, C\Theta_2\, S\Theta_4) $$

$$ r_{12} = -C\Theta_5(S\Theta_4(C\Theta_3\, S\Theta_1 - C\Theta_1 S\Theta_2\, S\Theta_3) + C\Theta_1\, C\Theta_2\, S\Theta_4) - $$

$$ S\Theta_5(C\Theta_4(C\Theta_3\, S\Theta_1 - C\Theta_1 S\Theta_2\, S\Theta_3) - C\Theta_1\, C\Theta_2\, S\Theta_4) $$

$$ r_{13} = S\Theta_1\, S\Theta_3 + C\Theta_1 C\Theta_3\, S\Theta_2 $$

$$ \boldsymbol{r_{14}} = l_2 S\Theta_1 + l_4 C\Theta_4(C\Theta_3 S\Theta_1 - C\Theta_1 S\Theta_2 S\Theta_3) + l_3 C\Theta_3 S\Theta_1 + $$

$$ l_5 C\Theta_5(C\Theta_4(C\Theta_1 S\Theta_1 - C\Theta_1 S\Theta_2 S\Theta_3) - C\Theta_1 C\Theta_2 S\Theta_4) - $$

$$ -l_5 S\Theta_5(S\Theta_4(C\Theta_3 S\Theta_1 - C\Theta_1 S\Theta_2 S\Theta_3) + C\Theta_1 C\Theta_2 C\Theta_4) - l_4 C\Theta_1 C\Theta_2 S\Theta_4 - $$

$$ l_3 C\Theta_1 S\Theta_2 S\Theta_3 $$

$$ r_{21} = S\Theta_5(S\Theta_4(C\Theta_1\, C\Theta_3 + S\Theta_1 S\Theta_2\, S\Theta_3) - C\Theta_2\, C\Theta_4\, S\Theta_1) - $$

$$ C\Theta_5(C\Theta_4(C\Theta_1\, C\Theta_3 + S\Theta_1 S\Theta_2\, S\Theta_3) + C\Theta_2\, S\Theta_1\, S\Theta_4) $$

$$ r_{22} = C\Theta_5(S\Theta_4(C\Theta_1\, C\Theta_3 + S\Theta_1 S\Theta_2\, S\Theta_3) - C\Theta_2\, C\Theta_4\, S\Theta_1) + $$

$$ S\Theta_5(C\Theta_4(C\Theta_1\, C\Theta_3 + S\Theta_1 S\Theta_2\, S\Theta_3) + C\Theta_2\, S\Theta_1\, S\Theta_4) $$

$$ r_{23} = C\Theta_1 S\Theta_1\, S\Theta_2 - C\Theta_1\, S\Theta_3 $$

$$\boldsymbol{r_{24}}= \quad l_5 S\Theta_5(S\Theta_4(C\Theta_1 C\Theta_3 + S\Theta_1 S\Theta_2 S\Theta_3) - C\Theta_2 C\Theta_4 S\Theta_1) -$$

$$l_4 C\Theta_4(C\Theta_1 C\Theta_3 + S\Theta_1 S\Theta_2 S\Theta_3) - l_3 C\Theta_1 C\Theta_3 - l_5\, C\Theta_5(C\Theta_4(C\Theta_1\, C\Theta_3 +$$

$$S\Theta_1 S\Theta_2\, S\Theta_3) + C\Theta_2\, S\Theta_1\, S\Theta_4) - l_2 C\Theta_1 - l_4 C\Theta_2 S\Theta_1\, S\Theta_4 - l_3 S\Theta_1 S\Theta_2\, S\Theta_3$$

$$r_{31} = -C\Theta_5(S\Theta_2\, S\Theta_4 - C\Theta_2 C\Theta_4\, S\Theta_3) - S\Theta_5(C\Theta_4\, S\Theta_2 + C\Theta_2 S\Theta_3\, S\Theta_4)$$

$$r_{32} = S\Theta_5(S\Theta_2\, S\Theta_4 - C\Theta_2 C\Theta_4\, S\Theta_3) - C\Theta_5(C\Theta_4\, S\Theta_2 + C\Theta_2 S\Theta_3\, S\Theta_4)$$

$$r_{33} = -C\Theta_2\, C\Theta_3$$

$$r_{34} = l_1 - l_5 C\Theta_5(S\Theta_2 S\Theta_4 - C\Theta_2 C\Theta_4 S\Theta_3) - l_5 S\Theta_5(C\Theta_4 S\Theta_2 + C\Theta_2 S\Theta_3 S\Theta_4) +$$

$$l_3 C\Theta_2 S\Theta_3 - l_4 S\Theta_2 S\Theta_4 + l_4 C\Theta_2 C\Theta_4\, S\Theta_3$$

4.2 Inverse kinematics

A humanoid robot control requires knowledge of the toe position and orientation for the instantaneous location of each joint as well as knowledge of the joint displacement required to place the foot at a new location. Closed form solution, joint displacement is determined as explicit functions of the position and orientation of the toe. The closed form in the present context means a solution method based on analytical algebraic or kinematic approach. In the present model five unknown joints parameter are to be determined[116]. The overall transformation matrix ${}^{0}_{5}T$ and end effector foot matrix T_{foot} represent the same transformations.

$$T_{foot} = \begin{bmatrix} n_x & o_x & a_x & \boldsymbol{p_x} \\ n_y & o_y & a_y & \boldsymbol{p_y} \\ n_z & \boldsymbol{n_z} & \boldsymbol{a_z} & \boldsymbol{p_z} \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^{0}_{5}T \qquad (4.7)$$

Where n, o, a is the unit vector along the normal, sliding and approach. The inverse kinematics o the system helps to take the static gait and dynamic gait during walking and the running of humanoid robot. The step size of the humanoid robot depends on the distance between the object and the robot.

Applying the closed form solution and analytical algebraic approach resulted in no solution. To obtain the solution for the joint displacement.

It is possible to view the description of tool frame {5}, with respect to frame {1}, the foot end point frame, along two different paths.

Path1 : frame{0}------frame{4}--------frame{5}

Along this path the transformation ${}^{0}_{5}T$ can be obtained as

$$ {}^{0}_{5}T = {}^{0}_{1}T * {}^{3}_{4}T * {}^{4}_{5}T = \begin{bmatrix} n_x & o_x & a_x & \boldsymbol{p_x} \\ n_y & o_y & a_y & \boldsymbol{p_y} \\ n_z & \boldsymbol{n_z} & \boldsymbol{a_z} & \boldsymbol{p_z} \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (4.8) $$

Since Θ_1, Θ_4 and Θ_5 are unknown. The inverse of matrix ${}^{4}_{5}T$ can be computed. Substituting these values and post multiplying with the matrix ${}^{0}_{5}T$ can be computed.

$$ {}^{0}_{5}T * {}^{4}_{5}T^{-1} = {}^{0}_{1}T * {}^{3}_{4}T \qquad (4.9) $$

$$ \begin{bmatrix} n_x * C\Theta_5 - o_x * S\Theta_5 & o_x * C\Theta_5 + n_x * S\Theta_5 & a_x & \boldsymbol{p_x} - l_5 * n_x \\ n_y * C\Theta_5 - o_y * S\Theta_5 & o_y * C\Theta_5 + n_y * S\Theta_5 & a_y & \boldsymbol{p_y} - l_5 * n_y \\ S\Theta_4 & C\Theta_4 & 0 & \boldsymbol{p_z} - l_5 * n_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = $$

$$ \begin{bmatrix} C\Theta_1 * C\Theta_4 & -C\Theta_1 * S\Theta_4 & S\Theta_1 & l_4 * C\Theta_1 * C\Theta_4 \\ C\Theta_4 * S\Theta_4 & -S\Theta_1 * S\Theta_4 & -C\Theta_1 & l_4 * C\Theta_4 * S\Theta_1 \\ S\Theta_4 & C\Theta_4 & 0 & l_1 + l_4 * S\Theta_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ \Theta_1 = \tan^{-1} \frac{(\boldsymbol{p_y} - l_5 * n_y)}{(\boldsymbol{p_x} - l_5 * n_x)} $$

$$\Theta_4 = \tan^{-1} \frac{(\boldsymbol{p_z} - l_5 * n_z - l_1)}{\left(\pm\sqrt{((\boldsymbol{p_x} - l_5 * n_x)^2)} + \left((\boldsymbol{p_y} - l_5 * n_y)^2\right)\right)}$$

Since Θ_5 is unknown. The inverse of matrix ${}^{3}_{4}T$ can be computed. Substituting these values and post multiplying with the matrix ${}^{0}_{5}T * {}^{4}_{5}T^{-1}$ can be computed.

$${}^{0}_{5}T * {}^{4}_{5}T^{-1} * {}^{3}_{4}T^{-1} = {}^{0}_{1}T \qquad (4.10)$$

$$\begin{bmatrix} n_x * C\Theta_{45} - o_x * S\Theta_{45} & o_x * C\Theta_{45} + n_x * S\Theta_{45} & a_x & \boldsymbol{p_x} - l_5 n_x - l_4 n_x C\Theta_5 + l_4 o_x S\Theta_5 \\ n_y * C\Theta_{45} - o_y * S\Theta_{45} & o_y * C\Theta_{45} + n_y * S\Theta_{45} & a_y & \boldsymbol{p_y} - l_5 n_y - l_4 n_y C\Theta_5 + l_4 o_y S\Theta_5 \\ n_z * C\Theta_{45} - o_z * S\Theta_{45} & o_z * C\Theta_{45} + n_z * S\Theta_{45} & a_z & \boldsymbol{p_z} - l_5 n_z - l_4 n_z C\Theta_5 + l_4 o_z S\Theta_5 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} C\Theta_1 & 0 & S\Theta_1 & 0 \\ S\Theta_1 & 0 & -C\Theta_1 & 0 \\ 0 & 0 & 0 & l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\Theta_5 = \tan^{-1} \frac{\{(\boldsymbol{p_y} + l_5 n_y) l_4 n_x - (\boldsymbol{p_x} + l_5 n_x) l_4 n_y\}}{\{(\boldsymbol{p_x} + l_5 n_x) l_4 o_y + (\boldsymbol{p_y} + l_5 n_y) l_4 o_x\}}$$

Path2: frame{0}------frame{1}--------frame{2}-------frame{3}-----Frame{4}------frame{5}

$${}^{0}_{5}T = {}^{0}_{1}T * {}^{1}_{2}T * {}^{2}_{3}T * {}^{3}_{4}T * {}^{4}_{5}T = \begin{bmatrix} n_x & o_x & a_x & \boldsymbol{p_x} \\ n_y & o_y & a_y & \boldsymbol{p_y} \\ n_z & \boldsymbol{n_z} & \boldsymbol{a_z} & \boldsymbol{p_z} \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (4.11)$$

Since Θ_2 and Θ_3 are unknown. The inverse of matrix ${}^{4}_{5}T$ and ${}^{3}_{4}T$ can be computed. Substituting these values and post multiplying with the matrix ${}^{0}_{5}T$ can be computed.

$${}^{0}_{5}T * {}^{4}_{5}T^{-1} * {}^{3}_{4}T^{-1} = {}^{0}_{1}T * {}^{1}_{2}T * {}^{2}_{3}T \qquad (4.12)$$

$$\begin{bmatrix} n_x * C\Theta_{45} - o_x * S\Theta_{45} & o_x * C\Theta_{45} + n_x * S\Theta_{45} & a_x & \boldsymbol{p_x} - l_5 n_x - l_4 n_x C\Theta_5 + l_4 o_x S\Theta_5 \\ n_y * C\Theta_{45} - o_y * S\Theta_{45} & o_y * C\Theta_{45} + n_y * S\Theta_{45} & a_y & \boldsymbol{p_y} - l_5 n_y - l_4 n_y C\Theta_5 + l_4 o_y S\Theta_5 \\ n_z * C\Theta_{45} - o_z * S\Theta_{45} & o_z * C\Theta_{45} + n_z * S\Theta_{45} & a_z & \boldsymbol{p_z} - l_5 n_z - l_4 n_z C\Theta_5 + l_4 o_z S\Theta_5 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} C\Theta_3 S\Theta_1 - C\Theta_1 S\Theta_2 S\Theta_3 & -C\Theta_1 C\Theta_2 & S\Theta_1 S\Theta_3 + C\Theta_1 C\Theta_3 S\Theta_2 & l_2 S\Theta_1 + l_3 C\Theta_3 S\Theta_1 - l_3 C\Theta_1 S\Theta_2 S\Theta_3 \\ -C\Theta_1 C\Theta_3 - S\Theta_1 S\Theta_2 S\Theta_3 & -C\Theta_2 S\Theta_1 & C\Theta_3 S\Theta_1 S\Theta_2 - C\Theta_1 S\Theta_3 & -l_2 C\Theta_1 - l_3 C\Theta_1 C\Theta_3 - l_3 S\Theta_1 S\Theta_2 S\Theta_3 \\ C\Theta_2 S\Theta_3 & -S\Theta_2 & -C\Theta_2 C\Theta_3 & l_1 + l_3 C\Theta_2 S\Theta_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Values of Θ_2 and Θ_3 are obtained as follows.

$$\Theta_2 = \tan^{-1} \frac{\left(-(o_z * C\Theta_{45} + n_z * S\Theta_{45})\right)}{\left(\pm\sqrt{(o_x * C\Theta_{45} + n_x * S\Theta_{45})^2 + \left((o_y * C\Theta_{45} + n_y * S\Theta_{45})^2\right)}\right)}$$

$$\Theta_3 = \tan^{-1} \frac{(n_z * C\Theta_{45} - o_z * S\Theta_{45})}{(-a_z)}$$

4.3Jacobian

The transformation from joint velocities to the end effector velocity is described by a matrix, called Jacobian. The jacobian matrix, which is dependent on manipulator configuration is a linear mapping from velocities in joint space to velocities in Cartesian space. The Jacobin is one of the most important tools for characterization of differential motions of the manipulator[117] .The mapping between differential changes is linear and can be expressed as

$$Ve(t) = J(q)\dot{q} \qquad (4.13)$$

Where *Ve(t)*= Cartesian velocity vector

J(q)=6X1 manipulator jacobian or jacobian matrix

$\dot{q}$= nX1 vector of n joints

Eqn(12) can be written in column vectors of the jacobian , that is,

$$Ve(t) = [J1(q)\ J2(q) J3(q) J4(q) J5(q) J6(q)] \qquad (4.14)$$

$$Ve(t) = \begin{bmatrix} v \\ w \end{bmatrix} = \begin{bmatrix} \dot{d} \\ \dot{\Theta} \end{bmatrix} = J(q)\,\dot{q} \qquad (4.15)$$

The rotary Jacobian can be written as follows

$$J_i(q)= \begin{bmatrix} P_{i-1}X\ {}^{i-1}_{n}P \\ P_{i-1} \end{bmatrix} \quad (4.16)$$

For computing J_i , vectors P_{i-1} and ${}^{i-1}_{n}P$ are needed.

The origin of frame{n} at the end effector is $O_n= [0 \quad 0 \quad 0 \quad 1]^T$. This applies for origin of any frame i.e. for any value of n. The above vector equation can be written as

$${}^{i-1}_{n}P = {}^{O}_{n}TO_n - \quad {}^{O}_{i-1}TO_n \quad (4.17)$$

Each column of jacobian matrix is computed separately and all the column are combined to from the total jacobian matrix.

$$J_1(q)= \begin{bmatrix} P_0X\ {}^{0}_{5}P \\ P_0 \end{bmatrix} \ , \ J_2(q)= \begin{bmatrix} P_1X\ {}^{1}_{5}P \\ P_1 \end{bmatrix}, \ J_3(q)= \begin{bmatrix} P_2X\ {}^{2}_{5}P \\ P_2 \end{bmatrix}$$

$$J_4(q)= \begin{bmatrix} P_3X\ {}^{3}_{5}P \\ P_3 \end{bmatrix} , \ J_5(q)= \begin{bmatrix} P_4X\ {}^{4}_{5}P \\ P_4 \end{bmatrix}$$

The calculation for the individual joint jacobian is shown in Appendix C

The jacobian for the manipulator for right leg is as follows

$$J= [J_1 \quad J_2 \quad J_3 \quad J_4 \quad J_5] \quad (4.18)$$

4.4 Singularity Posture

At a singular configuration, the humanoid robot loses one or more degree of freedom. The singular configuration is classified into two categories based on the location of the leg in the workspace. The boundary singularities occur when the leg is on the boundary of the workspace i.e. the leg is either fully stretched out or fully retracted. The interior singularities occur when the leg is inside the reachable workspace of the humanoid robot. They can occur anywhere in the reachable workspace. Other than the boundary singularities and configurations singularities, the actuation singularities effect the motion of the humanoid robot[118].

The actuation singularities does not allow the humanoid robot to take higher step, so that the balancing of the body can be maintain throughout the movement.

4.4.1 Computation of Singularities

The computation of internal singularities can be carried out by analyzing the rank of the jacobian matrix. The jacobian matrix loses its rank and becomes ill conditioned at values of joint variables, that is

$$|J| = 0 \qquad (4.19)$$

In the present model, the manipulators have five degree of freedom, it is possible to simplify the problem of singularity computation by dividing the problem into two separate problems;

- Computation of singularities resulting from motion of first, fourth and fifth joint are called leg singularities
- Computation of singularities resulting from the motion of second and third joints are called torso

This is achieved by partitioning the jacobian matrix into 3X3 submatrices and 2X2 submatrices.

$$J_{legsingularity} = \begin{bmatrix} J_{11} & J_{12} & J_{13} \\ J_{41} & J_{42} & J_{43} \\ J_{51} & J_{52} & J_{53} \end{bmatrix} \qquad (4.20)$$

$$J_{torsosingularity} = \begin{bmatrix} J_{21} & J_{22} \\ J_{31} & J_{32} \end{bmatrix} \qquad (4.21)$$

The advantage of singularity posture of the joint provides the safety to the system. If any complicated case arise, in that condition the structures of the system comes to the locking state and provides safety the system. The singularity posture avoided by the controller and do not aloe the link to achieve particular point.

The singularity point can be computed by taking the determinant of the equation and substituting it to zero.

$$det(J_{legsingularity}) = 0 \qquad (4.22)$$

$$0.000353S\theta_2S\theta_5 - 0.000839S\theta_2S\theta_4 + 0.000353C\theta_3S\theta_5 + 0.000821C\theta_4S\theta_2S\theta_5 + 0.000839C\theta_5C\theta_5S\theta_5S\theta_2S\theta_4 + 0.000839C\theta_2C\theta_4S\theta_3 - 0.000839C\theta_2C\theta_4C\theta_5C\theta_5S\theta_3 + 0.000839C\theta_4C\theta_5S\theta_2S\theta_5 + 0.000821C\theta_2S\theta_3S\theta_4S\theta_5 + 0.000833C\theta_2C\theta_5S\theta_3S\theta_4S\theta_5 = 0$$

$$\det(J_{torsosingularity}) = 0 \qquad (4.23)$$

$$0.000353S\theta_2S\theta_5 - 0.000839S\theta_2S\theta_4 + 0.000353C\theta_3S\theta_2S\theta_5 + 0.000821C\theta_4S\theta_2S\theta_5 + 0.000839C\theta_5C\theta_5S\theta_2S\theta_4 + 0.000839C\theta_2C\theta_4S\theta_3 - 0.000839C\theta_2C\theta_4C\theta_5C\theta_5S\theta_3 + 0.000839C\theta_4C\theta_5S\theta_2S\theta_5 + 0.000821C\theta_2S\theta_3S\theta_4S\theta_5 + 0.000833C\theta_2C\theta_5S\theta_3S\theta_4S\theta_5 = 0$$

Since the determinant values of these two expressions depends upon coupled values of joint position. A practical approach has been established to calculate the singular posture of the body. The Multibody proposed model helps to compute the singularity posture of the body to avoid the collision of two legs.

4.5 Static forces in humanoid robot joint

Assuming that the self-weight of the body $\boldsymbol{F}$ is acting at the torso of frame {0} and is given by

$$F = [f_x \quad f_y \quad f_z \quad 0 \quad 0 \quad 0]^T \qquad (4.24)$$

The joint torques are a function of the configuration and applied forces and are given by equation as

$$\tau_{static} = J(q)^T \boldsymbol{F} \qquad (4.25)$$

The computation of static torque shown in Appendix D.

The stall torque on each joint can be obtained as

$$\tau_{static} = [\tau_{1static} \quad \tau_{2static} \quad \tau_{3static} \quad \tau_{4static} \quad \tau_{5static}]^T \qquad (4.26)$$

Humanoid body is dynamically balance. For making the body dynamically balance the forces are applied by the joint.

This static torques are capable to handle the humanoid upper body weight and lower body weight in vertically upward direction. So that the zero-moment point is not disturbing, and body is in equilibrium condition.

The values of the stall torque are obtained for the self-weight of the body.

Table 4.2shows the stall torque required on the several joints for the initial joint position of joint.

Table 4.2:Stall joint torque for the initial position of lower body

Joint		Stall Torque (in Newton-meter)
Torso	Pitch	0
	Yaw	0
	Roll	-0.13748
Knee	Pitch	0
Ankle	Pitch	0

4.6 Humanoid robot equation of motion

The Lagrange –Euler formulation is a systematic procedure for obtaining the dynamic model of an n-DOF manipulator. The n-DOF open kinematic chain serial link manipulator has **n** joint position or displacement variables $q=[q_1,q_2,q_3,\ldots\ldots q_n]^T$. The L.E formulation established the relation between the joint positions, velocities ,accelerations , and the generalized torque applied to the manipulator[119]. Since the power to weight ratio in electrical actuator is very high. The actuator weight is not considered in the proposed system.

The Lagragian , L=K-P is given by

$$L = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=i}^{i}\sum_{k}^{i} Tr\left[\left({}_{j-1}^{0}T\ Q_j\ {}_{i}^{j-1}T\right) I_i \left({}_{k-1}^{0}T\ Q_k\ {}_{i}^{k-1}T^T\right)\right]\dot{q}_j\dot{q}_k + \sum_{i=1}^{n} m_i g\, {}_{i}^{0}T\, {}_{i}^{i}\bar{r}$$

(4.27)

Where L is the Lagrangian , K is the kinetic energy of the system , and P is the potential energy of the system . The Lagrangian can now be differentiated in order to from the dynamic equation of motion.

According to the Lagrangian- Euler dynamics formulation, the generalized torque τ_i of the actuator at joint i,to drive link i of the manipulator is given by

$$\tau_i = \frac{d}{dt}\left(\frac{\delta L}{\delta \dot{q}_i}\right) - \frac{\delta L}{\delta q_i} \qquad (4.28)$$

By substituting L and carrying out the differentiation, the generalized torque τ_i applied to link i to n-DOF manipulator is obtained.

$$\tau_i = \sum_{j=1}^{n} M_{ij}(q)\ddot{q}_j + \sum_{j=1}^{n}\sum_{k=1}^{n} h_{ijk}\,\dot{q}_j\dot{q}_k + G_i \text{ for i=1,2,……n} \qquad (4.29)$$

where

$$M_{ij} = \sum_{p=\max(i,j)}^{n} Tr\left[\ d_{pj}\ \ I_p\, {}_{pi}^{T}d\ \right]$$

$$h_{ijk} = \sum_{p=\max(i,j,k)}^{n} Tr\left[\frac{\partial (d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]$$

$$G_i = -\sum_{p=i}^{n} m_p\ g d_{pi}\ {}_{p}^{p}\bar{r}_p$$

and

$$d_{ij} = \begin{cases} {}_{j-1}^{0}T\ Q_j\ {}_{i}^{j-1}T & for\ j \le i \\ 0 & for\ \ j > i \end{cases}$$

and

$$\frac{\partial d_{ij}}{\partial q_k} = \begin{cases} {}^{O}_{j-1}TQ_j\,{}^{j-1}_{k-1}TQ_k\,{}^{k-1}_{i}T & for\ i \geq k \geq j \\ {}^{O}_{k-1}TQ_k\,{}^{k-1}_{j-1}T\,Q_j\,{}^{j-1}_{i}T & for\ i \geq j > k \\ 0 & for\ i < j\ or\ \ i < k \end{cases}$$

Equation is the dynamic model of the manipulator and gives a set of n nonlinear, coupled, second order ordinary differential equations for n-links of the n-DOF manipulator. The L.E formulation begins with the determination of kinematic model. Already we have the kinematic model for the humanoid robot. Her all the joints are revolute joint. Value of Q_j are obtained by the partial differentiation of the homogeneous transformation matrix.

So value of $Q_j = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$ *(4.30)*

Since all the joints are revolute, Hence

$$Q_1 = Q_2 = Q_3 = Q_4 = Q_5 = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

The inertia tensors I_1 , I_2 I_3, I_4 and I_5 for link of length l_1, l_2 l_3 l_4 l_5 with masses $m_1, m_2, m_3, m_4 and\ m_5$ at the centroid of the link and respect to frame {i}, i=1,2,3,4and 5 are computed with the help of Solid work software.

I_1 =

$$\begin{bmatrix} -0.0003276 & -0.0000484 & -0.000000000003 & -0.00459 \\ -0.0000484 & 0.0000614 & 0.00000000002 & 0.00152 \\ -0.000000000003 & 0.00000000002 & 0.0000436 & 0 \\ -0.00459 & 0.00152 & 0 & 0.142 \end{bmatrix}$$

$$I_2 = \begin{bmatrix} 0.000643 & -0.000355 & -0.000078 & -0.0053 \\ -0.000355 & 0.0002031 & 0.0000439 & 00.00292 \\ -0.000078 & 0.0000439 & 0.000013 & 0.000647 \\ -0.0053 & 0.00292 & 0.000647 & 0.04319 \end{bmatrix}$$

$$I_3 = \begin{bmatrix} 0.0000032 & -0.0000107 & -0.0000192 & 0.000258 \\ -0.0000107 & 0.0000525 & 0.0000806 & -0.00108 \\ -0.0000192 & 0.0000806 & 0.000147 & -0.00195 \\ 0.000258 & -0.00108 & -0.00195 & 0.026 \end{bmatrix}$$

$$I_4 = \begin{bmatrix} 0.0000130 & 0.000044 & 0.000044 & 0.00084 \\ 0.000044 & 0.000219 & 0.000186 & 0.00353 \\ 0.000044 & 0.0001860 & 0.000189 & 0.00355 \\ 0.00084 & 0.00353 & 0.00355 & 0.06733 \end{bmatrix}$$

$$I_5 = \begin{bmatrix} 0.0000149 & 0.000054 & 0.0000431 & 0.00091 \\ 0.0000548 & 0.000275 & 0.000188 & 0.00399 \\ 0.0000431 & 0.000188 & 0.000150 & 0.00312 \\ 0.00091 & 0.00399 & 0.00312 & 0.066 \end{bmatrix}$$

Table 4.3 and 4.4 shows the mass of the individual link and the centroid of link.

Table 4.3:Mass of the individual link

Link		Mass (in kg)
Torso	Pitch Link	0.142
	Yaw Link	0.04319
	Roll Link	0.026
Knee	Pitch Link	0.06733
Ankle	Pitch Link	0.066

Table 4.4:Centroid of the individual link

Link		Centroid (in meter)
Torso	Pitch Link	$[-0.032 \quad 0.01 \quad 0]$
	Yaw Link	$[-0.121 \quad 0.067 \quad 0.015]$
	Roll Link	$[0.00982 \quad -0.041 \quad -0.074]$
Knee	Pitch Link	$[0.012 \quad 0.052 \quad 0.052]$
Ankle	Pitch Link	$[0.013 \quad 0.060 \quad 0.047]$

The first step required the computation of matrices d_{ij}, which are required to compute all other coefficients. The computation of d_{ij} is shown in Appendix E.

Next step is applied to compute the elements of inertia matrix M_{ij} using the equation i.e

$$M_{ij} = \sum_{p=\max(i,j)}^{n} Tr\left[d_{pj} \quad I_p \, {}_{pi}^{T} d \right] \qquad (4.31)$$

The calculation of inertia matrix shown in appendix F

$$M_{ij} = \begin{bmatrix} M_{11} & M_{12} & M_{13} & M_{14} & M_{15} \\ M_{21} & M_{22} & M_{23} & M_{24} & M_{25} \\ M_{31} & M_{32} & M_{33} & M_{34} & M_{35} \\ M_{41} & M_{42} & M_{43} & M_{44} & M_{45} \\ M_{51} & M_{52} & M_{53} & M_{54} & M_{55} \end{bmatrix}$$

Next step, the Coriolis and centrifugal force coefficients, h_{ijk} for i,j,k=1,2,3,4,5are obtained from equation, i.e.

$$h_{ijk} = \sum_{p=\max(i,j,k)}^{n} Tr\left[\frac{\partial (d_{pk})}{\partial q_p} \; I_p \, {}_{pi}^{T} d \right] \qquad (4.32)$$

The calculation for Coriolis and centrifugal force coefficients shown in appendix G.

The Coriolis and centrifugal coefficient matrix H are therefore

$$H = \begin{bmatrix} H_1 \\ H_2 \\ H_3 \\ H_4 \\ H_5 \end{bmatrix} \qquad (4.33)$$

Next step is applied to compute the gravity loading at the joints and the computation of gravity matrix shown in the Appendix H

$G_i = -\sum_{p=i}^{n} m_p \, g d_{pi} \, {}^{p}_{p}\bar{r}_p$

Thus, the gravity coefficient matrix G is therefore

$$G = \begin{bmatrix} G_1 \\ G_2 \\ G_3 \\ G_4 \\ G_5 \end{bmatrix}$$

The complete dynamic model is obtained in the final step, by substituting the above results. The humanoid equation of motion in the matrix from are

$$\begin{bmatrix} \Gamma_1 \\ \Gamma_2 \\ \Gamma_3 \\ \Gamma_4 \\ \Gamma_5 \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} & M_{13} & M_{14} & M_{15} \\ M_{21} & M_{22} & M_{23} & M_{24} & M_{25} \\ M_{31} & M_{32} & M_{33} & M_{34} & M_{35} \\ M_{41} & M_{42} & M_{43} & M_{44} & M_{45} \\ M_{51} & M_{52} & M_{53} & M_{54} & M_{55} \end{bmatrix} * \begin{bmatrix} \ddot{\Theta}_1 \\ \ddot{\Theta}_2 \\ \ddot{\Theta}_3 \\ \ddot{\Theta}_4 \\ \ddot{\Theta}_5 \end{bmatrix} + \begin{bmatrix} H_1 \\ H_2 \\ H_3 \\ H_4 \\ H_5 \end{bmatrix} + \begin{bmatrix} G_1 \\ G_2 \\ G_3 \\ G_4 \\ G_5 \end{bmatrix} \qquad (4.34)$$

4.7 Trajectory of Humanoid robot

The basics for the trajectory is the Biomechanics. Biomechanics is the field of science that applies the laws of mechanics and physics to human performance. Biomechanics suggest that a humanoid should be form closed polygon when it is in motion. The best design for mechanical stability is closed loop polygon. Trajectory is the sequence of movement of individual joint.

Torque depends on the trajectory of joint. The torque value depends on the sine and cosine term. Hence the torque coming on the individual joint varies with respect to the time. So system torque varies from maximum to minimum over a time span[120]. Since the humanoid is walks freely, so trajectory of individual joint is assumed as follows.

It is essential to obtain the desired joint space trajectory given by the Cartesian space trajectory. Inverse kinematic deciding the joint values to move the torso from one position to another position. Afterwards controller is sending the repeated signal to the humanoid robot. Consider the first joint of humanoid robot. Table 2 gives the joint range values. Considering the initial velocity and final velocity of joint is zero as a boundary condition. Assuming the initial and final acceleration of the joint is also zero as boundary condition. Considering the fifth order polynomial to decide the trajectory.

For joint 1

Θ_1=$a_0 + a_1 t + a_2 t\text{^}2 + a_3\, t\text{^}3 + a_4 t\text{^}4 + a_5 t\text{^}5$

$\dot{\Theta}_1$= $a_1 + 2a_2 t + 3a_3\, t\text{^}2 + 4a_4 t\text{^}3 + 5a_5 t\text{^}4$

$\ddot{\Theta}_1$=$2a_2 + 6a_3 t + 12a_4\, t\text{^}2 + 20a_5 t\text{^}4$

Applying boundary condition

t_i=0, $\Theta_i = -45^o$; $t_f = tf$, $\Theta_f = 45^o$

t_i=0, $\dot{\Theta}_i = 0$; $t_f = tf$, $\dot{\Theta}_f$ =0

t_i=0, $\ddot{\Theta}_i = 0$; $t_f = tf$, $\ddot{\Theta}_f$ =0

Computing the coefficient, the joint trajectory is obtained as follows

Θ_1=$-45^o + 7.2\, t^3 - 2.16\, t^4 + .1728 t^5$ *(4.35)*

For joint 2

Θ_2=$b_0 + b_1 t + b_2 t\text{^}2 + b_3\, t\text{^}3 + b_4 t\text{^}4 + b_5 t\text{^}5$

$\dot{\Theta}_2 = b_1 + 2b_2t + 3b_3\,t\text{^}2 + 4b_4t\text{^}3 + 5b_5t\text{^}4$

$\ddot{\Theta}_2 = 2b_2 + 6b_3t + 12b_4\,t\text{^}2 + 20b_5t\text{^}4$

Applying boundary condition

t_i=0, $\Theta_i = -15^o$; $t_f = tf$, $\Theta_f = 100^o$

t_i=0, $\dot{\Theta}_i = 0$; $t_f = tf$, $\dot{\Theta}_f$ =0

t_i=0, $\ddot{\Theta}_i = 0$; $t_f = tf$, $\ddot{\Theta}_f$ =0

Computing the coefficient the joint trajectory is obtained as follows

$\Theta_2 = -15^o + 9.2t^3 - 2.76t^4 + .2208t^5$ *(4.36)*

For joint 3

$\Theta_3 = c_0 + c_1t + c_2t\text{^}2 + c_3\,t\text{^}3 + c_4t\text{^}4 + c_5t\text{^}5$

$\dot{\Theta}_3 = c_1 + 2c_2t + 3c_3\,t\text{^}2 + 4c_4t\text{^}3 + 5c_5t\text{^}4$

$\ddot{\Theta}_3 = 2c_2 + 6c_3t + 12c_4\,t\text{^}2 + 20c_5t\text{^}4$

Applying boundary condition

t_i=0, $\Theta_i = 0^o$; $t_f = tf$, $\Theta_f = 45^o$

t_i=0, $\dot{\Theta}_i = 0$; $t_f = tf$, $\dot{\Theta}_f$ =0

t_i=0, $\ddot{\Theta}_i = 0$; $t_f = tf$, $\ddot{\Theta}_f$ =0

Computing the coefficient, the joint trajectory is obtained as follows

$\Theta_3 = 3.6t^3 - 1.08t^4 + .0864t^5$ *(4.37)*

For joint 4

$\Theta_4 = d_0 + d_1t + d_2t\text{^}2 + d_3\,t\text{^}3 + d_4t\text{^}4 + d_5t\text{^}5$

$\dot{\Theta}_4 = d_1 + 2d_2t + 3d_3\,t\text{^}2 + 4d_4t\text{^}3 + 5d_5t\text{^}4$

$\ddot{\Theta}_4$=$2d_2 + 6d_3t + 12d_4\, t\text{^}2+20d_5t\text{^}4$

Applying boundary condition

t_i=0, $\Theta_i = 0^o$; $t_f = tf$, $\Theta_f = 120^o$

t_i=0, $\dot{\Theta}_i = 0$; $t_f = tf$, $\dot{\Theta}_f$ =0

t_i=0, $\ddot{\Theta}_i = 0$; $t_f = tf$, $\ddot{\Theta}_f$ =0

Computing the coefficient the joint trajectory is obtained as follows

$\Theta_4 = 9.6t^3 - 2.88t^4 + .2304t^5$ *(4.38)*

For joint 5

Θ_5=$e_0 + e_1t + e_2t\text{^}2 + e_3\, t\text{^}3+e_4t\text{^}4 + e_5t\text{^}5$

$\dot{\Theta}_5$= $e_1 + 2e_2t + 3e_3\, t\text{^}2+4e_4t\text{^}3 + 5e_5t\text{^}4$

$\ddot{\Theta}_5$=$2e_2 + 6e_3t + 12e_4\, t\text{^}2+20e_5t\text{^}4$

Applying boundary condition

t_i=0, $\Theta_i = -20^o$; $t_f = tf$, $\Theta_f = 40^o$

t_i=0, $\dot{\Theta}_i = 0$; $t_f = tf$, $\dot{\Theta}_f$ =0

t_i=0, $\ddot{\Theta}_i = 0$; $t_f = tf$, $\ddot{\Theta}_f$ =0

Computing the coefficient, the joint trajectory is obtained as follows

Θ_5=$-20 + 4.8t^3 - 1.44t^4 + .1152t^5$ *(4.39)*

The movement of the leg is biorhythmic process. The biorhythmic process is called as central pattern generator. The sequence of the trajectory generated with the help of oscillator.

4.8. Gait Design for humanoid robot:

In the proposed model the base of an industrial robot is fixed. In actual real situations, the base is not fixed for humanoid robot. The humanoid robot has to move in difficult conditions and need to plan a motion of humanoid robot to maintain contact between sole and the ground[121]. In standing condition, the weight of the body acts vertically downwards and reaction forces are vertically upwards. While there is no horizontal component of forces. The zero moment does not disturb, and humanoid maintains the static balance.

4.8.1 Walking

While walking in straight direction, the hip joint, knee joint and ankle joints are coming into the action. For a successful walk, the zero-moment point lies between the closed polygon formed by the leg and the torso. Based on the biomechanics and the real time senor data of the leg movement, the position of the joint with respect to time is obtained.Table4.5 shows the joint position of humanoid robot with respect to the time.

Table 4.5:Joint position with respect to gait time for straight walking

Gait time (in seconds)	Ankle joint position (in degree)	Knee joint position (in degree)	Hip joint position (in degree)
0	-7.5	10	-10
0.1333	10	-5	-7.5
0.2667	10	2.5	-15
0.4000	5	-10	10
0.5333	0	-10	15
0.6667	-10	15	10
0.8000	-7.5	10	-10

4.8.2 Turning

Two types of turning conditions exist in the humanoid robot. First condition consists of sudden turn during walk and the second type is of simple turn.

While taking turn, along with the hip joint, knee joint and ankle joints, the yaw joints are coming into motion momentarily. In sudden turn, the humanoid walking is going on with predefined values of joint, but the controller is sending the yaw joint values momentarily. Afterward the yaw joint values remain same, till the other joint of straight walking coming to the last position of the gait time.

Table4.6 shows the joint position of humanoid robot with respect to the time for the sudden turn.

Table 4.6:Joint position with respect to gait time for sudden turn

Gait time (in seconds)	Ankle joint position (in degree)	Knee joint position (in degree)	Hip joint position (in degree)	Yaw joint Position(in degree)
0	-7.5	10	-10	0
0.1333	10	-5	-7.5	5
0.2667	10	2.5	-15	5
0.4000	5	-10	10	5
0.5333	0	-10	15	5
0.6667	-10	15	10	5
0.8000	-7.5	10	-10	0

4.9 Simulink Model

Since all the joints of the lower body of humanoid robot are active kind. Joint torque is developed by the DC motor. While walking or turning, torque is required on the different joint to move from one position to another position. The estimated torque based on the mathematical expression obtained from the governing equation. The multi in multi out system (MIMO) of control system designed for the calculating the estimated torque. The input parameter for the control system is joint position, joint velocity and joint acceleration.

Figure 4.2 shows the Simulink block diagram of the estimated torque for the different joint. The simulation is carried out for the 5 seconds of planned trajectory of maximum joint values. The maximum value of the torque is computed for the selection of DC motor. The details of the Simulink model are described in the Appendix I.

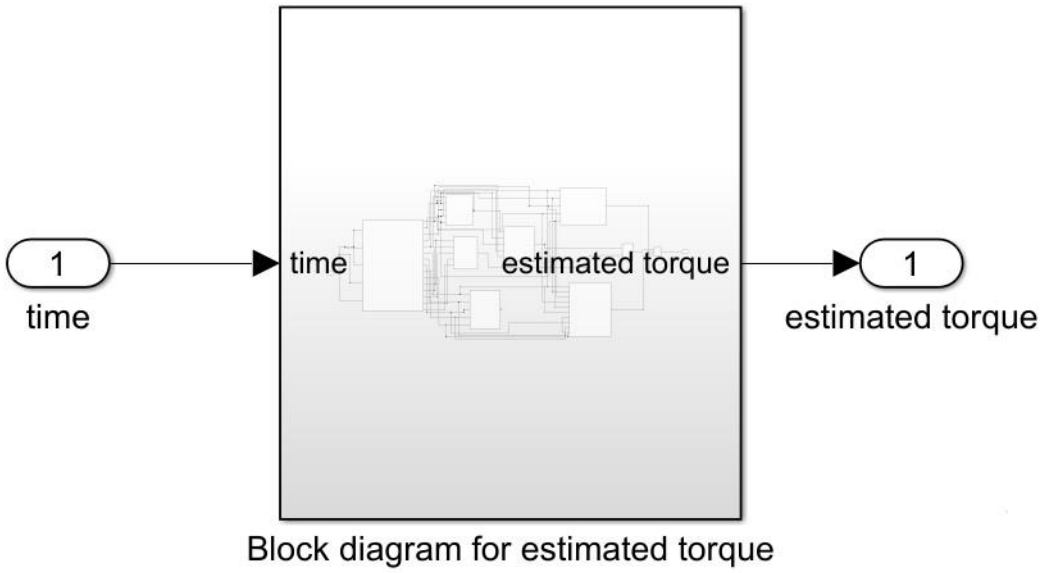


Figure 4.2:Simulink model for torque estimation

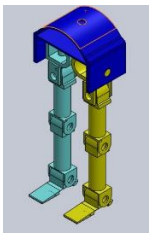
The maximum values of the torque obtained for the different joints after carrying the simulation. Table 4.7shows the maximum values of joint torque for planned trajectory.

Table 4.7:Joint torque values

Joint		Maximum Torque (in Newton-meter)
Torso	Pitch	0.0112
	Yaw	0.0299
	Roll	0.2756
Knee	Pitch	0.0540
Ankle	Pitch	0.0413

CHAPTER 5 CONTROL ARCHITECTURE

The control requires the knowledge of the mathematical model and some sort of intelligence. Whereas the required mathematical modeling is obtained from basic physical laws governing robot dynamics and associated devices. A robot performs the specified tasks in its environment, which can be divided into two classes: contact type task and non-contact type tasks. The non-contact type tasks involved manipulation of the end –effector in space to do desired work. While in contact type tasks the end effector interacts with the environment. The foot of the lower body of humanoid robot comes in contact with the ground and exerts the force on the ground. The ground force i.e the combination of friction force and normal reaction force is applied on the body of humanoid robot. Friction force is responsible to cause the motion in forward direction and the body is able to move on the ground. Individual joint of a manipulator is powered and driven by actuators that apply a force or a torque to cause motion of the links. The actuator commands that moves the manipulator and achieve the specified end effector motion, are provided by the manipulator control system. These commands are based on the control set points generated from the set of joint torque time histories obtained by the trajectory planner[122]. The actual joint and/or end effector positions and their derivative can be fed back to the control system to get accurate motion. The schematic diagram of a manipulator control system is shown in figure5.1. The dotted lines for the feedback indicate that the control system may be employ feedback of the actual joint locations and velocities. The parameter q , $\dot{q}$ and $\ddot{q}$ and τ and so on.



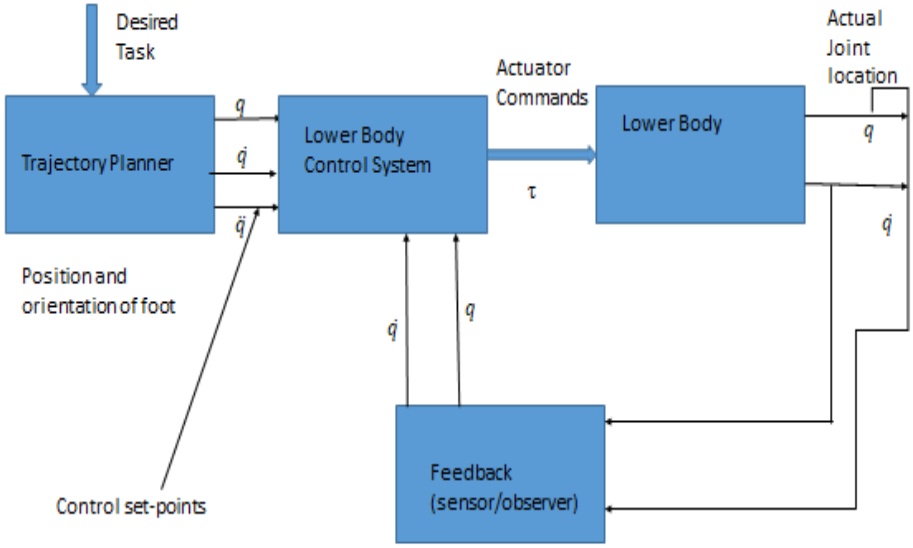


Figure 5.1:Block diagram of lower body control system

The dynamic control of a manipulator motions and /or interaction forces requires knowledge of forces or torque that must be exerted on the manipulator joints to move the kinks and the end effector from the present location to the desired location, with or without the constraints of a particular planned end –effector trajectory, and or planned end –effector force/torque.

In both the situations of the desired end –effector location and the desired end effector force, the control of the individual joint's location is important. Hence it is an obvious requirement that each joint is controlled by a position servo. If the lower body of robot is to move very slowly or to move one joint at a time, then the control is simple because coupled dynamics forces are negligible.

Each joint can be controlled independently by using a simple control system, which produces a joint actuator forces/torque proportional to the required change in the

joint variable. This is the proportional control algorithm. If the motion is fast, which are must for effective robot applications, all joints must move simultaneously. In this situation, the coupled dynamics forces are significant, and it is known that dynamics is nonlinear and complex. Joints cannot move independently, and complex control algorithm will be required. The typical robot control architecture for a n-DOF manipulator consists of a master control system to control and synchronize n-joints. This master control is responsible for sending "set point" information to command to each of the joint controllers. The n-joints controllers use the set point information to command the joint actuator to move the joint. The joint controller may employ feedback to the master controller. The schematic of typical control architecture for a manipulator control system is illustrated in figure 5.2.

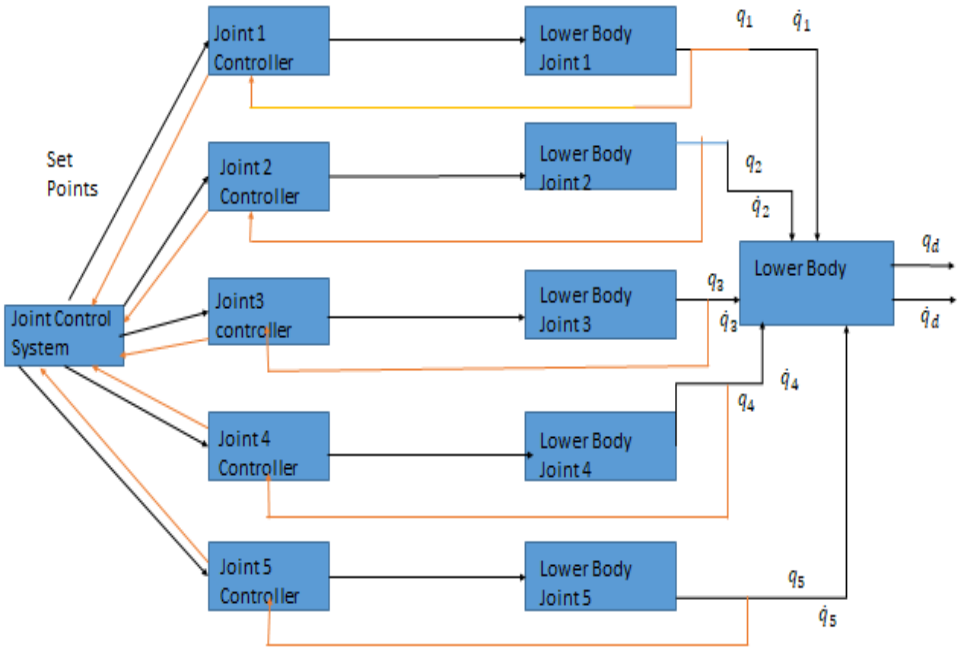


Figure 5.2:Lower body control architecture for 5-Dof manipulator

5.1Manipulator Control problem

The task to be performed by a manipulator may require either the execution of specific motions of the end effector in free space and/or the application of specific contact forces at the end effector, if the end effector is constrained by the environment. For a given manipulator, many techniques can be used to control it to perform in the desired manner. The performance and range of application of a manipulator is significantly influenced by the control scheme followed, as well the way it is implemented. The mechanical design and configuration design also greatly influence the control technique utilized and the performance of the manipulator. Depending on the control law employed to compute the joint torques, the control system is classified as either linear or nonlinear[123]. The design of the linear controllers based on the approximate linear model of the manipulator will be examined.

The effects of time varying inertia, interaction torques and nonlinear torques, of the dynamics model are ignored to simplify the control system. Servo control strategies such as proportional derivatives (PD) are considered. Finally, the method of computed torque control, which is based on a nonlinear control law will be considered.

5.2Effect of damping factor to the system.

The control of a system consists of varying the inputs so that the closed loop dynamics of the systems has the desired properties; say the output y follows the desired trajectory y_d. Feedback control explicitly uses the error signal , e= $y - y_d$, the difference between the measured output y and the desired output y_d, as a part of control law , so as to reduce the error and system sensitivity to the parameters in dynamic model. The block diagram of this system with unity negative feedback is drawn in figure5.3.

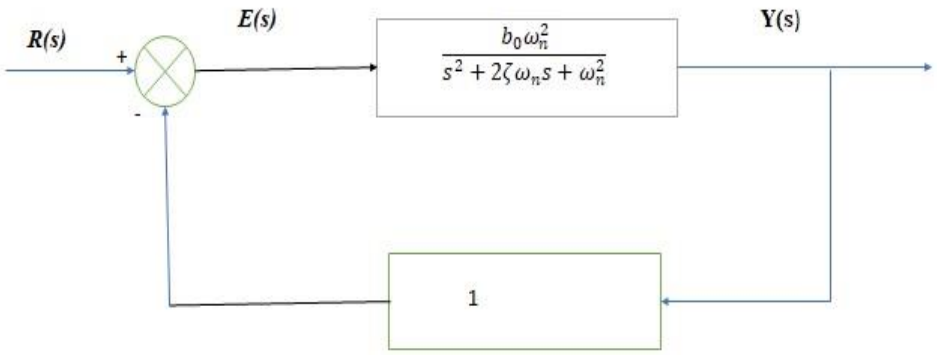


Figure 5.3:Block diagram of a second order linear system with unity feedback

If $R(s)=Y_d(s)$ is the Laplace transform of set -point input, the closed -loop transfer function of the control system is

$$\frac{Y(s)}{R(s)} = \frac{G(s)}{1+G(s)} = \frac{b_0\omega_n^2}{s^2+2\zeta\omega_n s+(1+b_0)\omega_n^2} \qquad (5.1)$$

The second order linear characteristics equation of the closed loop system, is written in standard form as

$$s^2 + 2\boldsymbol{\xi\omega_n s} + \boldsymbol{\omega_n^2}=\mathbf{0} \qquad (5.2)$$

where ξ and ω_n is the damping factor and natural frequency of the system.

The roots of the characteristics equation, $\boldsymbol{r_1, r_2}$ which characterize the system's time response to a given input, are

$$\boldsymbol{r_1, r_2} = -\boldsymbol{\xi\omega_n} \pm \boldsymbol{\omega_n}\sqrt{\boldsymbol{\xi^2 - 1}} \qquad (5.3)$$

These roots determine the nature of time response of the system. If roots are real, then the response of the system is sluggish and non-oscillatory. If roots are complex conjugate, then the response of the system is oscillatory. The system response analysis helps in fixing the controller parameters that ensure closest possible trajectory tracking performance of a joint[124]. Depending on the values of the damping ratio $\boldsymbol{\xi}$ **,** these roots can be real and unequal or real and equal or complex conjugate.

These three classes of the roots characterize the three types of time response of the systems. These three classes of roots arise with the value of damping ratio greater than unity, equal to unity and less than unity .

Case(i) Damping ratio greater than unity

A damping ratio greater than unity ($\boldsymbol{\xi > 1}$) gives the roots $\boldsymbol{r_1}$ and $\boldsymbol{r_2}$ as real and unequal. The roots $\boldsymbol{r_1}$ and $\boldsymbol{r_2}$ are

$$r_{1=} -\boldsymbol{\xi\omega_n} + \boldsymbol{\omega_n}\sqrt{\boldsymbol{\xi^2 - 1}}$$

$$r_{2=} -\boldsymbol{\xi\omega_n} - \boldsymbol{\omega_n}\sqrt{\boldsymbol{\xi^2 - 1}}$$

The time response is obtained by taking the inverse Laplace transform of equation, which gives

$$y(t) = 1 - \frac{e^{-\xi\omega_n t}}{2\sqrt{\xi^2-1}}\left[\left(\sqrt{\boldsymbol{\xi^2 - 1}} - \boldsymbol{\xi}\right)e^{-\boldsymbol{\omega_n}\sqrt{(\xi^2-1)}\boldsymbol{t}} + \left(\sqrt{\boldsymbol{\xi^2 - 1}} + \boldsymbol{\xi}\right)e^{\boldsymbol{\omega_n}\sqrt{(\xi^2-1)}\boldsymbol{t}}\right] \quad (5.4)$$

In this case, the system has a sluggish and nonconciliatory behavior and system is overdamped.

Case(ii) Damping ration equal to unity

For a damping ratio equal to unity ($\boldsymbol{\xi = 1}$)**,** both $\boldsymbol{r_1}$ and $\boldsymbol{r_2}$ **are** real and equal**.** The roots $\boldsymbol{r_1}$ and $\boldsymbol{r_2}$ are

$$r_{1=} -\boldsymbol{\omega_n}$$

$$r_{2=} -\boldsymbol{\omega_n}$$

and the time response of the system is

$$y(t) = 1 - [1 - e^{-\boldsymbol{\omega_n t}} - \boldsymbol{\omega_n t} e^{-\boldsymbol{\omega_n t}}] \quad (5.5)$$

Under this condition, the system exhibits the fastest possible non oscillatory, overshoot free response and it is critically damped.

Case(iii) Damping ratio less than unity

A damping ratio greater than unity ($\boldsymbol{\xi < 1}$) gives the roots $\boldsymbol{r_1}$ and $\boldsymbol{r_2}$ are complex conjugates of each other.

The time response is obtained by taking the inverse Laplace transform of equation, which gives

$$y(t) = 1 - \frac{e^{-\xi\omega_n t}}{2\sqrt{\xi^2-1}} sin\left(\boldsymbol{\omega_n\sqrt{(1-\xi^2)}t + \tan^{-1}\frac{\sqrt{(1-\xi^2)}}{\xi}}\right) \qquad (5.6)$$

In this case, the system has an oscillatory behavior and is underdamped.

From the above analysis, it is observed that the linear second order system is stable for all values of damping ratio, although the behavior may be oscillatory or non-oscillatory.

A humanoid control system cannot allow to have an oscillatory response for obvious reasons. For instance, in a pick and place operation, an oscillating end effector may strike against the object before picking it to manipulate. Hence highest possible speed of response and yet non oscillatory response dictates that the controller design parameters shall be chosen to have the damping ratio $\boldsymbol{\xi}$ **=1** or at least close to it but less than unity.

5.3 Model of a manipulator joint.

A common drive system for many industrial robots consists of an electric motor as the actuator and provide the torque to the joint. The schematic of an actuator gear link assembly of a rotary joint shown in figure5.4.

The gear ratio 'η' is defined as

η = number of teeth on the gear of link shaft/number of teeth on the gear of actuator shaft $=\frac{\eta_2}{\eta_1}$ *(5.7)*

The gear ratio η is greater than 1 for a reduction gear. The gear ratio causes a reduction of the actuator speed $\dot{\theta}_a$ to $\dot{\theta}$, the speed of the link and increase of the actuator torque τ_a to τ, the torque applied to the link (the load) as given by

$$\tau = \eta\tau_a \qquad (5.8)$$

$$\dot{\theta} = \frac{1}{\eta}\dot{\theta}_a \qquad (5.9)$$

$$\theta = \frac{1}{\eta}\theta_a \qquad (5.10)$$

Where θ_a is the angular displacement of the actuator shaft and θis the joint angular displacement.

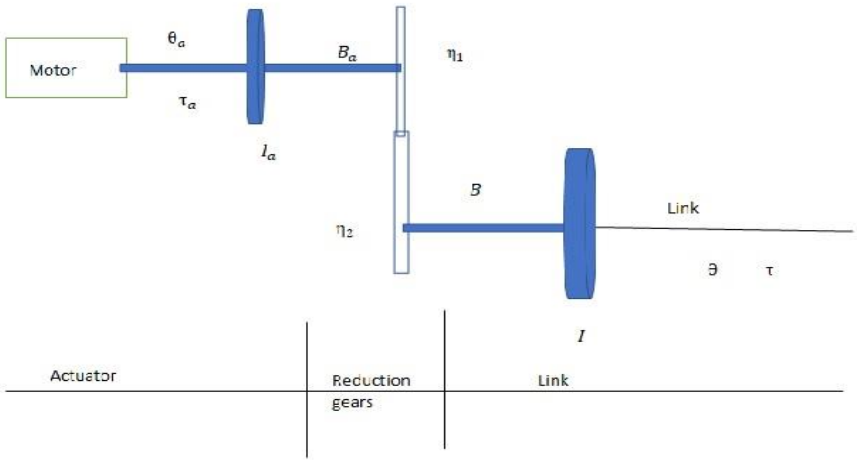


Figure 5.4:Actuator-gear-link mechanism

The link torque τ , when reflected to the actuator side, is scaled down by a factor η, the gear ratio. If I_a is the inertia of rotor of the actuator shaft and B_a is the viscous friction coefficient at the actuator bearings, for the rotational mass-damper system, the torque required at the actuator shaft is given by

$$\tau_a = I_a\ddot{\theta}_a + B_a\dot{\theta}_a + \frac{\tau}{\eta} \qquad (5.11)$$

$$\tau_a = I_a\ddot{\theta}_a + B_a\dot{\theta}_a + \frac{(I\ \ddot{\theta}+B\dot{\theta})}{\eta} \qquad (5.12)$$

Where I is the inertia of the load and B is the viscous friction coefficient for load. The *eqn(49)* can be written in terms of the actuator shaft variable as.

$$\tau_a = (I_a + \frac{I}{\eta^2})\ddot{\theta}_a + (B_a + \frac{B}{\eta^2})\dot{\theta}_a \qquad (5.13)$$

Or in terms of link variables as

$$\tau = (I + \eta^2 I_a)\ddot{\theta} + (B + \eta^2 B_a)\,\dot{\theta} \qquad (5.14)$$

The link torque τ is obtained from the dynamic model of the joint.The dynamic equation of the joint i is rewritten as below

$$\tau = \sum_{j=1}^{n} M_{ij}\,(q)\ddot{q}_j + \sum_{j=1}^{n}\sum_{k=1}^{n} h_{ijk}\,\dot{q}_j\dot{q}_k + G_i \text{ for i=1,2,……n} \qquad (5.15)$$

In above equation, gravity loading G is significant in gravity environment only and comes into effect whenever motion is against gravity. The equation does not include the effect of friction, backlash, and elastic deformation. Backlash is difficult to model and elastic deformation and static friction are highly nonlinear and complex to model. The viscous friction is directly proportional to the velocity and can be included in the model. Hence the equation is modified as below to include the contribution of viscous friction.

$$\tau = \sum_j M_j\ddot{\theta}_j + \sum_j\sum_k h_{jk}\dot{\theta}_j\dot{\theta}_k + B\dot{\theta} + G \qquad (5.16)$$

Where B is the viscous friction coefficient at the link bearings.

The term in equation are split into linear decoupled and nonlinear interacting terms.

The two linear decoupled terms are $M\ddot{\theta}$ from the first summation term for link i and the third term $B\dot{\theta}$. The remaining terms in equation are nonlinear interacting term and grouped together and denoted as '*d'* i.e

$$d = \sum_{j\neq i, j=1}^{n} M_j\ddot{\theta}_j + \sum_j\sum_k h_{jk}\dot{\theta}_j\dot{\theta}_k + G \qquad (5.17)$$

Thus the equation can be written as

$$\tau = M\ddot{\theta} + B\dot{\theta} + d \qquad (5.18)$$

Substituting τ from equation and rearranging, the actuator torque is obtained as

$$(I_a + \frac{M}{\eta^2})\ddot{\theta}_a + (B_a + \frac{B}{\eta^2})\dot{\theta}_a + \frac{d}{\eta} \qquad (5.19)$$

In the case of highly geared manipulators ($\eta \gg 1$), it is quite reasonable to ignore the contribution of the nonlinear interacting terms d and still guarantee a good trajectory tracking performance by the controller.

Neglecting the nonlinear terms, the actuator torque reduces to the simple form

$$\tau_a = (I_{eff})\ddot{\theta}_a + (B_{eff})\dot{\theta}_a \qquad (5.20)$$

Where I_{eff} is the effective inertia and B_{eff} is the effective damping with

$$I_{eff} = (I_a + \frac{M}{\eta^2})$$

and $B_{eff} = (B_a + \frac{B}{\eta^2})$

Because the configuration dependent terms M is reduced by a factor η^2, with $\eta \gg 1$), I_{eff} is almost constant for highly geared manipulators.

5.4 Model of a D C motor

The schematic diagram of an armature controlled permanent magnet DC motor is shown in figure5.5. Motors with field excitation provided by permanent magnets are preferred for robotics applications. Such motors are labelled permanent magnet DC motors.

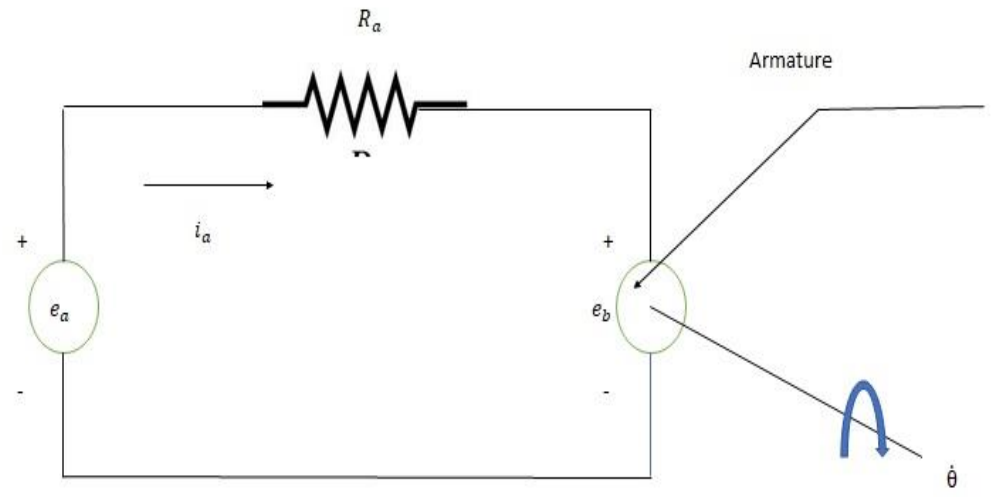


Figure 5.5:Permanent magnet DC motor

With references to the schematic diagram, the control signal to the DC motor is applied at the armature terminals in the form of armature voltagee_a. The motor torque τ_a is related to the armature current i_a as

$$\tau_a = K_T i_a \qquad (5.21)$$

where K_T is the motor torque constant.The armature current directly controls the torque generated by the actuator. The back emf e_b induced in the armature winding is given by

$$e_b = K_b \dot{\Theta}_a \qquad (5.22)$$

where K_b is the back emf constant. Applying the Kirchoff's voltage law for the circuit of figure gives

$$e_a = e_b + i_a R_a \qquad (5.23)$$

Where $e_a$is the emf across armature and R_a is the armature resistance. Substituting e_b from equation gives

$$e_a = K_b \dot{\Theta}_a + i_a R_a \qquad (5.24)$$

Thus, the armature voltage is adjusted, depending on the commands generated by the manipulator control system, which specify the torque required from the actuator. The continuous adjustment of the armature voltage e_a is carried out by the motor driver circuitry so that the desired current i_a flows through the armature windings. The schematic of a motor driver circuitry is shown in figure 5. 6.

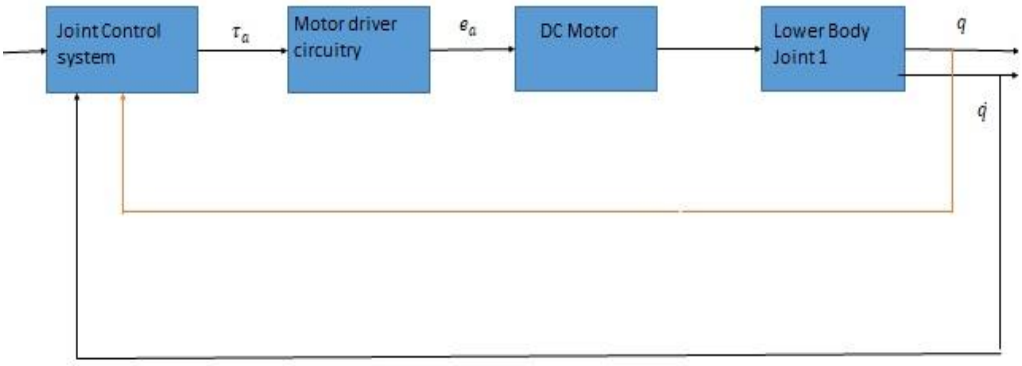


Figure 5.6:control of a manipulator joint driven by a DC motor

5.5 Partitioned PD Control scheme

A linear controller based on a partitioned proportional derivative control strategy is slightly different from a simple PD controller. It is useful for systems that are more complex. The controller is partitioned into two parts; a model-based portion and a servo portion, such that the joint parameters (I_{eff} and B_{eff}) appear only in the model-based portion. The approximate model of the manipulator joint developed in previous section is used to implement this controller. The block diagram of this controller is shown in figure5.7.

The portion to the right of the partitioned line is physical system, the model-based portion and to the left of the partitioned line is the servo based control system, usually implemented in a computer or microcontroller.

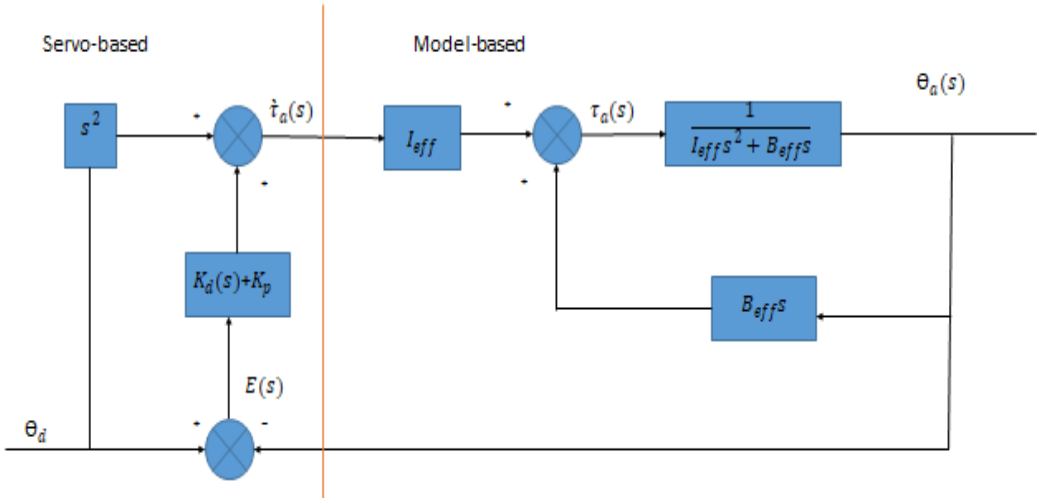


Figure 5.7:partitioned PD controller for trajectory control

The model-based portion of the control law computes the command torque based on the system parameter $I_{eff}$and B_{eff}. The Laplace of the approximate model of the joint is

$$\tau_a(s) = (I_{eff}s^2 + B_{eff}s)\,\theta_a(s) \qquad (5.25)$$

The model-based control law defines the actuator toque using a structure similar to that of equation, i.e

$$\tau_a(s) = I_{eff}\grave{\tau}_a(s) + B_{eff}s\,\theta_a(s) \qquad (5.26)$$

where $\grave{\tau}_a(s)$ is the torque value specified by the servo portion to control a unit inertia

From eqn*(62)* and *eqn(63)*

$$\grave{\tau}_a(s) = s^2\theta_a(s) \qquad (5.27)$$

This is the equation of motion for a system with unit inertia. Thus, the model based law effectively reduced the system to that of unit inertia system. In other words, the physical system appears to be a unit-inertia system to the servo portion.

The servo portion of the controller is based on the error in actuator position and its derivative. these servo errors are computed as

$$E(s) = \Theta_d(s) - \Theta_a(s) \qquad (5.28)$$

and $sE(s) = s\Theta_d(s) - s\Theta_a(s)$ *(5.29)*

where $\Theta_d(s)$is the desired joint displacement of the actuator shaft. The servo portion of the control law makes use of the feedback to modify the system behavior. Because the model –based portion of the control law design becomes very simple. The controller is simply required to control a system of unit inertia with no friction and stiffness. Assuming that the sensors are capable of detecting position and velocity, a proportional derivative control scheme can be used to control the system of mass /inertia. The control system using the PD control scheme with position and velocity feedback for the control of unit inertia is shown in figure 5.8.

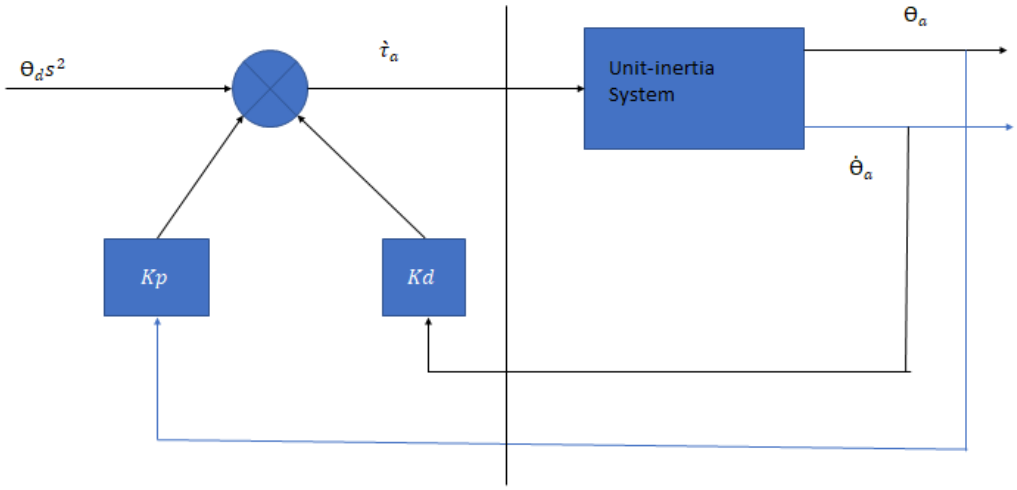


Figure 5.8:simplified PPD controller for control of unit-inertia system

The control law for the servo portion is specified with two controller gain parameters, the proportional constant K_p, and the derivative constants K_d as

$$\grave{\tau}_a(s) = s^2\Theta_d(s) + K_d sE(s) + K_p E(s) \qquad (5.30)$$

Combining equation *(5.28)-(5.30)* , the control law becomes

$$s^2\Theta_a(s) = s^2\Theta_d(s) + K_d sE(s) + K_p E(s) \qquad (5.31)$$

The error dynamics of the PPD controller using equation, is given by

$$s^2E(s)+ K_d sE(s) + K_pE(s)=0 \qquad (5.32)$$

*Eqn(69)*is identical to the characteristic equation of the second order linear system. Thus, the partitioned PD controller reduces the system to that of a linear second – order systems. Comparing equation with equation, gives the natural frequency of the system as;

$$\omega_n = \sqrt{K_p} \qquad (5.33)$$

and the damping ratio as

$$\boldsymbol{\xi}=\frac{K_d}{\mathbf{2}\sqrt{K_p}} \qquad (5.34)$$

From equation, it is clear that any desired second –order performance of the control system is possible by the choice of proper control gains, $K_p$and K_d

The controller for the manipulator should be critically damped, i.e $\boldsymbol{\xi = 1.}$

Hence for critically damped performance

$$K_d = 2\sqrt{K_p}=2\omega_n \qquad (5.35)$$

5.6 Setting of gain Parameter

The equation specifies the methodology for setting the methodology for setting the control gains K_p and K_d.A direct consequence of control law, equation , is that the servo error at any instant of time is zero provided there is no initial error and the computation time for the computer is zero, i.e the actuator torque is computed as a continuous function of time.

In reality, the time taken to compute the servo error, the PD law control gains and to command a new value of torque, is nonzero and is known as the cycle time. This the resulting command torque $\grave{\tau}_a$ is a staircase function and the servo error are non-zero at the beginning of each cycle.

The controller will reduce this nonzero servo error to zero during each cycle. Hence the actual trajectory tracked will be close to, but not exactly the same as desired trajectory. Apart from a damping ratio of unity, another factor that constraints the selection of control gains in the flexibility of links, which are assumed to be rigid bodies in the development of the joint model. The unmolded structural flexibility of the link and other mechanical elements produces resonance at frequencies other than natural frequency. Because these structural flexibilities have been ignored, the controller must be designed so as not to excite these unmolded resonances.

The lowest unmolded resonance, which corresponds to the maximum value of the effective inertia seen by the actuator,I_{max} has a resonance frequency

$$\omega_{res} = \omega_o \sqrt{\frac{I_o}{I_{max}}} \qquad (5.36)$$

where ω_o is the structural frequency when the effective inertia is I_o.

To prevent exciting these structural oscillations and also ensure structural stability, the controller natural frequency ω_n must be limited to $0.5\omega_{res}$.i,e

$$\omega_n \leq 0.5\omega_{res} \text{ and } K_p \leq (0.5\omega_o)^2 \frac{I_o}{I_{max}} \qquad (5.37)$$

Based on these parameters the control gain K_p and K_d are computed .

Table5.1 shows the approximate gain values for the different joint controller.

Table 5.1:Tuining parameter of the joint controller

Joint Controller(PD)	K_p	K_d
Joint Controller1	65	32
Joint Controller2	68	30
Joint Controller3	72	34
Joint Controller4	65	38
Joint Controller5	70	34

5.7 Computed torque control

Each joint of a manipulator was viewed as an individual system to be controlled for the simplified and linear controllers based on the PD and PID control scheme were examined for the simplified SISO model. More realistic problem is the control of an n-DOF manipulator as a single system. A control scheme that makes direct use of the complete dynamics model of the manipulator to cancel the effect of gravity, Coriolis and centrifugal force, friction and the manipulators inertia tensor[125].

The dynamic model of the n-DOF manipulator based on the rigid body dynamics gives the nx1 vector of joint torques.

$$\tau = M(q)\ddot{q} + H(q,\dot{q}) + \dot{G}(q) \qquad (5.38)$$

where $M(q)$is the nxn inertia matrix, $H(q,\dot{q})$ is the nx1 vector of centrifugal and Coriolis torques, $G(q)$ is the nx1 vectors of gravity torques and the $q(t)$, $q(\dot{t})$,and $q(\ddot{t})$ is the joint positions, velocities and accelerations respectively. Recall that q is the generalized joint variables.

Thus the Multi-Input –Multi output (MIMO), nonlinear dynamics model of n-DOF manipulator becomes

$$\tau = M(q)\ddot{q} + H(q,\dot{q}) + \dot{G}(q) \qquad (5.39)$$

The design of a nonlinear control system based on the complicated model given by the equation, is considered now.

Because this controller is based on a more accurate dynamic model of manipulator, it provides better trajectory performances than linear controllers do. The controller discussed here employs the computed torque control law to modify the system effectively decouple and linearize it. The computed torque control scheme also comprise two portions- a model based and a servo portion. The model-based portion defines the nx1 vector of control torques τ using a structure.

$$\tau = M(q)\grave{\tau} + H(q,\dot{q}) + \dot{G}(q) \qquad (5.40)$$

where $\grave{\tau}$ is the nx1 torque vector specified by the servo portion.

$$\grave{\tau} = \ddot{q} \qquad (5.41)$$

Thus, the model based portion effectively linearizes as well as decouples the system dynamics by employing a nonlinear feedback of the actual positions and velocities of joints. The schematic representation of this nonlinear control scheme is shown in figure5.9.

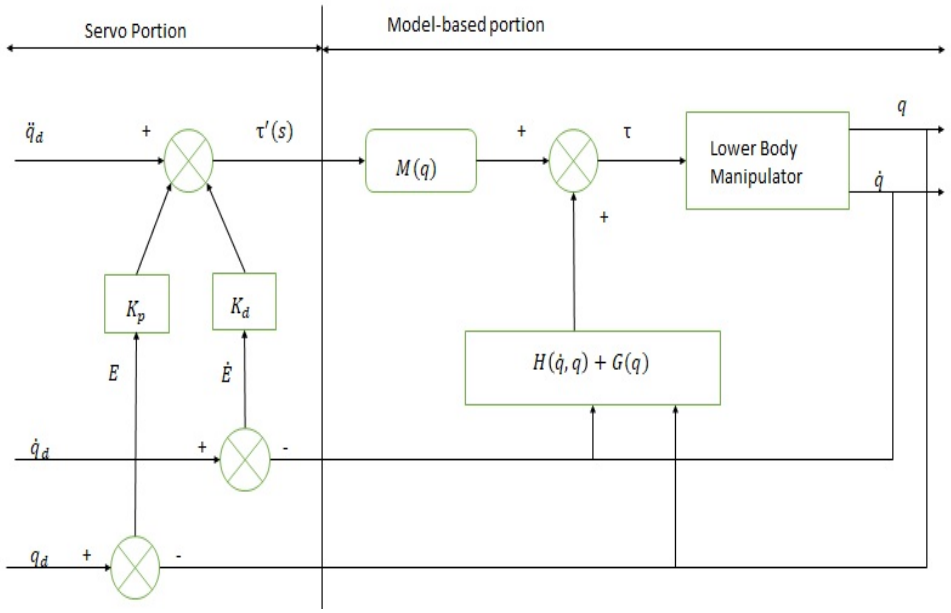


Figure 5.9:Computed torque control law

The control law of the servo portion based on the nx1 vectors E and $\dot{E}$ of the servo errors in joint positions and velocities, respectively.

The servo portion of the computed torque control scheme is, therefore, defined as

$$E(t) = q_d(t) - q(t) \qquad (5.42)$$

And $E(t) = \dot{q}_d\dot{(t)} - q\dot{(t)}$ *(5.43)*

Where q and q_d denote the nx1 vectors of actual and desired joints positions respectively. The servo portion of the computed torque control scheme is therefore, defined as

$$\grave{\tau} = \ddot{q}_d + K_d\dot{E} + \dot{K}_pE \qquad (5.44)$$

Where $K_p$and K_d are the nxn matrices of position and velocity gains, respectively.

Usually $K_p$and K_d are chosen as diagonal matrices with constant gains. This serves to decouple the error dynamics of the individual joints. The model of error behavior or error dynamics is obtained from

$$\ddot{q} = \ddot{q}_d + K_d\dot{E} + \vdots K_pE \qquad (5.45)$$

or with $\ddot{E} = \ddot{q}_d - \ddot{q}$ *(5.46)*

$$\ddot{E} + K_d\dot{E} + \dot{K}_pE = 0 \qquad (5.47)$$

This shows that the error dynamics of a closed –loop system is specified by as second order linear error equation. This vector equation is decoupled if K_p and K_d are diagonal matrices with constant gain. Hence the error equation could be written on a joint by joint basis. For joint i the error equation is

$$\ddot{e}_i + K_{di}\dot{e}_i + K_{pi}e_i = 0 \qquad (5.48)$$

Where $K_{pi}$and K_{di} are the position and velocity gains, respectively, for joint i.

For critically damped performance of joint i , the relationship between $K_{pi}$and K_{di} is obtained in equation i.e

$$K_{di} = 2\sqrt{K_{pi}} \qquad (5.49)$$

Observe that the computed torque controller employs nonlinear feedback to linearize dynamics and provides a better trajectory tracking performances than linear controllers, since it is based on a more accurate dynamics model, but the biggest limitation of this approach is that it is computationally more intensive and , hence highly expensive when compared to linear controllers.

Inaccuracies in the parameters of the manipulators in the dynamics model are other factors, which limit the manipulators performances.

5.8 Simulink Computed Torque control

Figure 5.10 shows the schematic Simulink model of control system. The Coriolis and centripetal torque do not allow the link to reach the desired position of the robot. Computed torque control measures the inertia of the system and sense the signal to different motors controlling the humanoid feet to reach the desired position. to reach to the desired position of humanoid feet to at actual position.

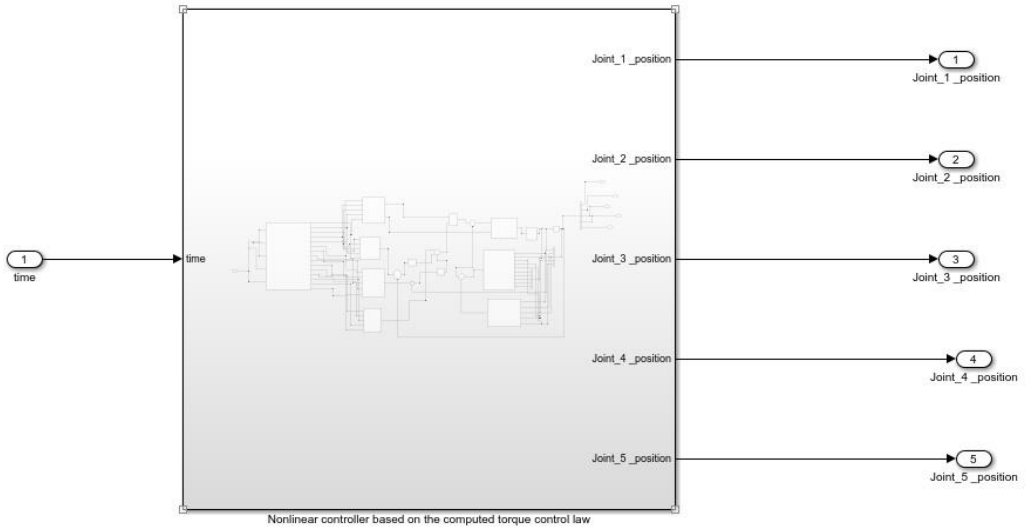


Figure 5.10:Simulink computed torque control

The input of the computed torque control is the input trajectory of joint. The nonlinear term of the CTC does not allow the link to reach the desired position. The CTC controller measure the error position and velocity of the joint. The PPD controller adjusted the torque and bring the joint into the desired position. The Coriolis and the centripetal torque are nonlinear in nature. The CTC measured this variation and modifying the controller parameter based on the unit inertia system. The Simulink signal interacting with the physical system. So that one converter used for the interface. For this purpose, the sps converter used to understand the interface the system. Figure 5.11 shows the interaction of the computed control to the lower body of the humanoid robot. The details of building Simulink block model is described in the Appendix J.

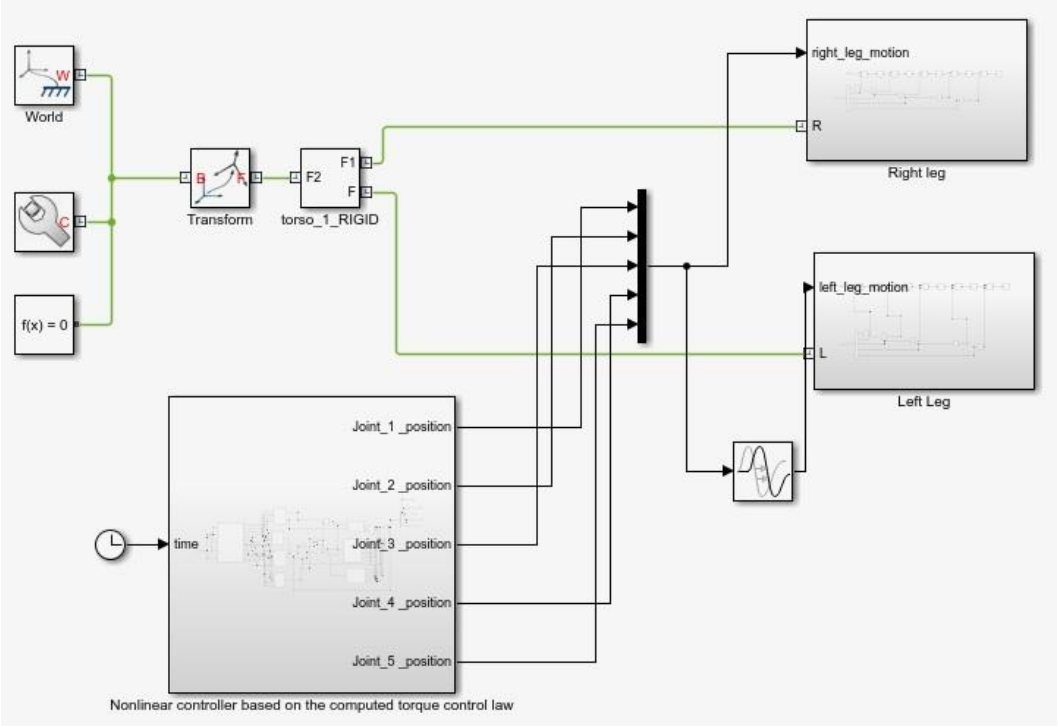


Figure 5.11:Interfacing of computed torque control to lower body of Humanoid

The interfacing of the CTC controller to the proposed model, the leg movement occurred for the plan trajectory and gait time has been noted. For a simple walk, the first, fourth and fifth joint trajectory has been implemented. During walk, the sequence of trajectory of hip joint, knee joint and ankle joint trajectory implemented, and movement of the leg observed for a gait period of 0.8 seconds. A successful walk obtained. For the stable sudden turn, the first, second, fourth and fifth joint of the system comes into the action. During sudden turn, the second joint motion produced for the momentarily and it will stop. Afterward the same trajectory repeated as usual for the normal walk.

CHAPTER 6 REINFORCEMENT LEARNING ALGORITHM

In pre-programmed robots, manipulator performs task in known environment. But,with the existence of dynamic condition, due to lack of decision making capabilites,these types of manipulator fail to do the task. In such cases the preprogrammed robot does not know what to do. The rienforcement algorithm helps to do the task in such type of unstructured and unknown environment.Reinforcement algorithm deals with the present state of the robot. It senses the present condition of the robot and takes the robot to next state. Reinforceemnt control algorithm is based on the value function, transition probabilities and the cost function reward. Reward function is the combination of positive and negative values. In present work , the reinforcement alogrithm is implementd for the hip joint, knee joint and ankle joint only. The same concept will be implementd for the whole body motion control.The positive value of reward is awarded when the manipulator is reaches to the next step successfully. Negative reward is not given to the action ,becuause when the humanoid is falls on the ground in order to reach to the desired position, hunamoid should bring the feet either in forwrad direction or negative direction . If negative reward is awarded to the state and action , next time the feet will not do the action to bring the feet in reverse direction. The reinforcement signal sense by the senor[126]. The accelerometer and the gyroscope sense the current position and the orientaion of the system. After sensing the current position , the signal will processes by the reinforcement controller. The controller looks the look-up table and compare the nexte state of the system. During switching the system some time fails to reach the desired state.In the reinforcement learning , the learning parameter decide the accuracy of the system.

6.1 Reinforcement Model

The RL model of lower body of Humanoid robot consists of

(i) A discrete set of environment states S

(ii) A discrete set of agent action A

(iii)A set of scalar reinforcement signals {0, 1} or the real numbers.

The two major characteristics of RL are: trial-and-error search and delayed reward. RL is defined not by characterizing learning methods, but by characterizing a learning problem.

One of the challenges in RL is the trade-off between exploration and exploitation. The agent has to exploit what it already knows in order to obtain reward, but it also has to explore in order to make better action selections in the future. A reinforcement learning agent is autonomous which means that its behavior is determined by its own experience. Learning is the mechanism through which an agent can increase its intelligence while performing operations. What is outside the agent is considered the environment. The states are parameters or features that describe the environment. An RL agent senses the environment and learns the optimal policy or near optimal policy by taking actions in each state of the environment. The agent must be aware of the states while interacting with the environment. An agent learns from reinforcement feedback received from its environment known as either reward or punishment signal. RL agents try to maximize the reward or minimize the punishment. Actions could affect the next state of the environment and subsequent rewards and have the ability to optimize the environment's state. Continuous learning and adapting through interaction with environment helps the agent to learn online in terms of performing the required task and improving its behavior in real time[127].

Figure 6.1 shows the model of the reinforcement model.

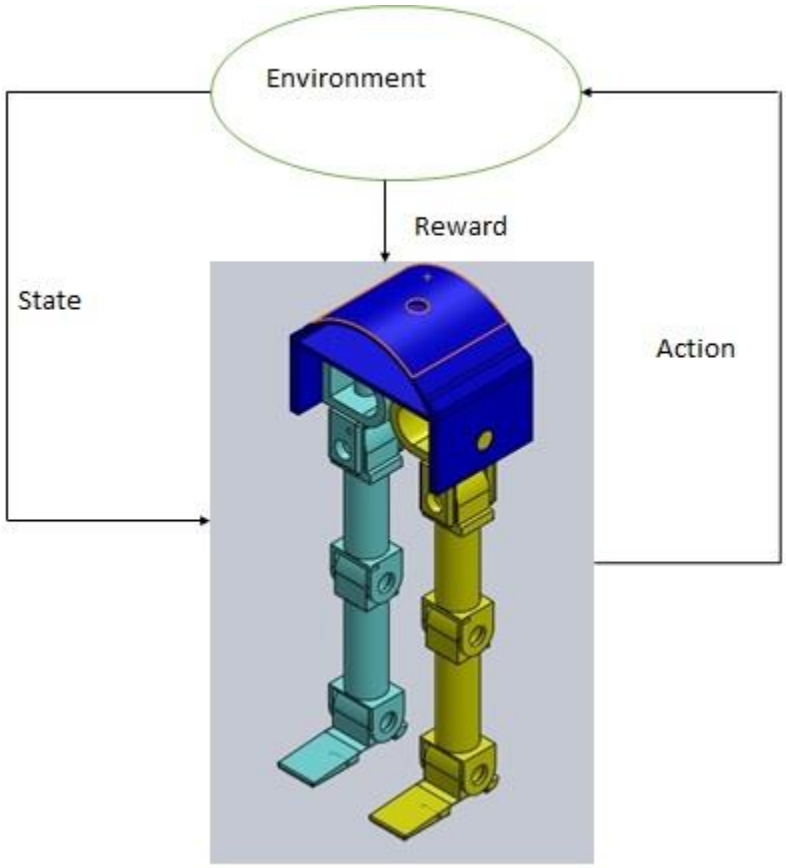


Figure 6.1: RL model of lower body of Humanoid robot

Stepwise execution of model of RL

i. Intelligent agent receives as input ‘i’, some indication of current state (s1) of the environment

ii. The intelligent agent then chooses an action from the set of actions (A) to generate as output

iii. The action changes the state of environment (s2) and the value of this state transition is communicated to the agent through a scalar reinforcement signal (r)

Reinforcement learning is mainly concerned with how an optimal policy can be obtained when a model is not known in advance.

The agent directly interacts with the environment to obtain information, which is processed through appropriate algorithm to produce an optimal policy. The two ways in which it can be proceed: Model-free method and model-based methods. In model-free method a controller is learnt without learning a model whereas in model-based method first the algorithm is learnt then it is used to derive a controller. The model-free methods are temporal difference methods, Q-learning, average rewards.

6.2 Q –Learning Method

Programming robots is a useful tedious task, so there is growing interest in building robots which can learn by themselves. Reinforcement Learning and teaching approach like Queue Learning (Q-Learning) to be implemented for robotics technology environment navigation and exploration. Q – Learning algorithm is one of the widely used online learning methods in robotics; it is simple, efficient, and not need to complex process as in adaptive system. The aim of this work is to empower the agent to learn a certain goal directed navigation strategy and to generate a shortest path in static environment which contain static obstacles; it uses one of the important intelligent search methods the “heuristic”. It makes a necessary modification for the search algorithm to suit the way of solving the problem.

In our approach of learning from demonstration, the robot learns a reward function from the demonstration and a task model from repeated attempts (trials) to perform the task.

A simplified reinforcement learning algorithm based on Q-Learning that is optimized in speed and memory consumption is proposed and implemented in MATLAB platform[128].

The body of Q-learning algorithm is as follows

1.Initialize $Q(s, a)$ to small random values

2.Observe state,s

3.Pick an action ,a, do it

4.Observe next state, s', and reward,r

5 $Q(s, a) \longleftarrow (1 - \alpha)\, Q(s, a) + \alpha\,(r + \gamma \max a\,, Q(s', a'))$

6.3 Pseudo code of Q-learning algorithm in MATALB

While developing the MATLAB code α is taken as epsilon and γ is taken as learning rate, s is taken as state and a is taken as actions and lastly reward r is taken as R.

```
learnRate=.90
epsilon=.6 // initial values
epsilonDecay=.8// need to decay the epsilon
discount=.9
successRate=1;
winBonus =100// if the goal point is achieved
startPt=[-40]//  input is  coming from the gyroscope sensor
goalPt=[0]// humanoid joint to reach the goal point  and it can varies depends on the user
maxEpi=50000;
action=1° // This value can be obtained from the smooth joint trajectory
actions=[0 , action]
x=linspace(startPt, goalPt,10)// controller crates the  eight intermediate step between the startPt and goalPt
states=zeros(length(x),1)
index=1;
for j=1:length(x)
states(index,1)=x(j)
```

```
index=index+1;
end
R=states*.1// reward is taken as linear here and it depends on the probabilities of happening the state.
Q=repmat(R,[1,3])// assigning the values of Q value for state and action
z1=startPt
For episodes=1:maxEpi
[~,SIdx]=min(sum(states-repmat(z1,[size(states,1),1])).^2,2))
if(rand()epsilon!!episodes=maxEpi)&&rand()<=successRate//pick according to the Q-matrix
[~,aIdx]=max(Q(sIdx,:) //best action
else
aIdx=randi(length(actions),1)
end
T=actions(aIdx)
z2=z1+T
z1=z2
if(z2=goalPt)
success=true
bonus=winBonus
[~,snewIdx]=min(sum(states-repmat(z1),[size(states,1),1])).^2,2))
Q(sIdx,aIdx)=Q(sIdx,aIdx)+learnRate*(R(snewIdx)+discount*max(Q(snewIdx,:)
)-Q(sIdx,aIdx)+bonus)
break
else
bonus=0
success=false
end
```

6.4 Reinforcement Controller

A reinforcement controller is implemented in MATLAB platform as shown in figure 6.2. The details of the Simulink block is described in the appendix K

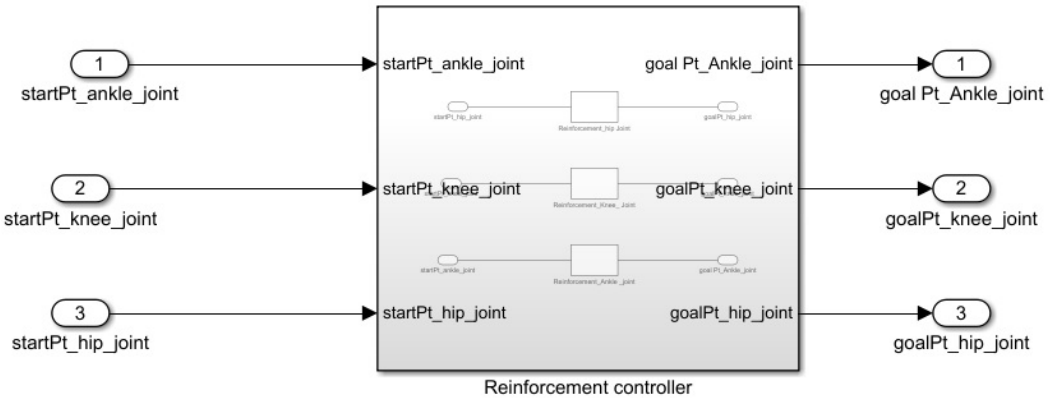


Figure 6.2:Simulink block diagram of reinforcement controller

Humanoid robot takes the decision of its own by knowing the present state and switches over to the next state without knowing the kinematics and dynamics of the system. In the present work lower body of humanoid robot is at certain position and that position is 0° to the ankle joint,-10° to the knee joint and -20° to the hip joint for the right leg and opposite of these vales to the left leg .When these values are sensed with the help of gyroscope sensor, robot has to travel the goal point. The goal point for the ankle joint, knee joint and hip joint are 10°,15° and 20° respectively. Time taking to reach these points are independent from the kinematics and dynamics calculation. In reinforcement controller time taken to reach next state, it is totally dependent on the processor capabilities and generating the control signal. The reinforcement controller controls the intermediate position of the joint between the start point and the target point.

Figure 6.3 shows the interaction of reinforcement controller to the lower body of Humanoid robot.

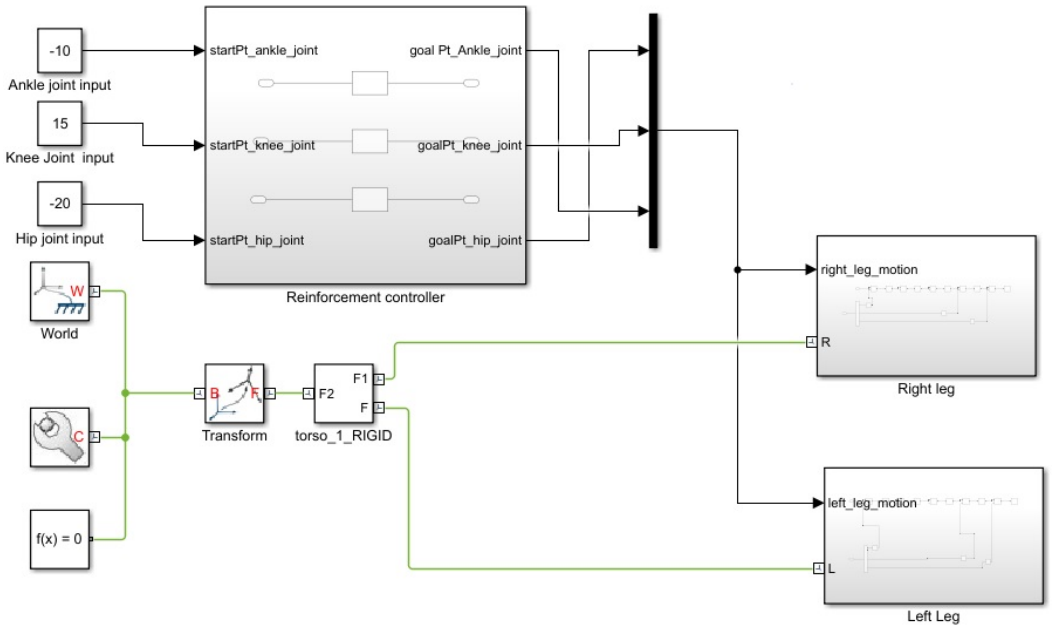


Figure 6.3:Interfacing of reinforcement controller to lower body of Humanoid robot

The randomness of the controller helps to take the action. The Q-learning algorithm allows the robot to go from one state to another state. The reward is gain during the switching by the state-action pair. After several running, the state -action pair of the reinforcement controller keeps the updated values. In the next time of running the system executes with the old values and try to switch to the next state.

CHAPTER 7 RESULTS & DISCUSSIONS

A. Planned Trajectory Results

The simulation was carried out for the planned trajectory of the different joint. The fifth order polynomial equation has been used for the trajectory planning. The boundary condition applied at the starting and end of the trajectory. Initially the joint velocity and acceleration are zero. Hence there is no jerk coming at the start and the end of the motion. The fifth order polynomial equation creates the smooth trajectory of the joint movement. The simulation was carried out for the 5 second for the minimum and maximum values of the individual joints. Figure 7.1 shows the variation of desired joint position with respect to time.

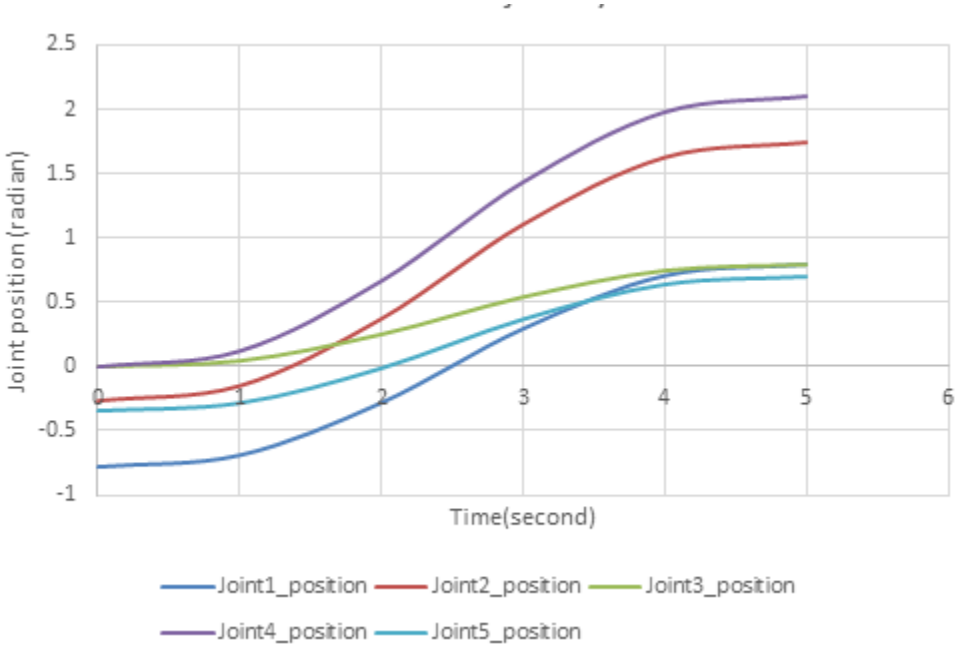


Figure 7.1:Joint position with respect to time

The desired trajectory has smooth curve, so that during motion of the body parts, nominal jerk is coming on the joints. Joint1 started from the -0.7875 radian and reached to 0.7875radian of the target point.

Starting point of joint 2 is -0.2625 radians and reached to target point of 1.75 radians. The joint 3 started from the and reached to target point of 0.7875 radians. Similarly the joint 4 started from zero radians and reached to 2.1 radians and lastly the joint 5 started from -0.35 radians and reached 0.7 radians During this trajectory , the joints are touching the intermediate position.

B. Workspace of leg

The workspace of the leg is simulated separately and assuming that there is no leg contact during interaction. This can be achieved by the maintaining the torso distance between both the leg is constant. The simulation is carried out for the workspace created by the right leg for the planned trajectory. Figure 7.2 shows the workspace created by the right leg in the space during the straight walking.

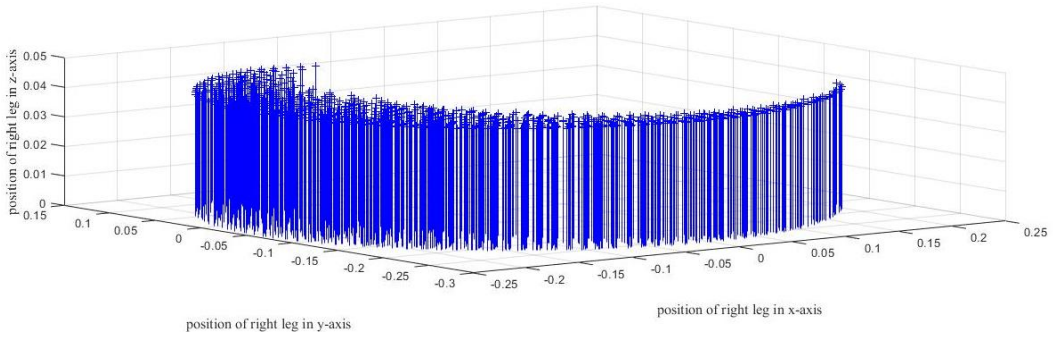


Figure 7.2:Workspace created by right leg during straight walk

During the straight walking motion, the joint1, joint4 and joint 5 are coming into action. The workspace created by the right leg is hemispherical in shape that is the combined movement of the joints. The reach of the right leg varies from the $[-0.1625 \ -0.2084\ 0.0413]$ to $[-0.0257 \ -0.1021 \ 0.0413]$ in x-axis, y-axis and z-axis.

During sudden turn, the joint1, joint2, joint 4 and joint 5 are coming into the action. The workspace created during the sudden turn is ellipsoid in shape. Figure 7.3 shows the workspace created for the right leg during sudden turn.

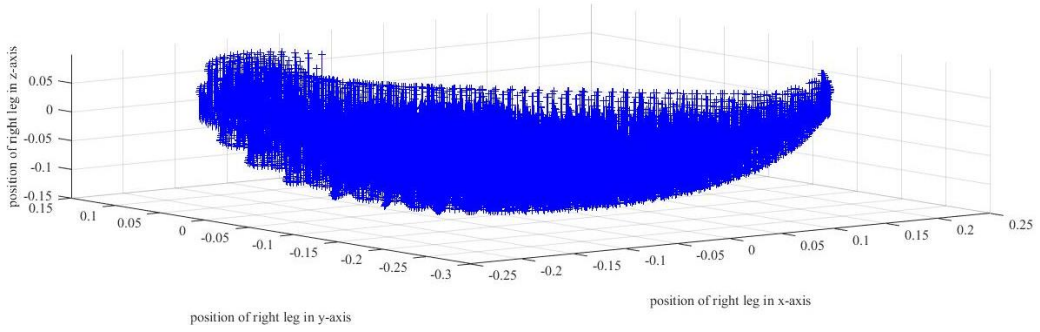


Figure 7.3:Workspace created during sudden turn

The reach of the right leg varies during sudden turning condition from the $[-0.1633 \ -0.2076 \ 0.0328]$ to $[-0.0256 \ 0.0533 \ -0.0701]$ in x-axis, y-axis and z-axis.

The complete workspace created by the right leg during movement of planned trajectory is shown in figure 7.4.

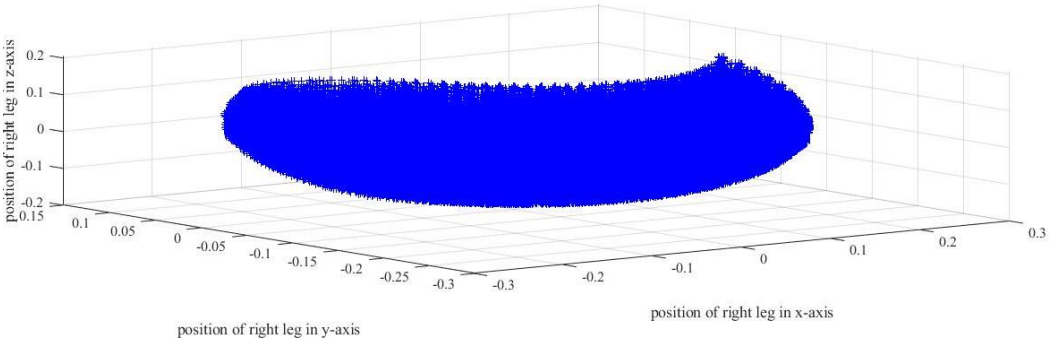


Figure 7.4:Workspace created by the right leg

The reach of the right leg varies in the workspace from the $[-0.1633 - 0.2076 \ 0.0328]$ to $[0.0414 \ 0.0806 - 0.0583]$ in x-axis, y-axis and z-axis.

C. Dynamic Torque

The simulation was carried out for the joint torque for the desired trajectory of the joint. The dynamic torque depends on the inertia of the body, centripetal and Coriolis, and the gravity. The magnitude of the dynamic torque is the selection parameter of the DC motor. Based on torque vales, the voltage and ampere characteristics of the DC motor were selected the nonlinear second order differential equation makes the system unstable. The damping factor creates the oscillation into the system. The oscillation effect discarded by introducing the critical damping into the system. Figure 7.5 shows the variation of different joint torque for the planned trajectory of joint movement with respect to time.

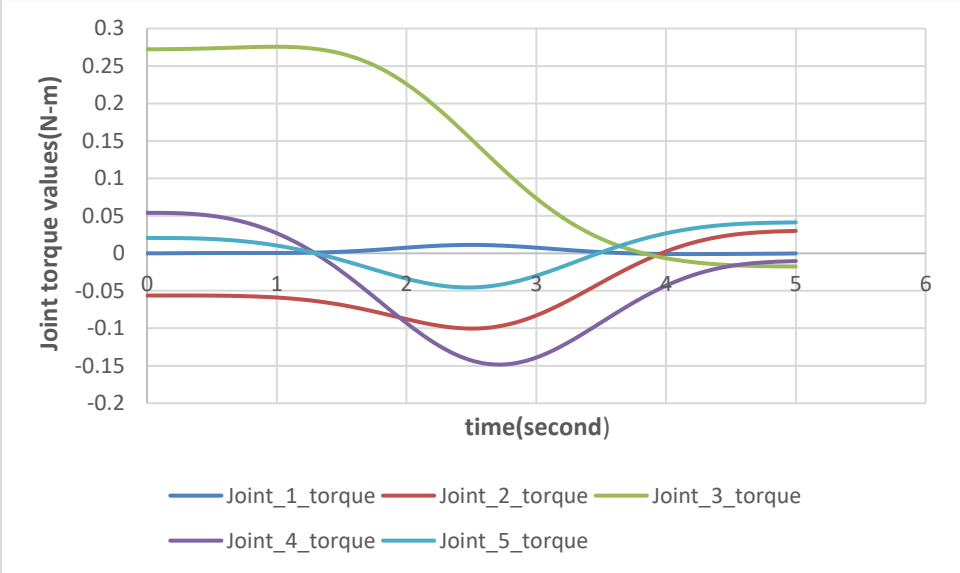


Figure 7.5:Joint torque varies with respect to time

Since the joint movement occurs in both the direction along the axis of movement. The torque values give the positive and negative values of the magnitudes. The positive values describe the forward movement and negative values describe the backward movement of joint. The joint1 torque varies from the 0N-m to -0.000181195 N-m to a maximum value of 0.011248351 N-m. The joint torque 2 varies from the -0.056275024 N-m to 0.029943409 N-m to a maximum value of 0.029943409 N-m. The joint roque 3 varies from the 0.272241706 N-m to -0.017557993 N-m to a maximum value of 0.275608884N-m.

The joint torque 4 varies from the 0.054035874 N-m to -0.01030846 N-m to a maximum value of 0.054035874N-m. The joint torque 5 varies from the 0.020501987 N-m to 0.04131821 N-m to a maximum value of 0.04131821 N-M.

D. Actual joint position

The simulation was carried out for the computed torque control law on the lower body of the Humanoid robot. The error between the desired joint position and the actual joint position evaluated for the planned trajectory of the individual joint.

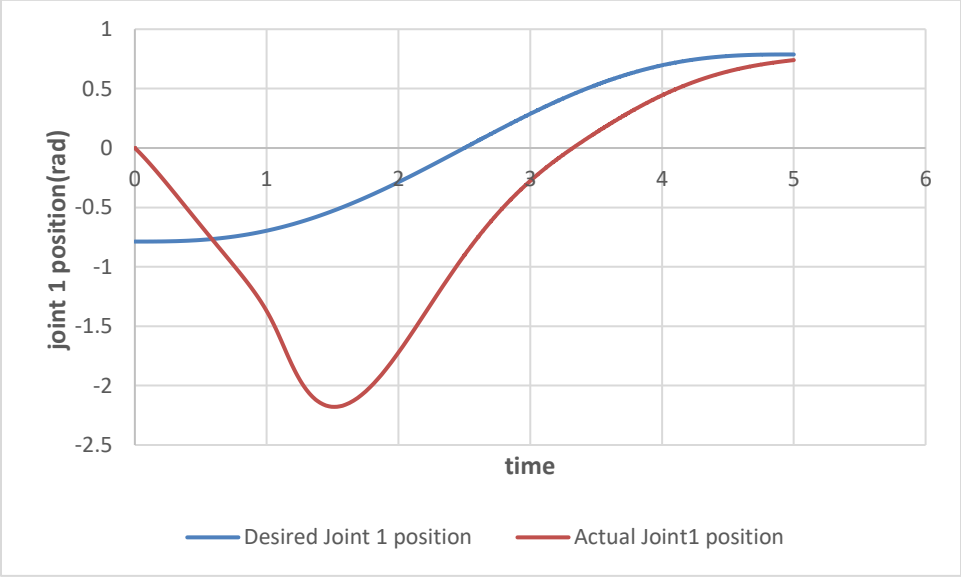


Figure 7.6:Comparision of joint 1 desired position to actual position

Figure7.6 shows the desired joint position and actual joint position of joint 1. The starting desired point of joint 1 is -0.7875 radian but the actual joint starting point of joint 1 is 0 radian. At the time of $t = 0$ seconds the joint 1 is starting from the -0.7875 radians, but the controller assumes the zero position of the actual joint1.Afterwards, the partitioned PD controller controls the joint position and reaching to the target point. During this movement there is overshoot of the actual joint position. This overshoot problem can be discarded by the assuming zero position of desired trajectory of joint 1. At the end of time $t = 5$ seconds the actual joint position 0.7875 radians achieved by the joint 1 without any oscillation.

Figure7.7 shows the desired joint position and actual joint position of joint 2. The starting desired point of joint 2 is -0.2625 radian but the actual joint starting point of joint 2 is 0 radian.

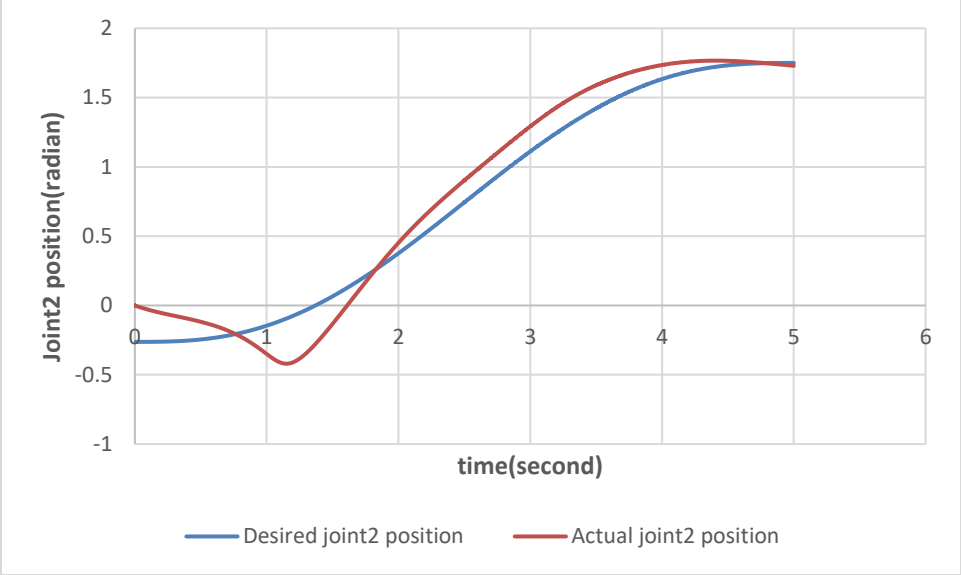


Figure 7.7:Comparision of joint 2 desired position to actual position

At the time of $t = 0$ seconds the joint 2 is starting from the -0.2625 radians, but the controller assumes the zero position of the actual joint2.Afterwards the partitioned PD controller controls the joint position and reaching to the target point. During this movement there is overshoot of the actual joint position. This overshoot problem can be discarded by the assuming zero position of desired trajectory of joint 1. At the end of time $t = 5$ seconds the actual joint position 1.3 radians, that is not achieved by the joint 2. There is an error of 0.02 radians. This error can be discarded by the tuning of PD controller parameter.

Figure7.8 shows the desired joint position and actual joint position of joint 3. The starting desired point of joint 2 is 0 radian and the actual joint starting point of joint 3 is 0 radian.

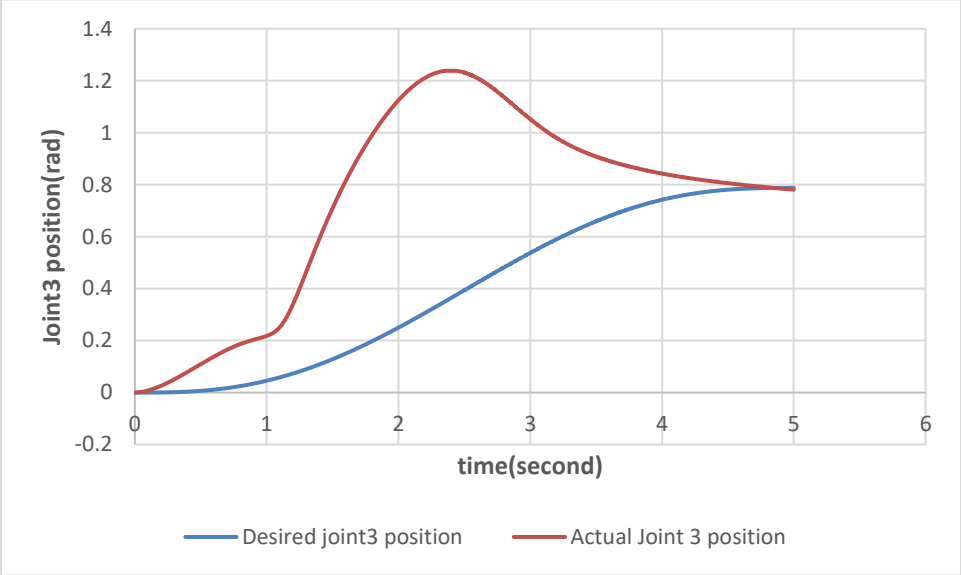


Figure 7.8:Comparision of joint 3 desired position to actual position

At the time of $t = 0$ seconds the joint 3 is started from the zero radian and targeted to goal point of 0.7875 radians. The partitioned PD controller controls the joint position and reaching to the target point. During this movement there is overshoot of the actual joint position. At the end of time $t = 5$ seconds the actual joint position 3 is 0.781 radians, which is slightly far away from the target desired position. There is an error of 0.006 radians. This error can be discarded by controlling the joint velocity of joint 3 by adjusting the velocity tuning parameter of PD controller.

Figure7.9 shows the desired joint position and actual joint position of joint 4. The starting desired point of joint 4 is 0 radian and the actual joint starting point of joint 4 is 0 radian.

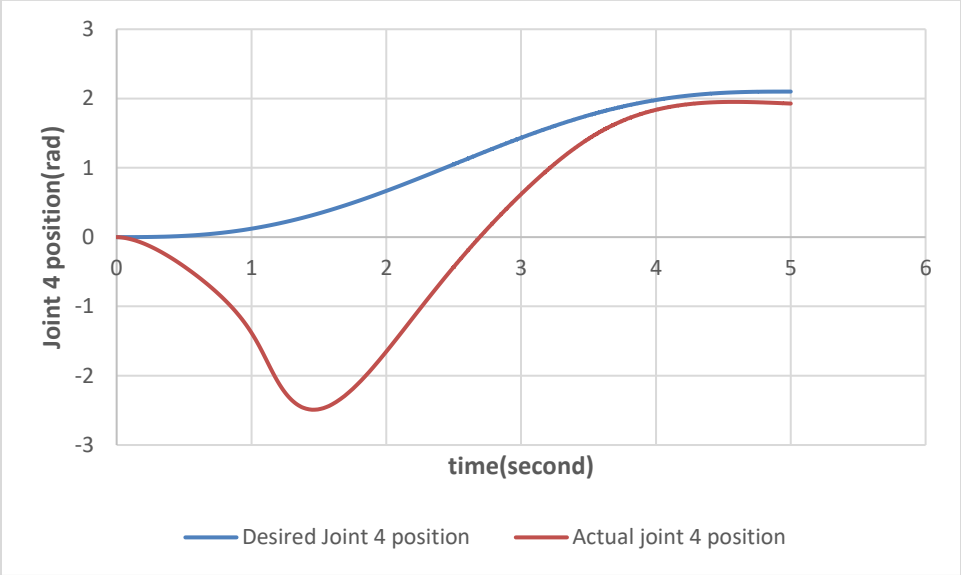


Figure 7.9:Comparision of joint 4 desired position to actual position

At the time of $t = 0$ seconds the joint 4 is started from the zero radian and targeted to goal point of 2.1 radians. The partitioned PD controller controls the joint position and reaching to the target point. During this movement there is overshoot of the actual joint position. At the end of time $t = 5$ seconds the actual joint position 4is 1.92 radians, which is slightly far away from the target desired position. There is an error of 0.18 radians. This error can be discarded by controlling the joint velocity of joint 4 by adjusting the tuning parameter of PD controller. During this movement of the joint, there is no oscillation in the movement.

Figure7.10 shows the desired joint position and actual joint position of joint 5. The starting desired point of joint 5 is -0.35 radian and the actual joint starting point of joint 5 is 0 radian.

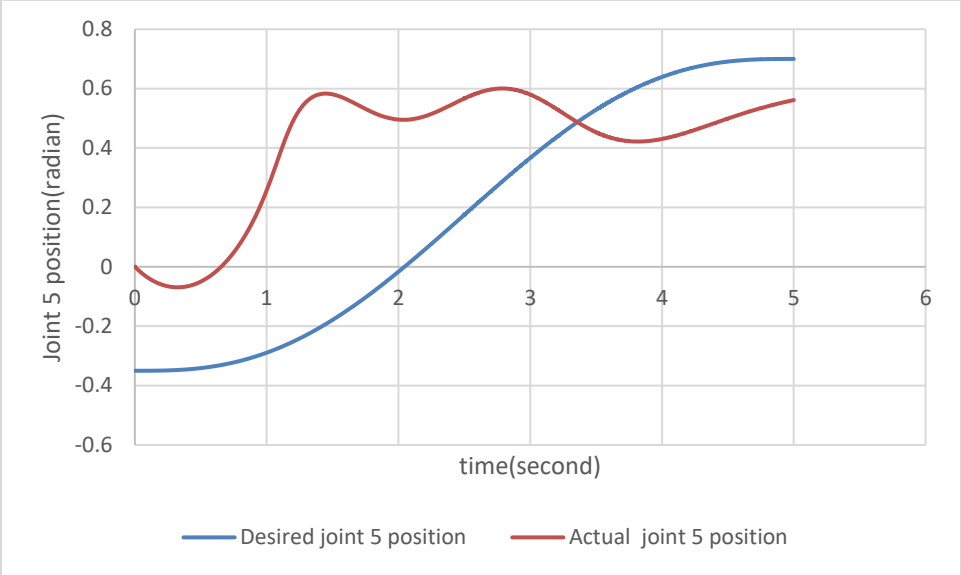


Figure 7.10:Comparision of joint 5 desired position to actual position

At the time of $t = 0$ seconds the joint 4 is started from the -0.35 radian and targeted to goal point of 0.7 radians. The partitioned PD controller controls the joint position and reaching to the target point. During this movement there is overshoot of the actual joint position. At the end of time $t = 5$ seconds the actual joint position 4is 0.562 radians, which is slightly far away from the target desired position. There is an error of 0.138 radians. This error can be discarded by controlling the joint velocity of joint 5 by adjusting the tuning parameter of PD controller. During this movement of the joint, there is no oscillation in the movement.

E. Learning of joint1

The simulation was carried out for the learning behavior of different joints. In proposed model, simulation of learning position and random number generation for joint 1, joint 4 and joint 5 presented below.

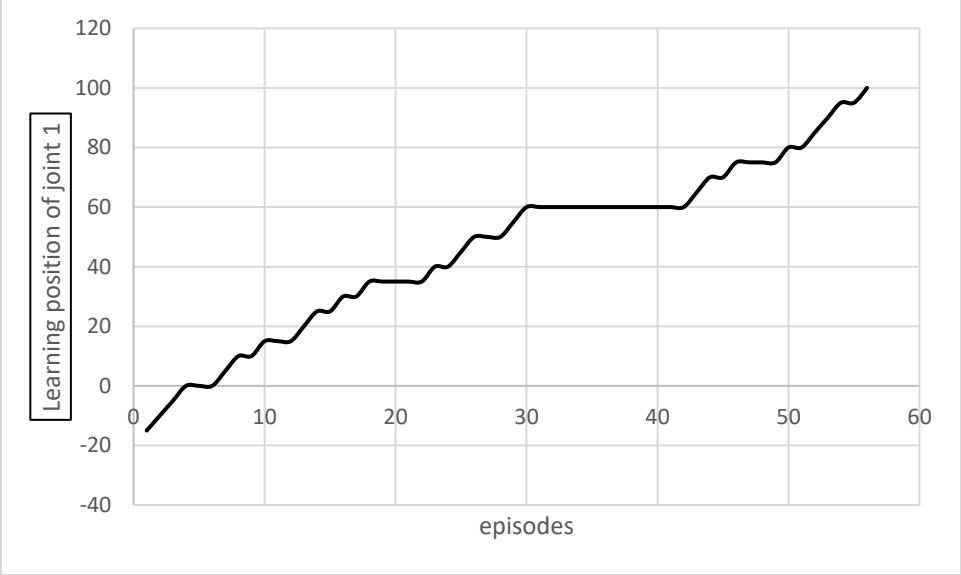


Figure 7.11:Learning of joint 1 position to target with respect to episodes

Figure 7.11 shows the learning of joint 1 with respect to episodes. After sensing the current position, the joint1 switch to the next sate. The starting point of joint 1 is -15° and target point is 100°. During this transition the joint 1 touches the intermediate position. The intermediated position achieved by taking the action. During switching the joint 1 remains in same position. The time taken to reach initial position to target point depends upon the processor capabilities and generation of random number. During the episodes of 4- 6 the joint1 remains at 0° and during episodes 45-49 it remains at 75°.The intermediate position depends on the random number. The random numbers are generated by the controller. User does not have any control over the generation of random number. The switching time can be controlled by the several experiment. During experiment the minimum number of episodes noted.

Figure 7.12 shows the random number generated by the processor with respect to episodes.

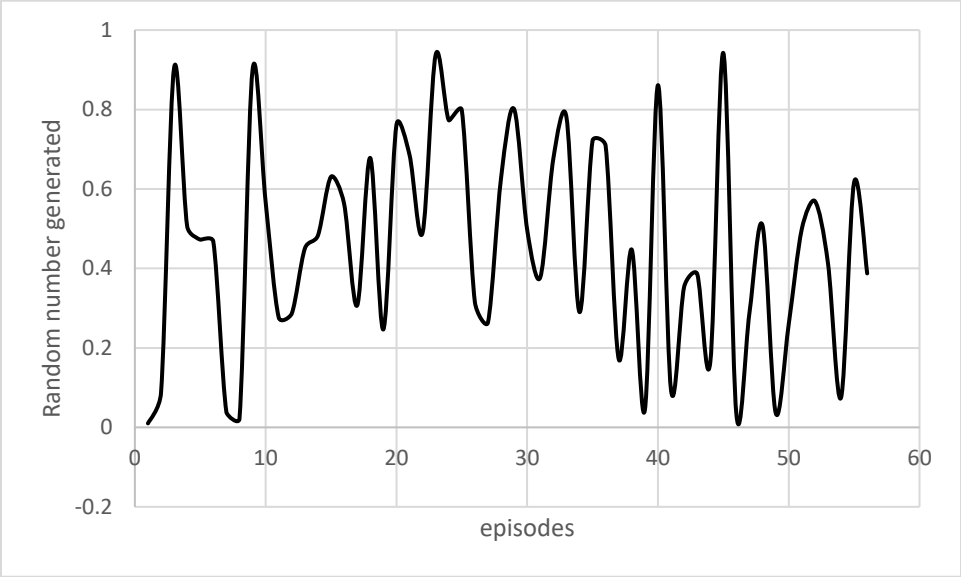


Figure 7.12:Random number generation to reach target point of joint 1 with respect to episodes

The average random number 0.456301 has been taken for the implementation of the reinforcement controller to decide the action. After successfully reaching to the target point the reward had been provided to the state and action. Table 7.1 shows the reward values of the state and action pair of the intermediate position.

Table 7.1:Q-optimal policy for joint 1

State	Action
-1.5	-1.5
1.375	1.375
4.25	4.25
7.125	7.125
10	117.91

F. Learning of joint4

Figure 7.13 shows the learning of joint 4with respect to episodes. After sensing the current position, the joint4 switch to the next sate.

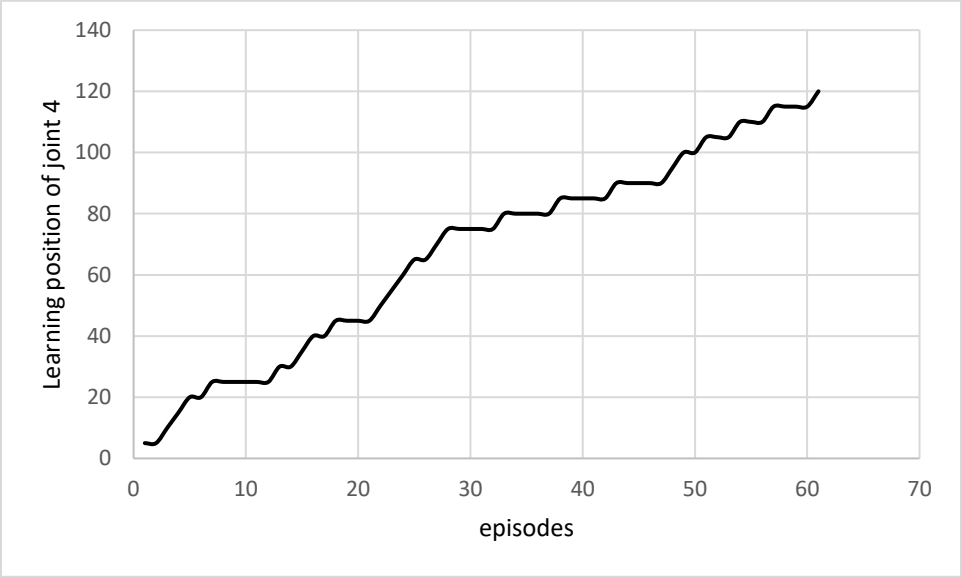


Figure 7.13:Learning of joint 4 position to target with respect to episodes

The starting point of joint 4 is 0° and target point 120°. During this transition the joint 4 touches the intermediate position. The intermediated position is achieved by taking the action during switching the joint 4 remains in same position. The time taken to reach initial position to target point depends on the processor capabilities and generation of random number. During the episodes of 7-12 the joint4 remains at 25° and during episodes 54-60 it remains at 110°.The intermediate position depends on the random number. The random number is generated by the controller. User does not have any control over the generation of random number. The switching time can be controlled by the several experiment. During experiment the minimum number of episodes noted.

Figure 7.14 shows the random number generated by the processor with respect to episodes.

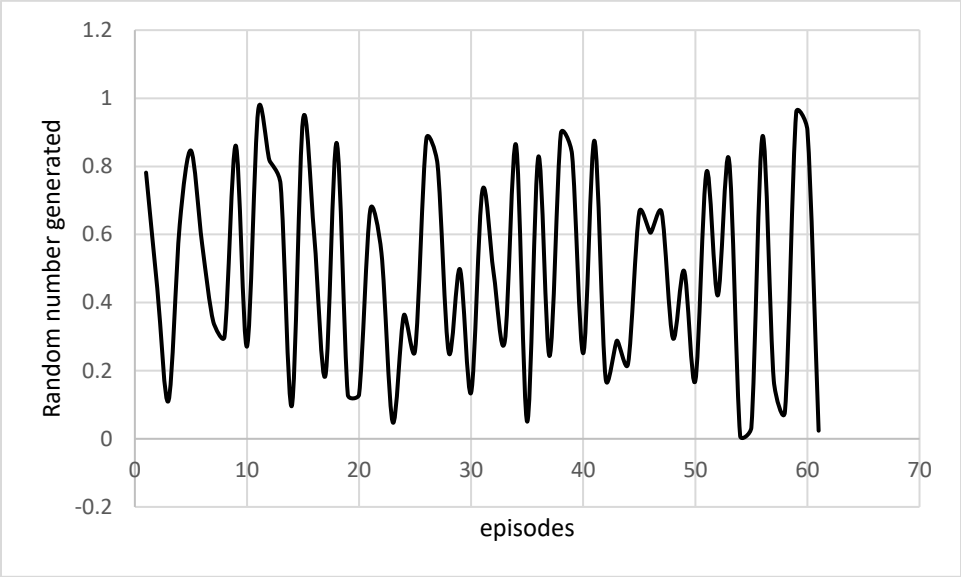


Figure 7.14:Random number generation to reach target point of joint 4 with respect to episodes

The average random number 0.493718has been taken for the implementing the reinforcement controller to decide the action. After successfully reaching to the target point the reward had been provided to the state and action. Table 7.2 shows the reward values of the state and action pair of the intermediate position.

Table 7.2:Q-optimal policy for joint 4

State	Action
0	0
3	3
6	6
9	9
12	121.692

G. Learning of joint5

Figure 7.15 shows the learning of joint 5with respect to episodes. After sensing the current position, the joint4 switch to the next sate.

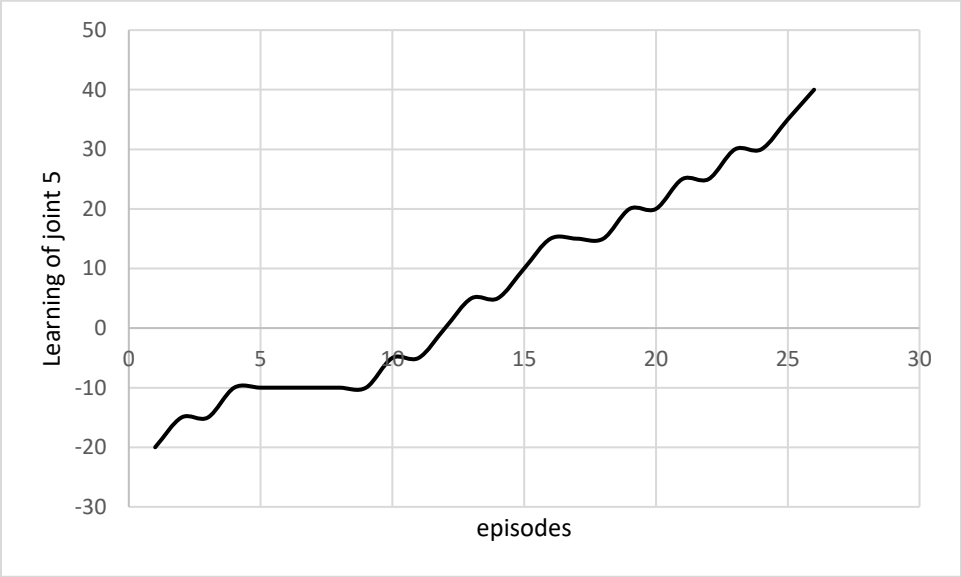


Figure 7.15:Learning of joint 5position to target with respect to episodes

The starting point of joint 5 is -20° and target point 40°. During this transition the joint 5 touches the intermediate position. The intermediated position achieved by taking the action. During switching the joint 5 remains in same position. The time taken to reach initial position to target point depends on the processor capabilities and generation of random number. During the episodes of 4-9 the joint5 remains at -10° and during episodes 16-18 it remains at 15°.The intermediate position depends on the random number. The random number generates by the controller. User does not have any control over the generation of random number. The switching time can be controlled by the several experiment. During experiment the minimum number of episodes noted.

Figure 7.16 shows the random number generated by the processor with respect to episodes.

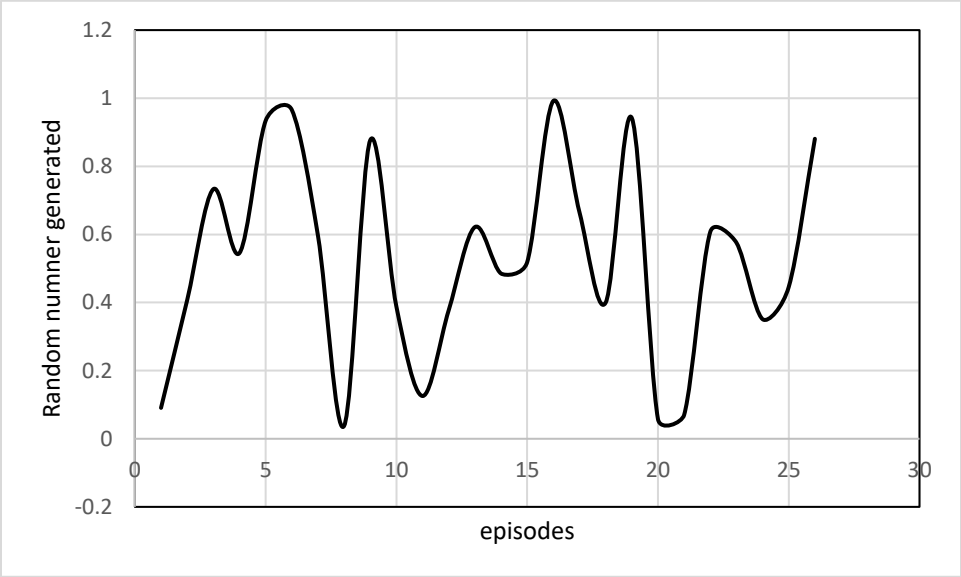


Figure 7.16:Random number generation to reach target point of joint 5with respect to episodes

The average random number 0.526571 has been taken for the implementing the reinforcement controller to decide the action. After successfully reaching to the target point the reward had been provided to the state and action. Table 7.3 shows the reward values of the state and action pair of the intermediate position.

Table 7.3:Q-optimal policy for joint 5

State	Action
-2	-2
-0.5	-0.5
1	1
2.5	2.5
4	106.564

H. Switching of state

The simulation experiment was carried out for the reinforcement controller. Figure 38 shows the initial position of lower body of humanoid robot. The initial position of joint 1 is taken as -10°, initial position of joint 4 is taken as 15° and the initial position of joint 5 is -20°. Due to software constraint, the initial position is taken as 0° for all the joint. The initial position of the lower body is vertically upward position and assuming no deviation in other joint 2 and joint 3. Figure 7.17. shows the initial posture of the lower body.

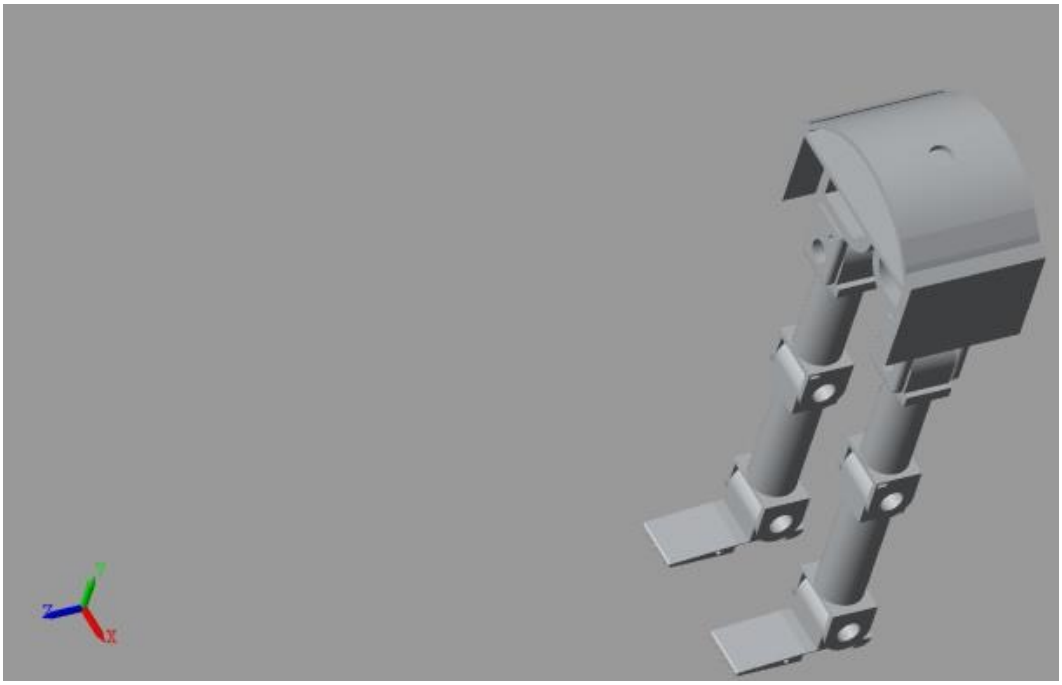


Figure 7.17:Initial state of lower body of humanoid robot

Reinforcement controller takes the input of joint position of lower body of the humanoid robot as a starting point. The goal point of the joint 1 is 10°, the goal point of joint 4 is -120° and the goal point of joint 4 is 0°. These goal points decided the one of the states of lower body. The controller takes the input and executes the algorithm.

Figure 7.18 shows the next state of lean of the lower body of humanoid robot.

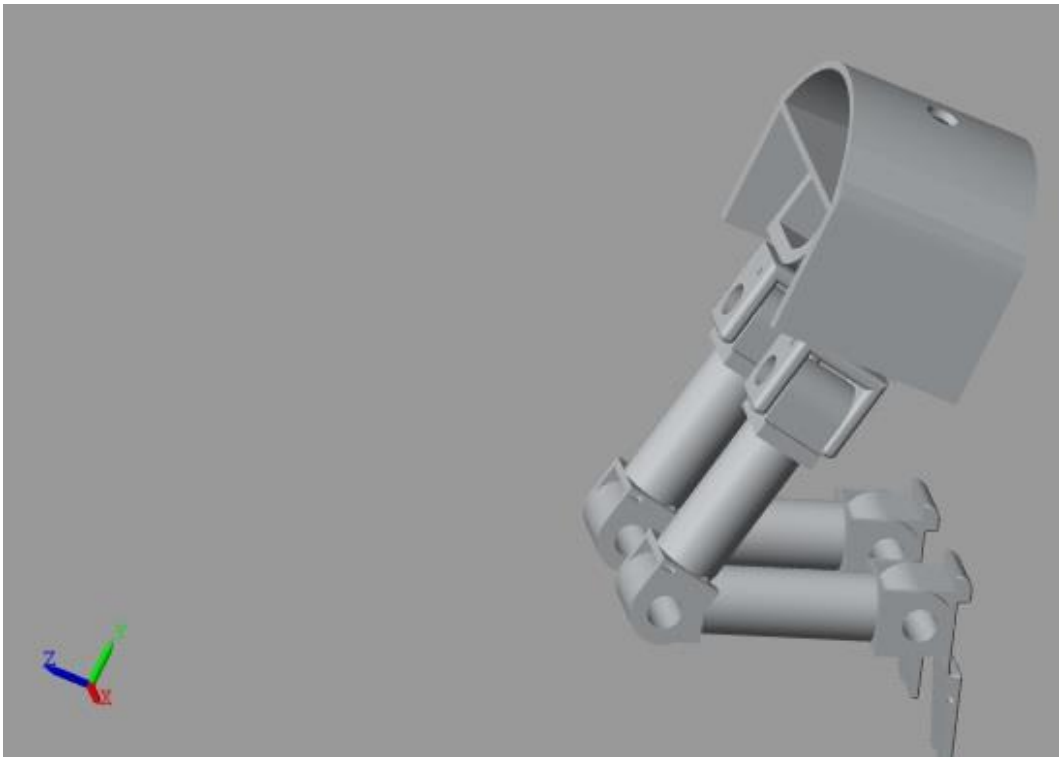


Figure 7.18:Next state of lower body of humanoid robot

During this transition from one state to next state, there is no kinematics and dynamics involved. Controller takes its own decision to switch over. There is no control over the motion of joint by any user or human interface. The switching is totally depending on the controller decision. The reinforcement controller does not depend upon the preprogram of the robot. It is totally dependent upon the transition of probabilities.

CHAPTER 8 CONCLUSION AND FUTURE SCOPE

One of the main issues in the humnaoid robot is the mechanical design . Due to several linkages and joint, the system become complicated and it reduces the dexterity of the humnaoid robot. The lioterature review found the kinematic configuartion of the humnaoid robot. With the existing kinemactic configuaration, the fast movement and sudden turn becomes challenging for the stability and control for the humnaoid robot.The other issue is self capable decision making in the unstructured and unknown environment.Till today, the limited success has been achived towards the self decision making robot. These two problems were identified.

The proposed model considered the biomechanics paramter of the human.A modified kinematic configuation was proposed. The first joint,fourth joint and the fifthe joint of the proposed model attached in the same plane. The motion of these joints are not disturbing to each other and the fast movement will be achieved without any deviation in the statbility. The second joint attached on the different plane. The motion of second joint helps to take the turn very efficiently and stabilize the body. The third joint has different plane of rotation other than the first two joints. These joints help the humnaoid robot to stay back in the original position, when sudden impact comes from the side.

The mathematical expression were derived for the modified kinematcic configuarartion of the humnaoid robot. The mathemeatical expression described the kinematics and dynamics of the lower body. It calaculate the singularity posture of the body.For the proposed model , during walking , there is no singularity posture obtained.

A simple distributed master and slave control architecture adopted fir the joint movement control. The partitioned PD control law controls the non linear system very efficientrly.The controller gain paramter were selected on the material stiffness and the natural frequecny of the system. The systen were designed for the crtitical damped,so that fast non oscillatory behavior of the joint is obtained.
The reinforcement algorithms implemented to take the decison in the unstructed and unknown enevironment . The Q-algorithm adpoted for the developing the set of instructions. The partial Markove Decision process model were considerd for the developing the reinforcement controller. The randomness of the controller deos not depend on the other human assiatance and the user. The simulation were carreid out for transition of state. The result found that the humanoid robot changes from the current state to the next state without any model based informantion.

8.1 Future work

At the beginning of this work, the fabrication of full scale of humanoid robot to implement. Due to machining error inside the workshop laboratory, there existed mechanical errors. Rhino DC servo motor supposed to control the joint position. Due to less magnitude of the stall torque, body does not hold the position in the upright condition. These are the failures of this project. The future of this project will include

- The full-scale humanoid robot should be fabricated with new kinematic configuration
- The first layer of the control architecture consists the machine vision and behavioral based interaction. These two features were required to control the smooth motion and increasing the stability.
- MATALB generate the C-code on the MATALAB editor. These codes were implemented in the embedded code in the form of task description languages.

CHAPTER 9 REFERENCE

[1] J. Denny, M. Elyas, S. A. Dcosta, and R. D. DSouza, "Humanoid Robots – Past , Present and the Future Humanoid Robots – Past , Present and the Future," Eur. J. Adv. Eng. Technol., vol. 3, no. 5, pp. 8–15, 2016.

[2] T. Asfour et al., "Toward humanoid manipulation in human-centred environments," Rob. Auton. Syst., vol. 56, no. 1, pp. 54–65, 2008.

[3] C. Ott et al., "A humanoid two-arm system for dexterous manipulation," Proc. 2006 6th IEEE-RAS Int. Conf. Humanoid Robot. HUMANOIDS, pp. 276–283, 2006.

[4] K. Okada, M. Kojima, Y. Sagawa, T. Ichino, K. Sato, and M. Inaba, "Vision based behavior verification system of humanoid robot for daily environment tasks," Proc. 2006 6th IEEE-RAS Int. Conf. Humanoid Robot. HUMANOIDS, vol. 00, pp. 7–12, 2006.

[5] P. Michel, J. Chestnutt, J. Kuffner, and T. Kanade, "Vision-guided humanoid footstep planning for dynamic environments," Proc. 2005 5th IEEE-RAS Int. Conf. Humanoid Robot., vol. 2005, pp. 13–18, 2005.

[6] S. Kagami, K. Nishiwaki, J. J. Kuffner, Y. Kuniyoshi, M. Inaba, and H. Inoue, "Online 3D vision, motion planning and bipedal locomotion control coupling system of humanoid robot: H7," no. October, pp. 2557–2562, 2003.

[7] A. Song, G. Song, D. Constantinescu, L. Wang, and Q. Song, "Sensors for Robotics 2015," J. Sensors, vol. 2015, pp. 1–2, 2015.

[8] O. Khatib, "Robotics and Autonomous Systems Mobile manipulation: The robotic assistant," Rob. Auton. Syst., vol. 26, pp. 175–183, 1999.

[9] N. Sawasaki, T. Nakajima, A. Shiraishi, S. Nakamura, K. Wakabayashi, and Y. Sugawara, "Application of humanoid robots to building and home management services," pp. 2992–2997, 2004.

[10] T. Dahl and M. Boulos, "Robots in Health and Social Care: A Complementary Technology to Home Care and Telehealthcare?," Robotics, vol. 3, no. 1, pp. 1–21, 2013.

[11] H. Robinson, B. MacDonald, and E. Broadbent, “The Role of Healthcare Robots for Older People at Home: A Review,” Int. J. Soc. Robot., vol. 6, no. 4, pp. 575–591, 2014.

[12] I. Maurtua, A. Ibarguren, J. Kildal, L. Susperregi, and B. Sierra, “Human-robot collaboration in industrial applications: Safety, interaction and trust,” Int. J. Adv. Robot. Syst., vol. 14, no. 4, pp. 1–10, 2017.

[13] Y. Tanaka, H. Lee, D. Wallace, Y. Jun, P. Oh, and M. Inaba, “Toward deep space humanoid robotics inspired by the NASA Space Robotics Challenge,” 2017 14th Int. Conf. Ubiquitous Robot. Ambient Intell. URAI 2017, pp. 14–19, 2017.

[14] H. K. Kalra and R. Chadha, “TC,” vol. 4, no. 3, pp. 31–33, 2018.

[15] M. Iklé et al., “Using Tononi Phi to measure consciousness of a cognitive system while reading and conversing ?,” CEUR Workshop Proc., vol. 2287, pp. 1–6, 2018.

[16] D. Hanson et al., “Realistic humanlike robots for treatment of ASD, social training, and research; shown to appeal to youths with ASD, cause physiological arousal, and Increase Human- to-Human Social Engagement,” 5th Int. Conf. PErvasive Technol. Relat. to Assist. Environ., pp. 1–7, 2012.

[17] S. Maniatopoulos, P. Schillinger, V. Pong, D. C. Conner, and H. Kress-gazit, “Reactive High-level Behavior Synthesis for an Atlas Humanoid Robot,” 2016 IEEE Int. Conf. Robot. Autom., pp. 4192–4199, 2016.

[18] H. Motor, “Asimo,” Public Relations Div., no. September, 2007.

[19] V. Tikhanoff, P. Fitzpatrick, F. Nori, L. Natale, G. Metta, and A. Cangelosi, “The iCub Humanoid Robot Simulator,” vol. 1.

[20] M. Lapeyre, S. N’Guyen, A. Le Falher, and P. Y. Oudeyer, “Rapid morphological exploration with the Poppy humanoid platform,” IEEE-RAS Int. Conf. Humanoid Robot., vol. 2015-Febru, pp. 959–966, 2015.

[21] N. Pateromichelakis et al., “Head-eyes system and gaze analysis of the humanoid robot Romeo,” IEEE Int. Conf. Intell. Robot. Syst., no. Iros, pp. 1374–1379, 2014.

[22] A. K. Pandey et al., “Romeo2 project: Humanoid robot assistant and companion for everyday life: I. situation assessment for social intelligence,” CEUR Workshop Proc., vol. 1315, no. November, pp. 140–147, 2014.

[23] G. Nelson *et al.*, “PETMAN: A Humanoid Robot for Testing Chemical Protective Clothing,” *J. Robot. Soc. Japan*, vol. 30, no. 4, pp. 372–377, 2012.

[24] S. Shamsuddin, L. I. Ismail, H. Yussof, N. I. Zahari, and S. Bahari, “Humanoid Robot NAO : Review of Control and Motion Exploration,” *2011 IEEE Int. Conf. Control Syst. Comput. Eng.*, pp. 511–516, 2011.

[25] S. Shamsuddin, H. Yussof, H. Robots, and B. Hurobs, “Initial Response of Autistic Children in Human-Robot Interaction Therapy with Humanoid Robot NAO,” 2012 IEEE 8th Int. Colloq. Signal Process. its Appl., pp. 188–193, 2012.

[26] I. M. Verner, A. Polishuk, and N. Krayner, “Science Class with RoboThespian: Using a Robot Teacher to Make Science Fun and Engage Students,” IEEE Robot. Autom. Mag., vol. 23, no. june 2016, pp. 74–80.

[27] Y. Kondo, K. Takemura, J. Takamatsu, and T. Ogasawara, “A Gesture-Centric Android System for Multi-Party Human-Robot Interaction,” J. Human-Robot Interact., vol. 2, no. 1, pp. 133–151, 2013.

[28] S. C. Lim and G. H. Yeap, “The locomotion of bipedal walking robot with six degree of freedom,” Procedia Eng., vol. 41, no. Iris, pp. 8–14, 2012.

[29] J. Park, “General ZMP Preview Control for Bipedal Walking,” EEE Int. Conf. Robot. Autom. Roma, Italy, 10-14 April 2007, no. April, pp. 10–14, 2007.

[30] N. Ken’ichiro, “Acquisition of Visually Guided Swing Motion Based on Genetic Algorithms and Neural Networks in Two-Armed Bipedal,” Proc. 1997 IEEE lntemational Conf. Robot. Autom., no. April, pp. 2944–2949, 1997.

[31] L. Yang, C. M. Chew, and A. N. Poo, “Adjustable Bipedal Gait Generation using Genetic,” Proc. 2006 IEEE/RSJ Int. Conf. Intell. Robot. Syst. Oct. 9 - 15, 2006, Beijing, China, pp. 4435–4440, 2006.

[32] S. C. Y. Kim and S. Hutchinson, “An Improved Hierarchical Motion Planner for Humanoid Robots,” 2008 8th IEEE-RAS Int. Conf. Humanoid Robot. December 1 -- 3, 2008 / Daejeon, Korea, pp. 654–661, 2008.

[33] R. Niiyama, S. Nishikawa, and Y. Kuniyoshi, “Athlete Robot with Applied Human Muscle Activation Patterns for Bipedal Running,” 2010 10th IEEE-RAS Int. Conf. Humanoid Robot., pp. 498–503, 2010.

[34] H. Inada, “Behavior Generation of Bipedal Robot Using Central Pattern Generator(CPG) (1st Report: CPG Parameters Searching Method by Genetic Algorithm),” Proc. 01 2003 IEEWRSJ lntl Conterence ntell gent Robot. ana Syst. Las, no. October, pp. 2179–2184, 2003.

[35] B. Thuilot, A. Goswami, and B. Espiau, “Bifurcation and chaos in a simple passive bipedal gait,” Proc. 1997 IEEE Int. Conf. Robot. Autom. Albuquerque, New Mex. - April 1997 Bifurc., no. April, pp. 792–798, 2002.

[36] D. Nishino and A. Takanishi, “Development of a Bipedal Humanoid Robot Having Antagonistic Driven Joints and Three DOF Trunk,” Roceedings 1998 IEEE/RSJ htl. Conf. Intell. Robot. Syst. Victoria, B.C., Canada Oct. 1998, no. October, pp. 96–101, 1998.

[37] S. Inoue and A. Takanishi, “Development of a Bipedal Humanoid Robot - Control Method of Whole Body Cooperative Dynamic Biped Walking -,” Proc. 1999 EEE Int. Conf. Robot. Autom. Detroit, Michigan May 1999 -, no. May, pp. 368–374, 1999.

[38] T. Komatsu, “Dynamic Walking and Running of a Bipedal Robot Using Hybrid Central Pattern Generator Method,” IEEE Int. Conf. Mechatronics Autom. 2005, vol. 2, no. July, pp. 987-992 Vol. 2, 2005.

[39] T. Reil and P. Husbands, “Evolution of Central Pattern Generators for Bipedal Walking in a Real-Time Physics Environment,” IEEE Trans. Evol. Comput. VOL. 6, NO. 2, April 2002, vol. 6, no. 2, pp. 159–168, 2002.

[40] J. J. Kuffner, “Footstep Planning Among Obstacles for Biped Robots,” Proeocdings 2001 IEEE/RSJ Int. Conf. Intell. Robot. Syst. Maui, Hawaii, USA, pp. 500–505, 2001.

[41] D. Tlalolini, C. Chevallereau, and Y. Aoustin, “Human-Like Walking : Optimal Motion of a Bipedal Robot With Toe-Rotation Motion,” IEEE/ASME Trans. MECHATRONICS, vol. 16, no. 2, pp. 310–320, 2011.

[42] M. Rostami and G. Bessonnet, “Impactless sagittal gait of a biped robot during the single support phase,” Proc. 1998 IEEE Int. Conf. Robot. L? Autom., no. May, pp. 1385–1391, 1998.

[43] F. R. Copyright, F. S. Company, and F. E. Only, “Integrated Motion Control,” Proc. lthe 2004 IEEE lnternatlonal Conf. Robot. Autom., no. C, pp. 2005–2007, 2007.

[44] D. G. Caldwel and C. J. Bowler, “Investigation of Bipedal Robot Locomotion using Pneumatic Muscle Actuators,” Proc. 1997 IEEE Int. Conf. Robot. Autom., 1997.

[45] J. W. Grizzle, J. Hurst, B. Morris, H. Park, and K. Sreenath, “MABEL , A New Robotic Bipedal Walker and Runner,” 2009 Am. Control Conf., pp. 2030–2036, 2009.

[46] Y. Kuroki, K. Kato, K. Nagasaka, A. Miyamoto, K. Ueno, and J. Yamaguchi, “Motion creating system for a small biped entertainment robot,” no. October, pp. 3809-3814 Vol.4, 2004.

[47] Y. Huang et al., “Step length and velocity control of a dynamic bipedal walking robot with adaptable compliant joints,” IEEE/ASME Trans. Mechatronics, vol. 18, no. 2, pp. 598–611, 2013

[48] G. Garofalo, C. Ott, and A. Albu-Schäffer, “Walking control of fully actuated robots based on the bipedal SLIP model,” *Proc. - IEEE Int. Conf. Robot. Autom.*, pp. 1456–1463, 2012.

[49] Z. Yu et al., “Design and Development of the Humanoid Robot BHR-5,” Adv. Mech. Eng., vol. 2014

[50] C. Borst et al., “A humanoid upper body system for two-handed manipulation,” Proc. - IEEE Int. Conf. Robot. Autom., no. April, pp. 2766–2767, 2007.

[51] N. Kanehira et al., “Design and experiments of advanced leg module (HRP-2L) for humanoid robot (HRP-2) development,” vol. 2, no. October, pp. 2455–2460, 2003.

[52] J. H. Oh, D. Hanson, W. S. Kim, I. Y. Han, J. Y. Kim, and I. W. Park, “Design of android type humanoid robot Albert HUBO,” IEEE Int. Conf. Intell. Robot. Syst., pp. 1428–1433, 2006.

[53] H. Iwata and S. Sugano, “Design of human symbiotic robot TWENDY-ONE,” Proc. - IEEE Int. Conf. Robot. Autom., pp. 580–586, 2009.

[54] N. Fukaya and S. Toyama, “Design of the TUAT / Karlsruhe Humanoid Hand,” Proc. 2000 IEEE/RSJ Int. Conf. Intell. Robot. Syst., no. March, pp. 1754–1759, 2000.

[55] Y. Ogura et al., “Development of a new humanoid robot WABIAN-2,” Proc. - IEEE Int. Conf. Robot. Autom., vol. 2006, no. May, pp. 76–81, 2006.

[56] K. Akachi et al., “Development of humanoid robot HRP-3P,” Proc. 2005 5th IEEE-RAS Int. Conf. Humanoid Robot., vol. 2005, pp. 50–55, 2005.

[57] I. Ha, Y. Tamura, H. Asama, J. Han, and D. W. Hong, “Development of open humanoid platform DARwIn-OP,” Annu. Conf. Soc. Instrum. Control Eng. Japan, pp. 2178–2181, 2011.

[58] N. Endo et al., “Development of whole-body emotion expression humanoid robot,” Proc. - IEEE Int. Conf. Robot. Autom., pp. 2140–2145, 2008.

[59] S. Lohmeier, T. Buschmann, and H. Ulbrich, “Humanoid robot LOLA,” Proc. - IEEE Int. Conf. Robot. Autom., pp. 775–780, 2009.

[60] I. W. Park, J. Y. Kim, J. Lee, and J. H. Oh, “Mechanical design of humanoid robot platform KHR-3 (KAIST humanoid robot - 3: HUBO),” Proc. 2005 5th IEEE-RAS Int. Conf. Humanoid Robot., vol. 2005, pp. 321–326, 2005.

[61] R. O. Ambrose et al., “Robonaut : NASA ’ s Space Humanoid,” 1973.

[62] F. Yamasaki, T. Matsui, T. Miyashita, and H. Kitano, “PINO The Humanoid that Walk,” Proc. First IEEE-RAS Int. Conf. Humanoid Robot., 2000.

[63] R. Tellez et al., “Reem-B: an autonomous lightweight human-size humanoid robot,” 2008 8th IEEE-RAS Int. Conf. Humanoid Robot. December 1 -- 3, 2008/ Daejeon, Korea WP1-25, pp. 462–468, 2008.

[64] N. G. Tsagarakis, Z. Li, J. Saglia, and D. G. Caldwell, “The design of the lower body of the compliant humanoid robot ‘cCub,’” Proc. - IEEE Int. Conf. Robot. Autom., pp. 2035–2040, 2011.

[65] M. Gienger, K. Loffler, and F. Pfeiffer, “Towards the design of a biped jogging robot,” pp. 4140–4145, 2002.

[66] Z. Mohamed and G. Capi, “Development of a New Mobile Humanoid Robot for Assisting Elderly People,” vol. 41, no. Iris, pp. 345–351, 2012.

[67] G. Wyeth, D. Kee, M. Wagstaff, N. Brewer, and P. Art, “Design of an Autonomous Humanoid Robot,” Proc.2001 Aust. Conf. Robot. Autom., no. November, pp. 14–15, 2001.

[68] J. Kim et al., “Development of the Lower Limbs for a Humanoid Robot,” 2012 IEEE/RSJ Int. Conf. Intell. Robot. Syst., pp. 4000–4005, 2012.

[69] C. Burghart et al., “A Cognitive Architecture for a Humanoid Robot : A First Approach,” 5th IEEE-RAS Int. Conf. Humanoid Robot. 2005., pp. 357–362, 2005.

[70] T. Kanda, I. Hiroshi, M. Imai, T. Ono, and K. Mase, “A constructive approach for developing interactive humanoid robots,” Proc. 2002 IEEElRSJ Intl. Conf. Intell. Robot. Syst. EPFL, Lausanne, Switz., no. October, pp. 1265–1270, 2002.

[71] J. K. Rosenblatt and D. W. Payton, “A Fine-Grained Alternative to the Subsumption Architecture for Mobile Robot Control,” Hughes Artif. Intell. Cent. Hughes Res. Lab. 301 1 Malibu Canyon Road Malibu, Calif. 90265.

[72] R. A. Brooks, “A HARDWARE RETARGETABLE DISTRIBUTED LAYERED ARCHITECTURE FOR MOBILE ROBOT CONTROL,” CH2413-3/87/3000/0106~3~.03 0 1987 IEEE, 1987.

[73] D. N. Ly, K. Regenstein, and T. Asfour, “A Modular and Distributed Embedded Control Architecture for Humanoid Robots,” Proc. 01 2004 IEEWRSJ lnternstional Conf. IntelligentRobots Syst. Sendai, Japan, pp. 2775–2780, 2004.

[74] Y. Wang and S. E. Butner, “A New Architecture for Robot Control,” CH2413-3/87/0000/0664$01.00 Q 1987 IEEE 664, no. 08421415, pp. 664–670, 1987.

[75] Y. Kuroki, M. Fujita, T. Ishida, K. Nagasaka, and J. Y. I, “A Small Biped Entertainment Robot Exploring Attractive Applications,” Proc. 2003 IEEE lnternatiooal Conf. Robot. Autom., pp. 471–476, 2003.

[76] R. Simmons and D. Apfelbaum, “Task Description Language for Robot Control,” Proc. 1998 IEEEYRSJ Intl. Conf. Intell. Robot. Syst., no. October, pp. 1931–1937, 1998.

[77] O. Khatib, “A Unified Approach for Motion and Force Control of Robot Manipulators: The Operational Space Formulation,” IEEE J. Robot. Autom., vol. 3, no. 1, pp. 43–53, 1987.

[78] N. Mansard, O. Stasse, P. Evrard, and A. Kheddar, “A versatile Generalized Inverted Kinematics implementation for collaborative working humanoid robots: The Stack Of Tasks,” Adv. Robot. 2009. ICAR 2009. Int. Conf., no. 8, pp. 1–6, 2009.

[79] J. L. Posadas, J. L. Poza, J. E. Simó, G. Benet, and F. Blanes, “Agent-based distributed architecture for mobile robot control,” Eng. Appl. Artif. Intell., vol. 21, no. 6, pp. 805–823, 2008.

[80] S. Erhart, D. Sieber, and S. Hirche, “An impedance-based control architecture for multi-robot cooperative dual-arm mobile manipulation,” IEEE Int. Conf. Intell. Robot. Syst., pp. 315–322, 2013.

[81] T. Asfour et al., “ARMAR-III: An integrated humanoid platform for sensory-motor control,” Proc. 2006 6th IEEE-RAS Int. Conf. Humanoid Robot. HUMANOIDS, pp. 169–175, 2006.

[82] D. Feil-Seifer and M. J. Matarić, “B3IA: A control architecture for autonomous robot-assisted behavior intervention for children with autism spectrum disorders,” Proc. 17th IEEE Int. Symp. Robot Hum. Interact. Commun. RO-MAN, pp. 328–333, 2008.

[83] C. Galindo, J. Gonzalez, and J. A. Fernández-Madrigal, “Control architecture for human-robot integration: Application to a robotic wheelchair,” IEEE Trans. Syst. Man, Cybern. Part B Cybern., vol. 36, no. 5, pp. 1053–1067, 2006.

[84] M. Naumann, K. Wegener, R. D. Schraft, and F. Ipa, “Control Architecture for Robot Cells to Enable Plug ’ n ’ Produce,” IEEE Int. Conf. Robot. Autom., no. April, pp. 10–14, 2007.

[85] H. Liu, T. Iberall, and G. A. Bekey, “Neural Network Architecture for Robot Hand Control,” IEEE Control Syst. Mag., vol. 5, no. April, pp. 183–190, 1989.

[86] J. Y. Kim, I. W. Park, J. Lee, M. S. Kim, B. K. Cho, and J. H. Oh, “System design and dynamic walking of humanoid robot KHR-2,” Proc. - IEEE Int. Conf. Robot. Autom., vol. 2005, no. April, pp. 1431–1436, 2005.

[87] P. Yokohama and T. Takashima, “Open Architecture Humanoid Robotics Platform,” Proc. 2002 IEEE Int. Conf. Robot. Autom. Washington, DC, pp. 24–30, 2002.

[88] E. Rohmer, S. P. N. Singh, and M. Freese, “V-REP: A versatile and scalable robot simulation framework,” IEEE Int. Conf. Intell. Robot. Syst., pp. 1321–1326, 2013.

[89] G. Velentzas, T. Tsitsimis, I. Rañó, C. Tzafestas, and M. Khamassi, “Adaptive reinforcement learning with active state-specific exploration for engagement maximization during simulated child-robot interaction,” Paladyn, vol. 9, no. 1, pp. 235–253, 2018.

[90] D. Katic and M. Vukobratovic, “Intelligent control techniques for humanoid robots,” Robot. Lab. Mihailo Pupin Inst. P.O.Box 15, Volgina 15, 11000 Belgrade, Yugosl., pp. 1839–1844.

[91] J. Kober and J. Peters, “PolicySearchforMotorPrimitivesinRobotics,” Max Planck Inst. Biol. Cybern. Spemannstr.38 72076 Tübingen, Ger., vol. 2, no. 1, pp. 87–99, 2006.

[92] U. Kartoun, H. Stern, and Y. Edan, “A human-robot collaborative reinforcement learning algorithm,” J. Intell. Robot. Syst. Theory Appl., vol. 60, no. 2, pp. 217–239, 2010.

[93] P. Wawrzyński, “Autonomous reinforcement learning with experience replay for humanoid gait optimization,” Procedia Comput. Sci., vol. 13, pp. 205–211, 2012.

[94] D. Kati and M. Vukobratovi, “Control Algorithm for Humanoid Walking Based on Fuzzy Reinforcement Learning Model of the Robot ’ s Mechanism,” SISY 2006 • 4th Serbian-Hungarian Jt. Symp. Intell. Syst., pp. 81–93, 2006.

[95] M. Frank, J. Leitner, M. Stollenga, A. Forster, and J. Schmidhuber, “Curiosity driven reinforcement learning for motion planning on humanoids,” Front. Neurorobot., vol. 7, no. JAN, pp. 1–15, 2014.

[96] S. Christen and S. Stevˇ, “Demonstration-Guided Deep Reinforcement Learning of Control Policies for Dexterous Human-Robot Interaction,” AIT Lab, Dep. Comput. Sci. ETH Zurich, 8092 Zurich, Switz., no. i, pp. 2161–2167, 2019.

[97] R. Lober et al., “Efficient Reinforcement Learning for Humanoid Whole-Body Control,” IEEE-RAS Int. Conf. Humanoid Robot. Nov 2016, Cancun, Mex. hal-01377831 HAL, 2016.

[98] T. Hester, M. Quinlan, and P. Stone, “Generalized model learning for reinforcement learning on a humanoid robot,” Proc. - IEEE Int. Conf. Robot. Autom., no. May, pp. 2369–2374, 2010.

[99] M. Riedmiller, T. Gabel, R. Hafner, and S. Lange, “Reinforcement learning for robot soccer,” Auton. Robots, vol. 27, no. 1, pp. 55–73, 2009.

[100] S. Iida, “Humanoid Robot Control Based on Reinforcement Learning,” Micro-Nanomechatronics Hum. Sci. 2004 Fourth Symp. Micro-Nanomechatronics Information-Based Soc. 2004., pp. 353–358, 2004.

[101] D. U. Sko, “Reinforcement learning control algorithm for humanoid robot walking,” Int. J. Inf. Syst. Sci., vol. 4, no. 2, pp. 256–267, 2008.

[102] M. Danel, “Reinforcement Learning for Humanoid Robot Control,” POSTER,PRAGUE, pp. 1–5, 2017.

[103] J. Peters, S. Vijayakumar, and S. Schaal, “Reinforcement Learning for Humanoid Robotics,” Third IEEE-RAS Int. Conf. Humanoid Robot. Karlsruhe, Ger., 2003.

[104] F. Guenter, M. Hersch, S. Calinon, and A. Billard, “Reinforcement learning for imitating constrained reaching movements,” Adv. Robot., vol. 21, no. 13, pp. 1521–1544, 2007.

[105] K. Lowrey, S. Kolev, J. Dao, A. Rajeswaran, and E. Todorov, “Reinforcement learning for non-prehensile manipulation: Transfer from simulation to physical system,” 2018 IEEE Int. Conf. Simulation, Model. Program. Auton. Robot. SIMPAR 2018, pp. 35–42, 2018.

[106] J. Kober, E. Oztop, and J. Peters, “Reinforcement learning to adjust robot movements to new situations,” IJCAI Int. Jt. Conf. Artif. Intell., pp. 2650–2655, 2011.

[107] I. J. Silva, D. H. Perico, A. H. Costa, and R. A. Bianchi, “Using Reinforcement Learning To Optimize Gait Generation,” XIII Simp´osio Bras. Autom. Intel., pp. 288–294, 2017.

[108] S. K. Kim, E. A. Kirchner, A. Stefes, and F. Kirchner, “Intrinsic interactive reinforcement learning-Using error-related potentials for real world human-robot interaction,” Sci. Rep., vol. 7, no. 1, pp. 1–16, 2017.

[109] S. Agarwal, S. Hyder, H. Zaidi, and S. K. Agarwal, “Correlation Of Body Height By Foot Length And Knee Height Measurements In Population Of North India,” Inter, vol. 3, no. 3, pp. 1225–1229, 2015.

[110] M. Cenciarini and A. M. Dollar, “Biomechanical considerations in the design of lower limb exoskeletons,” *IEEE Int. Conf. Rehabil. Robot.*, pp. 1–6, 2011.

[111] C. Hernández-Santos, E. Rodriguez-Leal, R. Soto, and J. L. Gordillo, “Kinematics and dynamics of a new 16 DOF humanoid biped robot with active toe joint,” Int. J. Adv. Robot. Syst., vol. 9, 2012.

[112] M. Silva, R. Barbosa, and T. Castro, “Multi-legged walking robot modelling in MATLAB/simmechanics TM and its simulation,” Proc. - 8th EUROSIM Congr. Model. Simulation, EUROSIM 2013, pp. 226–231, 2015.

[113] H. Dallali et al., “Development of a dynamic simulator for a compliant humanoid robot based on a symbolic multibody approach,” 2013 IEEE Int. Conf. Mechatronics, ICM 2013, pp. 598–603, 2013.

[114] P. I. Corke, “A Simple and Systematic Approach to Assigning Denavit–Hartenberg Parameters Abs,” IEEE Trans. Robot., vol. 48, no. 4, pp. 766–769, 2001.

[115] D. Bharadwaj and M. Prateek, “Kinematics and Dynamics of Lower Body of Autonomous Humanoid Biped Robot,” no. 4, pp. 141–146, 2019.

[116] Pyung Chang, “A closed-form solution for inverse kinematics of robot manipulators with redundancy,” IEEE J. Robot. Autom., vol. 3, no. 5, pp. 393–403, 2009.

[117] J. P. Merlet, “Jacobian, Manipulability, Condition Number, and Accuracy of Parallel Robots,” J. Mech. Des., vol. 128, no. 1, p. 199, 2006.

[118] F. C. Park and J. W. Kim, “Manipulability and singularity analysis of multiple robot systems: A geometric approach,” Proc. - IEEE Int. Conf. Robot. Autom., vol. 2, no. May, pp. 1032–1037, 1998.

[119] Y. Nakamura and M. Ghodoussi, “Dynamics Computation of Closed-Link Robot Mechanisms with Nonredundant and Redundant actuators,” IEE Trans. Robot. Autom., vol. 5, p. 194, 1989.

[120] A. Piazzi and A. Visioli, “Global minimum-jerk trajectory planning of robot manipulators,” IEEE Trans. Ind. Electron., vol. 47, no. 1, pp. 140–149, 2000.

[121] S. Kajita, H. Hirukawa, K. Harada, and K. Yokoi, Introduction to Humanoids. Springer Tracts in Advanced Robotics.

[122] K. Oussama and B. Joel, “Motion and Force Control of Robot Manipulators Oussama Khatib and Joel Burdick Artificial Intelligence Laboratory Stanford University Computer Science Department,” Architecture, no. 5, pp. 1381–1386, 1986.

[123] H. Kensuke, Y. Eiichi, and Y. Kazuhito, Motion Planning For Humnaoid Robots. Springer.

[124] A. B. Bishop, “Second-Order Systems,” Introd. to Discret. Linear Control., no. 0, pp. 278–329, 2014.

[125] J. H. Park and K. D. Kim, “Biped robot walking using gravity-compensated inverted pendulum mode and computed torque control,” Proc. - IEEE Int. Conf. Robot. Autom., vol. 4, no. May, pp. 3528–3533, 1998.

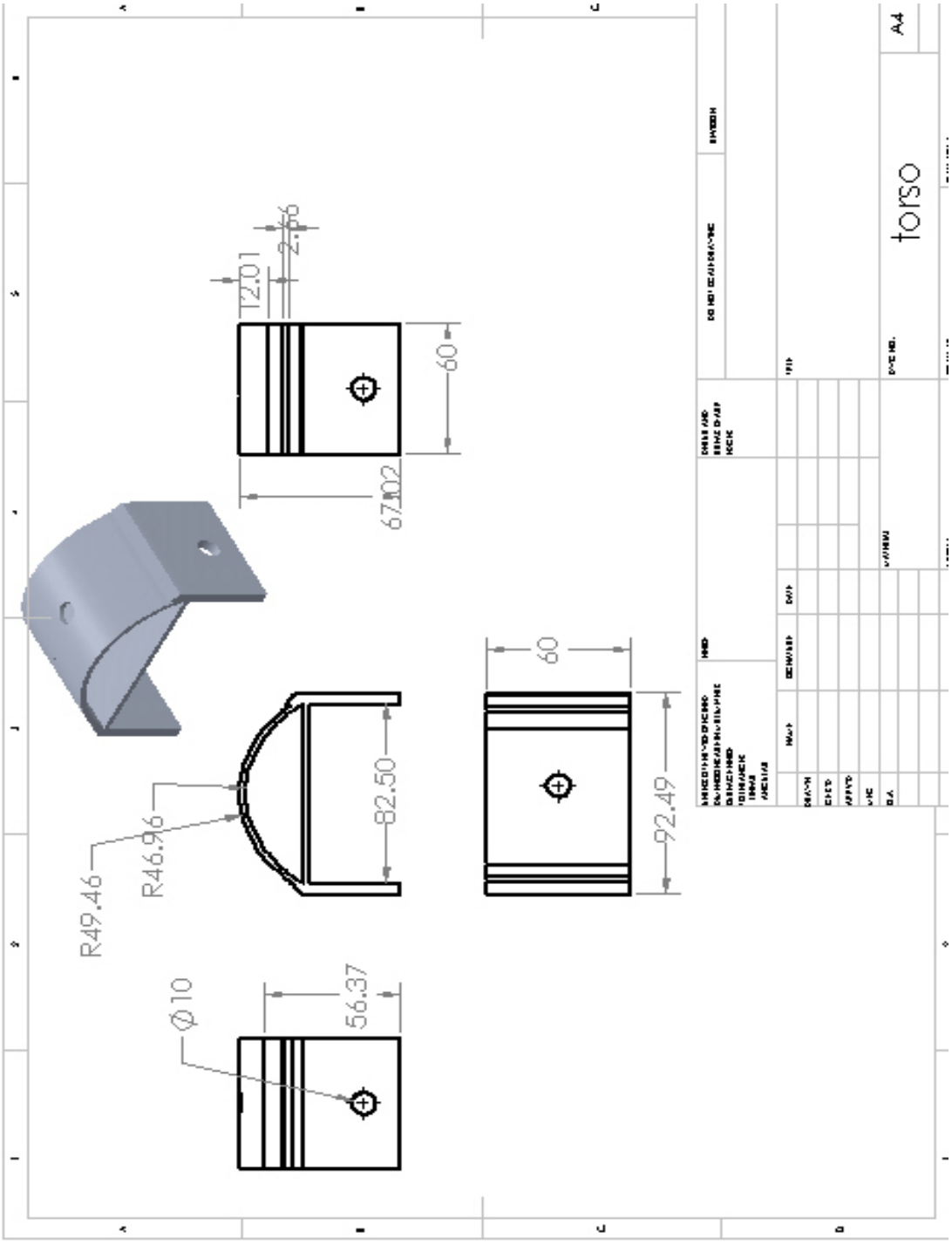
[126] R. S. & B. Sutton, “Reinforcement Learning: An Introduction,” 1998.

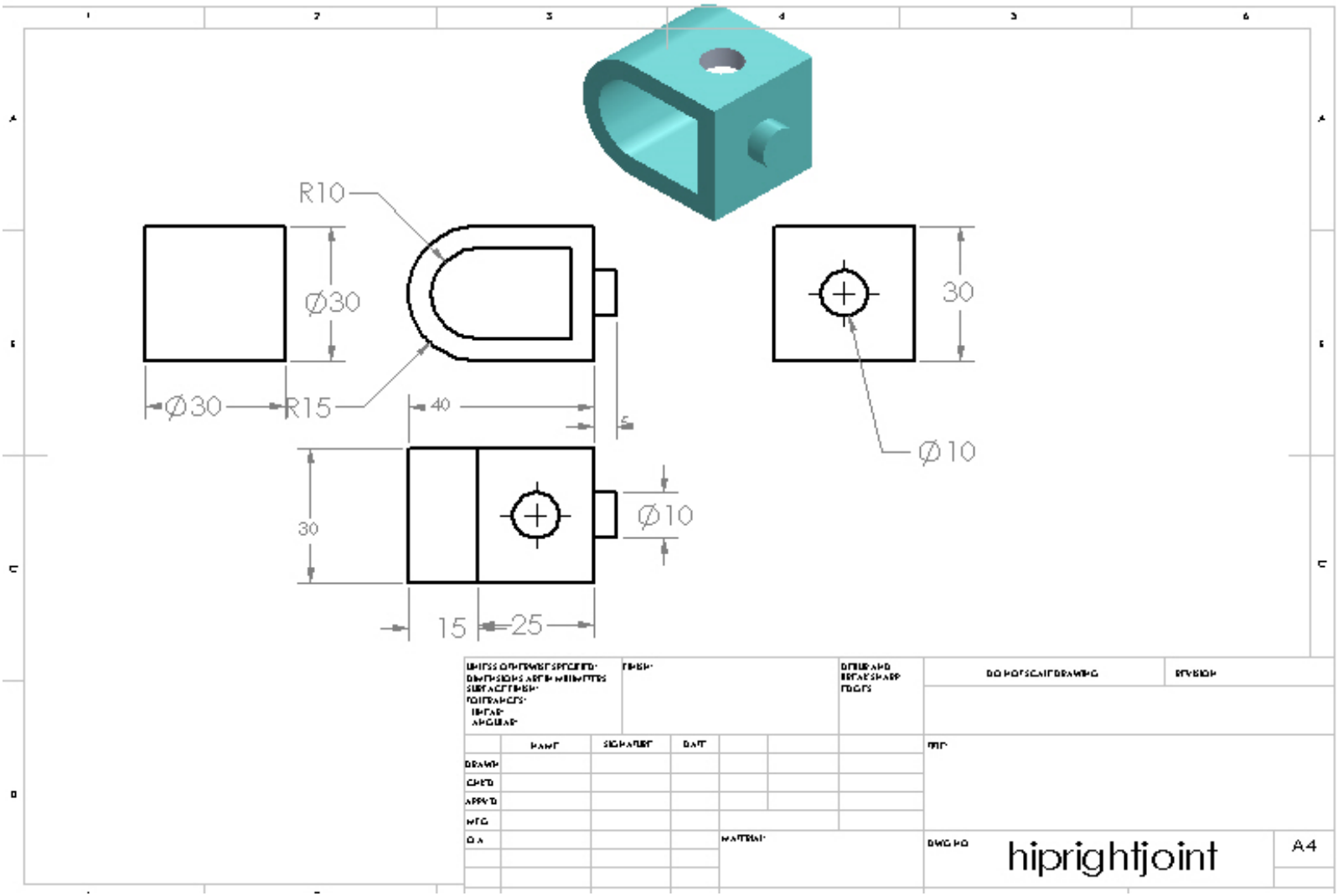
[127] D. Bharadwaj, M. Prateek, and R. Sharma, “Development of Reinforcement Control Algorithm of Lower Body of Autonomous Humanoid Robot,” no. 1, pp. 915–919, 2019.

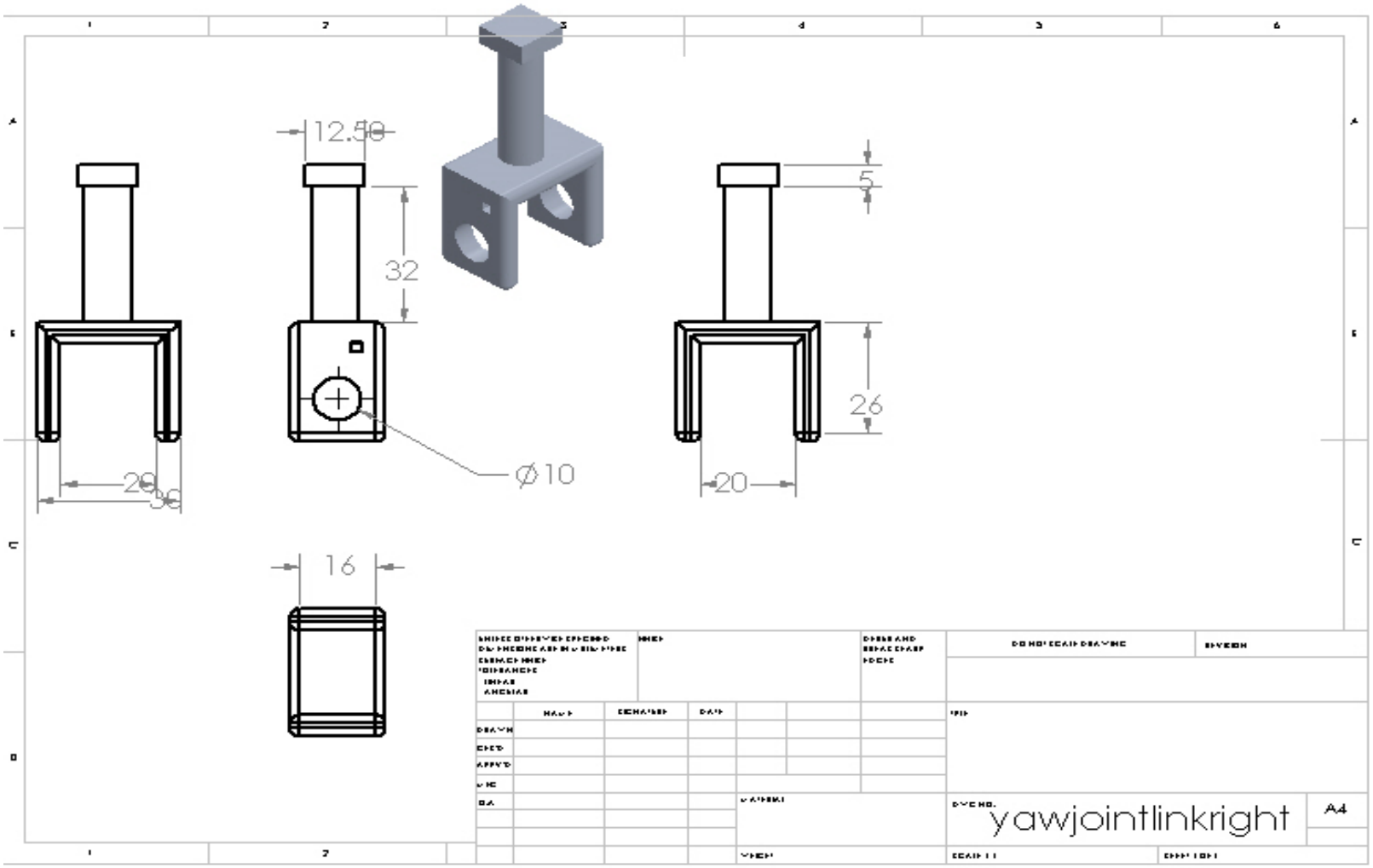
[128] R. Sharma, I. Singh, D. Bharadwaj, and M. Prateek, “Incorporating Forgetting Mechanism in Q-learning Algorithm for Locomotion of Bipedal Walking Robot,” no. 7, pp. 1782–1787, 2019.

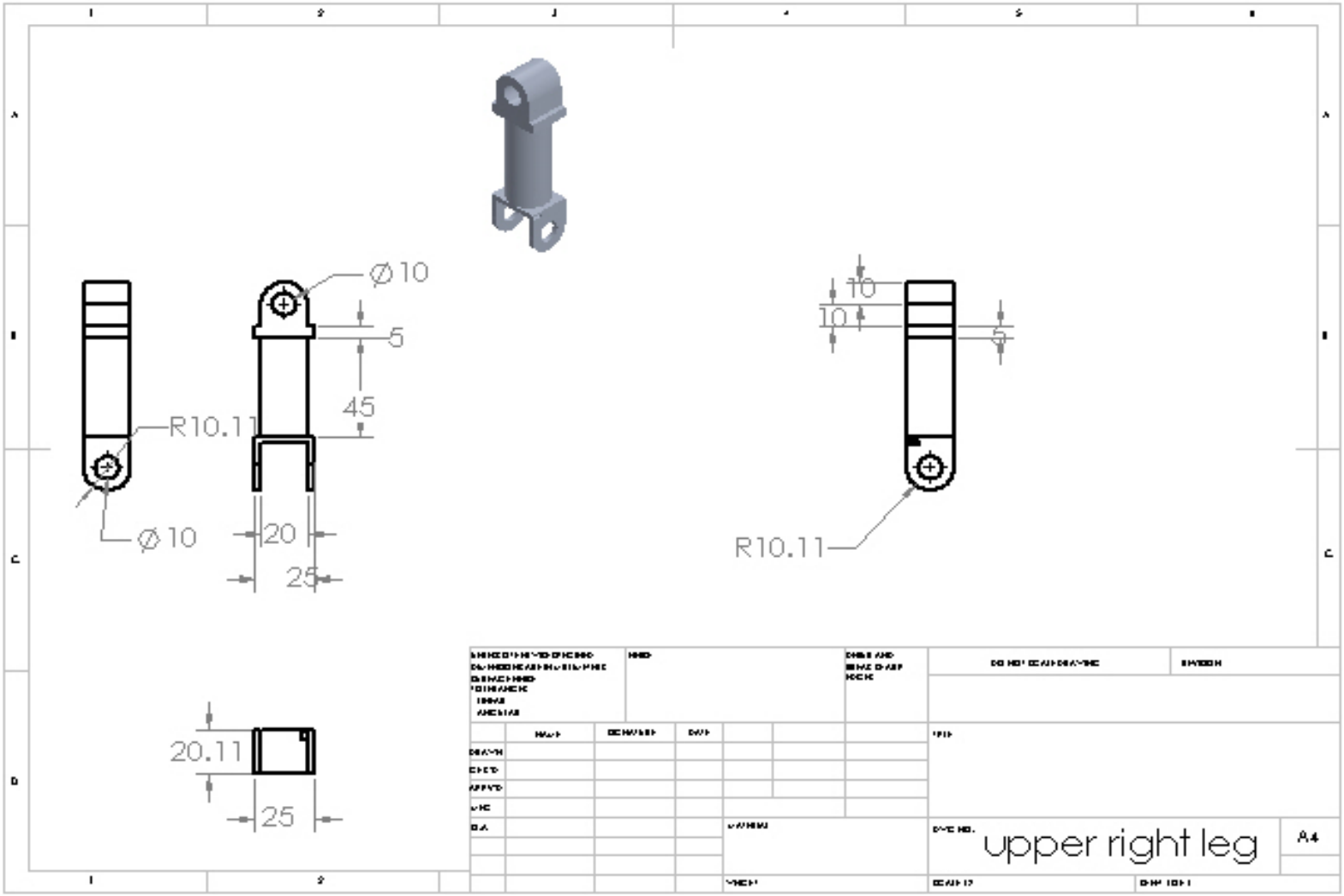
APPENDIX A

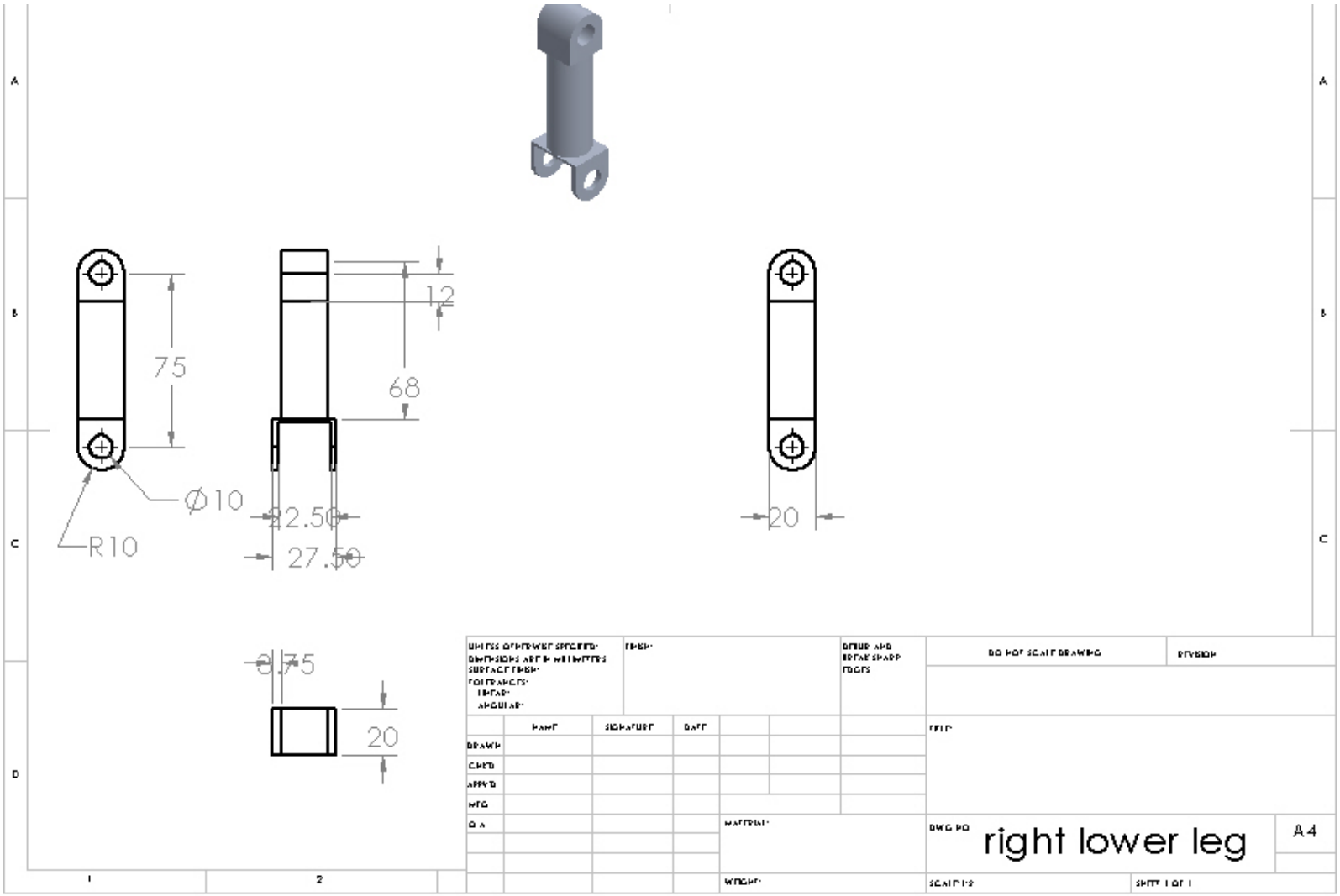
Engineering drawing of the proposed model

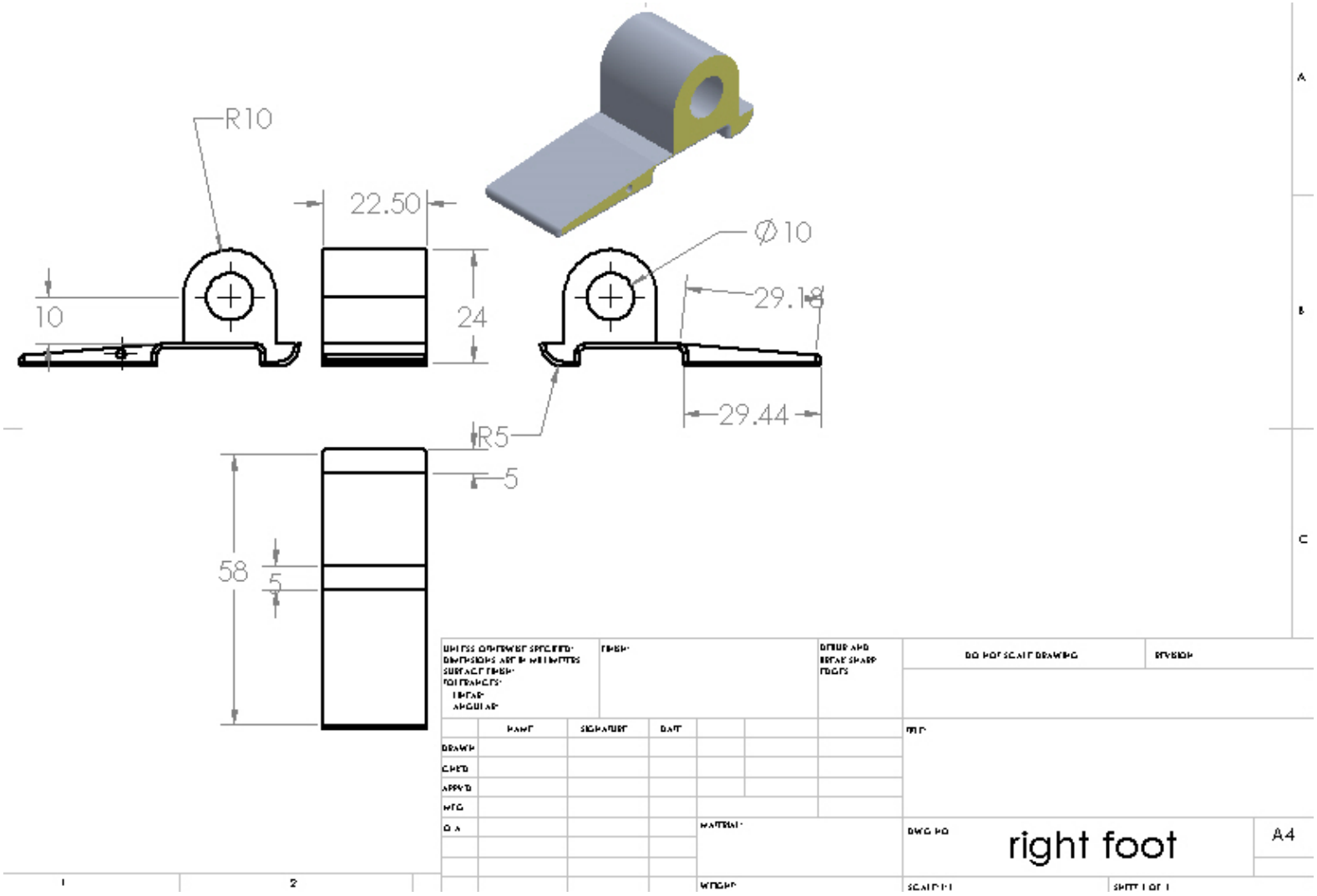



R10
Ø30
Ø30
R15
40
5
30
Ø10
30
Ø10
15
25
DO NOT SCALE DRAWING
REVISION
DRAWN
CHK'D
APPV'D
MFG
Q.A
MATERIAL
DWG NO
hiprightjoint
A4


12.50
32
5
26
Ø10
20
16
yawjointlinkright
A4

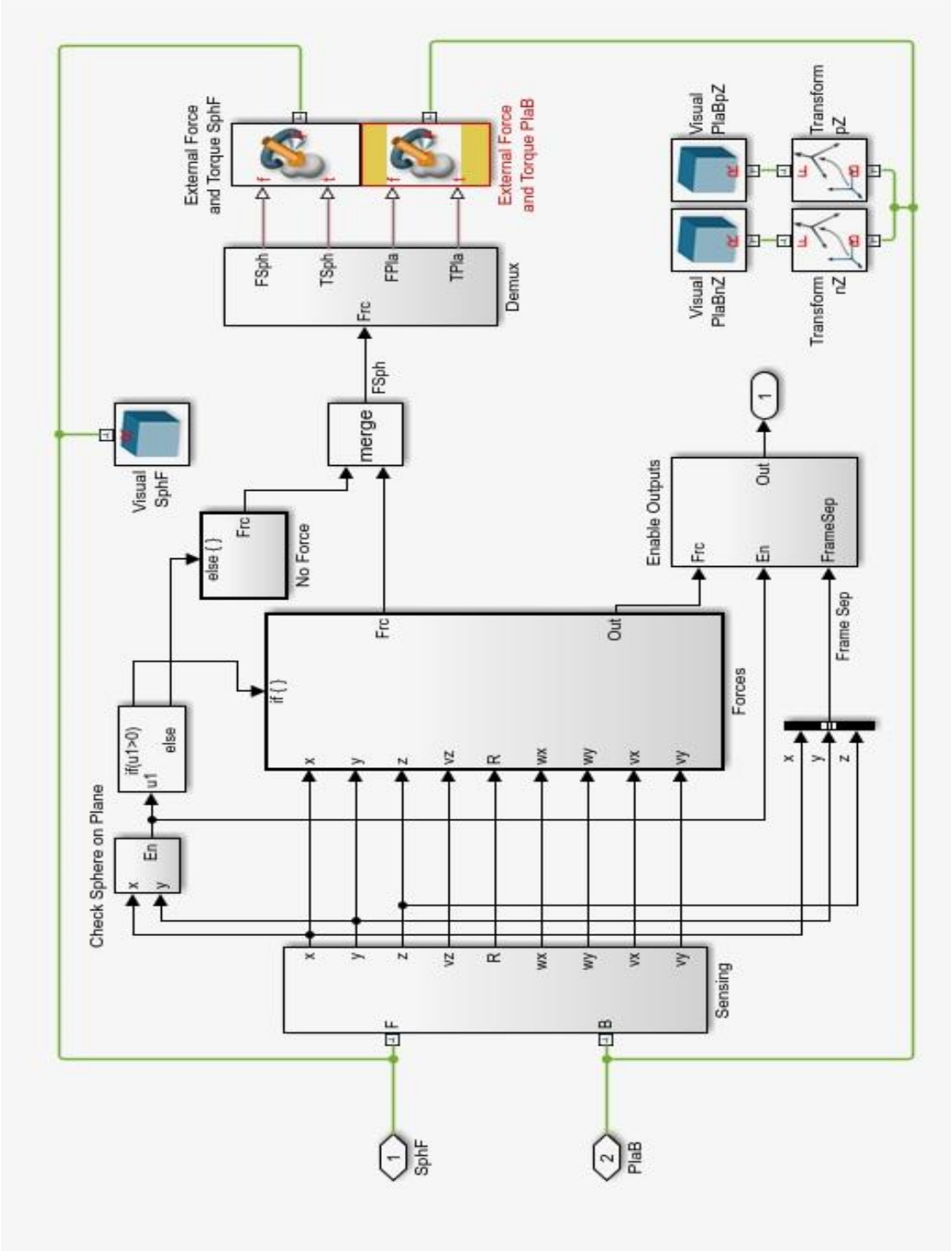

Ø10
5
45
R10.11
Ø10
20
25
10
10
5
R10.11
20.11
25
upper right leg
A4


12
75
68
Ø10
R10
2.50
27.50
20
8.75
20
right lower leg
A4


R10
22.50
Ø10
10
24
29.18
29.44
R5
5
58
5
A
B
C
1
2
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MILLIMETERS
SURFACE FINISH:
TOLERANCES:
LINEAR:
ANGULAR:
FINISH:
DEBUR AND
BREAK SHARP
EDGES
DO NOT SCALE DRAWING
REVISION
NAME
SIGNATURE
DATE
DRAWN
CHKD
APPV'D
MFG
Q.A
MATERIAL:
WEIGHT:
DWG NO
right foot
A4
SCALE:1:1
SHEET 1 OF 1

APPENDIX B

Simulink block of ground force contact



APPENDIX C

Calculation of Jacobian

The jacobian matrix column J_1 for joint 1 ,which is a rotary joint ,is determined as follows:

The joint axis vector P_0(P_{i-1} for i=1)

$P_0={}^0_0R\hat{u}$ where $\hat{u}$=[0 0 1]

The transformation matrix ${}^0_0T$ and rotation matrix ${}^0_0R$ are the identity matrix . Thus

$$P_0=\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}\begin{bmatrix}0\\0\\1\end{bmatrix}=\begin{bmatrix}0\\0\\1\end{bmatrix}$$

The toe position vector (for i=1 and n=5) is determined as follows

$${}^0_5P={}^O_5TO_n-\ {}^O_0TO_n={}^O_5T\begin{bmatrix}0\\0\\0\\1\end{bmatrix}-\ {}^O_0T\begin{bmatrix}0\\0\\0\\1\end{bmatrix}$$

The jacobian for joint 1 can be computes as follows

$$J_1(q)=\begin{bmatrix}P_0X\ {}^0_5P\\P_0\end{bmatrix}=$$

$$\begin{bmatrix}\{l_2C\Theta_1+l_4C\Theta_4(C\Theta_1C\Theta_3+S\Theta_1S\Theta_2S\Theta_3)+l_3C\Theta_1C\Theta_3+l_5C\Theta_5(C\Theta_4(C\Theta_1C\Theta_3+S\Theta_1S\Theta_2S\Theta_3)+C\Theta_2S\Theta_1S\Theta_4)\\-l_5S\Theta_5(S\Theta_4(C\Theta_1C\Theta_3+S\Theta_1S\Theta_2S\Theta_3)-C\Theta_2C\Theta_4S\Theta_1)+l_4C\Theta_2S\Theta_1S\Theta_4+l_3S\Theta_1S\Theta_2S\Theta_3\}\\\{l_2S\Theta_1+l_4C\Theta_4(C\Theta_3S\Theta_1-C\Theta_1S\Theta_2S\Theta_3)+l_3C\Theta_3S\Theta_1+l_5C\Theta_5(C\Theta_4(C\Theta_3S\Theta_1-C\Theta_1S\Theta_2S\Theta_3)-C\Theta_1C\Theta_2S\Theta_4)\\-l_5S\Theta_5(S\Theta_4(C\Theta_3S\Theta_1-C\Theta_1S\Theta_2S\Theta_3)+C\Theta_1C\Theta_2C\Theta_4)-l_4C\Theta_1C\Theta_2S\Theta_4-l_3C\Theta_1S\Theta_2S\Theta_3\}\\0\\0\\0\\1\end{bmatrix}$$

The jacobian matrix column J_2 for joint 2 ,which is a rotary joint ,is determined as follows:

The joint axis vector P_1(P_{i-1} for i=2)

$$P_1 = {}^0_1R\hat{u} = \begin{bmatrix} C\Theta_1 & 0 & S\Theta_1 \\ S\Theta_1 & 0 & -C\Theta_1 \\ 0 & 0 & 0 \end{bmatrix}\begin{bmatrix}0\\0\\1\end{bmatrix} = \begin{bmatrix} S\Theta_1 \\ -C\Theta_1 \\ 0 \end{bmatrix}$$

The toe position vector (for i=2 and n=5) is determined as follows

$${}^1_5P = {}^O_5TO_n - {}^O_1TO_n = {}^O_5T\begin{bmatrix}0\\0\\0\\1\end{bmatrix} - {}^O_1T\begin{bmatrix}0\\0\\0\\1\end{bmatrix}$$

The jacobian for joint 2 can be computes as follows

$$J_2(q) = \begin{bmatrix} P_1X\ {}^1_5P \\ P_1 \end{bmatrix} =$$

$$\begin{bmatrix} l_5C_5(C_1\,S_2S_4 - C_1C_2C_4S_3) + l_5S_5(C_1C_4S_2 + C_1C_2S_3S_4) - l_3C_1C_2S_3 + l_4C_1S_2S_4 - l_4C_1C_2C_4S_3 \\ l_5S_5(C_4\,S_1S_2 + C_2S_1S_3S_4) + l_5C_5(S_1S_2S_4 - C_2C_4S_1S_3) - l_3C_2S_1S_3 + l_4S_1S_2S_4 - l_4C_2C_4S_1S_3 \\ -l_5C_5(C_2\,S_4 + C_4S_2S_3) - l_5S_5(C_2C_4 - S_2S_3S_4) - l_4C_2S_4 - l_3S_2S_3 - l_4C_4S_2S_3 \\ S\Theta_1 \\ -C\Theta_1 \\ 0 \end{bmatrix}$$

The jacobian matrix column J_3 for joint 3 ,which is a rotary joint ,is determined as follows:

The joint axis vector P_2(P_{i-1} for i=3)

$$P_2 = {}^0_2R\hat{u} = \begin{bmatrix} -C\Theta_1C\Theta_2 \\ -C\Theta_2S\Theta_1 \\ -S\Theta_2 \end{bmatrix}$$

The toe position vector (for i=3 and n=5) is determined as follows

$${}^2_5P = {}^O_5TO_n - {}^O_2TO_n = {}^O_5T\begin{bmatrix}0\\0\\0\\1\end{bmatrix} - {}^O_2T\begin{bmatrix}0\\0\\0\\1\end{bmatrix}$$

The jacobian for joint 3 can be computes as follows

$$J_3(q) = \begin{bmatrix} P_2X\ {}^2_5P \\ P_2 \end{bmatrix} =$$

$$\begin{bmatrix} l_5S_4S_5(S_1S_3 + C_1C_3S_2) - l_3S_1S_3 - l_5C_1C_5(S_1S_3 + C_1C_3S_2) - l_4C_4(S_1S_3 + C_1C_3S_2) - l_3C_1C_3S_2 \\ l_4C_4(C_1S_3 - C_3S_1S_2) + l_3C_1S_3 + l_5C_4C_5(C_1S_3 - C_3S_1S_2) - l_5S_4S_5(C_1S_3 - C_3S_1S_2) - l_3C_3S_1S_2 \\ l_3C_2C_3 + l_4C_2C_3C_4 + l_5C_2C_3C_4C_5 - l_5C_2C_3S_4S_5 \\ -C\Theta_1C\Theta_2 \\ -C\Theta_2S\Theta_1 \\ -S\Theta_2 \end{bmatrix}$$

Similarly

$J_4(q) = \begin{bmatrix} P_3X \ {}^3_5P \\ P_3 \end{bmatrix} =$

$$\begin{bmatrix} -l_4S_4(C_3S_1 - C_1S_2S_3) - l_5C_5(S_4(C_3S_1 - C_1S_2S_3) + C_1C_2C_4) - l_5S_5(C_4(C_3S_1 - C_1S_2S_3) - C_1C_2S_4) - l_4C_1C_2C_4 \\ l_4S_4(C_1C_3 + S_1S_2S_3) + l_5C_5(S_4(C_1C_3 + S_1S_2S_3) - C_2C_4S_1) + l_5S_5(C_4(C_1C_3 + S_1S_2S_3) + C_2S_1S_4) - l_4C_2C_4S_1 \\ l_5S_5(S_2S_4 - C_2C_4S_3) - l_5C_5(C_4S_2 + C_2S_3S_4) - l_4C_4S_2 - l_4C_2S_3S_4 \\ S\Theta_1S\Theta_3 + C\Theta_1C\Theta_3S\Theta_2 \\ C\Theta_3S\Theta_1S\Theta_2 - C\Theta_1S\Theta_3 \\ -C\Theta_2C\Theta_3 \end{bmatrix}$$

$J_5(q) = \begin{bmatrix} P_4X \ {}^4_5P \\ P_4 \end{bmatrix} =$

$$\begin{bmatrix} -l_5C_5(S_4(C_3S_1 - C_1S_2S_3) + C_1C_2S_4) - l_5S_5(C_4(C_3S_1 - C_1S_2S_3) - C_1C_2S_4) \\ l_5C_5(S_4(C_1C_3 + S_1S_2S_3) - C_2C_4S_1) + l_5S_5(C_4(C_1C_3 + S_1S_2S_3) + C_2S_1S_4) \\ l_5S_5(S_2S_4 - C_2C_4S_3) - l_5C_5(C_4S_2 + C_2S_3S_4) \\ S\Theta_1S\Theta_3 + C\Theta_1C\Theta_3S\Theta_2 \\ C\Theta_3S\Theta_1S\Theta_2 - C\Theta_1S\Theta_3 \\ -C\Theta_2C\Theta_3 \end{bmatrix}$$

APPENDIX D

Calculation of stall torque

Assuming that the weight of the body $\boldsymbol{F}$ is acting at the torso of frame {0} and is given by

$F = [f_x \quad f_y \quad f_z \quad 0 \quad 0 \quad 0]^T$

The joint torques are a function of the configuration and applied forces and are given by equation as

$\tau_{static} = J(q)^T \boldsymbol{F}$

The torque on each joint can be obtained as

$\tau_{static} = [\tau_{\mathbf{1}static} \quad \tau_{\mathbf{2}static} \quad \tau_{\mathbf{3}static} \quad \tau_{\mathbf{4}static} \quad \tau_{\mathbf{5}static}]^T$

The mathematical expression obtained for the stall torque is described as

$\tau_{\mathbf{1}static} = \{l_2 C\Theta_1 + l_4 C\Theta_4(C\Theta_1 C\Theta_3 + S\Theta_1 S\Theta_2 S\Theta_3) + l_3 C\Theta_1 C\Theta_3 + l_5 C\Theta_5(C\Theta_4(C\Theta_1 C\Theta_3 + S\Theta_1 S\Theta_2 S\Theta_3) + C\Theta_2 S\Theta_1 S\Theta_4) - l_5 S\Theta_5(S\Theta_4(C\Theta_1 C\Theta_3 + S\Theta_1 S\Theta_2 S\Theta_3) - C\Theta_2 C\Theta_4 S\Theta_1) + l_4 C\Theta_2 S\Theta_1 S\Theta_4 + l_3 S\Theta_1 S\Theta_2 S\Theta_3\}\boldsymbol{f_x} + \{l_2 S\Theta_1 + l_4 C\Theta_4(C\Theta_3 S\Theta_1 - C\Theta_1 S\Theta_2 S\Theta_3) + l_3 C\Theta_3 S\Theta_1 + l_5 C\Theta_5(C\Theta_4(C\Theta_3 S\Theta_1 - C\Theta_1 S\Theta_2 S\Theta_3) - C\Theta_1 C\Theta_2 S\Theta_4) - l_5 S\Theta_5(S\Theta_4(C\Theta_3 S\Theta_1 - C\Theta_1 S\Theta_2 S\Theta_3) + C\Theta_1 C\Theta_2 C\Theta_4) - l_4 C\Theta_1 C\Theta_2 S\Theta_4 - l_3 C\Theta_1 S\Theta_2 S\Theta_3\}\boldsymbol{f_y}$

$\tau_{\mathbf{2}static} = \{ l_5 C_5(C_1 S_2 S_4 - C_1 C_2 C_4 S_3) + l_5 S_5(C_1 C_4 S_2 + C_1 C_2 S_3 S_4) - l_3 C_1 C_2 S_3 + l_4 C_1 S_2 S_4 - l_4 C_1 C_2 C_4 S_3 \} \boldsymbol{f_x} + \{ l_5 S_5(C_4 S_1 S_2 + C_2 S_1 S_3 S_4) + l_5 C_5(S_1 S_2 S_4 - C_2 C_4 S_1 S_3) - l_3 C_2 S_1 S_3 + l_4 S_1 S_2 S_4 - l_4 C_2 C_4 S_1 S_3 \} \boldsymbol{f_y} + \{ -l_5 C_5(C_2 S_4 + C_4 S_2 S_3) - l_5 S_5(C_2 C_4 - S_2 S_3 S_4) - l_4 C_2 S_4 - l_3 S_2 S_3 - l_4 C_4 S_2 S_3\}\boldsymbol{f_z}$

$\tau_{\mathbf{3}static} = \{ l_5 S_4 S_5(S_1 S_3 + C_1 C_3 S_2) - l_3 S_1 S_3 - l_5 C_1 C_5(S_1 S_3 + C_1 C_3 S_2) - l_4 C_4(S_1 S_3 + C_1 C_3 S_2) - l_3 C_1 C_3 S_2 \} \boldsymbol{f_x} + \{ l_4 C_4(C_1 S_3 - C_3 S_1 S_2) + l_3 C_1 S_3 +$

$l_5C_4C_5(C_1S_3 - C_3S_1S_2) - l_5S_4S_5(C_1S_3 - C_3S_1S_2) - l_3C_3S_1S_2\ \}\ \boldsymbol{f_y}\ +\{\ l_3C_2C_3 + l_4C_2C_3C_4 + l_5C_2C_3C_4C_5 - l_5C_2C_3S_4S_5\}\boldsymbol{f_z}$

$\tau_{\boldsymbol{4static}}\ =\ \{\ -l_4S_4(C_3S_1 - C_1S_2S_3) - l_5C_5(S_4(C_3S_1 - C_1S_2S_3) + C_1C_2C_4) - l_5S_5(C_4(C_3S_1 - C_1S_2S_3) - C_1C_2S_4) - l_4C_1C_2C_4\ \}\ \boldsymbol{f_x}\ +\{\ l_4S_4(C_1C_3 + S_1S_2S_3) + l_5C_5(S_4(C_1C_3 + S_1S_2S_3) - C_2C_4S_1) + l_5S_5(C_4(C_1C_3 + S_1S_2S_3) + C_2S_1S_4) - l_4C_2C_4S_1\ \}\ \boldsymbol{f_x}\ +\{\ l_5S_5(S_2S_4 - C_2C_4S_3) - l_5C_5(C_4S_2 + C_2S_3S_4) - l_4C_4S_2 - l_4C_2S_3S_4\}\boldsymbol{f_z}$

$\tau_{\boldsymbol{5static}}\ =\{\ -l_5C_5(S_4(C_3S_1 - C_1S_2S_3) + C_1C_2S_4) - l_5S_5(C_4(C_3S_1 - C_1S_2S_3) - C_1C_2S_4)\}\ \boldsymbol{f_x} + \{l_5C_5(S_4(C_1C_3 + S_1S_2S_3) - C_2C_4S_1) + l_5S_5(C_4(C_1C_3 + S_1S_2S_3) + C_2S_1S_4)\}\boldsymbol{f_y} + \{l_5S_5(S_2S_4 - C_2C_4S_3) - l_5C_5(C_4S_2 + C_2S_3S_4)\}\boldsymbol{f_z}$

APPENDIX E

Computation of d_{ij} matrix

The computation of matrices d_{ij}, which are required to compute all other coefficients.

For *i* ,*j*=1,2,3,4,5 the d_{ij} can be computed now.

$$d_{11} = {}^{0}_{0}T\ Q_1\ {}^{0}_{1}T = Q_1\ {}^{0}_{1}T = \begin{bmatrix} -S_1 & 0 & C_1 & 0 \\ C_1 & 0 & S_1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$d_{21} = {}^{0}_{0}T\ Q_1\ {}^{0}_{2}T = \begin{bmatrix} S_1S_2 & -C_1 & C_2S_1 & l_2C_1 \\ -C_1S_2 & -S_1 & -C_1C_2 & l_2S_1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$, \quad d_{31} = {}^{0}_{0}T\ Q_1\ {}^{0}_{3}T$$

$$= \begin{bmatrix} C_1C_3 + S_1S_2S_3 & C_2S_1 & C_1S_3 - C_3S_1S_2 & l_2C_1 + l_3C_1C_3 + l_3S_1S_2S_3 \\ C_3S_1 - C_1S_2S_3 & -C_1C_2 & S_1S_3 + C_1C_3S_2 & l_2S_1 + l_3C_3S_1 - l_3C_1S_2S_3 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$d_{41} = {}^{0}_{0}T\ Q_1\ {}^{0}_{4}T = \begin{bmatrix} a11 & b11 & c11 & d11 \\ e11 & f11 & g11 & h11 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$a11 = C_4(C_1C_3 + S_1S_2S_3) + C_2S_1S_4$$

$$b11 = C_2C_4S_1 - S_4(C_1C_3 + S_1S_2S_3)$$

$$c11 = C_1S_3 - C_3S_1S_2$$

$$d11 = l_2C_1 + l_4C_4(C_1C_3 + S_1S_2S_3) + l_3C_1C_3 + l_4C_2S_1S_4 + l_3S_1S_2S_3$$

$$e11 = C_4(C_3S_1 - C_1S_2S_3) - C_1C_2S_4$$

$$f11 = -S_4(C_3S_1 - C_1S_2S_3) - C_1C_2C_4$$

$g11 = S_1S_3 + C_1C_3S_2$

$h11 = l_2S_1 + l_4C_4(C_3S_1 - C_1S_2S_3) + l_3C_3S_1 - l_4C_1C_2S_4 - l_3C_1S_2S_3$

$d_{51} = {}^0_0T\ Q_1\ {}^0_5T = \begin{bmatrix} A & B & C & D \\ E & F & G & H \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$

A=$C_5(C_4(C_1C_3 + S_1S_2S_3) + C_2S_1S_4) - S_5(S_4(C_1C_3 + S_1S_2S_3) - C_2C_4S_1)$

B=$-C_5(S_4(C_1C_3 + S_1S_2S_3) - C_2C_4S_1) - S_5(C_4(C_1C_3 + S_1S_2S_3) + C_2S_1S_4)$

C=$C_1S_3 - C_3S_1S_2$

D= $l_2C_1 + l_4C_4(C_1C_3 + S_1S_2S_3) + l_3C_1C_3 + l_5C_5(C_4(C_1C_3 + S_1S_2S_3) + C_2S_1S_4) - l_5S_5(S_4(C_1C_3 + S_1S_2S_3) - C_2C_4S_1) + l_4C_2S_1S_4 + l_3S_1S_2S_3$

E=$C_5(C_4(C_3S_1 - C_1S_2S_3) - C_1C_2S_4) - S_5(S_4(C_3S_1 - C_1S_2S_3) + C_1C_2C_4)$

F=$-C_5(S_4(C_3S_1 - C_1S_2S_3) + C_1C_2C_4) - S_5(C_4(C_3S_1 - C_1S_2S_3) - C_1C_2S_4)$

G=$S_1S_3 + C_1C_3S_2$

H= $l_2S_1 + l_4C_4(C_3S_1 - C_1S_2S_3) + l_3C_3S_1 + l_5C_5(C_4(C_3S_1 - C_1S_2S_3) - C_1C_2S_4) - l_5S_5(S_4(C_3S_1 - C_1S_2S_3) + C_1C_2C_4) - l_4C_1C_2S_4 - l_3C_1S_2S_3$

$d_{22} = {}^0_1T\ Q_2\ {}^1_2T = \begin{bmatrix} -C_1C_2 & 0 & C_1S_2 & 0 \\ -C_2S_1 & 0 & S_1S_2 & 0 \\ -S_2 & 0 & -C_2 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$

$d_{32} = {}^0_1T\ Q_2\ {}^1_3T = \begin{bmatrix} -C_1C_2S_3 & C_1S_2 & C_1C_1C_3 & -l_3C_1C_2S_3 \\ -C_2S_1S_3 & S_1S_2 & C_2C_3S_1 & -l_3C_2S_1S_3 \\ -S_2S_3 & -C_2 & C_3S_2 & -l_3S_2S_3 \\ 0 & 0 & 0 & 0 \end{bmatrix}$

$d_{42} = {}^0_1T\ Q_2\ {}^1_4T =$

$$\begin{bmatrix} C_1S_2S_4 - C_1C_2C_4S_3 & C_1C_4S_2 + C_1C_2S_3S_4 & C_1C_2C_3 & l_4C_1S_2S_4 - l_3C_1C_2S_3 - l_4C_1C_2C_4S_3 \\ S_1S_2S_4 - C_2C_4S_1S_3 & C_4S_1S_2 + C_2S_1S_3S_4 & C_2C_3S_1 & l_4S_1S_2S_4 - l_3C_2S_1S_3 - l_4C_2C_4S_1S_3 \\ -C_2S_4 - C_4S_2S_3 & S_2S_3S_4 - C_2C_4 & C_3S_2 & -l_4C_2S_4 - l_3S_2S_3 - -l_4C_4S_2S_3 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$d_{52}= {}^{0}_{1}T\ Q_2\ {}^{1}_{5}T= \begin{bmatrix} I & J & K & L \\ M & N & O & P \\ Q & R & S & T \\ 0 & 0 & 0 & 0 \end{bmatrix}$

I=$C_5(C_1S_2S_4 - C_1C_2C_4S_3) + S_5(C_1C_4S_2 + C_1C_2S_3S_4)$

J=$C_5(C_1C_4S_2 + C_1C_2S_3S_4) - S_5(C_1S_2S_4 - C_1C_2C_4S_3)$

K=$C_1C_2C_3$

L= $l_5C_5(C_1S_2S_4 - C_1C_2C_4S_3) + l_5S_5(C_1C_4S_2 + C_1C_2S_3S_4) - l_3C_1C_2S_3 + l_4C_1S_2S_4 - l_4C_1C_2C_4\ S_3$

M= $C_5(S_1S_2S_4 - C_2C_4S_1S_3) + S_5(C_4S_1S_2 + C_2S_1S_3S_4)$

N= $C_5(C_4S_1S_2 + C_2S_1S_3S_4) - S_5(S_1S_2S_4 - C_2C_4S_1S_3)$

O= $C_2C_3S_1$

P= $l_5S_5(C_4S_1S_2 + C_2S_1S_3S_4) + l_5C_5(S_1S_2S_4 - C_2C_4S_1S_3) - l_3C_2S_1S_3 + l_4S_1S_2S_4 - l_4C_2C_4S_1\ S_3$

Q= $-C_5(C_2S_4 + C_4S_2S_3) - S_5(C_2C_4 - S_2S_3S_4)$

R= $S_5(C_2S_4 + C_4S_2S_3) - C_5(C_2C_4 - S_2S_3S_4)$

S= C_3S_2

T=$-l_5C_5(C_2S_4 + C_4S_2S_3) - l_5S_5(C_2C_4 - S_2S_3S_4) - l_4C_2S_4 - l_3S_2S_3 - l_4C_4S_2\ S_3$

$d_{33}= {}^{0}_{2}T\ Q_3\ {}^{2}_{3}T= \begin{bmatrix} -S_1S_3 - C_1C_3S_2 & 0 & C_3S_1 - C_1S_2S_3 & -l_3S_1S_3 - l_3C_1C_3S_2 \\ C_1S_3 - C_3S_1S_2 & 0 & -C_1C_3 - S_1S_2S_3 & l_3C_1S_3 - l_3C_3S_1S_2 \\ C_2C_3 & 0 & C_2S_3 & l_3C_2C_3 \\ 0 & 0 & 0 & 0 \end{bmatrix}$

$d_{43}= {}^{0}_{2}T\ Q_3\ {}^{2}_{4}T=$

$$\begin{bmatrix} -C_4(S_1S_3 + C_1C_3S_2) & S_4(S_1S_3 + C_1C_3S_2) & C_3S_1 - C_1S_2S_3 & -l_4C_4(S_1S_3 + C_1C_3S_2) - l_3S_1S_3 - l_3C_1C_3S_2 \\ C_4(C_1S_3 - C_3S_1S_2) & -S_4(C_1S_3 - C_3S_1S_2) & -C_1C_3 - S_1S_2S_3 & l_4C_4(C_1S_3 - C_3S_1S_2) + l_3C_1S_3 - l_3C_3S_1S_2 \\ C_2C_3C_4 & -C_2C_3S_4 & C_2S_3 & l_3C_2C_3 + l_4C_2C_3C_4 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$d_{53} = {}^0_2T\ Q_3\ {}^2_5T = \begin{bmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$a = S_4S_5(S_1S_3 + C_1C_3S_2) - C_4C_5(S_1S_3 + C_1C_3S_2)$

$b = C_4S_5(S_1S_3 + C_1C_3S_2) + C_5S_4(S_1S_3 + C_1C_3S_2)$

$c = C_3S_1 - C_1S_2S_3$

$d = l_5S_4S_5(S_1S_3 + C_1C_3S_2) - l_3S_1S_3 - l_5C_4C_5(S_1S_3 + C_1C_3S_2) - l_4C_4(S_1S_3 + C_1C_3S_2) - l_3C_1C_3S_2$

$e = C_4C_5(C_1S_3 - C_3S_1S_2) - S_4S_5(C_1S_3 - C_3S_1S_2)$

$f = -C_4S_5(C_1S_3 - C_3S_1S_2) - C_5S_4(C_1S_3 - C_3S_1S_2)$

$g = -C_1C_3 - S_1S_2S_3$

$h = l_4C_4(C_1C_3 - C_3S_1S_2) + l_3C_1S_3 + l_5C_4C_5(C_1S_3 - C_3S_1S_2) - l_5S_4S_5(C_1S_3 - C_3S_1S_2) - l_3C_3S_1S_2$

$i = C_2C_3C_4C_5 - C_2C_3S_4S_5$

$j = -C_2C_3C_4S_5 - C_2C_3C_5S_4$

$k = C_2S_3$

$l = l_3C_2C_3 + l_4C_2C_3C_4 + l_5C_2C_3C_4C_5 - l_5C_2C_3S_4S_5$

$$d_{44} = {}^0_3T\ Q_4\ {}^3_4T =$$

$$\begin{bmatrix} -S_4(C_3S_1 - C_1S_2S_3) - C_1C_2C_4 & C_1C_2S_4 - C_4(C_3S_1 - C_1S_2S_3) & 0 & -l_4S_4(C_3S_1 - C_1S_2S_3) - l_4C_1C_2C_4 \\ S_4(C_1C_3 + S_1S_2S_3) - C_2C_4S_1 & C_4(C_1C_3 + S_1S_2S_3) + C_2S_1S_4 & 0 & l_4S_4(C_1C_3 + S_1S_2S_3) - l_4C_2C_4S_1 \\ -C_4S_2 - C_2S_3S_4 & S_2S_4 - C_2C_4S_3 & 0 & -l_4C_4S_2 - l_4C_2S_3S_4 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$d_{54}= {}^{0}_{3}T\ Q_4\ {}^{3}_{5}T=\begin{bmatrix} m & n & 0 & p \\ q & r & 0 & t \\ u & v & 0 & x \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$m= -C_5(S_4(C_3S_1 - C_1S_2S_3) + C_1C_2C_4) - S_5(C_4(C_3S_1 - C_1S_2S_3) - C_1C_2S_4)$$

$$n=S_5(S_4(C_3S_1 - C_1S_2S_3) + C_1C_2C_4) - C_5(C_4(C_3S_1 - C_1S_2S_3) - C_1C_2S_4)$$

$$p = -l_4S_4(C_3S_1 - C_1S_2S_3) - l_5C_5(S_4(C_3S_1 - C_1S_2S_3) + C_1C_2C_4) - l_5S_5(C_4(C_3S_1 - C_1S_2S_3) - C_1C_2S_4) - l_4C_1C_2C_4$$

$$q=C_5(S_4(C_1C_3 + S_1S_2S_3) - C_2C_4S_1) + S_5(C_4(C_1C_3 + S_1S_2S_3) + C_2S_1S_4)$$

$$r= C_5(C_4(C_1C_3 + S_1S_2S_3) + C_2S_1S_4) - S_5(S_4(C_1C_3 + S_1S_2S_3) - C_2C_4S_1)$$

$$t=l_5S_5(S_2S_4 - C_2C_4S_3) - l_5C_5(C_4S_2 + C_2S_3S_4) - l_4C_4S_2 - l_4C_2S_3S_4$$

$$u=S_5(S_2S_4 - C_2C_4S_3) - C_5(C_4S_2 + C_2S_3S_4)$$

$$v= C_5(S_2S_4 - C_2C_4S_3) + S_5(C_4S_2 + C_2S_3S_4)$$

$$x= l_5S_5(S_2S_4 - C_2C_4S_3) - l_5C_5(C_4S_2 + C_2S_3S_4) - l_4C_4S_2 - l_4C_2S_3S_4$$

$$d_{55}= {}^{0}_{4}T\ Q_5\ {}^{4}_{5}T=\begin{bmatrix} a1 & b1 & 0 & c1 \\ d1 & e1 & 0 & f1 \\ g1 & h1 & 0 & i1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$a1= -C_5(S_4(C_3S_1 - C_1S_2S_3) + C_1C_2C_4) - S_5(C_4(C_3S_1 - C_1S_2S_3) - C_1C_2S_4)$$

$$b1= S_5(S_4(C_3S_1 - C_1S_2S_3) + C_1C_2C_4) - C_5(C_4(C_3S_1 - C_1S_2S_3) - C_1C_2S_4)$$

$$c1 = -l_5C_5(S_4(C_3S_1 - C_1S_2S_3) + C_1C_2C_4) - l_5S_5(C_4(C_3S_1 - C_1S_2S_3) - C_1C_2S_4)$$

$$d1= C_5(S_4(C_1C_3 + S_1S_2S_3) - C_2C_4S_1) + S_5(C_4(C_1C_3 + S_1S_2S_3) + C_2S_1S_4)$$

$e1= C_5(C_4(C_1C_3 + S_1S_2S_3) + C_2S_1S_4) - S_5(S_4(C_1C_3 + S_1S_2S_3) - C_2C_4S_1)$

$f1=l_5C_5(S_4(C_1C_3 + S_1S_2S_3) - C_2C_4S_1)+ l_5S_5(C_4(C_1C_3 + S_1S_2S_3) + C_2S_1S_4)$

$g1=S_5(S_2S_4 - C_2C_4S_3) - C_5(C_4S_2 + C_2S_3S_4)$

$h1= C_5(S_2S_4 - C_2C_4S_3) + S_5(C_4S_2 + C_2S_3S_4)$

$i1= l_5S_5(S_2S_4 - C_2C_4S_3) - l_5C_5(C_4S_2 + C_2S_3S_4)$

APPENDIX F

Calculation of inertia Matrix

$M_{ij} = \sum_{p=\max(i,j)}^{n} Tr\left[\; d_{pj} \quad I_p \, {}^{T}_{pi}d \; \right]$

$$M_{ij} = \begin{bmatrix} M_{11} & M_{12} & M_{13} & M_{14} & M_{15} \\ M_{21} & M_{22} & M_{23} & M_{24} & M_{25} \\ M_{31} & M_{32} & M_{33} & M_{34} & M_{35} \\ M_{41} & M_{42} & M_{43} & M_{44} & M_{45} \\ M_{51} & M_{52} & M_{53} & M_{54} & M_{55} \end{bmatrix}$$

For i=1,j=1

M_{11}=Tr$(d_{11}I_1d_{11}^T)$+ Tr$(d_{21}I_2d_{21}^T)$+ Tr$(d_{31}I_3d_{31}^T)$+ Tr$(d_{41}I_4d_{41}^T)$+ Tr$(d_{51}I_5d_{51}^T)$

For i=1,j=2

M_{12}=Tr$(d_{22}I_2d_{21}^T)$+ Tr$(d_{32}I_3d_{31}^T)$+ Tr$(d_{42}I_4d_{41}^T)$+ Tr$(d_{52}I_5d_{51}^T)$

For i=1,j=3

M_{13}=Tr$(d_{33}I_3d_{31}^T)$+ Tr$(d_{43}I_4d_{41}^T)$+ Tr$(d_{53}I_5d_{51}^T)$

For i=1,j=4

M_{14}=Tr$(d_{44}I_4d_{41}^T)$+ Tr$(d_{54}I_5d_{51}^T)$

For i=1,j=5

M_{15}=Tr$(d_{55}I_5d_{51}^T)$

Since coupling inertia are same . Hence

$M_{21} = M_{12}$, $M_{31} = M_{13}$, $M_{41} = M_{14}$, $M_{51} = M_{15}$,

Similarly

For i=2,j=2

M_{22}= Tr$(d_{22}I_2d_{22}^T)$+ Tr$(d_{32}I_3d_{32}^T)$+ Tr$(d_{42}I_4d_{42}^T)$+ Tr$(d_{52}I_5d_{52}^T)$

For i=2,j=3

M_{23}=Tr($d_{33}I_3d_{32}^T$)+ Tr($d_{43}I_4d_{42}^T$)+ Tr($d_{53}I_5d_{52}^T$)

For i=2,j=4

M_{24}=Tr($d_{44}I_4d_{42}^T$)+ Tr($d_{54}I_5d_{52}^T$)

For i=2,j=5

M_{25}=Tr($d_{55}I_5d_{52}^T$)

Since coupling inertia are same . Hence

M_{32} =M_{23}, M_{42} =M_{24}, M_{52} =M_{25},

For i=3,j=3,

M_{33}=Tr($d_{33}I_3d_{33}^T$)+ Tr($d_{43}I_4d_{43}^T$)+ Tr($d_{53}I_5d_{53}^T$)

For i=3,j=4,

M_{34}=Tr($d_{44}I_4d_{43}^T$)+ Tr($d_{54}I_5d_{53}^T$)

For i=3,j=5

M_{35}=Tr($d_{55}I_5d_{53}^T$)

Since coupling inertia are same. Hence

M_{43} =M_{34}, M_{53} =M_{35},

For i=4, j=4

M_{44}=Tr($d_{44}I_4d_{44}^T$)+ Tr($d_{54}I_5d_{54}^T$)

For i=4, j=5

M_{45}=Tr($d_{55}I_5d_{54}^T$)

Since coupling inertia are same. Hence

M_{54} =M_{45},

For i=5,j=5

M_{55}=Tr$(d_{55}I_5d_{55}^T)$

From the inertia coefficients obtained above , the inertia matrix for the manipulator is

$$M_{ij}=\begin{bmatrix} M_{11} & M_{12} & M_{13} & M_{14} & M_{15} \\ M_{21} & M_{22} & M_{23} & M_{24} & M_{25} \\ M_{31} & M_{32} & M_{33} & M_{34} & M_{35} \\ M_{41} & M_{42} & M_{43} & M_{44} & M_{45} \\ M_{51} & M_{52} & M_{53} & M_{54} & M_{55} \end{bmatrix}$$

APPENDIX G

Calculation of Coriolis and centrifugal force coefficients

The Coriolis and centrifugal force coefficients , h_{ijk} for i,j,k=1,2,3,4,5are obtained from equation, i.e

$$h_{ijk} = \sum_{p=\max(i,j,k)}^{n} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]$$

Thus , the Coriolis and centrifugal coefficient matrix H is therefore

$$H = \begin{bmatrix} H_1 \\ H_2 \\ H_3 \\ H_4 \\ H_5 \end{bmatrix}$$

The centrifugal acceleration coefficients and the Coriolis coefficients are

For i=1,j=1,k=1

$$h_{111} = \sum_{p=\max(1,1,1)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right] = \text{Tr}\left(\frac{\partial(d_{11})}{\partial q_1}\ I_1\ {}_{11}^{T}d\right) + \text{Tr}\left(\frac{\partial(d_{21})}{\partial q_2}\ I_2\ {}_{21}^{T}d\right) +$$

$$\text{Tr}\left(\frac{\partial(d_{31})}{\partial q_3}\ I_3\ {}_{31}^{T}d\right) +$$

$$\text{Tr}\left(\frac{\partial(d_{41})}{\partial q_4}\ I_4\ {}_{41}^{T}d\right) + \text{Tr}\left(\frac{\partial(d_{51})}{\partial q_5}\ I_5\ {}_{51}^{T}d\right) = \text{Tr}\left({}_{0}^{0}T\ Q_1\ Q_1\ {}_{1}^{0}T\ I_1\ {}_{11}^{T}d\right) +$$

$$\text{Tr}\left({}_{0}^{0}T\ Q_1\ Q_2\ {}_{2}^{0}T I_2\ {}_{21}^{T}d\right) + \text{tr}\left({}_{0}^{0}T\ Q_1\ Q_3\ {}_{3}^{0}T\ I_3\ {}_{31}^{T}d\right) + \text{tr}\left({}_{0}^{0}T\ Q_1\ Q_4\ {}_{4}^{0}T I_4\ {}_{41}^{T}d\right) +$$

$$\text{Tr}\left({}_{0}^{0}T\ Q_1 Q_5\ {}_{5}^{0}T\ I_5\ {}_{51}^{T}d\right) = \text{Tr}\left({}_{0}^{0}T\ Q_1\ Q_1\ {}_{1}^{0}T\ I_1\ {}_{11}^{T}d\right) + \text{Tr}\left({}_{0}^{0}T\ Q_1\ {}_{1}^{0}T\ Q_2\ {}_{2}^{1}T I_2\ {}_{21}^{T}d\right) +$$

$$\text{Tr}\left({}_{0}^{0}T\ Q_1\ {}_{1}^{0}T\ {}_{2}^{1}T\ Q_3\ {}_{3}^{2}T I_3\ {}_{31}^{T}d\right) + \text{Tr}\left({}_{0}^{0}T\ Q_1\ {}_{1}^{0}T\ {}_{2}^{1}T\ {}_{3}^{2}T\ Q_4\ {}_{4}^{3}T\ I_4\ {}_{41}^{T}d\right) +$$

$$\text{Tr}\left({}_{0}^{0}T\ Q_1\ {}_{1}^{0}T\ {}_{2}^{1}T\ {}_{3}^{2}T\ {}_{4}^{3}T\ Q_5\ {}_{5}^{4}T\ I_5\ {}_{51}^{T}d\right)$$

For i=1,j=1,k=2

$h_{112} = \sum_{p=\max(1,1,2)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p} I_p \, {}_{pi}^{T}d\right] =$ Tr($\frac{\partial(d_{22})}{\partial q_2} I_2 \, {}_{21}^{T}d$)+ Tr($\frac{\partial(d_{32})}{\partial q_3} I_3 \, {}_{31}^{T}d$)+ Tr($\frac{\partial(d_{42})}{\partial q_4} I_4 \, {}_{41}^{T}d$)+

Tr($\frac{\partial(d_{52})}{\partial q_5} I_5 \, {}_{51}^{T}d$)=

tr(${}_{1}^{0}T \; Q_2 \; Q_2 \; {}_{2}^{1}T \; I_2 \, {}_{21}^{T}d$)+tr(${}_{1}^{0}T \; Q_2 \; Q3 \; {}_{3}^{1}T I_3 \, {}_{31}^{T}d$) +tr(${}_{1}^{0}T \; Q_2 \; Q4 \; {}_{4}^{1}T \; I_4 \, {}_{41}^{T}d$)+tr(${}_{1}^{0}T \; Q_2 \; Q5 \; {}_{5}^{1}T \; I_5 \, {}_{51}^{T}d$)=

tr(${}_{1}^{0}T \; Q_2 \; Q_2 \; {}_{2}^{1}T \; I_2 \, {}_{21}^{T}d$)+ tr(${}_{1}^{0}T \; Q_2 * {}_{2}^{1}T \; Q_3 \; {}_{3}^{2}T I_3 \, {}_{31}^{T}d$)+ tr(${}_{1}^{0}T \; Q_2 \; {}_{2}^{1}T \; {}_{3}^{2}T \; Q_4 \; {}_{4}^{3}T \; I_4 \, {}_{41}^{T}d$)

+ tr(${}_{1}^{0}T \; Q_2 \; {}_{2}^{1}T \; {}_{3}^{2}T \; {}_{4}^{3}T \; Q_5 \; {}_{5}^{4}T \; I_5 \, {}_{51}^{T}d$)

For i=1,j=1,k=3

h_{113}= Tr($\frac{\partial(d_{33})}{\partial q_3} I_3 \, {}_{31}^{T}d$)+ Tr($\frac{\partial(d_{43})}{\partial q_4} I_4 \, {}_{41}^{T}d$)+Tr($\frac{\partial(d_{53})}{\partial q_5} I_5 \, {}_{51}^{T}d$)=

Tr(${}_{2}^{0}T \; Q_3 \; Q_3 \; {}_{3}^{2}T I_3 \, {}_{31}^{T}d$)+tr(${}_{2}^{0}T \; Q_3 \; Q4 \; {}_{4}^{2}T \; I_4 \, {}_{41}^{T}d$) +tr(${}_{2}^{0}T \; Q_3 \; Q5 \; {}_{5}^{2}T \; I_5 \, {}_{51}^{T}d$) =

Tr(${}_{1}^{0}T \; {}_{2}^{1}T \; Q_3 \; Q_3 {}_{3}^{2}T I_3 \, {}_{31}^{T}d$)+tr(${}_{1}^{0}T \; {}_{2}^{1}T Q_3 {}_{3}^{2}T \; Q_4 \; {}_{4}^{3}T I_4 \, {}_{41}^{T}d$) +tr(${}_{1}^{0}T \; {}_{2}^{1}T \; Q_3 {}_{3}^{2}T \; {}_{4}^{3}T \; Q_5 \; {}_{5}^{4}T \; I_5 \, {}_{51}^{T}d$)

For i=1,j=1,k=4

h_{114} = Tr($\frac{\partial(d_{44})}{\partial q_4} I_4 \, {}_{41}^{T}d$)+Tr($\frac{\partial(d_{54})}{\partial q_5} I_5 \, {}_{51}^{T}d$)=

tr(${}_{3}^{0}T \; Q_4 \; Q_4 \; {}_{4}^{3}T \; I_4 \, {}_{41}^{T}d$) +tr(${}_{3}^{0}T \; Q_4 \; Q_5 \; {}_{5}^{3}T \; I_5 \, {}_{51}^{T}d$) =

Tr(${}_{1}^{0}T \; {}_{2}^{1}T {}_{3}^{2}T \; Q_4 \; Q_4 \; {}_{4}^{3}T \; I_4 \, {}_{41}^{T}d$) +tr(${}_{1}^{0}T \; {}_{2}^{1}T {}_{3}^{2}T \; Q_4 {}_{4}^{3}T Q_5 {}_{5}^{4}T \; I_5 \, {}_{51}^{T}d$)

For i=1,j=1,k=5

h_{115}= Tr($\frac{\partial(d_{55})}{\partial q_5} I_5 \, {}_{51}^{T}d$)= tr(${}_{4}^{0}T \; Q_5 \; Q_5 \; {}_{5}^{4}T \; I_5 \, {}_{51}^{T}d$) =tr(${}_{1}^{0}T \; {}_{2}^{1}T {}_{3}^{2}T {}_{4}^{3}T Q_5 Q_5 \; {}_{5}^{4}T \; I_5 \, {}_{51}^{T}d$)

Since the effects of one joint over another joint are equal ,hence

h_{112}=h_{121}, h_{113}= h_{131} , h_{114}=h_{141}, h_{115}=h_{151}

For **i=1,j=2,k=2**

h_{122}= Tr($\frac{\partial(d_{22})}{\partial q_2}\ I_{2\ 2}{}_1^T d$)+ Tr($\frac{\partial(d_{32})}{\partial q_3}\ I_{3\ 3}{}_1^T d$)+ Tr($\frac{\partial(d_{42})}{\partial q_4}\ I_{4\ 4}{}_1^T d$)+Tr($\frac{\partial(d_{52})}{\partial q_5}\ I_{5\ 5}{}_1^T d$)

For i=1,j=2,k=3

h_{123}= Tr($\frac{\partial(d_{33})}{\partial q_3}\ I_{3\ 3}{}_1^T d$)+ Tr($\frac{\partial(d_{43})}{\partial q_4}\ I_{4\ 4}{}_1^T d$)+Tr($\frac{\partial(d_{53})}{\partial q_5}\ I_{5\ 5}{}_1^T d$)

For i=1,j=2,k=4

h_{124}= = Tr($\frac{\partial(d_{44})}{\partial q_4}\ I_{4\ 4}{}_1^T d$)+Tr($\frac{\partial(d_{54})}{\partial q_5}\ I_{5\ 5}{}_1^T d$)

For i=1,j=2,k=5

h_{125}= Tr($\frac{\partial(d_{55})}{\partial q_5}\ I_{5\ 5}{}_1^T d$)

h_{123}=h_{132}, h_{124}= h_{142}, h_{125}=h_{152}

For i=1,j=3,k=3

h_{133}= Tr($\frac{\partial(d_{33})}{\partial q_3}\ I_{3\ 3}{}_1^T d$)+ Tr($\frac{\partial(d_{43})}{\partial q_4}\ I_{4\ 4}{}_1^T d$)+Tr($\frac{\partial(d_{53})}{\partial q_5}\ I_{5\ 5}{}_1^T d$)

For i=1,j=3,k=4

h_{134}= Tr($\frac{\partial(d_{44})}{\partial q_4}\ I_{4\ 4}{}_1^T d$)+Tr($\frac{\partial(d_{54})}{\partial q_5}\ I_{5\ 5}{}_1^T d$)

For i=1 ,j=3,k=5

h_{135}= Tr($\frac{\partial(d_{55})}{\partial q_5}\ I_{5\ 5}{}_1^T d$)

h_{134}=h_{143} , h_{135}=h_{153}

For i=1 ,j=4,k=4

h_{144}= Tr($\frac{\partial(d_{44})}{\partial q_4}\ I_{4\ 4}{}_1^T d$)+Tr($\frac{\partial(d_{54})}{\partial q_5}\ I_{5\ 5}{}_1^T d$)

For i=1 ,j=4,k=5

h_{145}= Tr($\frac{\partial(d_{55})}{\partial q_5}\ I_5\ {}_5^1T d$)

h_{145}=h_{154}

For i=1 ,j=5,k=5

h_{155}= Tr($\frac{\partial(d_{55})}{\partial q_5}\ I_5\ {}_5^1T d$)

The Coriolis and centrifugal coefficient terms are computed using the series summation.

For i=1,

$H_1 = \sum_{j=1}^{n}\sum_{k=1}^{n} h_{ijk}\,\dot{\theta}_j\dot{\theta}_k = h_{111}\dot{\theta}_1^{\,2} + h_{112}\dot{\theta}_1\dot{\theta}_2 + h_{113}\dot{\theta}_1\dot{\theta}_3 + h_{114}\dot{\theta}_1\dot{\theta}_4 + h_{115}\dot{\theta}_1\dot{\theta}_5 + h_{121}\dot{\theta}_2\dot{\theta}_1 + h_{122}\dot{\theta}_2^{\,2} + h_{123}\dot{\theta}_2\dot{\theta}_3 + h_{124}\dot{\theta}_2\dot{\theta}_4 + h_{125}\dot{\theta}_2\dot{\theta}_5 + h_{131}\dot{\theta}_3\dot{\theta}_1 + h_{132}\dot{\theta}_3\dot{\theta}_2 + h_{133}\,\dot{\theta}_3^{\,2} + h_{134}\dot{\theta}_3\dot{\theta}_4 + h_{135}\dot{\theta}_3\dot{\theta}_5 + h_{141}\dot{\theta}_4\dot{\theta}_1 + h_{142}\dot{\theta}_4\dot{\theta}_2 + h_{143}\dot{\theta}_4\dot{\theta}_3 + h_{144}\,\dot{\theta}_4^{\,2} + h_{145}\dot{\theta}_4\dot{\theta}_5 + h_{151}\dot{\theta}_5\dot{\theta}_1 + h_{152}\dot{\theta}_5\dot{\theta}_2 + h_{153}\dot{\theta}_5\dot{\theta}_3 + h_{154}\dot{\theta}_5\dot{\theta}_4 + h_{155}\,\dot{\theta}_5^{\,2}$

The centrifugal acceleration coefficients and the Coriolis coefficients are

For i=2,j=1,k=1

$h_{211} = \sum_{p=\max(2,1,1)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_p^1T d\right]$ =Tr($\frac{\partial(d_{21})}{\partial q_2}\ I_2\ {}_2^1T d$)+Tr($\frac{\partial(d_{31})}{\partial q_3}\ I_3\ {}_3^2T d$)+ Tr($\frac{\partial(d_{41})}{\partial q_4}\ I_4\ {}_4^2T d$)+tr($\frac{\partial(d_{51})}{\partial q_5}\ I_5\ {}_5^2T d$) =

Tr(${}_0^0T\ Q_1\ Q_2\ {}_2^0T I_2\ {}_2^2T d$)+ tr(${}_0^0T\ Q_1\ Q_3\ {}_3^0T\ I_3\ {}_3^2T d$)+tr(${}_0^0T\ Q_1\ Q_4\ {}_4^0T I_4\ {}_4^2T d$)+

Tr(${}_0^0T\ Q_1 Q_5\ {}_5^0T\ I_5\ {}_5^2T d$) =

Tr(${}_0^0T\ Q_1 {}_1^0T\ Q_2 {}_2^1T I_2\ {}_2^2T d$)+

Tr(${}_0^0T\ Q_1 {}_1^0T\ {}_2^1T\ Q_3\ {}_3^2T I_3\ {}_3^2T d$)+Tr(${}_0^0T\ Q_1 {}_1^0T\ {}_2^1T\ {}_3^2T\ Q_4\ {}_4^3T\ I_4\ {}_4^2T d$)+

$\mathrm{Tr}({}^{0}_{0}T\ Q_1\ {}^{0}_{1}T\ {}^{1}_{2}T\ {}^{2}_{3}T\ {}^{3}_{4}T\ Q_5\ {}^{4}_{5}T\ I_5\ {}^{T}_{52}d\ \ )$

For i=2 ,j=1,k=2

$h_{212}\ =\ \sum_{p=\max(2,1,2)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}^{T}_{pi}d\right]$ =Tr($\frac{\partial(d_{22})}{\partial q_2}\ I_2\ {}^{T}_{22}d\ \ )$+Tr($\frac{\partial(d_{32})}{\partial q_3}\ I_3\ {}^{T}_{32}d\ \ )$+

Tr($\frac{\partial(d_{42})}{\partial q_4}\ I_4\ {}^{T}_{42}d$)+tr($\frac{\partial(d_{52})}{\partial q_5}\ I_5\ {}^{T}_{52}d$)=

tr(${}^{0}_{1}T\ Q_2\ Q_2\ {}^{1}_{2}T\ I_2\ {}^{T}_{22}d$)+tr(${}^{0}_{1}T\ Q_2\ Q3\ {}^{1}_{3}T I_3\ {}^{T}_{32}d)$ +tr(${}^{0}_{1}T\ Q_2\ Q4\ {}^{1}_{4}T$ $I_4\ {}^{T}_{42}d)$+tr(${}^{0}_{1}T\ Q_2\ Q5\ {}^{1}_{5}T\ I_5\ {}^{T}_{52}d)$=

tr(${}^{0}_{1}T\ Q_2\ Q_2\ {}^{1}_{2}T\ I_2\ {}^{T}_{22}d)$+ tr(${}^{0}_{1}T\ Q_2 * {}^{1}_{2}T\ Q_3\ {}^{2}_{3}T I_3\ {}^{T}_{32}d)$+ tr(${}^{0}_{1}T\ Q_2\ {}^{1}_{2}T\ {}^{2}_{3}T\ Q_4\ {}^{3}_{4}T\ \ I_4\ {}^{T}_{42}d)$

+ tr(${}^{0}_{1}T\ Q_2\ {}^{1}_{2}T\ {}^{2}_{3}T\ {}^{3}_{4}T\ Q_5\ {}^{4}_{5}T\ I_5\ {}^{T}_{52}d\ \ )$

For i=2 ,j=1,k=3

h_{213}= Tr($\frac{\partial(d_{33})}{\partial q_3}\ I_3\ {}^{T}_{32}d)$+ Tr($\frac{\partial(d_{43})}{\partial q_4}\ I_4\ {}^{T}_{42}d)$+Tr($\frac{\partial(d_{53})}{\partial q_5}\ I_5\ {}^{T}_{52}d)$

For i=2 ,j=1,k=4

h_{214}= Tr($\frac{\partial(d_{44})}{\partial q_4}\ I_4\ {}^{T}_{42}d)$+Tr($\frac{\partial(d_{54})}{\partial q_5}\ I_5\ {}^{T}_{52}d)$

For i=2 ,j=1,k=5

h_{215}= Tr($\frac{\partial(d_{55})}{\partial q_5}\ I_5\ {}^{T}_{52}d)$

Since the effects of one joint over another joint are equal. Hence

h_{221}=h_{212}, h_{231}=h_{213}, h_{241}=h_{214}, h_{251}=h_{215},

For i=2 ,j=2,k=2

$h_{222} = \sum_{p=\max(2,2,2)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p} I_p \, {}^{T}_{pi}d\right]$ =Tr($\frac{\partial(d_{22})}{\partial q_2} I_2 \, {}^{T}_{22}d$)+Tr($\frac{\partial(d_{32})}{\partial q_3} I_3 \, {}^{T}_{32}d$)+ Tr($\frac{\partial(d_{42})}{\partial q_4} I_4 \, {}^{T}_{42}d$)+tr($\frac{\partial(d_{52})}{\partial q_5} I_5 \, {}^{T}_{52}d$)

For i=2 ,j=2,k=3

$h_{223} = \sum_{p=\max(2,2,3)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p} I_p \, {}^{T}_{pi}d\right]$ = Tr($\frac{\partial(d_{33})}{\partial q_3} I_3 \, {}^{T}_{32}d$)+ Tr($\frac{\partial(d_{43})}{\partial q_4} I_4 \, {}^{T}_{42}d$)+tr($\frac{\partial(d_{53})}{\partial q_5} I_5 \, {}^{T}_{52}d$)

For i=2 ,j=2,k=4

$h_{224}=\sum_{p=\max(2,2,4)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p} I_p \, {}^{T}_{pi}d\right]$= Tr($\frac{\partial(d_{44})}{\partial q_4} I_4 \, {}^{T}_{42}d$)+tr($\frac{\partial(d_{54})}{\partial q_5} I_5 \, {}^{T}_{52}d$)

For i=2 ,j=2,k=5

$h_{225}=\sum_{p=\max(2,2,5)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p} I_p \, {}^{T}_{pi}d\right]$= tr($\frac{\partial(d_{55})}{\partial q_5} I_5 \, {}^{T}_{52}d$)

$h_{223}=h_{232}$, $h_{224}=h_{242}$, $h_{225}=h_{252}$

For i=2 ,j=3,k=3

$h_{233} = \sum_{p=\max(2,3,3)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p} I_p \, {}^{T}_{pi}d\right]$ = Tr($\frac{\partial(d_{33})}{\partial q_3} I_3 \, {}^{T}_{32}d$)+ Tr($\frac{\partial(d_{43})}{\partial q_4} I_4 \, {}^{T}_{42}d$)+tr($\frac{\partial(d_{53})}{\partial q_5} I_5 \, {}^{T}_{52}d$)

For i=2 ,j=3,k=4

$h_{234}=\sum_{p=\max(2,3,4)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p} I_p \, {}^{T}_{pi}d\right]$= Tr($\frac{\partial(d_{44})}{\partial q_4} I_4 \, {}^{T}_{42}d$)+tr($\frac{\partial(d_{54})}{\partial q_5} I_5 \, {}^{T}_{52}d$)

For i=2 ,j=3,k=5

$h_{235}=\sum_{p=\max(2,3,5)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=$ tr$\left(\frac{\partial(d_{55})}{\partial q_5}\ I_5\ {}_{52}^{T}d\right)$

$h_{234}=h_{243}$, $h_{235}=h_{253}$

For i=2 ,j=4,k=4

$h_{244}=\sum_{p=\max(2,4,4)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=$ Tr$\left(\frac{\partial(d_{44})}{\partial q_4}\ I_4\ {}_{42}^{T}d\right)$+tr$\left(\frac{\partial(d_{54})}{\partial q_5}\ I_5\ {}_{52}^{T}d\right)$

For i=2 ,j=4,k=5

$h_{245}=\sum_{p=\max(2,4,5)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=$ tr$\left(\frac{\partial(d_{55})}{\partial q_5}\ I_5\ {}_{52}^{T}d\right)$

$h_{245}= h_{254}$

For i=2 ,j=5,k=5,

$h_{255}=\sum_{p=\max(2.5.5)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=$ tr$\left(\frac{\partial(d_{55})}{\partial q_5}\ I_5\ {}_{52}^{T}d\right)$

The centrifugal acceleration coefficients and the Coriolis coefficients are

For i=2

$$H_2 = \sum_{j=1}^{n}\sum_{k=1}^{n} h_{ijk}\dot{\theta}_j\dot{\theta}_k = h_{211}\dot{\theta}_1^{\,2} + h_{212}\dot{\theta}_1\dot{\theta}_2 + h_{213}\dot{\theta}_1\dot{\theta}_3 + h_{214}\dot{\theta}_1\dot{\theta}_4 + h_{215}\dot{\theta}_1\dot{\theta}_5 +$$

$$h_{221}\dot{\theta}_2\dot{\theta}_1 + h_{222}\dot{\theta}_2^{\,2} + h_{223}\dot{\theta}_2\dot{\theta}_3 + h_{224}\dot{\theta}_2\dot{\theta}_4 + h_{225}\dot{\theta}_2\dot{\theta}_5 + h_{231}\dot{\theta}_3\dot{\theta}_1 + h_{232}\dot{\theta}_3\dot{\theta}_2 +$$

$$h_{233}\,\dot{\theta}_3^{\,2} + h_{234}\dot{\theta}_3\dot{\theta}_4 + h_{235}\dot{\theta}_3\dot{\theta}_5 + h_{241}\dot{\theta}_4\dot{\theta}_1 + h_{242}\dot{\theta}_4\dot{\theta}_2 + h_{243}\dot{\theta}_4\dot{\theta}_3 + h_{244}\,\dot{\theta}_4^{\,2} +$$

$$h_{245}\dot{\theta}_4\dot{\theta}_5 + h_{251}\dot{\theta}_5\dot{\theta}_1 + h_{252}\dot{\theta}_5\dot{\theta}_2 + h_{253}\dot{\theta}_5\dot{\theta}_3 + h_{254}\dot{\theta}_5\dot{\theta}_4 + h_{255}\,\dot{\theta}_5^{\,2}$$

The centrifugal acceleration coefficients and the Coriolis coefficients are

For i=3,j=1,k=1

$h_{311} = \sum_{p=\max(3,1,1)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p} I_p \, {}_{pi}^{T}d\right] = \mathrm{Tr}\left(\frac{\partial(d_{31})}{\partial q_3} I_3 \, {}_{33}^{T}d\right) +$

$\mathrm{Tr}\left(\frac{\partial(d_{41})}{\partial q_4} I_4 \, {}_{43}^{T}d\right) + \mathrm{tr}\left(\frac{\partial(d_{51})}{\partial q_5} I_5 \, {}_{53}^{T}d\right)$

For i=3,j=1,k=2

$h_{312} = \sum_{p=\max(3,1,2)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p} I_p \, {}_{pi}^{T}d\right] = \mathrm{Tr}\left(\frac{\partial(d_{32})}{\partial q_3} I_3 \, {}_{33}^{T}d\right) +$

$\mathrm{Tr}\left(\frac{\partial(d_{42})}{\partial q_4} I_4 \, {}_{43}^{T}d\right) + \mathrm{tr}\left(\frac{\partial(d_{52})}{\partial q_5} I_5 \, {}_{53}^{T}d\right)$

For i=3,j=1,k=3

$h_{313} = \sum_{p=\max(3,1,3)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p} I_p \, {}_{pi}^{T}d\right] = \mathrm{Tr}\left(\frac{\partial(d_{33})}{\partial q_3} I_3 \, {}_{33}^{T}d\right) +$

$\mathrm{Tr}\left(\frac{\partial(d_{43})}{\partial q_4} I_4 \, {}_{43}^{T}d\right) + \mathrm{tr}\left(\frac{\partial(d_{53})}{\partial q_5} I_5 \, {}_{53}^{T}d\right)$

For i=3,j=1,k=4

$h_{314} = \sum_{p=\max(3,1,4)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p} I_p \, {}_{pi}^{T}d\right] = \mathrm{Tr}\left(\frac{\partial(d_{44})}{\partial q_4} I_4 \, {}_{43}^{T}d\right) + \mathrm{tr}\left(\frac{\partial(d_{54})}{\partial q_5} I_5 \, {}_{53}^{T}d\right)$

For i=3,j=1,k=5

$h_{315} = \sum_{p=\max(3,1,5)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p} I_p \, {}_{pi}^{T}d\right] = \mathrm{tr}\left(\frac{\partial(d_{55})}{\partial q_5} I_5 \, {}_{53}^{T}d\right)$

$h_{312} = h_{321}$, $h_{313} = h_{331}$, $h_{314} = h_{341}$, $h_{315} = h_{351}$

For i=3,j=2,k=2

h_{322} = $\sum_{p=\max(3,2,2)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p} I_p \, {}_{pi}^{T}d\right]$ = Tr($\frac{\partial(d_{32})}{\partial q_3} I_3 \, {}_{33}^{T}d$)+

Tr($\frac{\partial(d_{42})}{\partial q_4} I_4 \, {}_{43}^{T}d$)+tr($\frac{\partial(d_{52})}{\partial q_5} I_5 \, {}_{53}^{T}d$)

For i=3,j=2,k=3

h_{323} = $\sum_{p=\max(3,2,3)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p} I_p \, {}_{pi}^{T}d\right]$ = Tr($\frac{\partial(d_{33})}{\partial q_3} I_3 \, {}_{33}^{T}d$)+

Tr($\frac{\partial(d_{43})}{\partial q_4} I_4 \, {}_{43}^{T}d$)+tr($\frac{\partial(d_{53})}{\partial q_5} I_5 \, {}_{53}^{T}d$)

For i=3,j=2,k=4

h_{324}=$\sum_{p=\max(3,2,4)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p} I_p \, {}_{pi}^{T}d\right]$= Tr($\frac{\partial(d_{44})}{\partial q_4} I_4 \, {}_{43}^{T}d$)+tr($\frac{\partial(d_{54})}{\partial q_5} I_5 \, {}_{53}^{T}d$)

For i=3,j=2,k=5

h_{325}=$\sum_{p=\max(3,2,5)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p} I_p \, {}_{pi}^{T}d\right]$= tr($\frac{\partial(d_{53})}{\partial q_5} I_5 \, {}_{53}^{T}d$)

h_{323}=h_{332}, h_{324}=h_{342}, h_{325}=-h_{352}

For i=3,j=3,k=3

h_{333} = $\sum_{p=\max(3,3,3)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p} I_p \, {}_{pi}^{T}d\right]$ = Tr($\frac{\partial(d_{33})}{\partial q_3} I_3 \, {}_{33}^{T}d$)+

Tr($\frac{\partial(d_{43})}{\partial q_4} I_4 \, {}_{43}^{T}d$)+tr($\frac{\partial(d_{53})}{\partial q_5} I_5 \, {}_{53}^{T}d$)

For i=3,j=3,k=4

h_{334}=$\sum_{p=\max(3,3,4)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p} I_p \, {}_{pi}^{T}d\right]$= Tr($\frac{\partial(d_{44})}{\partial q_4} I_4 \, {}_{43}^{T}d$)+ tr($\frac{\partial(d_{54})}{\partial q_5} I_5 \, {}_{53}^{T}d$)

For i=3,j=3,k=5

$h_{335}=\sum_{p=\max(3,3,5)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=\ \mathrm{Tr}(\frac{\partial(d_{55})}{\partial q_5}\ I_5\ {}_{53}^{T}d)$

$h_{334}=h_{343}$, $h_{335}=h_{353}$

For i=3,j=4,k=4

$h_{344}=\sum_{p=\max(3,4,4)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=\ \mathrm{Tr}(\frac{\partial(d_{44})}{\partial q_4}\ I_4\ {}_{43}^{T}d)+\ \mathrm{tr}(\frac{\partial(d_{54})}{\partial q_5}\ I_5\ {}_{53}^{T}d)$

For i=3,j=4,k=5

$h_{345}=\sum_{p=\max(3,4,5)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=\ \mathrm{Tr}(\frac{\partial(d_{55})}{\partial q_5}\ I_5\ {}_{53}^{T}d)$

$h_{345}=h_{354}$,

For i=3,j=5,k=5

$h_{355}=\sum_{p=\max(3,5,5)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=\ \mathrm{Tr}(\frac{\partial(d_{55})}{\partial q_5}\ I_5\ {}_{53}^{T}d)$

The centrifugal acceleration coefficients and the Coriolis coefficients are

For i=3

$H_3 = \sum_{j=1}^{n}\sum_{k=1}^{n} h_{ijk}\,\dot{\theta}_j\dot{\theta}_k = h_{311}\dot{\theta}_1^{\,2} + h_{312}\dot{\theta}_1\dot{\theta}_2 + h_{313}\dot{\theta}_1\dot{\theta}_3 + h_{314}\dot{\theta}_1\dot{\theta}_4 + h_{315}\dot{\theta}_1\dot{\theta}_5 +$

$h_{321}\dot{\theta}_2\dot{\theta}_1 + h_{322}\dot{\theta}_2^{\,2} + h_{323}\dot{\theta}_2\dot{\theta}_3 + h_{324}\dot{\theta}_2\dot{\theta}_4 + h_{325}\dot{\theta}_2\dot{\theta}_5 + h_{331}\dot{\theta}_3\dot{\theta}_1 + h_{332}\dot{\theta}_3\dot{\theta}_2 +$

$h_{333}\,\dot{\theta}_3^{\,2} + h_{334}\dot{\theta}_3\dot{\theta}_4 + h_{335}\dot{\theta}_3\dot{\theta}_5 + h_{341}\dot{\theta}_4\dot{\theta}_1 + h_{342}\dot{\theta}_4\dot{\theta}_2 + h_{343}\dot{\theta}_4\dot{\theta}_3 + h_{344}\,\dot{\theta}_4^{\,2} +$

$h_{345}\dot{\theta}_4\dot{\theta}_5 + h_{351}\dot{\theta}_5\dot{\theta}_1 + h_{352}\dot{\theta}_5\dot{\theta}_2 + h_{353}\dot{\theta}_5\dot{\theta}_3 + h_{354}\dot{\theta}_5\dot{\theta}_4 + h_{355}\,\dot{\theta}_5^{\,2}$

The centrifugal acceleration coefficients and the Coriolis coefficients are

For i=4,j=1,k=1

$$h_{411}=\sum_{p=\max(4,1,1)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}^{T}_{pi}d\right]=\ \mathrm{Tr}(\frac{\partial(d_{41})}{\partial q_4}\ I_4\ {}^{T}_{44}d)+\ \mathrm{tr}(\frac{\partial(d_{51})}{\partial q_5}\ I_5\ {}^{T}_{54}d)$$

For i=4,j=1,k=2

$$h_{412}=\sum_{p=\max(4,1,2)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}^{T}_{pi}d\right]=\ \mathrm{Tr}(\frac{\partial(d_{42})}{\partial q_4}\ I_4\ {}^{T}_{44}d)+\ \mathrm{tr}(\frac{\partial(d_{52})}{\partial q_5}\ I_5\ {}^{T}_{54}d)$$

For i=4,j=1,k=3

$$h_{413}=\sum_{p=\max(4,1,3)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}^{T}_{pi}d\right]=\ \mathrm{Tr}(\frac{\partial(d_{43})}{\partial q_4}\ I_4\ {}^{T}_{44}d)+\ \mathrm{tr}(\frac{\partial(d_{53})}{\partial q_5}\ I_5\ {}^{T}_{54}d)$$

For i=4,j=1,k=4

$$h_{414}=\sum_{p=\max(4,1,4)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}^{T}_{pi}d\right]=\ \mathrm{Tr}(\frac{\partial(d_{44})}{\partial q_4}\ I_4\ {}^{T}_{44}d)+\ \mathrm{tr}(\frac{\partial(d_{54})}{\partial q_5}\ I_5\ {}^{T}_{54}d)$$

For i=4,j=1,k=5

$$h_{415}=\sum_{p=\max(4,1,5)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}^{T}_{pi}d\right]=\ \mathrm{tr}(\frac{\partial(d_{55})}{\partial q_5}\ I_5\ {}^{T}_{54}d)$$

$h_{412} = h_{421}, h_{413} = h_{431}, h_{414} = h_{441}$

For i=4,j=2,k=2

$$h_{422}=\sum_{p=\max(4,2,2)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}^{T}_{pi}d\right]=\ \mathrm{Tr}(\frac{\partial(d_{42})}{\partial q_4}\ I_4\ {}^{T}_{44}d)+\ \mathrm{tr}(\frac{\partial(d_{52})}{\partial q_5}\ I_5\ {}^{T}_{54}d)$$

For i=4,j=2,k=3

$$h_{423}=\sum_{p=\max(4,2,3)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}^{T}_{pi}d\right]=\ \mathrm{Tr}(\frac{\partial(d_{43})}{\partial q_4}\ I_4\ {}^{T}_{44}d)+\ \mathrm{tr}(\frac{\partial(d_{53})}{\partial q_5}\ I_5\ {}^{T}_{54}d)$$

For i=4,j=2,k=4

$h_{424}=\sum_{p=\max(4,2,4)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=\ \mathrm{Tr}(\frac{\partial(d_{44})}{\partial q_4}\ I_4\ {}_{44}^{T}d)+\ \mathrm{tr}(\frac{\partial(d_{54})}{\partial q_5}\ I_5\ {}_{54}^{T}d)$

For i=4,j=2,k=5

$h_{425}=\sum_{p=\max(4,2,5)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=\ \mathrm{tr}(\frac{\partial(d_{55})}{\partial q_5}\ I_5\ {}_{54}^{T}d)$

$h_{423}=h_{432}, h_{424}=h_{442}, h_{425}=h_{452}$

For i=4,j=3,k=3

$h_{433}=\sum_{p=\max(4,3,3)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=\ \mathrm{Tr}(\frac{\partial(d_{43})}{\partial q_4}\ I_4\ {}_{44}^{T}d)+\ \mathrm{tr}(\frac{\partial(d_{53})}{\partial q_5}\ I_5\ {}_{54}^{T}d)$

For i=4,j=3,k=4

$h_{434}=\sum_{p=\max(4,3,4)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=\ \mathrm{Tr}(\frac{\partial(d_{44})}{\partial q_4}\ I_4\ {}_{44}^{T}d)+\ \mathrm{tr}(\frac{\partial(d_{54})}{\partial q_5}\ I_5\ {}_{54}^{T}d)$

For i=4,j=3,k=5

$h_{435}=\sum_{p=\max(4,3,5)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=\ \mathrm{tr}(\frac{\partial(d_{55})}{\partial q_5}\ I_5\ {}_{54}^{T}d)$

$h_{434}=h_{443}, h_{435}=h_{453}$

For i=4,j=4,k=4

$h_{444}=\sum_{p=\max(4,4,4)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=\ \mathrm{Tr}(\frac{\partial(d_{44})}{\partial q_4}\ I_4\ {}_{44}^{T}d)+\ \mathrm{tr}(\frac{\partial(d_{54})}{\partial q_5}\ I_5\ {}_{54}^{T}d)$

For i=4,j=4,k=5

$h_{445}=\sum_{p=\max(4,4,5)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}^{T}_{pi}d\right]=\ \mathrm{tr}(\frac{\partial(d_{55})}{\partial q_5}\ I_5\ {}^{T}_{54}d)$

$h_{445} = h_{454},$

For i=4,j=5,k=5

$h_{455}=\sum_{p=\max(4,5,5)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}^{T}_{pi}d\right]=\ \mathrm{tr}(\frac{\partial(d_{55})}{\partial q_5}\ I_5\ {}^{T}_{54}d)$

The centrifugal acceleration coefficients and the Coriolis coefficients are

For i=4

$H_4 = \sum_{j=1}^{n}\sum_{k=1}^{n} h_{ijk}\,\dot{\theta}_j\dot{\theta}_k = h_{411}\dot{\theta}_1^{\,2} + h_{412}\dot{\theta}_1\dot{\theta}_2 + h_{413}\dot{\theta}_1\dot{\theta}_3 + h_{414}\dot{\theta}_1\dot{\theta}_4 + h_{415}\dot{\theta}_1\dot{\theta}_5 +$

$h_{421}\dot{\theta}_2\dot{\theta}_1 + h_{422}\dot{\theta}_2^{\,2} + h_{423}\dot{\theta}_2\dot{\theta}_3 + h_{424}\dot{\theta}_2\dot{\theta}_4 + h_{425}\dot{\theta}_2\dot{\theta}_5 + h_{431}\dot{\theta}_3\dot{\theta}_1 + h_{432}\dot{\theta}_3\dot{\theta}_2 +$

$h_{433}\,\dot{\theta}_3^{\,2} + h_{434}\dot{\theta}_3\dot{\theta}_4 + h_{435}\dot{\theta}_3\dot{\theta}_5 + h_{441}\dot{\theta}_4\dot{\theta}_1 + h_{442}\dot{\theta}_4\dot{\theta}_2 + h_{443}\dot{\theta}_4\dot{\theta}_3 + h_{444}\,\dot{\theta}_4^{\,2} +$

$h_{445}\dot{\theta}_4\dot{\theta}_5 + h_{451}\dot{\theta}_5\dot{\theta}_1 + h_{452}\dot{\theta}_5\dot{\theta}_2 + h_{453}\dot{\theta}_5\dot{\theta}_3 + h_{454}\dot{\theta}_5\dot{\theta}_4 + h_{455}\,\dot{\theta}_5^{\,2}$

The centrifugal acceleration coefficients and the Coriolis coefficients are

For i=5,j=1,k=1

$h_{511}=\sum_{p=\max(5,1,1)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}^{T}_{pi}d\right]=\ \mathrm{tr}(\frac{\partial(d_{51})}{\partial q_5}\ I_5\ {}^{T}_{55}d)$

For i=5,j=1,k=2

$h_{512}=\sum_{p=\max(5,1,2)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}^{T}_{pi}d\right]=\ \mathrm{tr}(\frac{\partial(d_{52})}{\partial q_5}\ I_5\ {}^{T}_{55}d)$

For i=5,j=1,k=3

$$h_{513}=\sum_{p=\max(5,1,3)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=\ \mathrm{tr}(\frac{\partial(d_{53})}{\partial q_5}\ I_5\ {}_{55}^{T}d)$$

For i=5,j=1,k=4

$$h_{514}=\sum_{p=\max(5,1,4)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=\ \mathrm{tr}(\frac{\partial(d_{54})}{\partial q_5}\ I_5\ {}_{55}^{T}d)$$

For i=5,j=1,k=5

$$h_{515}=\sum_{p=\max(5,1,5)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=\ \mathrm{tr}(\frac{\partial(d_{55})}{\partial q_5}\ I_5\ {}_{55}^{T}d)$$

$h_{512} = h_{521}, h_{513} = h_{531}, h_{514} = h_{541}, h_{515} = h_{551}$

For i=5,j=2,k=2

$$h_{522}=\sum_{p=\max(5,2,2)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=\ \mathrm{tr}(\frac{\partial(d_{52})}{\partial q_5}\ I_5\ {}_{55}^{T}d)$$

For i=5,j=2,k=3

$$h_{523}=\sum_{p=\max(5,2,3)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=\ \mathrm{tr}(\frac{\partial(d_{53})}{\partial q_5}\ I_5\ {}_{55}^{T}d)$$

For i=5,j=2,k=4

$$h_{524}=\sum_{p=\max(5,2,4)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=\ \mathrm{tr}(\frac{\partial(d_{54})}{\partial q_5}\ I_5\ {}_{55}^{T}d)$$

For i=5,j=2,k=5

$$h_{525}=\sum_{p=\max(5,2,5)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=\ \mathrm{tr}(\frac{\partial(d_{55})}{\partial q_5}\ I_5\ {}_{55}^{T}d)$$

$h_{523} =\ h_{532}, h_{524} = h_{542}, h_{525} = h_{552}$

For i=5,j=3,k=3

$h_{533}=\sum_{p=\max(5,3,3)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=\ \mathrm{tr}(\frac{\partial(d_{53})}{\partial q_5}\ I_5\ {}_{55}^{T}d)$

For i=5,j=3,k=4

$h_{534}=\sum_{p=\max(5,3,4)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=\ \mathrm{tr}(\frac{\partial(d_{54})}{\partial q_5}\ I_5\ {}_{55}^{T}d)$

For i=5,j=3,k=5

$h_{535}=\sum_{p=\max(5,3,5)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=\ \mathrm{tr}(\frac{\partial(d_{55})}{\partial q_5}\ I_5\ {}_{55}^{T}d)$

$h_{534} = h_{543}, h_{535} = h_{553}$

For i=5,j=4,k=4

$h_{544}=\sum_{p=\max(5,4,4)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=\ \mathrm{tr}(\frac{\partial(d_{54})}{\partial q_5}\ I_5\ {}_{55}^{T}d)$

For i=5,j=4,k=5

$h_{545}=\sum_{p=\max(5,4,5)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=\ \mathrm{tr}(\frac{\partial(d_{55})}{\partial q_5}\ I_5\ {}_{55}^{T}d)$

$h_{545} = h_{554},$

For i=5,j=5,k=5

$h_{555}=\sum_{p=\max(5,5,5)}^{5} Tr\left[\frac{\partial(d_{pk})}{\partial q_p}\ I_p\ {}_{pi}^{T}d\right]=\ \mathrm{tr}(\frac{\partial(d_{55})}{\partial q_5}\ I_5\ {}_{55}^{T}d)$

The centrifugal acceleration coefficients and the Coriolis coefficients are

For i=5

$$H_5 = \sum_{j=1}^{n}\sum_{k=1}^{n} h_{ijk}\,\dot{\theta}_j\dot{\theta}_k = h_{511}\dot{\theta_1}^{\,2} + h_{512}\dot{\theta}_1\dot{\theta}_2 + h_{513}\dot{\theta}_1\dot{\theta}_3 + h_{514}\dot{\theta}_1\dot{\theta}_4 + h_{515}\dot{\theta}_1\dot{\theta}_5 +$$

$$h_{521}\dot{\theta}_2\dot{\theta}_1 + h_{522}\dot{\theta_2}^{\,2} + h_{523}\dot{\theta}_2\dot{\theta}_3 + h_{524}\dot{\theta}_2\dot{\theta}_4 + h_{525}\dot{\theta}_2\dot{\theta}_5 + h_{531}\dot{\theta}_3\dot{\theta}_1 + h_{532}\dot{\theta}_3\dot{\theta}_2 +$$

$$h_{533}\,\dot{\theta_3}^{\,2} + h_{534}\dot{\theta}_3\dot{\theta}_4 + h_{535}\dot{\theta}_3\dot{\theta}_5 + h_{541}\dot{\theta}_4\dot{\theta}_1 + h_{542}\dot{\theta}_4\dot{\theta}_2 + h_{543}\dot{\theta}_4\dot{\theta}_3 + h_{544}\,\dot{\theta_4}^{\,2} +$$

$$h_{545}\dot{\theta}_4\dot{\theta}_5 + h_{551}\dot{\theta}_5\dot{\theta}_1 + h_{552}\dot{\theta}_5\dot{\theta}_2 + h_{553}\dot{\theta}_5\dot{\theta}_3 + h_{554}\dot{\theta}_5\dot{\theta}_4 + h_{555}\,\dot{\theta_5}^{\,2}$$

Thus , the Coriolis and centrifugal coefficient matrix H is therefore

$$H = \begin{bmatrix} H_1 \\ H_2 \\ H_3 \\ H_4 \\ H_5 \end{bmatrix}$$

APPENDIX H

Calculation of gravity Matrix

G_i = -$\sum_{p=i}^{n} m_p \; g d_{pi} \; {}^{p}_{p}\bar{r}_p$

Thus, the gravity coefficient matrix G is therefore

$$G = \begin{bmatrix} G_1 \\ G_2 \\ G_3 \\ G_4 \\ G_5 \end{bmatrix}$$

For i=1

G_1=-$\sum_{p=1}^{5} m_p \; g d_{pi} \; {}^{p}_{p}\bar{r}_p$=

-($m_1 \; g \; d_{11} \; {}^{1}_{1}\bar{r_1} + m_2 \; g \; d_{21} \; {}^{2}_{2}\bar{r_2} + m_3 \; g \; d_{31} \; {}^{3}_{3}\bar{r_3} + m_4 \; g \; d_{41} \; {}^{4}_{4}\bar{r_4} + m_5 \; g \; d_{51} \; {}^{5}_{5}\bar{r_5}$)

The mass of the link has been assumed to be concentrated at the centroid of the links. The centroid is located from the origins of frame{1} is

$${}^{1}_{1}\bar{r_1} = \begin{bmatrix} 0 & \frac{l_1}{2} & 0 & 1 \end{bmatrix}^{\mathrm{T}}$$

Similarly for the frame{2}, frame{3}, frame{4} and frame{5} are

$${}^{2}_{2}\bar{r_2} = \begin{bmatrix} 0 & 0 & \frac{l_2}{2} & 1 \end{bmatrix}^{\mathrm{T}} , {}^{3}_{3}\bar{r_3} = \begin{bmatrix} 0 & 0 & \frac{l_3}{2} & 1 \end{bmatrix}^{\mathrm{T}} , {}^{4}_{4}\bar{r_4} = \begin{bmatrix} 0 & 0 & \frac{l_4}{2} & 1 \end{bmatrix}^{\mathrm{T}}$$

$${}^{5}_{5}\bar{r_2} = \begin{bmatrix} 0 & 0 & \frac{l_5}{2} & 1 \end{bmatrix}^{\mathrm{T}}$$

And the gravity is in the positive z –direction giving $g = [0 \quad 0 \quad g \quad 1]$

For i=2

G_2 = -$\sum_{p=2}^{5} m_p \; g d_{pi} \; {}^{p}_{p}\bar{r}_p$= -($m_2 \; g \; d_{22} \; {}^{2}_{2}\bar{r_2} + m_3 \; g \; d_{32} \; {}^{3}_{3}\bar{r_3} + m_4 \; g \; d_{42} \; {}^{4}_{4}\bar{r_4}$ +

$m_5 \; g \; d_{52} \; {}^{5}_{5}\bar{r_5}$)

For i=3

G_3 = -$\sum_{p=3}^{5} m_p\ g d_{pi}\ {}^{p}_{p}\bar{r}_p$= -($m_3\ g\ d_{33}\ {}^{3}_{3}\bar{r_3} + m_4\ g\ d_{43}\ {}^{4}_{4}\bar{r_4} + m_5\ g\ d_{53}\ {}^{5}_{5}\bar{r_5}$)

For i=4

G_4 = -$\sum_{p=4}^{5} m_p\ g d_{pi}\ {}^{p}_{p}\bar{r}_p$= -($m_4\ g\ d_{44}\ {}^{4}_{4}\bar{r_4} + m_5\ g\ d_{54}\ {}^{5}_{5}\bar{r_5}$)

For i=5

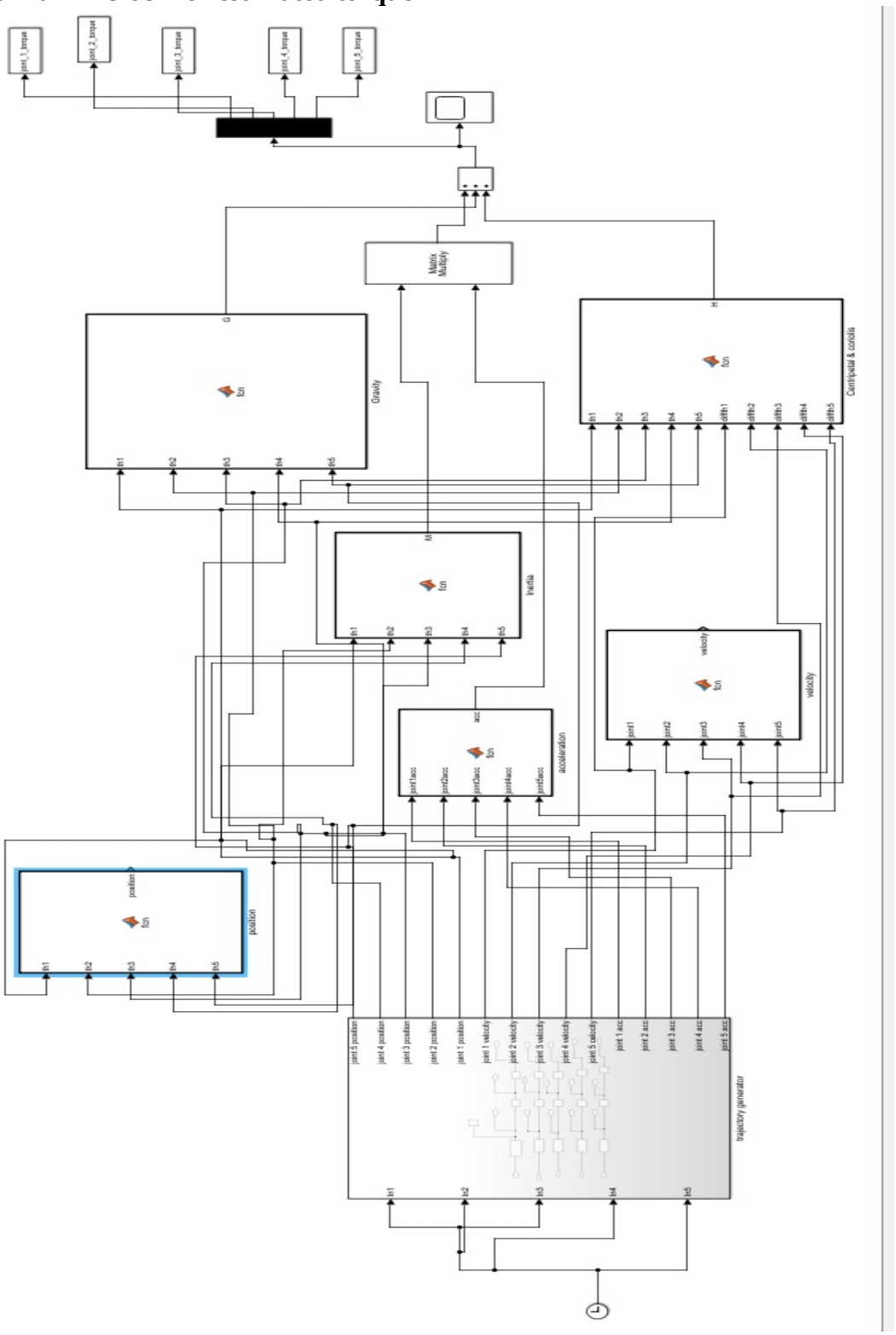
G_5 = -$\sum_{p=5}^{5} m_p\ g d_{pi}\ {}^{p}_{p}\bar{r}_p$= -($m_5\ g\ d_{55}\ {}^{5}_{5}\bar{r_5}$)

Thus , the gravity coefficient matrix G is therefore

$$G = \begin{bmatrix} G_1 \\ G_2 \\ G_3 \\ G_4 \\ G_5 \end{bmatrix}$$

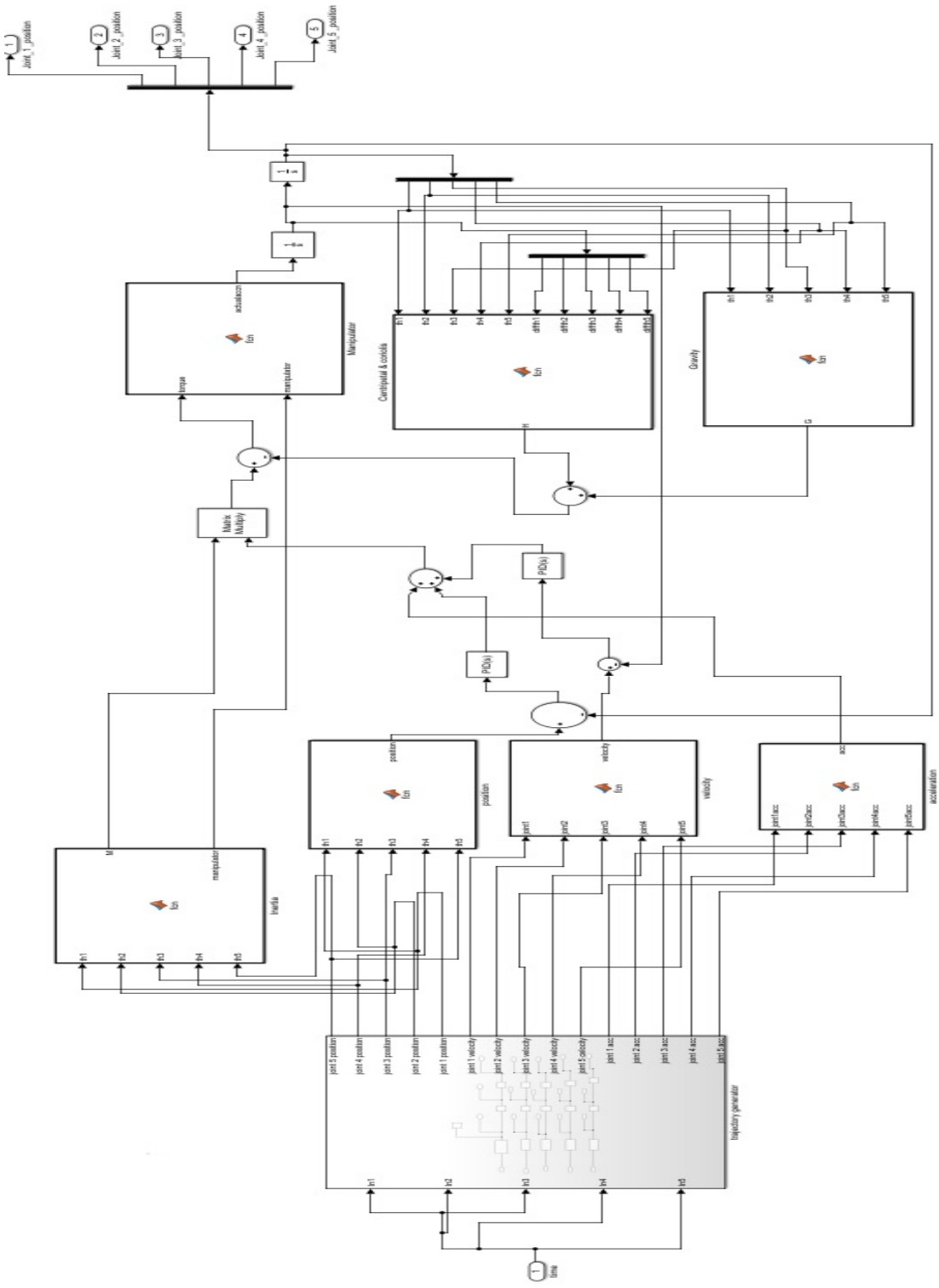
APPENDIX I

Simulink block for estimated torque



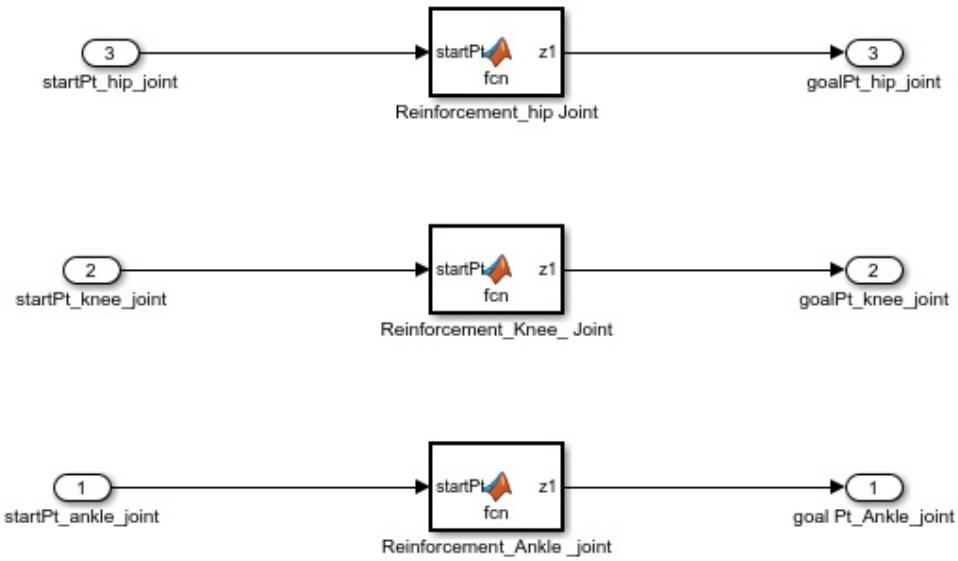
APPENDIX J

Simulink block diagram of computed torque control



APPENDIX K

Simulink Reinforcement Controller



MATLAB SCRIPT FOR EVALUATING THE KINEMATICS AND DYNAMICS

```
clear
clc
syms t  nx ny nz px py pz ax ay az ox oy oz real;
 m1=.5; m2=.3; m3=.4; m4=.2; m5=.2; g=-9.81;
 l1=.2; l2=.3; l3=.1; l4=.2; l5=.2; fx=0; fy=0; fz=-10;
% trajectory
 th1=deg2rad(-45+7.2*t.^3-2.16*t.^4+.1728*t.^5)
 th2=deg2rad(-15+9.2*t.^3-2.76*t.^4+.2208*t.^5)
 th3=deg2rad(3.6*t.^3-1.08*t.^4+.0864*t.^5)
 th4=deg2rad(9.6*t.^3-2.88*t.^4+.2304*t.^5)
 th5=deg2rad(-20+4.8*t.^3-1.44*t.^4+.1152*t.^5)
%forward kinematics
A0=[1 0 0 0 ;0 1 0 0;0 0 1 0;0 0 0 1]
A1=[cos(th1) -sin(th1) 0 0;sin(th1) cos(th1) 0 0;0 0 1 0;0 0 0 1]*[1
0 0 0;0 1 0 0;0 0 1 l1;0 0 0 1]*...
    [1 0 0 0;0 1 0 0;0 0 1 0;0 0 0 1]*[1 0 0 0;0 0 -1 0;0 1 0 0;0
0 0 1]
A2=[-sin(th2) -cos(th2) 0 0;cos(th2) -sin(th2) 0 0;0 0 1 0;0 0 0
1]*[1 0 0 0;0 1 0 0;0 0 1 l2;0 0 0 1]*...
    [1 0 0 0;0 1 0 0;0 0 1 0;0 0 0 1]*[1 0 0 0;0 0 1 0;0 -1 0 0;0
0 0 1]
A3=[sin(th3) cos(th3) 0 0;-cos(th3) sin(th3) 0 0;0 0 1 0;0 0 0 1]*[1
0 0 0;0 1 0 0;0 0 1 0;0 0 0 1]*...
    [1 0 0 l3;0 1 0 0;0 0 1 0;0 0 0 1]*[1 0 0 0;0 0 -1 0;0 1 0 0;0
0 0 1]
A4=[cos(th4) -sin(th4) 0 0;sin(th4) cos(th4) 0 0;0 0 1 0;0 0 0 1]*[1
0 0 0 ;0 1 0 0;0 0 1 0;0 0 0 1]*...
    [1 0 0 l4;0 1 0 0;0 0 1 0;0 0 0 1]*[1 0 0 0;0 1 0 0;0 0 1 0;0
0 0 1]
A5=[cos(th5) -sin(th5) 0 0;sin(th5) cos(th5) 0 0;0 0 1 0;0 0 0 1]*[1
0 0 0;0 1 0 0 ;0 0 1 0;0 0 0 1]*...
    [1 0 0 l5;0 1 0 0 ;0 0 1 0;0 0 0 1]*[1 0 0 0;0 1 0 0;0 0 1 0;0
0 0 1]
toe=A1*A2*A3*A4*A5
%inverse kinematics
```

```
theta1=atand((py-l5*ny)/(px-l5*nx))
theta41=atand((pz-l5*nz-l1)/(sqrt((px-l5*nx).^2)+((py-
l5*ny).^2)))
theta42=atand((pz-l5*nz-l1)/(-sqrt((px-l5*nx).^2)+((py-
l5*ny).^2)))
theta5=atand(((py+l5*ny)*l4*nx-
(px+l5*nx)*l4*ny)/((px+l5*nx)*l4*oy+(py+l5*ny)*l4*ox))
theta21=atand(-
(oz*cosd(theta41+theta5)+nz*sin(theta41+theta5))/(sqrt((ox*cosd(t
heta41+theta5)+nx*sin(theta41+theta5)).^2)+((oy*cosd(theta41+thet
a5)+ny*sin(theta41+theta5)).^2)))
theta22=atand(-
(oz*cosd(theta42+theta5)+nz*sin(theta42+theta5))/(sqrt((ox*cosd(t
heta42+theta5)+nx*sin(theta42+theta5)).^2)+((oy*cosd(theta42+thet
a5)+ny*sin(theta42+theta5)).^2)))
theta23=atand(-
(oz*cosd(theta41+theta5)+nz*sin(theta41+theta5))/(-
sqrt((ox*cosd(theta41+theta5)+nx*sin(theta41+theta5)).^2)+((oy*co
sd(theta41+theta5)+ny*sin(theta41+theta5)).^2)))
theta24=atand(-
(oz*cosd(theta42+theta5)+nz*sin(theta42+theta5))/(-
sqrt((ox*cosd(theta42+theta5)+nx*sin(theta42+theta5)).^2)+((oy*co
sd(theta42+theta5)+ny*sin(theta42+theta5)).^2)))
theta31=atand((nz*cos(theta41+theta5)+nz*sin(theta41+theta5))/-
az)
theta32=atand((nz*cos(theta42+theta5)+nz*sin(theta42+theta5))/-
az)
%calculation of jacobian1
P0=[1 0 0 ;0 1 0;0 0 1]*[0;0;1]
P5=A1*A2*A3*A4*A5*[0;0;0;1]-[1  0  0  0;0  1  0  0;0  0  1  0;0  0  0
1]*[0;0;0;1]
P5Jacob=diff(P5,th1)
P5Jacobb=P5Jacob(1:3,1)
J1=[P5Jacobb;P0]
calculation of jacobian2
P1=A1(1:3,1:3)*[0;0;1]
IP5=A1*A2*A3*A4*A5*[0;0;0;1]-A1*[0;0;0;1]
```

```
IP5jacob=diff(IP5,th2)
IP5jacobb=IP5jacob(1:3,1)
J2=[IP5jacobb;P1]
 %calculation of jacobian3
P2=A1(1:3,1:3)*A2(1:3,1:3)*[0;0;1]
IIP5=A1*A2*A3*A4*A5*[0;0;0;1]-A1*A2*[0;0;0;1]
IIP5jacob=diff(IIP5,th3)
IIP5jacobb=IIP5jacob(1:3,1)
J3=[IIP5jacobb;P2]
 %calculation of jacobian4
P3=A1(1:3,1:3)*A2(1:3,1:3)*A3(1:3,1:3)*[0;0;1]
IIIP5=A1*A2*A3*A4*A5*[0;0;0;1]-A1*A2*A3*[0;0;0;1]
IIIP5jacob=diff(IIIP5,th4)
IIIP5jacobb=IIIP5jacob(1:3,1)
J4=[IIIP5jacobb;P3]
%calculation of jacobian5
P4=A1(1:3,1:3)*A2(1:3,1:3)*A3(1:3,1:3)*A4(1:3,1:3)*[0;0;1]
IVP5=A1*A2*A3*A4*A5*[0;0;0;1]-A1*A2*A3*A4*[0;0;0;1]
IVP5jacob=diff(IVP5,th5)
IVP5jacobb=IVP5jacob(1:3,1)
J5=[IVP5jacobb;P4]
J=[J1 J2 J3 J4 J5]
 %calculation of static torque
F=[fx fy fz 0 0 0]'
T=J'*F
%calculation of dynamic  torque
%since all the joints are revolute
Q1=[0 -1 0 0;1 0 0 0;0 0 0 0;0 0 0 0]
Q2=[0 -1 0 0;1 0 0 0;0 0 0 0;0 0 0 0]
Q3=[0 -1 0 0;1 0 0 0;0 0 0 0;0 0 0 0]
Q4=[0 -1 0 0;1 0 0 0;0 0 0 0;0 0 0 0]
Q5=[0 -1 0 0;1 0 0 0;0 0 0 0;0 0 0 0]
I1=[1/3*m1*l1.^2 0 0 -1/2*m1*l1;0 0 0 0;0 0 0 0;-1/2*m1*l1 0 0 m1]
I2=[1/3*m2*l2.^2 0 0 -1/2*m2*l2;0 0 0 0;0 0 0 0;-1/2*m2*l2 0 0 m2]
I3=[1/3*m3*l3.^2 0 0 -1/2*m3*l3;0 0 0 0;0 0 0 0;-1/2*m3*l3 0 0 m3]
I4=[1/3*m4*l4.^2 0 0 -1/2*m4*l4;0 0 0 0;0 0 0 0;-1/2*m4*l4 0 0 m4]
I5=[1/3*m5*l5.^2 0 0 -1/2*m5*l5;0 0 0 0;0 0 0 0;-1/2*m5*l5 0 0 m5]
```

```
d11=A0*Q1*A1
d21=A0*Q1*A1*A2
d31=A0*Q1*A1*A2*A3
d41=A0*Q1*A1*A2*A3*A4
d51=A0*Q1*A1*A2*A3*A4*A5
d22=A1*Q2*A2
d32=A1*Q2*A2*A3
d42=A1*Q2*A2*A3*A4
d52=A1*Q2*A2*A3*A4*A5
d33=A1*A2*Q3*A3
d43=A1*A2*Q3*A3*A4
d53=A1*A2*Q3*A3*A4*A5
d44=A1*A2*A3*Q4*A4
d54=A1*A2*A3*Q4*A4*A5
d55=A1*A2*A3*A4*Q5*A5
%calculation of inertia matrix
M11=trace(d11*I1*d11')+trace(d21*I2*d21')+trace(d31*I3*d31')+trac
e(d41*I4*d41')+trace(d51*I5*d51')
M12=trace(d22*I2*d21')+trace(d32*I3*d31')+trace(d42*I4*d41')+trac
e(d52*I5*d51')
M13=trace(d33*I3*d31')+trace(d43*I4*d41')+trace(d53*I5*d51')
M14=trace(d44*I4*d41')+trace(d54*I5*d51')
M15=trace(d55*I5*d51')
%since coupling inertia are same hence
M21=M12
M31=M13
M41=M14
M51=M15
M22=trace(d22*I2*d22')+trace(d32*I3*d32')+trace(d42*I4*d42')+trac
e(d52*I5*d52')
M23=trace(d33*I3*d32')+trace(d43*I4*d42')+trace(d53*I5*d52')
M24=trace(d44*I4*d42')+trace(d54*I5*d52')
M25=trace(d55*I5*d52')
%since coupling inertia are same hence
M32=M23
M42=M24
M52=M25
```

```
M33=trace(d33*I3*d33')+trace(d43*I4*d43')+trace(d55*I5*d53')
M34=trace(d44*I4*d43')+trace(d54*I5*d53')
M35=trace(d55*I5*d53')
%since coupling inertia are same hence
M43=M34
M53=M35
M44=trace(d44*I4*d44')+trace(d54*I5*d54')
M45=trace(d55*I5*d54')
%since coupling inertia are same hence
M54=M45
M55=trace(d55*I5*d55')
% calculation of inertia matrix
M1=[M11 M12 M13 M14 M15]
M2=[M21 M22 M23 M24 M25]
M3=[M31 M32 M33 M34 M35]
M4=[M41 M42 M43 M44 M45]
M5=[M51 M52 M53 M54 M55]
M=[M11 M12 M13 M14 M15;M21 M22 M23 M24 M25;M31 M32 M33 M34 M35;M41
M42 M43 M44 M45;M51 M52 M53 M54 M55]
%calculation of coriolis and centrifugal coefficients
h111=trace(A0*Q1*Q1*A1*I1*d11')+trace(A0*Q1*A1*Q2*A2*I2*d21')+tra
ce(A0*Q1*A1*A2*Q3*A3*I3*d31')+trace(A0*Q1*A1*A2*A3*Q4*A4*I4*d41')
+...
    trace(A0*Q1*A1*A2*A3*A4*Q5*A5*I5*d51')
h112=trace(A1*Q2*Q2*A2*I2*d21')+trace(A1*Q2*A2*Q3*A3*I3*d31')+tra
ce(A1*Q2*A2*A3*Q4*A4*I4*d41')+trace(A1*Q2*A2*A3*A4*Q5*I5*d51')
h113=trace(A1*A2*Q3*Q3*A3*I3*d31')+trace(A1*A2*Q3*A3*Q4*A4*I4*d41
')+trace(A1*A2*Q3*A3*A4*Q5*A5*I5*d51')
h114=trace(A1*A2*A3*Q4*Q4*A4*I4*d41')+trace(A1*A2*A3*Q4*A4*Q5*A5*
I5*d51')
h115=trace(A1*A2*A3*A4*Q5*Q5*A5*I5*d51')
%since the effects of one joint over another joints are equal hence
h121=h112
h131=h113
h141=h114
h151=h115
```

```
h122=trace(A1*Q2*Q2*A2*I2*d21')+trace(A1*Q2*A2*Q3*A3*I3*d31')+tra
ce(A1*Q2*A2*A3*Q4*A4*I4*d41')+trace(A1*Q2*A2*A3*A4*Q5*A5*I5*d51')
h123=trace(A1*A2*Q3*Q3*A3*I3*d31')+trace(A1*A2*Q3*A3*Q4*A4*I4*d41
')+trace(A1*A2*Q3*A3*A4*Q5*A5*I5*d51')
h124=trace(A1*A2*A3*Q4*Q4*A4*I4*d41')+trace(A1*A2*A3*Q4*A4*Q5*A5*
I5*d51')
h125=trace(A1*A2*A3*A4*Q5*Q5*A5*I5*d51')
h132=h123
h142=h124
h152=h125
h133=trace(A1*A3*Q3*Q3*A3*I3*d31')+trace(A1*A2*Q3*A3*Q4*A4*I4*d41
')+trace(A1*A2*Q3*A3*A4*Q5*A5*I5*d51')
h134=trace(A1*A2*A3*Q4*Q4*A4*I4*d41')+trace(A1*A2*A3*Q4*A4*Q5*A5*
I5*d51')
h135=trace(A1*A2*A3*A4*Q5*Q5*A5*I5*d51')
h143=h134
h153=h135
h144=trace(A1*A2*A3*Q4*Q4*A4*I4*d41')+trace(A1*A2*A3*Q4*A4*Q5*A5*
I5*d51')
h145=trace(A1*A2*A3*A4*Q5*Q5*A5*I5*d51')
h154=h145
h155=trace(A1*A2*A3*A4*Q5*Q5*A5*I5*d51')
H1=h111*diff(th1)*diff(th1)+h112*diff(th1)*diff(th2)+h113*diff(th
1)*diff(th3)+h114*diff(th1)*diff(th4)+h115*diff(th1)*diff(th5)+..
.

h121*diff(th2)*diff(th1)+h122*diff(th2)*diff(th2)+h123*diff(th2)*
diff(th3)+h124*diff(th2)*diff(th4)+h125*diff(th2)*diff(th5)+...

h131*diff(th1)*diff(th1)+h132*diff(th3)*diff(th2)+h133*diff(th3)*
diff(th3)+h134*diff(th3)*diff(th4)+h135*diff(th3)*diff(th5)+...

h141*diff(th4)*diff(th1)+h142*diff(th4)*diff(th2)+h143*diff(th4)*
diff(th3)+h144*diff(th4)*diff(th4)+h145*diff(th4)*diff(th5)+...

h151*diff(th5)*diff(th1)+h152*diff(th5)*diff(th2)+h153*diff(th5)*
diff(th3)+h154*diff(th5)*diff(th4)+h155*diff(th5)*diff(th5)
```

```
h211=trace(A0*Q1*A1*Q2*A2*I2*d22')+trace(A0*Q1*A1*A2*Q3*A3*I3*d32
')+trace(A0*Q1*A1*A2*A3*Q4*A4*I4*d42')+trace(A0*Q1*A1*A2*A3*A4*Q5
*A5*I5*d52')
h212=trace(A1*Q2*Q2*A2*I2*d22')+trace(A1*Q2*A2*Q3*A3*I3*d32')+tra
ce(A1*Q2*A2*A3*Q4*A4*I4*d42')+trace(A1*Q2*A2*A3*A4*Q5*A5*I5*d52')
h213=trace(A1*A2*Q3*Q3*A3*I3*d32')+trace(A1*A2*Q3*A3*Q4*A4*I4*d42
')+trace(A1*A2*Q3*A3*A4*Q5*A5*I5*d52')
h214=trace(A1*A2*A3*Q4*Q4*A4*I4*d42')+trace(A1*A2*A3*Q4*A4*Q5*A5*
I5*d52')
h215=trace(A1*A2*A3*A4*Q5*Q5*A5*I5*d52')
h221=h212
h231=h213
h241=h214
h251=h215
h222=trace(A1*Q2*Q2*A2*I2*d22')+trace(A1*Q2*A2*Q3*A3*I3*d32')+tra
ce(A1*Q2*A2*A3*Q4*A4*I4*d42')+trace(A1*Q2*A2*A3*A4*Q5*A5*I5*d52')
h223=trace(A1*A2*Q3*Q3*A3*I3*d32')+trace(A1*A2*Q3*A3*Q4*A4*I4*d42
')+trace(A1*A2*Q3*A3*A4*Q5*A5*I5*d52')
h224=trace(A1*A2*A3*Q4*Q4*A4*I4*d42')+trace(A1*A2*A3*Q4*A4*Q5*A5*
I5*d52')
h225=trace(A1*A2*A3*A4*Q5*Q5*A5*I5*d52')
h232=h223
h242=h224
h252=h225
h233=trace(A1*A2*Q3*Q3*A3*I3*d32')+trace(A1*A2*Q3*A3*Q4*A4*I4*d42
')+trace(A1*A2*Q3*A3*A4*Q5*A5*I5*d52')
h234=trace(A1*A2*A3*Q4*Q4*A4*I4*d42')+trace(A1*A2*A3*Q4*A4*Q5*A5*
I5*d52')
h235=trace(A1*A2*A3*A4*Q5*Q5*A5*I5*d52')
h243=h234
h253=h235
h244=trace(A1*A2*A3*Q4*Q4*A4*I4*d42')+trace(A1*A2*A3*Q4*A4*Q5*A5*
I5*d52')
h245=trace(A1*A2*A3*A4*Q5*Q5*A5*I5*d52')
h254=h245
h255=trace(A1*A2*A3*A4*Q5*Q5*A5*I5*d52')
```

```
H2=h211*diff(th1)*diff(th1)+h212*diff(th1)*diff(th2)+h213*diff(th
1)*diff(th3)+h214*diff(th1)*diff(th4)+h215*diff(th1)*diff(th5)+..
.

h221*diff(th2)*diff(th1)+h222*diff(th2)*diff(th2)+h223*diff(th2)*
diff(th3)+h224*diff(th2)*diff(th4)+h225*diff(th2)*diff(th5)+...

h231*diff(th1)*diff(th1)+h232*diff(th3)*diff(th2)+h233*diff(th3)*
diff(th3)+h234*diff(th3)*diff(th4)+h235*diff(th3)*diff(th5)+...

h241*diff(th4)*diff(th1)+h242*diff(th4)*diff(th2)+h243*diff(th4)*
diff(th3)+h244*diff(th4)*diff(th4)+h245*diff(th4)*diff(th5)+...

h251*diff(th5)*diff(th1)+h252*diff(th5)*diff(th2)+h253*diff(th5)*
diff(th3)+h254*diff(th5)*diff(th4)+h255*diff(th5)*diff(th5)
h311=trace(A0*Q1*A1*A2*Q3*A3*I3*d33')+trace(A0*Q1*A1*A2*A3*Q4*A4*
I4*d43')+trace(A0*Q1*A1*A2*A3*A4*Q5*A5*I5*d53')
h312=trace(A1*Q2*A2*Q3*A3*I3*d33')+trace(A1*Q2*A2*A3*Q4*A4*I4*d43
')+trace(A1*Q2*A3*A4*Q5*A5*I5*d53')
h313=trace(A1*A2*Q3*Q3*A3*I3*d33')+trace(A1*A2*Q3*A3*Q4*A4*I4*d43
')+trace(A1*A2*Q3*A3*A4*Q5*A5*I5*d53')
h314=trace(A1*A2*A3*Q4*Q4*A4*I4*d43')+trace(A1*A2*A3*Q4*A4*Q5*A5*
I5*d53')
h315=trace(A1*A2*A3*A4*Q5*Q5*A5*I5*d53')
h321=h312
h331=h313
h341=h314
h351=h315
h322=trace(A1*Q2*A2*Q3*A3*I3*d33')+trace(A1*Q2*A2*A3*Q4*A4*I4*d43
')+trace(A1*Q2*A2*A3*A4*Q5*A5*I5*d53')
h323=trace(A1*A2*Q3*Q3*I3*d33')+trace(A1*A2*Q3*A3*Q4*A4*I4*d43')+
trace(A1*A2*Q3*A3*A4*Q5*A5*I5*d53')
h324=trace(A1*A2*A3*Q4*Q4*A4*I4*d43')+trace(A1*A2*A3*Q4*A4*Q5*A5*
I5*d53')
h325=trace(A1*A2*Q3*A3*A4*Q5*A5*I5*d53')
h332=h323
h342=h324
```

```
h352=h325
h333=trace(A1*A2*Q3*Q3*A3*I3*d33')+trace(A1*A2*Q3*A3*Q4*A4*I4*d43
')+trace(A1*A2*Q3*A3*A4*Q5*A5*I5*d53')
h334=trace(A1*A2*A3*Q4*Q4*A4*I4*d43')+trace(A1*A2*A3*Q4*A4*Q5*A5*
I5*d53')
h335=trace(A1*A2*A3*A4*Q5*Q5*A5*I5*d53')
h343=h334
h353=h335
h344=trace(A1*A2*A3*Q4*Q4*A4*I4*d43')+trace(A1*A2*A3*Q4*A4*Q5*A5*
I5*d53')
h345=trace(A1*A2*A3*A4*Q5*Q5*A5*d53')
h354=h345
h355=trace(A1*A2*A3*A4*Q5*Q5*A5*I5*d53')
H3=h311*diff(th1)*diff(th1)+h312*diff(th1)*diff(th2)+h313*diff(th
1)*diff(th3)+h314*diff(th1)*diff(th4)+h315*diff(th1)*diff(th5)+..
.

h321*diff(th2)*diff(th1)+h322*diff(th2)*diff(th2)+h323*diff(th2)*
diff(th3)+h324*diff(th2)*diff(th4)+h325*diff(th2)*diff(th5)+...

h331*diff(th1)*diff(th1)+h332*diff(th3)*diff(th2)+h333*diff(th3)*
diff(th3)+h334*diff(th3)*diff(th4)+h335*diff(th3)*diff(th5)+...

h341*diff(th4)*diff(th1)+h342*diff(th4)*diff(th2)+h343*diff(th4)*
diff(th3)+h344*diff(th4)*diff(th4)+h345*diff(th4)*diff(th5)+...

h351*diff(th5)*diff(th1)+h352*diff(th5)*diff(th2)+h353*diff(th5)*
diff(th3)+h354*diff(th5)*diff(th4)+h355*diff(th5)*diff(th5)


h411=trace(A0*Q1*A1*A2*A3*Q4*A4*I4*d44')+trace(A0*Q1*A1*A2*A3*A4*
Q5*A5*I5*d54')
h412=trace(A1*Q2*A2*A3*Q4*A4*I4*d44')+trace(A1*Q2*A2*A3*A4*Q5*A5*
I5*d54')
h413=trace(A1*A2*Q3*A3*Q4*A4*I4*d44')+trace(A1*A2*Q3*A3*A4*Q5*A5*
I5*d54')
h414=trace(A1*A2*A3*Q4*Q4*A4*I4*d44')+trace(A1*A2*A3*Q4*A4*Q5*A5*
I5*d54')
```

```
h415=trace(A1*A2*A3*A4*Q5*Q5*A5*I5*d54')
h421=h412
h431=h413
h441=h414
h451=h415
h422=trace(A1*Q2*A2*A3*Q4*A4*I4*d44')+trace(A1*Q2*A2*A3*A4*Q5*A5*
I5*d54')
h423=trace(A1*A2*Q3*A3*Q4*A4*I4*d44')+trace(A1*A2*Q3*A3*A4*Q5*A5*
I5*d54')
h424=trace(A1*A2*A3*Q4*Q4*A4*I4*d44')+trace(A1*A2*A3*Q4*A4*Q5*A5*
I5*d54')
h425=trace(A1*A2*A3*A4*Q5*Q5*A5*I5*d54')
h432=h423
h442=h424
h452=h425
h433=trace(A1*A2*Q3*A3*Q4*A4*I4*d44')+trace(A1*A2*Q3*A3*A4*Q5*A5*
I5*d54')
h434=trace(A1*A2*A3*Q4*Q4*A4*I4*d44')+trace(A1*A2*A3*Q4*A4*Q5*A5*
I5*d54')
h435=trace(A1*A2*A3*A4*Q5*Q5*A5*I5*d54')
h443=h434
h453=h435
h444=trace(A1*A2*A3*Q4*Q4*A4*I4*d44')+trace(A1*A2*A3*Q4*A4*Q5*A5*
I5*d54')
h445=trace(A1*A2*A3*A4*Q5*Q5*A5*I5*d54')
h454=h445
h455=trace(A1*A2*A3*A4*Q5*Q5*A5*I5*d54')
H4=h411*diff(th1)*diff(th1)+h412*diff(th1)*diff(th2)+h413*diff(th
1)*diff(th3)+h414*diff(th1)*diff(th4)+h415*diff(th1)*diff(th5)+..
.

h421*diff(th2)*diff(th1)+h422*diff(th2)*diff(th2)+h423*diff(th2)*
diff(th3)+h424*diff(th2)*diff(th4)+h425*diff(th2)*diff(th5)+...

h431*diff(th1)*diff(th1)+h432*diff(th3)*diff(th2)+h433*diff(th3)*
diff(th3)+h434*diff(th3)*diff(th4)+h435*diff(th3)*diff(th5)+...
```

```
h441*diff(th4)*diff(th1)+h442*diff(th4)*diff(th2)+h443*diff(th4)*
diff(th3)+h444*diff(th4)*diff(th4)+h445*diff(th4)*diff(th5)+...

h451*diff(th5)*diff(th1)+h452*diff(th5)*diff(th2)+h453*diff(th5)*
diff(th3)+h454*diff(th5)*diff(th4)+h455*diff(th5)*diff(th5)
h511=trace(A0*Q1*A1*A2*A3*A4*Q5*A5*I5*d55')
h512=trace(A1*Q2*A2*A3*A4*Q5*A5*I5*d55')
h513=trace(A1*A2*Q3*A3*A4*Q5*A5*I5*d55')
h514=trace(A1*A2*A3*Q4*A4*Q5*A5*I5*d55')
h515=trace(A1*A2*A3*A4*Q5*Q5*A5*I5*d55')
h521=h512
h531=h513
h541=h514
h551=h515
h522=trace(A1*Q2*A2*A3*A4*Q5*A5*I5*d55')
h523=trace(A1*A2*Q3*A3*A4*Q5*A5*I5*d55')
h524=trace(A1*A2*A3*Q4*A4*Q5*A5*I5*d55')
h525=trace(A1*A2*A3*A4*Q5*Q5*A5*I5*d55')
h532=h523
h542=h524
h552=h525
h533=trace(A1*A2*Q3*A3*A4*Q5*A5*I5*d55')
h534=trace(A1*A2*A3*Q4*A4*Q5*A5*I5*d55')
h535=trace(A1*A2*A3*A4*Q5*Q5*A5*I5*d55')
h543=h534
h553=h535
h544=trace(A1*A2*A3*Q4*A4*Q5*A5*I5*d55')
h545=trace(A1*A2*A3*A4*Q5*Q5*A5*I5*d55')
h554=h545
h555=trace(A1*A2*A3*A4*Q5*Q5*A5*I5*d55')
H5=h511*diff(th1)*diff(th1)+h512*diff(th1)*diff(th2)+h513*diff(th
1)*diff(th3)+h514*diff(th1)*diff(th4)+h515*diff(th1)*diff(th5)+..
.

h521*diff(th2)*diff(th1)+h522*diff(th2)*diff(th2)+h523*diff(th2)*
diff(th3)+h524*diff(th2)*diff(th4)+h525*diff(th2)*diff(th5)+...
```

```
h531*diff(th1)*diff(th1)+h532*diff(th3)*diff(th2)+h533*diff(th3)*
diff(th3)+h534*diff(th3)*diff(th4)+h535*diff(th3)*diff(th5)+...

h541*diff(th4)*diff(th1)+h542*diff(th4)*diff(th2)+h543*diff(th4)*
diff(th3)+h544*diff(th4)*diff(th4)+h545*diff(th4)*diff(th5)+...

h551*diff(th5)*diff(th1)+h552*diff(th5)*diff(th2)+h553*diff(th5)*
diff(th3)+h554*diff(th5)*diff(th4)+h555*diff(th5)*diff(th5)
H=[H1;H2;H3;H4;H5]
% calculation of gravity term
r1=[0 l1/2 0 1]'
r2=[0 0 l2/2 1]'
r3=[0 0 l3/2 1]'
r4=[0 0 l4/2 1]'
r5=[0 0 l5/2 1]'
g=[0 0 g 1]
G1=-(m1*g*d11*r1+m2*g*d21*r2+m3*g*d31*r3+m4*g*d41*r4+m5*g*d51*r5)
G2=-(m2*g*d22*r2+m3*g*d32*r3+m4*g*d42*r4+m5*g*d52*r5)
G3=-(m3*g*d33*r3+m4*g*d43*r4+m5*g*d53*r5)
G4=-(m4*g*d44*r4+m5*g*d54*r5)
G5=-(m5*g*d55*r5)
G=[G1;G2;G3;G4;G5]
%evaluation of torque
theta=[diff(th1,2);diff(th2,2);diff(th3,2);diff(th4,2);diff(th5,2
)]
dynamicTorque1=M1*theta+H1+G1
dynamicTorque2=M2*theta+H2+G2
dynamicTorque3=M3*theta+H3+G3
dynamicTorque4=M4*theta+H4+G4
dynamicTorque5=M5*theta+H5+G5
```