Static and Dynamic Obstacle Avoidance with

Dynamic Analysis for Mobile Robots

A Thesis

Submitted in partial fulfilment of the requirements for the award of degree of

Doctor of Philosophy

in

Mechanical Engineering

Submitted by

Rajmeet Singh

Regd. No. 951208005

Under the guidance of

Dr. Tarun Kumar Bera

Associate Professor
Department of Mechanical Engineering
Thapar Institute of Engineering & Technology
(Deemed to be University), Patiala

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

Department of Mechanical Engineering
Thapar Institute of Engineering & Technology
(Deemed to be University)
Patiala-147004, India
October 2019

This thesis is dedicated to

Jasgun, Mani

and

Mom, Dad

DECLARATION

I certify that

a. the work contained in this Thesis is original and has been done by me under the guidance of my supervisor.
b. the work has not been submitted to any other Institute for any degree or diploma.
c. I have followed the guidelines provided by the Institute in preparing the Thesis.
d. I have confirmed to the norms and guidelines given in the Ethical Code of Conduct of the Institute.
e. whenever I have used materials (data, theoretical analysis, figures and text) from other sources, I have given due credit to them by citing them in the text of the Thesis and giving their details in the references. Further, I have taken permission from the copyright owners of the sources, whenever necessary.

17/10/19

(Rajmeet Singh)

CERTIFICATE

This is to certify that the Thesis entitled **'Static and Dynamic Obstacle Avoidance with Dynamic Analysis for Mobile Robots'**, submitted by **Mr. Rajmeet Singh** to Thapar Institute of Engineering & Technology (Deemed to be University), Patiala, India, is a record of bona fide research work carried under my supervision and is worthy of consideration for the award of the degree of Doctor of Philosophy of the Institute.

Tarun Kumar Bera

Dr. Tarun Kumar Bera

Associate Professor
Mechanical Engineering Department
Thapar Institute of Engineering & Technology (Deemed to be University),
Patiala, India, 147004
Email: tkbera@thapar.edu

Date: 17-10-2019

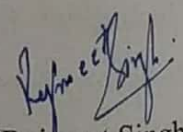
ACKNOWLEDGEMENT

First of all, I wish to thank God for given me an opportunity to carry present research work. I am greatly obliged to the authorities of Thapar Institute of Engineering & Technology (Deemed to be University), Patiala for providing me a chance of being a member of this research project and for providing the facilities for the completion of the work. Words are falling short to express my deepest regards and a sense of gratitude to my supervisor *Dr. Tarun Kumar Bera, Associate Professor, Department of Mechanical Engineering, Thapar Institute of Engineering & Technology (Deemed to be University), Patiala* for his fruitful suggestions, unfailing inspiration, whole-hearted cooperation, thought provoking discussions and painstaking supervision for the entire period of this work and making every tough work interesting. His deep insight into the problem and the ability to provide constructive suggestions have been of immense value in improving the quality of my research work at all stages.

I would be obliged to express my genuine thanks to Director, Thapar Institute of Engineering & Technology (TIET), Patiala, Dean, Research and Sponsored Projects, TIET, Patiala, Head, Mechanical Engineering Department, TIET, Patiala and PhD Coordinator, Mechanical Engineering Department, TIET, Patiala for their encouragement and support during this research work. I am also thankful to the authors whose research has helped me a lot. I also wish to express my deep sense of gratitude and respect to the members of Doctoral Committee, Dr. Anant Kumar Singh (Associate Professor, Mechanical Engineering Department, TIET, Patiala), Dr. Ashish Singla (Associate Professor, Mechanical Engineering Department, TIET, Patiala) and Dr. Naveen Kwatra (Professor, Civil Engineering Department, TIET, Patiala) for their unreserved guidance and productive discussions during the period of research. I also take this opportunity to thank the entire faculty and staff of Mechanical Engineering Department, TIET, Patiala, for their help, inspiration and moral support throughout this period.

In the end, I have the privilege to show regards and express a sincere thanks to my family, for supporting, motivating and encouraging me at every step of my work and always cooperating with me in every situation. No amount of words is enough to thank my wife *Manveen Kaur*, who is always a key behind my every successful work. She smoothly managed my house and also took care of my daughter, *Jasgun*. I had lost all hope and courage to do anything at the beginning of my research. But, it is the power of her support that I would have been able to complete my research work and dissertation with courage, confidence and zeal for hard work.

I would also like to thank my Father (S. Manjeet Singh), Mother (Rajinder Kaur), Father In-Law (S. Amarjeet Singh), Mother In- Law (Rajwinder Kaur) and Maninderpal Singh, Chandandeep Kaur, Jaskaran Singh, Navpreet Kaur and my beloved *Sahib Singh* and *Niyamat Kaur*, to encourage and support me during my research work. I express gratitude to Mr. Manarshhjot Singh, Mr. Vikram Singh Jamwal, Mr. Joypreet Singh, Mr. Jatinder Kataria, Mr. Taninder Singh, Ms. Rashmi Arora, Mr. Prakhar Jain and Ms. Garima Soharu for their help and open discussions. So, I finally thank all the people, who helped me directly or indirectly, in finishing this research work.



(Rajmeet Singh)

ABSTRACT

The self-driving car or robot is a key research area for the researchers due to increasing number of road accidents and erroneous driving. One of the major challenges in designing of a mobile robot is the autonomous travel of the robot. The key factor associated with the autonomous travel is the obstacle avoidance. The obstacle avoidance is classified as obstacle detection and avoidance control. Also, the dynamics of the mobile robot plays an important role during obstacle avoidance. A comprehensive literature review on mobile robot, obstacle detection and avoidance is done. Literature review on the overwhelming controller, physical model reduction, walking mobile robot, fault detection and isolation and bond graph methodology is also done. Accordingly, the objectives of the work are defined at the end of this literature review. The hybrid obstacle avoidance algorithm for static objects and fuzzy based obstacle avoidance algorithm for dynamic objects are proposed in this Thesis work. The model of bicycle vehicle model and four-wheel model of mobile robot are developed using bond graph theory. The effectiveness of the obstacle avoidance algorithm is tested on these models. The validation of the simulation results for obstacle detection and avoidance is also done with the experimentation-work. The obstacle avoidance for mobile robots is also done using overwhelming controller. Also, the quadruped robot based on Jansen linkage mechanism is developed and is used for obstacle avoidance application in a closed boundary-environment. The fuzzy logic controller with a set of rules is proposed to avoid the boundaries (static obstacle) of a closed environment. The simulation results are compared with the experimental result. The physical models for mobile robots are also reduced using eigenvalue sensitivity method to reduce the simulation time and system complexity. The response of the reduced model is compared with the full model. The dynamic model of the inverted pendulum with proportional-integral controller is developed and the results are validated with the experimental results. Further, the inverted pendulum model is extended to develop two-wheel mobile robot. The dynamic model of quadruped robot to study the performance of different gaits is done additionally. It is observed that the dynamically balanced gaits are faster and more energy efficient than statically stable gaits. Afterwards, the dynamic model for biped robot is developed and fault detection and isolation technique is implemented on this model introducing a fault in the leg's actuator during movement of the robot along a curved path. Finally, the conclusions of the Thesis are presented in the end with the scope for future research work.

Keywords: Mobile robot, obstacle avoidance, hybrid controller, fuzzy logic controller, overwhelming controller, model order reduction, fault detection, isolation and reconfiguration, bond graph.

LIST OF ABBREVIATIONS

ABS	Antilock Braking System
ALV	Autonomous Land Vehicle
ALVINN	Autonomous Land Vehicle in a Neural Network
ANN	Artificial Neural Network
ARMA	Auto Regressive Moving Average
ARR	Analytical Redundancy Relations
ATV	All-Terrain Vehicle
AV	Autonomous Vehicles
BG	Bond Graph
CAD	Computer Aided Design
CG	Centre of Gravity
CTF	Coordinate Transformation
DOF	Degrees-of-Freedom
DOT	Department of Transportation
De	Effort Detector
Df	Flow Detector
EGFO	Evolutionary Gabor Filter Optimization
EKFs	Extended Kalman Filters
EL	Euler-Lagrangian
FAR	Fault Accommodation by Reconfiguration
FDI	Fault Detection and Isolation
FFT	Fast Fourier Transform
FL	Fuzzy Logic
FSM	Fault Signature Matrix
FTC	Fault Tolerant Control
GA	Genetic Algorithm
GOLD	Generic Obstacle and Lane Detection
GPS	Global Positioning System

GY	Gyrator
IPM	Inverse Perspective Mapping
ITS	Intelligent Transportation Systems
IVI	Intelligent Vehicle Initiative Program
LIDAR	Laser Imaging Detection and Ranging
L-MSA	Low to Medium Speed Automation
LTI	Linear Time Invariant
MR	Mobile Robot
MTF	Modulated Transformer
NASA	National Aeronautics and Space Administration
NE	Newton-Euler
OAA	Obstacle Avoidance Algorithm
PI	Proportional Integral
PS	Physical Signal
PSO	Particle Swarm Optimization
R	Resistance
ROI	Regions of Interest
Se, SE	Source of Effort
Sf, SF	Source of Flow
SIMO	Single Input and Multiple Output
SPKF	Sigma-Point Kalman Filter
SPRHKF	Sigma-Point-Based Receding Hybrid Kalman Filter
SR-UKF	Square Root Unscented Kalman Filter
TELEMATICS	Telecommunications and Informatics
TF	Transformer
TFALDA	Three-Feature Based Automatic Lane Detection Algorithm
VFH	Vector Field Histogram
WMR	Wheeled Mobile Robot

NOMENCLATURE

a	Distance of front axle from the vehicle centre of gravity (m)
A	Cross sectional area (m^2)
$\{A\}$	Inertial frame
b	Distance of rear axle from vehicle centre of gravity (m)
B	Stiffness factor
$\{B\}$	Robot body frame
C	Capacitance (F)
d	Distance of the robot from target (m)
E	End point
F	Force (N)
Gi	Integral gain
Gp	Proportional gain
H	Height of vehicle CG from bushing reference point (m)
I	Moment of inertia (kg m^2)
J	Polar moment of inertia (kg m^2)
K	Stiffness (N/m)
l_1, l_2	Distance of front and rear wheel from centre of gravity (m)
L	Length (m)
L_m	Motor inductance (H)
M	Mass (kg)
Q	Electrical charge (Q)
R	Damping (Ns/m)
R_m	Resistance of motor (Ω)
T	Time (s)
U	Velocity in x direction (m/s)

V	Voltage (V)
x, y, z	Displacements in three directions (m)
$\dot{x}, \dot{y}, \dot{z}$	Velocities in three directions (m/s)
$\ddot{x}, \ddot{y}, \ddot{z}$	Accelerations in three directions (m/s^2)

Greek characters

ω	Angular rotation about y-direction (rad/s)
γ	Ramp angle (rad)
μ	Coefficient of friction
μ_{i}	Transformer modulus (i = 1…14)
μ_m	Motor torque constant (Nm/A)
θ	Heading angle of robot (rad)
$\dot{\theta}, \ddot{\theta}$	Angular velocity, acceleration (rad/s), (rad/s^2)
δ	Steering angle (rad)
Φ	Angle between target and heading angle of robot (rad)

Subscripts

A	Length of arm
B	Body
Bd	Brake drum
cfr, crr	Front and rear cornering force
cx, cy, cz x, y, z	Direction of vehicle body
C	Cylinder
f1	Front leg Link1
f2	Front leg Link2
Fr	Front
K	Capacitance
L	Left wheel

M	Mass
Mr	Mass of robot
nfr, nrr	Normal for front and rear wheel
O	Obstacle / object
p	Plant
P	Piston
Pad	Pad
r	Right wheel
r1	Rear leg link1
r2	Rear leg link2
Ref	Reference
Rr	Rear
S	Suspension
Stw	Steering wheel
t	Target
tfr, trr	Tangential (front and rear)
v	Vehicle
w	Wheel
1,2	Contact point number

Superscript

*	An estimate (virtual)

PUBLICATIONS FROM PRESENT WORK

Publications in SCI Journals (6)

1. **Singh R.** and Bera T. K., "Fault detection, isolation and reconfiguration of a bipedal-legged robot", *Simulation International*, *Sage Publishing House*, Vol. 95, No. 10, pp. 955–977, 2019 **IF-1.455**.
2. **Singh R.** and Bera T. K., "Fuzzy logic controller for obstacle avoidance of mobile robot", *International Journal of Nonlinear Sciences and Numerical Simulation*, DE GRUYTER, Vol. 20, No. 1, pp. 51–62, 2019 **IF-1.162**.
3. **Singh R.** and Bera T. K., "Bond Graph Approach for Dynamic Modeling of the Biped Robot and Application to Obstacle Avoidance", *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, *Springer*, Vol. 41, No. 10, pp. 1–12, 2019 **IF-1.743**.
4. **Singh R.**, Bera T. K., "Obstacle avoidance of bicycle vehicle model using overwhelming controller", *Arabian Journal for Science and Engineering, Springer*, Vol. 43, No. 9, pp. 4821–4833, 2018. **IF-1.518**
5. **Singh R.** and Bera T. K., "Walking Model of Jansen Mechanism Based Quadruped Robot and Application to Obstacle Avoidance", *Arabian Journal for Science and Engineering, Springer*, **IF-1.518.**
6. Soharu G., **Singh R.** and Bera T. K., "Bond Graph Modelling and Simulation of Quadruple Robot with Different Gaits", *Arabian Journal for Science and Engineering, Springer*, Vol. 44, No. 9, pp. 7385–7397, 2019 **IF-1.518**

Publications in Scopus Journals (3)

1. **Singh R.**, Singh M., and Bera T. K., "Model reduction in vehicle dynamic systems", *International Journal of Modelling and Simulation, Taylor and Francis*, Vol. 37, No. 2, pp. 67–81, 2017.
2. **Singh R.** and Bera T. K., "Obstacle avoidance of mobile robot using fuzzy logic and hybrid obstacle avoidance algorithm". *IOP Conference Series: Materials Science and Engineering*, Vol. 517, pp. 50–54, 2019.
3. **Singh R.** and Bera T. K., "Obstacle avoidance of a humanoid mobile robot based on multi-sensor information", *WJMS*, World Academic Press, Vol. 15, No. 2, pp. 116–127, 2019.

Publications in International Conference (4)

1. **Singh R.** and Bera T. K., "Obstacle avoidance of mobile robot using fuzzy logic and hybrid obstacle avoidance algorithm". 2nd International Conference on Robotics and Mechantronics (ICRoM 2018), Singapore, November 9–11, 2018.
2. Jain M., Sehgal M., Kalra Y., Jain P., Aggarwal N., **Singh R.**, Kumar D. and Bera T. K. "Object Detection and Gesture Control of Four-Wheel Mobile Robot" Fourth International Conference on Communication and Electronics Systems (ICCES 2019), Coimbatore, July 17–19, 2019.
3. **Singh R.**, Bera T. K. and Singh J., "Bond Graph Based Modeling of Four-Cylinder In-line Engine and Study the Various Engine Characteristics Parameters", National Conference on Internal Combustion Engines and Combustion, NIT Kurukhetra, November 1–4, 2019.
4. Kumar A., **Singh R.**, Bera T. K. and Singla A. "PI control based modelling of segway using bond graph", *iNaCoMM*, IIT Mandi, 2019.

Publications in Under Review (3)

1. **Singh R.** and Bera T. K., "Walking mechanism of quadruped robot on a side ramp using PI controller", *International Journal of Robotics and Automation*, ACTA Press.
2. Ashish Kumar, **Singh R.**, Bera T. K. and Singla A. "Heuristic Controller Design for Inverted Pendulum Using Bond Graph Approach", *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, Springer.
3. **Singh R.** and Bera T. K., "Navigation model for four wheel mobile robot", *International Journal of Robotics and Automation*, ACTA Press.

LIST OF FIGURES

LIST OF TABLES

TABLE OF CONTENTS

Chapter 1
Introduction

"A *Mobile Robot* is an automatic apparatus that is capable of self-drive in a specified environment". Mobility is one of the advantages of mobile robots (MR) as compared to the fixed base robot. As a result, large workspace was explored by the mobile robots. They may move on the ground surface, on the surface of water, under water and in the air space. The manoeuvrability of the mobile robots might be controlled by increasing number of sensors. They have been used to accomplish a number of tasks that are carried out by the people or a mechanism such as space, watch, housekeeping, care taker, surveillance etc. The aim of the mobile robots is to reach to the target-location safely while avoiding obstacles (static or dynamic). Several key tasks for mobile robots are

- Obstacle avoidance (static: building, wall, furniture etc. or dynamic: moving robots, moving pedestrian etc.)
- Locomotion mechanism with adaptation of environment
- Planning and navigation
- Perception of surrounding

The modelling and simulation are important aspects while designing the mobile robots. It is not easy to manufacture the real setup of mobile robots to perform various tasks for checking the feasibility of the robots. Therefore, the mathematical model is required to check the results under various conditions. The most important task for mobile robot is to avoid obstacles (static or dynamic) maintaining its dynamic stability. The static and dynamic obstacle avoidance of mobile robot using hybrid algorithm, fuzzy logic and overwhelming control is proposed in the Thesis. Further, for the dynamic analysis of various sub systems of mobile robot, the model reduction technique is implemented to solve the complex dynamic systems without altering its performances. The walking mechanism of mobile robot is also considered in the Thesis.

1.1 Mobile Robots

A mobile robot also known as driverless vehicle, self-driving vehicle or robot car is capable to move in known and unknown environment. A mobile robot senses its surrounding environment and navigates without human interference. MR, a multibody mechanism is modelled depending on the type of terrains, task time, target and environmental situations. The fundamental building blocks for the functionality of the mobile robots are shown in Fig. 1.1.

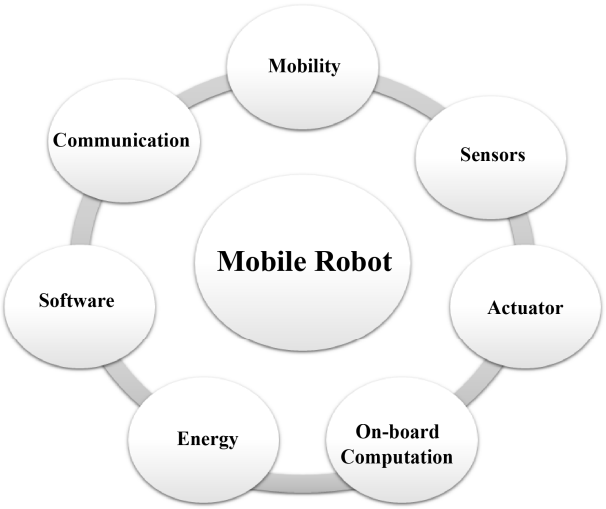


Fig. 1.1 Fundamental building blocks of mobile robots [Chen *et al*., 2009]

- ***Mobility***: Mobility for mobile robots is the ability to move by interacting to its surrounding environment. For plane surface, wheeled robot is preferable. For un-even and rough surface, legged robot is desirable.
- ***Sensors***: For mapping the surrounding information, different sensors are available. Sensors such as chemical sensors, accelerometers, inertial measurement unit (IMU), global positioning system (GPS) etc. may be used for monitoring purposes.
- ***Actuators***: They are used to achieve other jobs apart from mobility. Actuators may be electric motor for rotary motion and piston-cylinder arrangement or lead screw for linear motion.

- ***On-board computation***: The on-board computation system is required for the function of a mobile robot. Small robot just needs distance sensor for data calculation whereas complex robot interacting with humans requires advanced computational boards.
- ***Software***: The software is used to make the interface link between the hardware and the algorithm depending upon the application of the robots.
- ***Energy***: Robot usually works in closed surroundings where energy sources are limited. The applications such as space exploration, the solar panels as energy sources for robots are created to complete the assignment.
- ***Communication***: The communication is required for the robot to communicate with other robots or systems. The concept of platooning is accomplished by proper communication between mobile robots.

1.1.1 Classification of Mobile Robots

Several factors are considered for classifications of mobile robots. Some of the factors are listed below:

- Mode of operation
    - Tele-operation mode (semi-automatic)
    - Autonomous mode (fully-automatic)
- Operation environment
    - Land based (wheeled robots, tracked robots, legged robots)
    - Aquatic/Underwater robots
    - Aerial robots (drone)
- Nature of ground surface
    - Even/smooth surface (wheeled robot)
    - Un-even/rough surface (tracked or legged robots)
- Environment
    - Indoor workspace (localization is done by overhead cameras or roof mounted beacons)
    - Outdoor workspace (GPS is used to locate the robot position)

- Application
    - Industrial application (logistics, manufacturing)
    - Services (medical, home, defence, space exploration)

1.1.2 Advantages of Mobile Robots

Increased use of mobile robot makes it possible to achieve multiple benefits such as:

- Work under hazardous conditions which are not suitable for humans to work
- Aid to the military solider for uplifting heavy armours in the battle field
- Look after elder person in the home
- Provide home services such as cleaning, dusting etc.
- Upgrade the autonomous mobile robot for reduction of road accidents

1.1.3 Applications of Autonomous Vehicle

Mobile robots can be used for surveillance, defence, or home services. Also, unapproachable or bad situations such as aquatic, space rovers or robots operating in dirty situations [Nehmzow, 2003] are the application-fields of robots. Making artificial pets is very attractive now-a-days. Automated inspection in the industry is an application of mobile robots. As manual inspection is a very costly and tedious to a human operator, it is ignored to avoid the risk of errors. Now-a-days, areas such as artificial intelligence, cognitive science and psychology are emerging research fields in robotics. Intelligent robots deal with an exceptional way of examining hypotheses about intellectual performance, perception and cognition.

1.2 Background and Motivation

The word ‘Mobile’ basically means able to move or to be moved freely or easily. The mobile robots have much greater advantages in comparison with the fixed robots. In everyday life, the popularity of mobile robots is increasing because of their more flexible nature, large workspace, greater accuracy etc. as compared to those of fixed robots. It can be used for performing a number of jobs like surveillance, trajectory tracking, material handling etc.

During World War II, the first mobile robots arose as a consequence of advancement in the field of computer science and cybernetics. They were generally flying bombs (smart bombs) that were navigated with the help of guiding system and radar control. Two autonomous mobile robots which are equipped with a light sensor to explore their surrounding environment are developed by Walter [1948]. In 1961, Johns Hopkins University developed the self-survival mobile robot "Beast" and the sonar sensor was used to move around. Four-wheel Stanford cart with spider was developed for solving navigation problem [Moravec, 1980]. In 1980 onwards, the people have shown interests in the robots for home use.

From 1980s to 1990s, lot of researches were carried out to control the motion of the mobile robots. The robots like car with a speed of 55 mph on empty streets were developed and tested by Dickmanns [2007]. During this period, the technology such as IBM series, speech command, electroencephalogram **(**EEG) signals and electrooculogram (EOG) signals were used to control the mobile robots. An autonomous small MR for a purpose of research work *Khepera* was developed by Mondada *et al*. [1999] with the support of LAMI-EPFL lab. The legged based walking robots were developed by Carnegie Mellon University to explore the live volcanoes [Bares *et al*., 1999].

The concept of the swarm robots was introduced in many disciplines such as military, navigation etc. [Sahin *et al.*, 2002]. The mobile robot for home-floor cleaning was developed by Bell *et al*. [2003]. The Infrared sensors were used to detect the upcoming obstacles (walls, furniture etc.). A team of 100 small mobile robots *Centibots* were used to work together to map the unknown environment and to identify objects within the environment [Charlie *et al*., 2004]. The Boston Dynamics developed a legged robot for lifting weighty loads across the rough land [Playter *et al.*, 2006]. The team of autonomous vehicles was developed to map a huge vigorous urban location, to find and track the beings and to avoid the unreceptive things [Heieh & Lacroix, 2012]. The multi-function agile remote controlled robot was developed for military purposes to explore and diffuse the explosive devices by William & Zemach [2016]. The quadruped robot consisting of LIDAR sensor to act pictorial, audio and thermal inspection on an oil and gas location was developed by Hutter *et al*. [2017].

Now-a-days, the increasing number of accidents on road due to troublesome and fatigue driving is a major concern; the field of mobile robots or autonomous vehicles became very popular. A small attempt is made in this Thesis work to develop the obstacle avoidance algorithm to avoid static and dynamic obstacles for mobile robots. The hybrid obstacle avoidance algorithm by combining the benefits of line following, wall following and tangent bug algorithm is proposed and this is applicable on the biped robot to detect and avoid static obstacle in indoor environment. Also, fuzzy logic controller is proposed for four-wheel mobile robot to avoid static as well as dynamic obstacles. The phenomenon of lane change is also considered in this work. The obstacle avoidance controller for Jansen mechanism [Komoda *et al*., 2011] based quadruped robot is also developed using fuzzy logic.

A powerful indirect approach of doing system inversion called overwhelming control was used in robot-control in earlier days. This scheme works well if there are presence of noises, uncertainties in parameters and measurements. This strategy has already been applied to many serial manipulators [Pathak *et al*., 2005] and parallel manipulators [Bera *et al*., 2010]. This Thesis involves the application of overwhelming control approach in obstacle avoidance of mobile robot moving on a predefined path.

Mobile robotic systems that contain mechanical structure, electrical, computer and electronics parts or sub-components are complex in nature. A reduced model of the system can be obtained to reduce high computational time, without compromising on the accuracy. The dynamics of the reduced model is same as that of original system. The saving in computational time is the motivation to develop a reduced model of bicycle vehicle and four-wheel mobile robot.

The principle of inverted pendulum is same as the swinging bar of the two-wheel mobile robot. The modelling and simulation of inverted pendulum were carried out in this Thesis work. The heuristic and proportional integral (PI) based controller was developed to control the motion of the inverted pendulum. The simulation results of the inverted pendulum were validated with the experimentation-results. The linear inverted pendulum setup developed by Googol Tech, Hong Kong was used for experimentation-work. The above validated control-strategy was implemented on the two-wheel mobile robot model.

The advantages of legged mobile robot over wheeled mobile robots are the motivation to develop the walking mechanism for legged robot. The modelling of the quadruped robot is proposed and simulated for different gaits such as trot, bound, pace and amble. From the results, it is seen that present model is stable irrespective of the gaits being followed during forward motion. As noticed from the comparison of the displacement achieved by the gaits implemented, it is observed that the dynamically balanced gaits are faster and more energy efficient than statically stable gaits as they traverse more distance than statically balanced gaits. Another observation is that the trot gait is the most energy efficient as it traverses the maximum distance among all the gaits.

Fault detection and isolation (FDI) field has received significant attention during last three decades [Chen & Patton, 1999; Isermann, 2005]. The concept of fault detection and isolation was applied to many mechatronics systems used in robotics, aircraft etc. [Abdi, 2010; Chi *et al*., 2010]. The fault accommodation through reconfiguration of four legged robots is proposed by Krishnan *et al*. [2011]. No work has been reported in the literature to develop the FDI model through reconfiguration for two legged mobile robots.

For modelling complex and large multibody systems, the BG tool is proposed in this Thesis work. Effective algorithms for control can be developed by investigating a structure made in bond graph and hence, model can be suitably changed. The bond graph's causal structure illustrates how computer simulation of the system can be produced by combining element's constitutive relations. In bond graph, it is very simple to construct the inverse dynamics of any system by doing different causal orientations in the mirror image of the actual system. This is the motive behind the rapidly growing use of bond graph modelling in design of mechatronic systems. Hence, the modelling and simulation are performed in the Thesis through bond graph modelling technique.

1.3 Contribution of the Thesis

The main contribution of the work presented in the dissertation is the development of control strategies for autonomous navigation of mobile robots (wheeled and legged) in the presence of static and dynamic obstacles. In addition, dynamic analysis of mobile robots like fault detection, isolation and reconfiguration, model order reduction etc. are also done.

1.4 Associated Developments for Autonomous Navigation of Mobile Robots during Obstacle Avoidance

The major developments to reach the major contributions are as follows:

- A hybrid obstacle avoidance algorithm has been developed by combining the merits of line following, wall following and tangent bug algorithm. The obstacle avoidance algorithm is used to avoid the static obstacle for biped mobile robot. For this purpose, modelling of the mobile robot is done. The proposed algorithm is applied to biped robot to avoid static obstacle.
- The fuzzy logic controller is developed for avoiding dynamic and static obstacles. The inputs to the controller are the distance of the robot from obstacle, distance from target and heading angle of the robot. The fuzzy based obstacle avoidance algorithm is implemented on four-wheel mobile robot. The modelling of the four-wheel mobile robot is developed using bond graph tool.
- The trajectory tracking problem is also considered in the Thesis work. An accurate and efficient overwhelming based controller of bicycle vehicle system is proposed. The developed controller is used to avoid the obstacles with different shapes and geometries.
- The obstacle avoidance controller of Jansen mechanism based quadruped robot developed is implemented using fuzzy logic.
- The model reduction technique is implemented in this Thesis work to reduce the computational time for solving the complex multibody systems. The model reduction of various components is done using eigenvalue separation and

sensitivity methods. Also, model reduction of full vehicle model is developed. The results of reduced model are similar to the results of actual system.

- The proportional integral (PI) based controller is developed to control the motion of an inverted pendulum. The results of the proposed controller are validated with the experimentation-results. This controller is further applicable in the two-wheel mobile robot (segway). Bond graph model of a segway and its sub-systems are developed.
- The bond graph based modelling of quadruped robot is done. The proposed model is simulated for various gaits such as trot, bound, pace and amble.
- The fault detection and isolation concept is considered in this work. The bond graph model of the three-dimensional biped legged robot is developed. Also, the fault detection, isolation and reconfiguration of biped robot are considered when fault occurs in one of the leg's actuator.

1.5 Organization of the Thesis

The dissertation includes six chapters in total including this chapter followed by a list of references referred in the work. An outline of the role of other five chapters is discussed as follows:

Chapter 2 presents an extensive review of the literature related to the mobile robot, obstacle detection, overwhelming controller and physical model reduction. Literature on quadruped robot, fault detection and isolation and bond graph modelling is also explained in this chapter. Finally, the objectives of the present work are identified from the observations illustrated in the literature.

Chapter 3 describes the control strategies on obstacle avoidance for the legged mobile robot, bicycle vehicle model and four-wheel mobile robot. Bond graph representations of bicycle vehicle model, four-wheel mobile robot are proposed. The hybrid obstacle avoidance algorithm is developed for static obstacles and is implemented on the legged mobile robot. The fuzzy logic controller is also used to avoid dynamic obstacles and is applied to the bond graph model of the four-wheel mobile robot. The developed model is validated by considering different case studies. The basic concepts of overwhelming

control and its properties are derived. Bond graph implementation of an overwhelming controller with an inverse dynamics model for the bicycle vehicle model is also presented. The path generated by the obstacle avoidance algorithm is used as the input to the inverse bicycle vehicle model to track the trajectory avoiding obstacles by the forward bicycle vehicle model while reaching the target.

The obstacle avoidance controller for Jansen mechanism based quadruped robot is implemented using fuzzy logic in Chapter 4. The model reduction of vehicle model through eigenvalue separation and sensitivity methods are also developed. The model reduction technique is applied to the various components of the bicycle vehicle and four-wheel vehicle model. The model is reduced such that dynamics of the system remains unchanged though its performance stays unaffected. The performance of the full model with the reduced model is compared. Thereafter, bond graph model of two-wheel mobile robot (segway) is developed. The inverted pendulum model based on bond graph is proposed for the swinging bar of the segway. The simulation results are validated with the experimental results. The PI controller is used to control the motion of the two-wheel mobile robot.

Chapter 5 concerns with the walking mechanism of the mobile robots. The quadruped robot model using bond graph approach is proposed here. The model is used to validate the motion of the quadruped robot for various gaits. The fault detection and isolation model for the biped robot is also developed using bond graph approach. The fault detection and isolation is considered when one leg's actuator is under faulty condition. Finally, the PI based controller for quadruped robot is proposed. The proposed controller is applied on the four legged robot to walk on the side ramp.

Chapter 6 concentrates the conclusions derived from the research work carried out in the Thesis. The scope for future research work is also presented in this chapter.

Afterwards, a list of references pertinent to the whole Thesis is presented in the end. These references have already been cited in all chapters of the Thesis.

Chapter 2

Literature Review

In this chapter, a survey of literature on modelling, simulation and control of mobile robot is presented. Afterwards, the literature on obstacle avoidance algorithms is reviewed. A special focus is made on model reduction techniques as these strategies are applied in the Thesis for model reduction of the bicycle model and four-wheel model. Some literature on modelling of two-wheel mobile robot is also mentioned. Later on, a literature review on walking mechanism of mobile robots is discussed. Afterwards, the detailed literature survey on bond graph modelling is done and this shows how bond graph has been used for modelling of various systems in different fields. In the last, the objectives of the present research work are summarised based on the observations made from the literature.

The first *contribution* of the work is the development of control strategies for autonomous navigation of mobile robots (wheeled and legged) in presence of static and dynamic obstacles. This obstacle avoidance algorithm is developed based on three different controllers i.e. hybrid obstacle avoidance algorithm (Line following, Tangent bug and Wall following), fuzzy logic controller using distance of the robot from target, heading angle and distance of the robot from the obstacle and overwhelming controller using inverse model of the actual system for obstacle avoidance. These three controllers were used in different applications. The hybrid obstacle avoidance algorithm was used for biped robot to avoid static obstacles. The fuzzy logic controller was used for four-wheel mobile robot to avoid both static and dynamic objects and the same controller was implemented for Jansen based quadruped (legged) to avoid static obstacles. The overwhelming controller along with inverse model and hybrid controller were used to avoid static obstacles of different shapes and locations in case of bicycle vehicle model based mobile robot.

In addition, dynamic analyses of mobile robots were performed for better performance (reducing simulation time, reconfiguration of the mobile robots during faulty situation, gait analysis).

The different obstacle avoidance algorithms for mobile robots, applications of these algorithms in different areas and performance analysis of mobile robots are closely related and those are explained in the Thesis.

2.1 Mobile Robots

Mobile robot is an emerging research area today at different parts of the world. The key research area for mobile robots is obstacle avoidance. The University of Minnesota has developed the *Safe Truck* concept [Zheng *et al.*, 2004] which is equipped with virtual sensor bumper. A vision-based method to detect the obstacles for autonomous land vehicle (ALV) is proposed by Hyo *et al.* [2012]. The pattern recognition scheme is used to find collision-free paths in unknown environments. In some applications where environmental conditions are aggressive to the presence of human beings, for example active military zones, robust mobile systems are used. Hence, a new strategy is proposed by LeSage & Longoria [2014]. An attempt has been made in this Thesis for the development of hybrid obstacle avoidance algorithm for static obstacles and this algorithm is implemented on the legged mobile robot. Also, fuzzy based obstacle avoidance controller has been developed and implemented on four-wheel mobile robot to avoid dynamic and static obstacles. Different control techniques are used now-a-days for obstacle avoidance of mobile robots. Inspired by human behaviour, the scheme is proposed for navigation of the robot in an unknown dynamic environment [Kakoty *et al.,* 2018]. The surrounding information is generated by on-board sonar sensor using occupancy grid map algorithm. The algorithm based on the behaviour of ants searching for a food in unknown environment is proposed for wheeled mobile robot's navigation [Mordechai *et al*., 2017]. The modified artificial potential field method to generate trajectory for wheeled mobile robot is presented by Sudhakara *et al.* [2018]. From this method, the more accurate workspace is generated as compared with classical artificial potential field method. To find shortest path with obstacle avoidance for wheeled mobile

robot based on fuzzy, ant colony optimization technique is proposed by Yen *et al*. [2018]. The wheel mobile robot using arduino board with ultrasonic sensor and compass (HMC5883L) is developed for avoiding static obstacles [Mohomed *et al*., 2018]. The proposed model of the robot is tested in the outdoor environment. The method for predicting the movement trends of the obstacle using movement laws of the dynamic obstacle is proposed for mobile robot to plan the obstacle free path [Liu *et al*., 2018]. The fuzzy logic obstacle avoidance approach is proposed for safe navigation of humanoid robot by Rath *et al*. [2018]. The obstacle distance and bearing angle of the robot are considered as input function to the fuzzy logic controller. The proposed algorithm is validated on NAO humanoid robot in V-REP simulation environment. The fuzzy logic using "recognition memory" method is developed for optimization of local navigation of autonomous mobile robot by Xu *et al*. [2018]. This method is widely applicable for repetitive jobs. The *Webots* and *Matlab* platform are used to validate the proposed method. The adaptive particle swarm optimization based method is presented for a robust path planning for mobile robot by Dewang *et al.* [2018]. The two objective functions are considered such as distance between goal and obstacles for optimization of path length and simulation time. The fuzzy Markov decision based algorithm is proposed for path planning of humanoid robot in unknown environment [Fakoor *et al*., 2016].

A grid mapping method for static environment with object detection is presented by Lategahn *et al.* [2011]. Due to non-parametric nature, the grid method is used for static environment. The mobile robot with an on-board vision system used for obstacle avoidance to detect the objects is presented in Jun *et al.*, [2010]. An image processing method is developed to find distances between the robot and the obstacles. An algorithm for object detection using stereo camera is mentioned in Lin *et al.*, [2010]. In this work, the 3-D spatial information is obtained by stereo matching algorithm. A wheeled mobile robot (WMR) with vision, ultrasonic sensor, and positioning system for object detection and parking control is presented by Hamid *et al.* [2009]. The image processing technique is used to compute the distance of the robot from the object. Fuzzy approach and genetic algorithm based model is applied to monitor the robot motion. A three-dimensional sight analysis method from a dynamic object combined with sight geometry estimation, two-

dimension car and pedestrian detection, three-dimension navigation and trajectory calculation are presented by Leibe *et al.* [2007]. Remarkably, this work proposed a multi-view obstacle recognition method in a real world. Also, 2-D vehicle human detection is changed into 3-D remarks. A multi-resolution object recognition method using three fixed ROIs is discussed in Broggi *et al.* [2004]. In Zarate *et al.* [2002], an evolutionary gabor filter optimization (EGFO) method for obstacle recognition is introduced. In this method, selected Gabor filters using a trained SVM classifier is used.

A surrounding detection and mapping approach for autonomous driving in both rural and off-road environments is discussed by Ching & Tsai [1999]. The LIDAR sensor is used to develop the local position map for obstacle detection. A vehicle following approach by finding the symmetry axis of vehicle is proposed by Du & Papanikolopoulos [1997]. Moreover, this method has many demerits, such as large computing task. In this work, an attempt was made for modelling and simulation of biped walking robot, two-wheel mobile robot, four-wheel mobile robot, bicycle vehicle model and quadruped robot. Also, the prototype model of biped and quadruped robot is developed in the research work.

2.2 Obstacle Detection and Avoidance

The use of multi sensor (stereo matching) to extract the multi feature of the surrounding or environment for path planning of the vehicle is proposed by Linhui *et al.* [2009]. The fuzzy rule is used to integrate the slope and roughness of the road to calculate the traversing ability of the vehicle. The vector field histogram algorithm (VFH) is applied to develop the object/obstacle avoidance algorithm (OAA). Sensors such as camera, laser range finder and stereo vision system are used to develop the 3-D environment of the surroundings. The path planning of the vehicle using cubic B-spline is presented in Nagel *et al.*, [2006]. Two LIDAR sensors are fixed on the top of the vehicle and scan the area. The data from LIDAR were combined with GPS to generate the surroundings.

Chu [2012] developed the on line path-planning algorithm for off-road vehicle driving with static obstacles detection. The predefined waypoints are used to compute the optimal path. The algorithm for unforeseen obstacle avoidance by an autonomous

sweeping robot is proposed by Kurabayashit & Koga [1998]. Sweeping motion is the motion required to cover a 2-D region. When a mobile robot detects an obstacle, it not only avoids the obstacle but also sweeps the area. A particle swarm optimization method is proposed for a path planning of soccer robot by Wang *et al*. [2006]. The particle swarm optimization (PSO) technique is used for path planning. The method for better detection, prediction and escaping of dynamic objects in urban environment is discussed in Ferguson *et al.*, [2008]. The data of road structure in the form of lanes can be obtained from aerial camera and processed in an offline manner. The static obstacle is considered as two-dimensional grid. Each object is represented by box and a point model.

Collision free algorithm for all-terrain vehicle (ATV) is developed by Ghangrekar & Conrad [2009]. The surrounding is mapped with the help of LIDAR and ultrasound sensor. The optimum path factor is considered in this work. The pattern of navigation is developed based on grid and then, static obstacle is avoided. ATV follows the shortest path avoiding the obstacle to reach the destination. An application of robot in agricultural sector is discussed by Pingzeng *et al.* [2011]. For navigation and positioning of robot, the GPS system is developed. A fuzzy set is used to select the area around the obstacle. The ultrasonic sensor, infrared distance sensor and laser scanner are used to avoid the object. The OAA based on geometric modelling of robot workspace is proposed by Ghoshray & Yen [1996]. The work space is categorised into four quadrants using geometric modelling approach. Safe path for mobile robot is calculated by solving the obstacle-quad tree.

An artificial neural network (ANN) approach is developed by Ghoshray & Yen [1996] to control a robot locomotion in dynamically changing environments. The hybrid planning method is used for robot's path planning. The sensors are used for obstacle detection. The past and future position is used to calculate the present position of the robot. The past positions give robot's previous position and future positions give robot's goal. Based on the above information, robots do not lose its path while avoiding the obstacles. A mobile robot with GPS navigation system and object detection system with low cost structure, positioning module and sonar is developed by Zarate *et al.*[2002]. The GPS and Sonar calculate the exact position of the mobile robot. The way points are provided for better navigation of the robot. The daisy chaining method is concentrated on

the electronic circuit rather than on software part. The sensors are chained together i.e. output of one sensor is an input to the other. Mobile robot can manoeuvre through desired waypoint and parallel of the object avoidance task is applied. The real time simulation is shown by developing the mobile robot using radio control (RC) truck in combination with GPS and sonar sensor. The hybrid object detection algorithm based on the merits of line following, wall following and tangent bug algorithm is proposed in this work and is applied to the biped robot. Also, fuzzy based controller to avoid static and dynamic objects for four-wheel mobile robot is presented.

2.3 Overwhelming Control

An overwhelming control approach is a type of indirect approach for system inversion and is used in robot control. This scheme is powerful with respect to uncertainties in parameters and measurements. This strategy has already been applied to many serial manipulators [Pathak *et al.*, 2005] and parallel manipulators [Bera *et al.*, 2010] for trajectory tracking. The inverse model of the hybrid parallel manipulator is proposed by using overwhelming control strategy [Rashmi *et al.*, 2018]. Thumb and index finger of a human hand are taken as example of hybrid manipulator. The simulation and experimental work validate the novelty of the proposed controller. An attempt has been made in this Thesis to develop an overwhelming controller and object detection algorithm for a desired trajectory of bicycle vehicle model in mobile robots. The feasibility of obstacle avoidance algorithm was checked for obstacles depending upon various parameters.

2.4 Physical Model Reduction

In today's life, most of the systems are multi-domain systems comprising of mechanical, electrical and thermal components or sub-modules. Due to complexity of the multi domain system, it requires huge simulation time and computational cost. Therefore, it is necessary to perform the model order reduction by maintaining the dynamics and accuracy of the actual system. This results in significant reduction of simulation time and number of iterations. It is possible to quickly solve the reduced model in comparison to

bigger complex systems. From the last three decades, a lot of researches on model reduction technique have been illustrated by numerous researchers.

Two methods are carried out for model order reduction of any system: (1) Resolve the complete model, then apply mathematical procedures and (2) Solve the reduced system. Numerous methods for reducing the order of model are presented by Orbak *et al*. [2004]. One such approach is singular value decomposition method [Zhou *et al.*, 1996; Singh, 2015] based on the coordinate transformations. Here, a transformation matrix containing the dynamic characteristics of the system is split into three singular values. The system is first rotated into an intermediate space. Then, it is scaled and rotated into the desired space. Another method based on coordinate transformations is the balancing approach studied by Moore [1981]. Here, the system is converted to a particular stable structure to obtain the reduced model. It targets at the worst case scenarios and hence, reduces the order of the transfer matrix between command and response. The shortcomings of the previous method [Moore, 1981] are improved by Safonov & Chiang [1989]. They have applied a Schur method to do stable truncation model reduction. Here, two specific methods are developed to compute random bases for the left and right eigenvalue spaces linked with the high eigenvalue of the product of the observability and controllability and then projections are defined in terms of these arbitrary bases. A similar type of algorithm to decompose an unstable continuous/discrete time system into fully balanced and unbalanced subsystems with use of actual Schur transformation approach was developed by Nagar & Singh [2004]. This algorithm was applied on unstable linear time invariant (LTI) plant. The structural internal information of the system is not known in this method.

There are many approaches which provide good approximations and these are generally known as time and frequency domain methods. One of the commonly used time domain methods is moment matching method proposed by Davidson & Walters [1988]. It employs singular value decomposition approximation and eliminates some time moments with its help. Another similar method is derived by Lalonde *et al*. [1992], named as least squares model reduction method. A lesser order equation to predict autoregressive moving average (ARMA) is computed from an ARMA equation having bigger order for

this method and power of curve fitting is used to reduce the model. There are many methods related to the frequency domain existing in the literature. One of such methods is component cost analysis to reduce a model used by Skelton & Yousuff [1983]. Component cost analysis determines the contribution of each system element in the overall dynamic performance of the system. The technique for separate treatment of low-frequency and mid frequency ranges is proposed by Xiheng [1987] in Pade approximation method. In this method, the matrix continued fraction is generated by expanding the matrix transfer function and by using the matrix for Routh algorithm with the inversion process and continued fraction expansion.

All the above approaches are totally mathematical in nature. In addition to these methods, another approach of model reduction in physical domain [Ye & Youcef-Toumi, 2000] is presented to show that the energy exchange pattern of dynamic elements in a system is utilized to identify their importance and lesser important elements can be easily removed from the complex system to make it smaller. This approach can be used for both linear and non-linear systems. Another technique of model reduction [Orbak *et al*., 2003] is applied to calculate model reduction by eliminating few elements from the real system. Dynamic behaviour of a model is not dependent upon all the components of a system. So, these components are eliminated from the actual system [Louca *et al*., 1997]. Sueur & Dauphin-Tanguy [1991] developed model order reducing approach based on the singular perturbation. This approach decreases the dimension of the system by deleting the elements from the bond graph models. Some other techniques like eigenvalue separation and eigenvalue sensitivity technique are also proposed by Orbak [2010].

Slone *et al*. [2002] discussed the circuit analysis based model reduction techniques. An overview to model reduction process is discussed by Schilders [2008]. Without any change on the accuracy of the reduced system, it is helpful for solving complex problems so that calculation time and cost would be saved. The model reduction technique has been used in the present work to develop a model in bond graph with lesser order for bicycle and four-wheel vehicle model. Then, its performance is compared with full dynamic model for longitudinal velocity and pitch angle of vehicle.

2.5 Modelling of Walking Mobile Robot

The popularity of walking robots has increased significantly as compared to wheeled robots due to its ability to walk in all terrains. The walking pattern of quadruped robot is similar to the animal's walking style and this makes them to walk on difficult terrains. The desired locomotion for robot is key element while designing the walking robot. The walking mechanism of robot is classified into two phases i.e. stance phase and flight phase [Iida et *al*., 2005]. The model is developed to control the behaviour of body dynamics through force applied on robot's feet by ground [Raibert, 2008]. The model based on the behaviour of human muscle and tendons is developed for walking of the biped robot [Remy *et al*., 2012]. The proportional integral derivative (PID) based model for 4-DOF biped robot is developed and validated for uneven surfaces by Liaquat et *al*. [2008]. The biological based controller is developed for quadruped robot with trot gait [Phuc et *al*., 2010]. An effort has been made in the present work to conduct the dynamic analysis of the biped robot and quadruped robot and to validate the simulation results with the prototype model. Also, a comparison is made for walking of the quadruped robot with different gaits.

2.6 Fault Detection and Isolation

Practical processes are becoming more and more difficult and complex and safety may hamper if a fault occurs in a system, especially in the processes like trains, power plant, aircraft, automobiles and chemical plants [Chen & Patton, 1999; Isermann, 2005]. Therefore, fault detection and isolation (FDI) field has received significant attention during the last three-decades. Most of the research related to fault tolerant control in walking mobile robots is based on the hypothesis that the redundancy does not occur in the system [Yang *et al*., 2000; Yang, 2005; Yang, 2008a; Yang, 2008b]. The fault tolerance through reconfiguration with combination of swarm-intelligence based method for inverse kinematics modelling is developed by Jakimovski et *al*. [2008]. This approach is applicable for more than two legged robots. Recently, for 5-DOF robotic manipulator, the concept of motor fault accommodation through reconfiguration is applied by Abdi [2010]. The FDI model is used for on-line diagnosis of fault occurrence in sensors of

flight control system using analytical redundancy relations (ARR) [Chi *et al*., 2010]. Fault detection and isolation (FDI) algorithms are very popular in robotics and this is applicable to other areas like chemical reactors [Ould Bouamama *et al*., 2012]. The work on fault detection of actuator has been done by Bera et *al*. [2012]. The bond graph methodology is used to derive the variables for system behaviour. ARRs are continuously examined to generate residual signals and the errors in these signals are monitored for actuator's fault detection and isolation. The model of fault accommodation through reconfiguration of a quadruped walking robot is developed by Krishnan *et al*. [2011].

2.7 Bond Graph Modelling

Modelling and simulation of physical systems is complex task. It may be time-consuming and tidy task to solve complex and multi-disciplinary systems. A unified pictorial representation to the physical modelling of multi-disciplinary system based on conservation of power and causality is called bond graph (BG) [Paynter, 1961; Breedveld, 1984; Gawthrop & Smith, 1996; Karnopp *et al*., 2000; Thoma & Ould Bouamama, 2000; Dauphin-Tanguy, 2000; Borutzky, 2004; Mukherjee *et al*., 2006; Brown, 2007; Samantaray & Ould Bouamama, 2008].

In this work, BG has been used to represent various physical systems. A BG model is a concise graphical demonstration of the system dynamics. It also allows representation of information exchange between different sub-components when sensors, actuators and control systems are part of the system [Gawthrop, 1995; Dauphin-Tanguy *et al*., 1999; Margolis & Shim, 2001]. The energy and information flow based unified approach for solving multi energy domain systems is proposed by Samantaray & Ould Bouamama [2008]. The modelling using BG technique of large physical system is proposed by Borutzky [1999]. Multi BG is used for concise illustration of complex multibody systems [Bos, 1986 ; Cho, 1998]. Dymola [Otter *et al*., 1996] is an object oriented modelling language, which provides a modular approach to model of large numbers of inter-connected sub-systems. Apart from this, accurate control method is computed from the analysis of the bond graph [Gawthrop, 1995; Dauphin-Tanguy *et al*., 1999; Ngwompo & Gawthrop, 1999; Pathak *et al*., 2008]. For this reason, bond graph approach is generally

implemented in mechatronic systems [Gawthrop, 1991; Karnopp *et al*., 2000; Granda, 2002]. The modelling of virtual reality application may be done by BG technique [Lin *et al*., 2006; Bruns & Erbe, 2007; Gawthrop *et al*., 2009a]. The dynamic study based on bond graph technique for proportional control piloted relief valve have been studied and validated with the experimentation results by Dasgupta *et al.* [2005]. Bond graph method is appropriate for modelling of large systems by creating sub-models and afterwards combining them to form the global model. The bond graphs are also able to do modelling and simulation in case of electrical fields like switched power junction, a three-phase inverter, a series DC motor, rotary electric machines, AC motor etc. [Junco *et al*., 2007, Junco & Donaire, 2011]. The planar mechanisms can be very easily modelled with the help of bond graph approach [Khadabadi & Pilli, 2005]. Free rail wheel set rolling on flexible curved track was modelled by Banerjee *et al*. [2007] using BG. The area of magnetic levitation has also been explored using the bond graph modelling offering the control of magnetic force in a levitating ball by electromagnet [Mishra *et al*., 2013]. The bond graph tool can also be combined with functional analysis for intelligent and autonomous systems [Chatti *et al*., 2013].

The bond graph approach is also used to calculate the estimation [Merzouki *et al*., 2007a], identification [Gawthrop *et al*., 1992], fault detection [Ould Bouamama *et al*., 2005; Ould Bouamama *et al*., 2006; Merzouki *et al*., 2007b; Medjaher *et al*., 2006] *etc* in the mechatronic systems. This technique is applied to develop the feed-back system of the CNC machine [Tomar *et al*., 2007]. Hence, it is observed that the BG approach is well suited to model a big complicated system by developing sub-models and connecting them together afterwards to develop a final model. The control algorithms for electrically controlled brake and steering were proposed by Margolis & Shim [2001]. A 3-dimension dynamic model of coupled body is presented by Pacejka [1985]. The engine model with drive train is proposed in Louca *et al*., [2001]. Some multibody systems having parallel structure with closed loops like Stewart platform were kinematically formulated by Saha & Schiehlen [2001]. Tyre force in full vehicle body with six degrees-of- freedom (DOF) for vehicle chassis and six-DOF of each wheel are considered by Pacejka [2001]. These tyre forces were used to calculate the lateral, vertical and longitudinal forces and motion.

The model of autonomous vehicle (AV) by using the bond graph was developed by Bera *et al*. [2012] for the implementation of FDI approach. The bond graph model of different systems of AV is interrelated with the controller to achieve the objective. The pure-pursuit algorithm is developed for path tacking. The handling performance of vehicle with different combinations of actuators (when some actuator is damaged) is discussed.

In the present work, the BG model of sliding actuators, legged robot, bicycle vehicle model, segway model, quadruple robot, and a full vehicle model etc. are developed. The S*YMBOLS Shakti* software is used for bond graph modelling.

2.8 Observations from Literature

After a detailed study of literature on mobile robots, object detection and avoidance, overwhelming control, model reduction, walking of mobile robot, fault detection and isolation and bond graph modelling, the observations from the literature are as follows:

- The different algorithms have been used individually but not much work has been done with regard to the combination of line following, wall following and tangent bug algorithms
- As per author's knowledge, the fuzzy logic controller for obstacle avoidance of four-wheel mobile robot was not developed earlier. The four wheel mobile robot is nonholonomic in nature as velocity relations are directly written to frame bond graph modelling of mobile robots. The kinematic properties are discussed in Chapter 3, 4 and 5.
- No work has been reported in the literature with regard to obstacle avoidance of mobile robot based on overwhelming controller. Overwhelming controller with inverse model of the actual system is used for obstacle avoidance of mobile robots.
- The dynamic analysis of mobile robots is performed for better performance. The model order reduction technique is required to reduce simulation time and real time implementation, where feedback delay can lead to instability. This will help

the mobile robot to reach quickly to the destination avoiding static and dynamic obstacles.

- Very less work has been reported on the study on different gaits for four-legged robot
- The fault detection, isolation (FDI) and reconfiguration for mobile robot avoiding different obstacles is challenging task. So, the FDI technique for mobile robots is adopted in this work.

2.9 Objectives and Assumptions of the Present Work

According to the literature review, it is found that BG is a unified method of modelling physical systems. So, the BG modelling technique [Mukherjee *et al*., 2006] is used in this Thesis work. The *objectives* of the present work are formulated and the summary of it is given below:

1. To develop obstacle avoidance algorithm for static objects.
2. To develop obstacle avoidance algorithm for dynamic objects.
3. To develop trajectory tracking model of mobile robot for predefined path.
4. Theoretical validation of various sub system of mobile robot using bond graph theory.
5. Walking mechanism of robots.

The mobile robots considered in this work are of two types: wheeled and legged mobile robots. The basic *assumptions* considered in the Thesis are stated as

1. The legs of the legged robot are designed based on oscillating cylinder mechanism.
2. The height of the robot is more than twice the length of the leg of the mobile robot. In this proposed model of the biped robot, the bigger foot size is designed so that maximum stability and load carrying capacity can be achieved. The bigger size foot has maximum surface contact with the ground as well as more weight and this results better and stable walking on the even or uneven surfaces.

3. The differential drive based yaw motion of wheeled mobile robot is adopted in this work.
4. The actual distance of the robot from the obstacle is the summation of the distance measured by ultrasonic sensor (d_o) mounted on the front chassis of the robot and half of the wheel base. In this calculation, the wheel base as well as width of the mobile robot is not considered.
5. The distance of the robot from the obstacle and the shape of the obstacle may be captured by overhead camera. But in this work, ultrasonic sensor is used to measure the distance and overhead camera is used for capturing shape of the obstacle.
6. The controller used for this work is Arduino UNO which may be replaced by Raspberry Pi or BeagleBone for faster processing.

Chapter 3

Obstacle Avoidance for Mobile Robot

3.1 Introduction

The area of autonomous vehicle (AV) is the key research field in intelligent transportation systems (ITS). The task of the AV became more challenging as compared to the indoor robots due to perception of real-time dynamic environment and physical modelling. AV has to accomplish the basic goals such as perceiving and modelling surrounding, localizing and building maps, planning routes, making decisions and governing the motion of vehicles.

A mobile robot (MR) is also known as an AV that has an ability of locomotion. MR provides an immense application in the area of manufacturing industries, medicines, process-control etc. It is capable to sense its surrounding and to navigate using sensors without human assistance. Some advanced control systems are used to interpret sensory data for identifying navigation paths as well as obstacles (static or dynamic). One of the key tasks for mobile robots is the detection and avoidance of obstacles during the navigation task. For the solution of the above challenges, many researchers introduced different algorithms and methods.

The organization of this chapter is as follows: At the beginning, the static obstacle avoidance algorithm for a legged robot is discussed. The kinematic model of the legged robot is established in *Matlab* and *Labview* environment and is simulated to avoid static obstacles. The simulation results are validated with the experimentation results. Thereafter, the dynamic obstacles are considered for the four-wheel mobile robot with fuzzy logic controller. The bond graph representation of four-wheel mobile robot is also detailed using *SYMBOLS Shakti* software. Finally, the bond graph based obstacle avoidance algorithm for the two-wheel mobile robot is developed using overwhelming controller.

3.2 Obstacle Avoidance for Legged Mobile Robot

In this section, static obstacle avoidance algorithm (OAA) for legged mobile robot is discussed. Emphasis is given to the development of obstacle avoidance algorithm. The results from the simulation are validated with the results of experimental work.

3.2.1 Modelling and Control of Legged Robot

The modelling and controls of legged robot used in the simulation and the experimental work are discussed. The legged robot is a two-legged robot where each leg is driven by an individual motor. The computer-aided design (CAD) model of the robot and kinematic motion controller is developed and discussed.

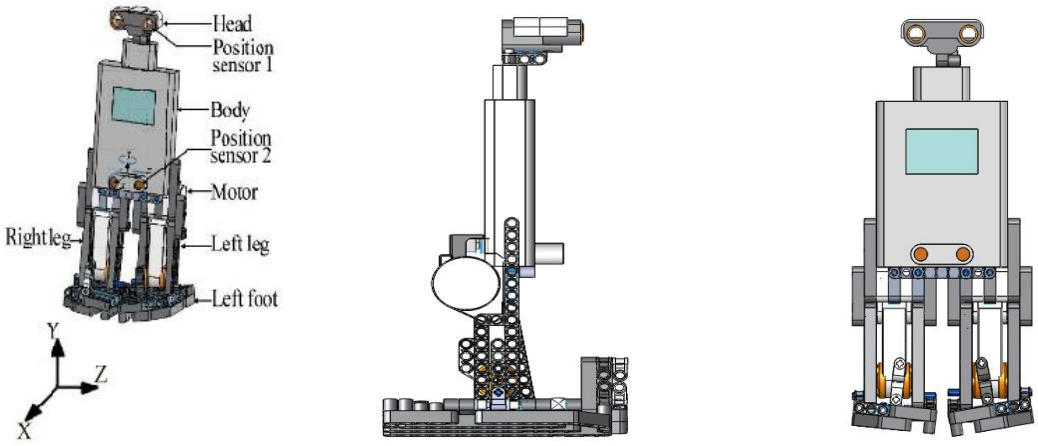
3.2.1.1 CAD Model of Legged Robot

The CAD model gives the detailed drawing of the robot. Each part (foot, leg, motor, body, sensors, and head) of the legged robot is designed in 2-D (sketch module) frame and then it is converted into 3-D (part module) frame. The detailed model of the legged robot is shown in Fig. 3.1. The forward and lateral motions of the robot are considered along *X*-direction and *Z*-direction, respectively. A class of novel biped mechanism with four links (l) and three planar lower joints (hinge joint, j) is presented in this work. These robots are able to walk stably down a gentle (maximum of 20°) slope as well as horizontal path. The degree of freedom of the biped robot as shown in Fig. 3.1(f) is 3 [$\text{DOF} = 3(l-1) - 2j - k = 3(4-1) - 2\times 3 - 1\times 0 = 3$)]. The walking of the robot is based on the oscillating cylinder mechanism as shown in Fig. 3.2. The piston rod of the cylinder is attached to the motor through the crank. The fixed end of the cylinder is attached to the body of the robot. The foot is connected to the end point (F) of the extended connecting rod. The piston reciprocates within the oscillating cylinder (robot legs) which results in the walking of the robot.

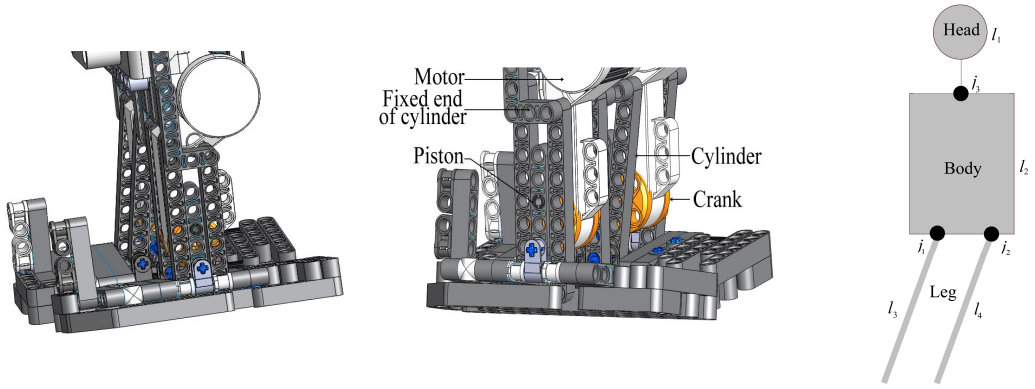
3.2.1.2 Simulink Model

The Simulink block diagram of the robot is depicted in Fig. 3.3. XML file of the model of the robot is generated by *Matlab* software using export command and a set of geometry files (STL) are imported into Simscape library of Simulink to generate a kinematic

model. The individual block of the joint of the robot is created in Simscape library. The world frame, mechanical configuration and solver configuration blocks are required for initial step of Simscape modelling. The world frame is connected to the robot body. The motion control unit is also attached to the world frame to generate the smooth motion of the robot. The right and left leg blocks are attached to the robot body with predefined points or coordinate frames by revolute joints. The revolute joint acts between two frames. This joint has a single rotational degree-of-freedom (DOF). Ports 'B' and 'F' are frame ports that denote the base and follower frames, respectively. The joint direction is defined by the motion of the follower frame relative to the base frame. The output of the revolute joint is the linear velocity which is used as the input to the motion control unit. PS-Simulink converter is used to transform the input physical signal (PS) into a unit less Simulink output signal. At fixed frame of the revolute joint1, the motor block is attached which is further attached to the robot leg. The same process is used for the left leg using revolute joint. Similarly, the right and left foot are attached to the respective legs. The data from the ultrasonic sensor is used to control the motion of right and left leg motor.



(a) (b) (c)



(d) (e) (f)

Fig. 3.1 CAD model of a legged robot (a) Isometric view with world coordinate system, (b) Side view, (c) Front view, (d) Leg with motor arrangement, (e) Leg considering as oscillating cylinder mechanism and (f) Kinematic model

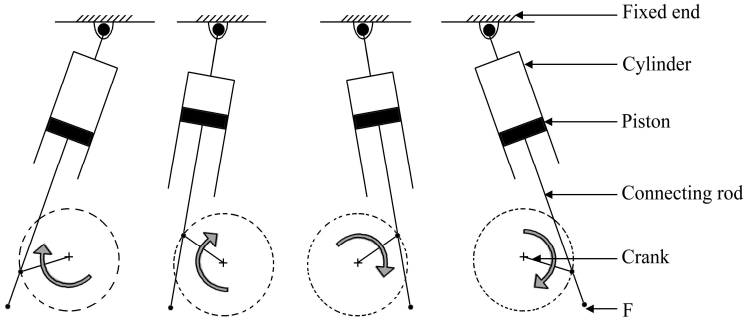


Fig. 3.2 Different motions of robot leg considered as oscillating cylinder mechanism

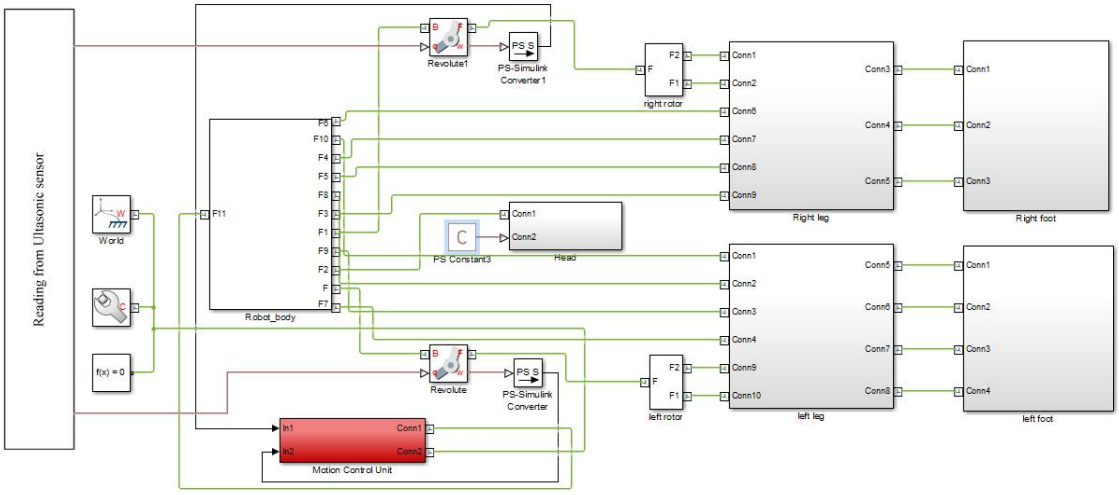


Fig. 3.3 Simulink block diagram of a legged robot

Figure 3.4 shows the motion control unit of the legged robot. The motion of the robot with respect to the inertial frame is done using kinematic sub-system model. The equations for the kinematic sub-system model of the robot are

$$V_{\text{avg}} = \frac{v_{\text{r}} + v_{\text{l}}}{2} \tag{3.1}$$

$$\dot{\theta}_Y = \frac{v_{\text{r}} - v_{\text{l}}}{\text{b}} \tag{3.2}$$

$$\theta_Y = \int \dot{\theta}_y dt \tag{3.3}$$

$$\dot{X} = \int \left(V_{\text{avg}} \times \sin\theta_Y \right) \tag{3.4}$$

$$\dot{Z} = \int \left(V_{\text{avg}} \times \cos\theta_Y \right) \tag{3.5}$$

where V_{r} and V_{l} are velocities of right and left wheel and b is the width of the robot. The inputs to the motion control unit are velocities of right and left leg and these velocities are further used to calculate the X and Z positions and angular displacement about the y-axis of the robot using above equations. Further, the planar joint block is used to convert the motion of the base coordinate frame into the follower frame. This block consists of 3 degrees-of-freedom (DOF) (one rotation and two linear). The function of PS-Simulink block is to transform the physical signal into the Simulink output signal. Scope block is used to plot the location of the robot body in X, Y and Z directions.

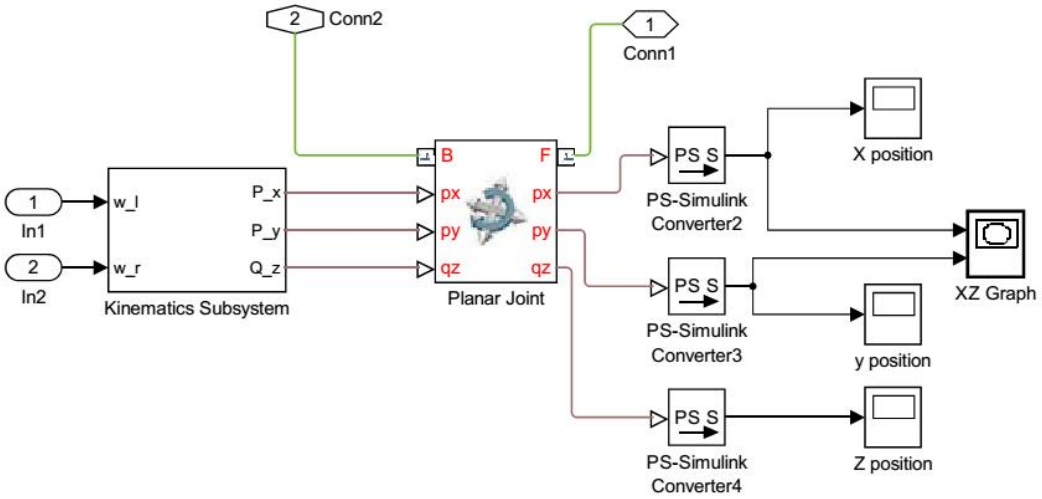


Fig. 3.4 Motion control unit

3.2.2 Obstacle Avoidance Algorithm

This section describes the algorithm for controlling the robot to avoid the obstacles. The control scheme is shown in Fig. 3.5. The assumptions of this algorithm are that the path and the obstacle geometry are known and the target and the obstacle are static in nature. The robot follows the predefined path as per line following algorithm and if any obstacle is detected by the ultrasonic sensor mounted on the head of the robot, current pose and orientation of the robot are calculated by the overhead camera using the image-processing technique. The data is fed to the control unit which calculates the minimum distance to reach the target by avoiding the obstacle using tangent bug algorithm. Then the robot surrounds the obstacle using wall following algorithm. After the obstacle is avoided, the robot again follows the predefined path to reach the target using line following algorithm.

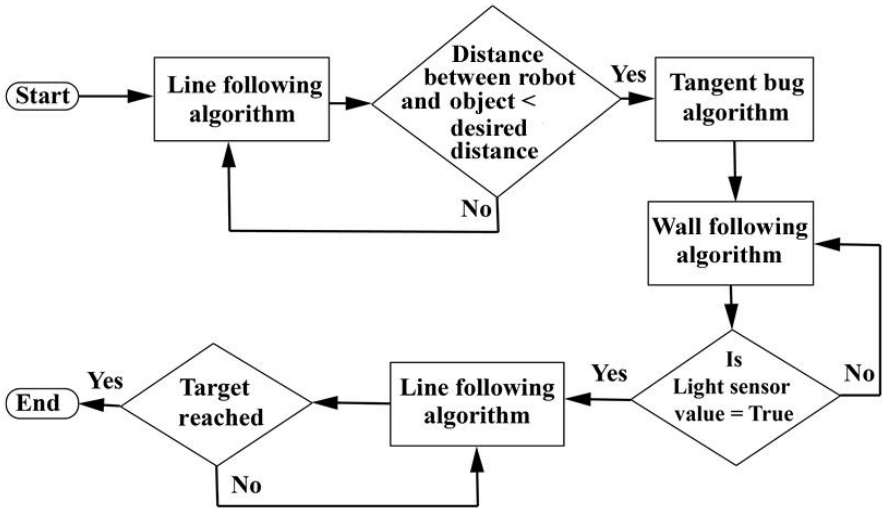


Fig. 3.5 Obstacle avoidance algorithm

The tangent bug algorithm decides whether the robot takes left or right turn to bypass the obstacle with the help of image captured from overhead camera. The detailed flow diagrams of the control algorithms are illustrated in Fig. 3.6. The two virtual sensors (ultrasonic) are mounted in front of the robot body. The sensor calculates the range of the robot body from upcoming obstacle. If the range is less than the desired value, the control algorithm is switched over to the tangent bug algorithm. This algorithm decides the least

distance of the robot from the target. The image from the overhead camera is processed using edge detection method and the least distance is calculated. As the output of the tangent bug algorithm is the shortest path to reach the target, the robot starts to move in the direction by desirable steer angle. Figure 3.7 shows the screenshots of the algorithm implemented in the *Labview* software.

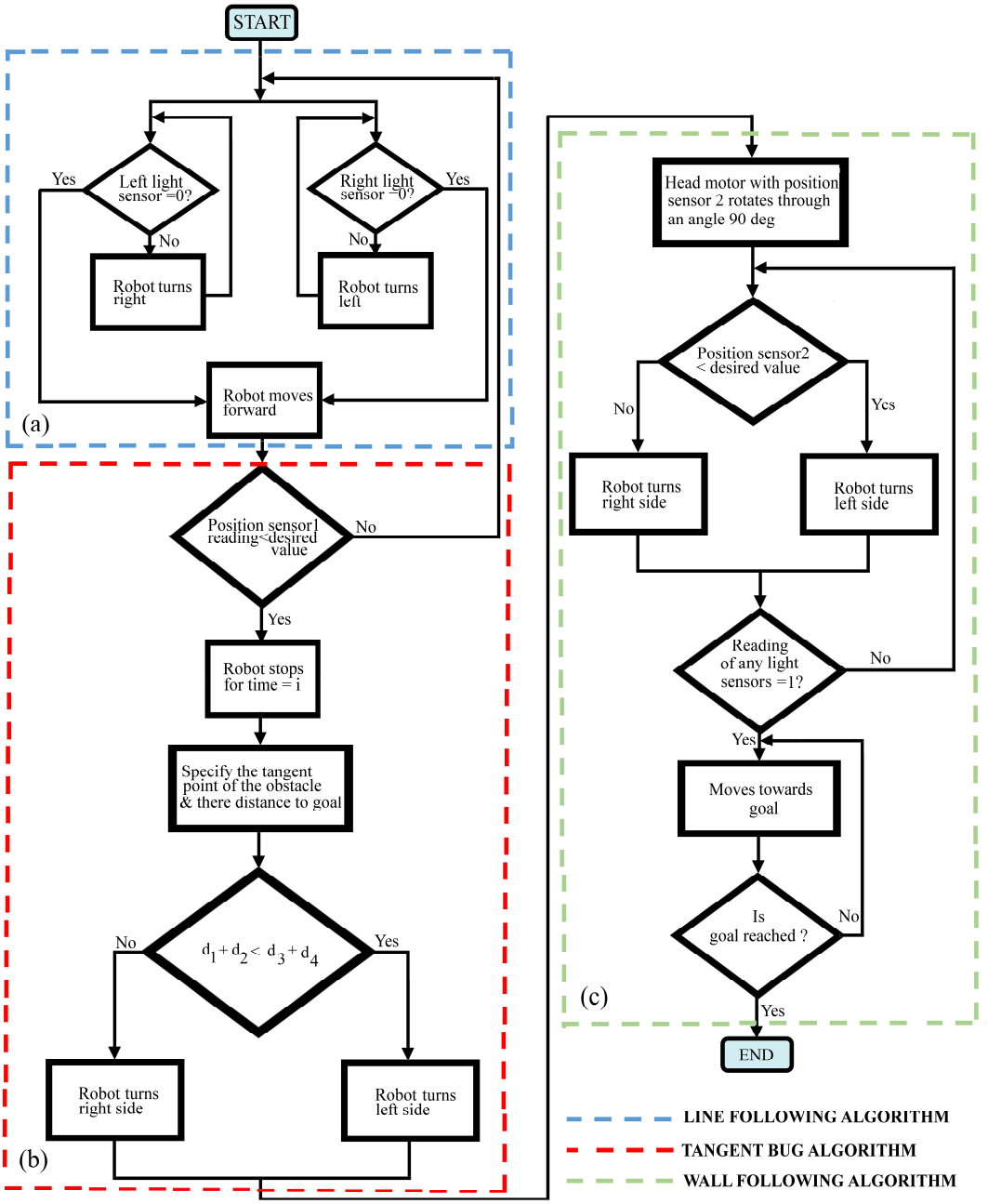


Fig. 3.6 Flow chart of obstacle avoidance algorithm: (a) Line following, (b) Tangent bug and (c) Wall following algorithm

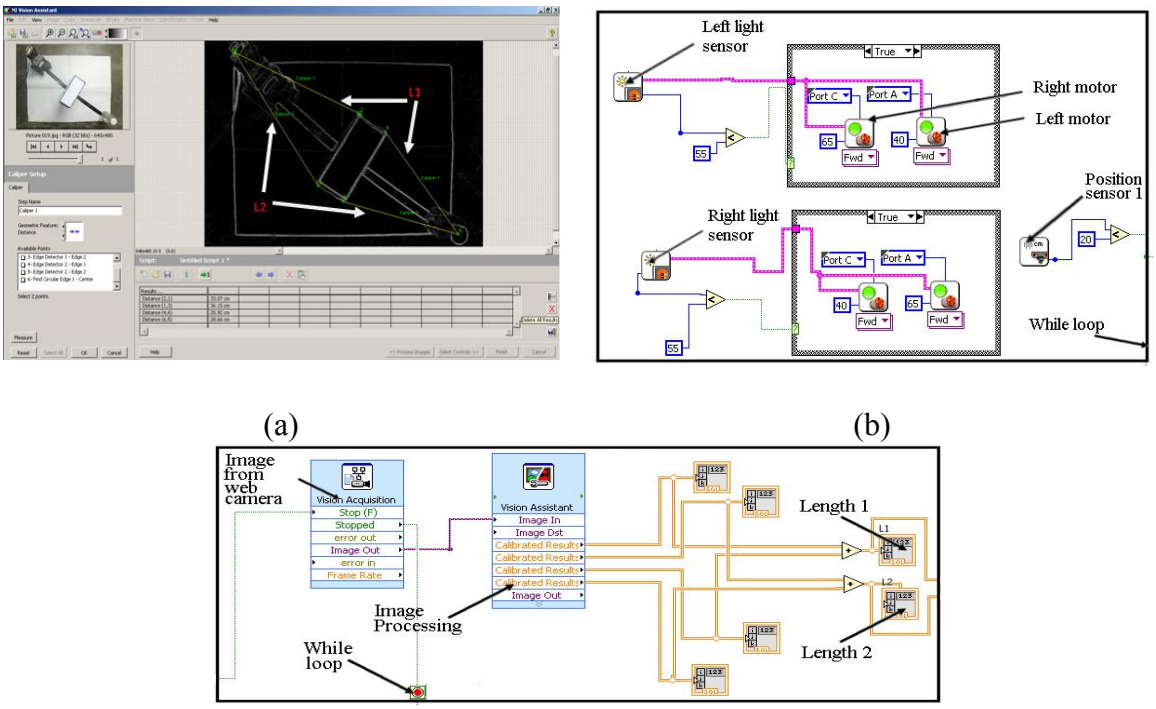


Fig. 3.7 Implementation of the algorithm: (a) Tangent bug algorithm (image processing), (b) Line following and (c) Tangent bug

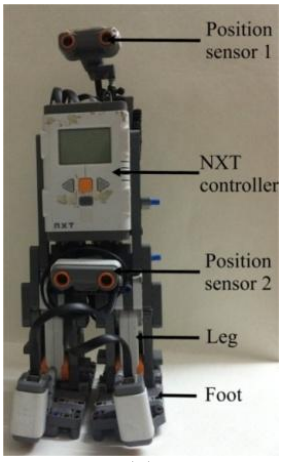
3.2.3 Experimentation Setup

The oscillating cylinder based leg mechanism for biped robot is developed in this work. Lego based hardware is used to develop the prototype model of the biped robot. The Lego NXT 2.0 controller can be easily interfaced with serial monitor for controlling purpose. Four sensors and three motors are connected with NXT controller. This is a low cost biped robot. The choice is based on the cost and the availability of experimental facilities. The inbuilt encoder servomotors are used to actuate the mechanism of the robot legs. The oscillating cylinder mechanism is used to move the robot leg. Two ultrasonic sensors (US_1 and US_2), one light sensor (LS) and two motors (M_1 and M_2) were used. The ultrasonic sensors, motor encoders and light sensor were used for obstacle detection and avoidance, prediction of position and detection of the predefined path, respectively. The codes were developed in Labview software. The communication between the robot and laptop is achieved by the communication port.

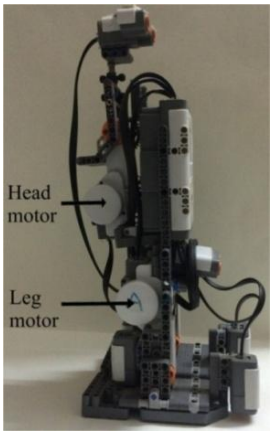
The prototype of the robot is shown in Fig. 3.8(a–c) and the environment, experimental setup, camera and laptop for data processing are shown in Fig. 3.8(d–g). The static obstacles, predefined start and target point and the known path are the assumptions for the arena. The overhead camera is used to monitor the current pose of the robot with respect to the obstacle and the target. Figure 3.8(d–g) shows the setup for the experimental work. Table 3.1 shows the detailed specifications of various sub-systems used in the experimental work.

Table 3.1 Specifications of sub-systems

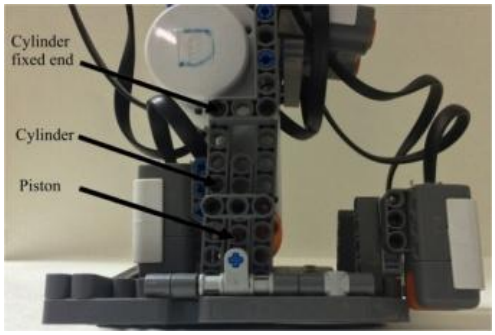
Subsystems	Specifications	Company name	Place of manufacture
Robot kit	Lego Mindstorms NXT 2.0	Mindstorms	United States
Camera	Night vision 5 G lens digital zoom	I ball	China
Ultrasonic sensor	0 to 255 cm with a precision of +/- 3 cm	Mindstorms	United States
Light sensor	1 to 2 cm, measuring light intensity	Mindstorms	United States
Controller	32-bit ARM7 microcontroller, 256 Kb FLASH, 64 Kb RAM	NXT	United States
Obstacle	0.2 m × 0.11 m × 0.37 m		
Arena	1.4 m × 0.6 m		



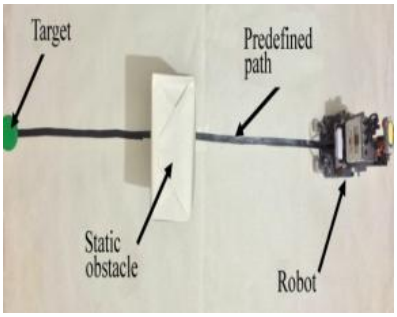
(a)



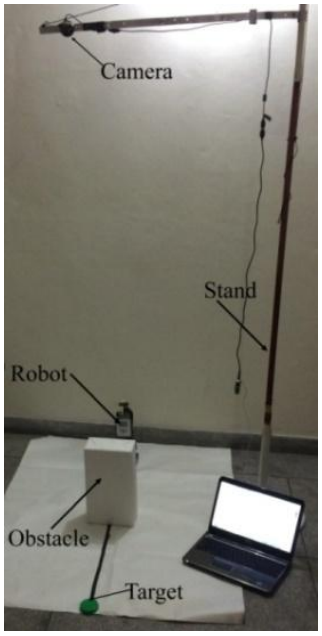
(b)



(c)

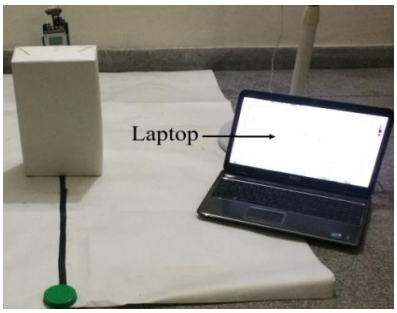


(d)



(e)

(f)



(g)

Fig. 3.8 Actual robot and experiment setup: (a) Front view, (b) Side view, (c) Leg mechanism, (d) Environment, (e) Experimental setup, (f) Overhead camera and (g) Laptop for data processing

3.2.4 Results and Discussions

The robot moves from the start position and stops at a distance (predefined) ahead from the obstacle, takes turn according to the calculated steering angle using tangent bug algorithm, avoids the obstacle and finally, reaches the target position. Figure 3.9 shows the different positions of the robot during an experimental run. The paths followed by the robot in simulation and experimental work are shown and are compared in Fig. 3.10(a). The grey line shows the experimental results whereas; the black line indicates the simulation results. The simulation and experimental results are shown in Z-X plane. The

pattern of simulation results is quite similar to experimental results. It is seen from Fig 3.10(a) that in the experiment, the robot takes left turn approximately 5 cm after the left turn of the robot in simulation result. The origins of inaccuracy of trajectory tracking as depicted from Fig. 3.10 (a) may be due to the following reasons:

1. Slow response of the controller and the overhead camera.
2. Walking speed of the biped robot is slightly more than the desired speed as the robot has less rigid legs.

The percentage errors in *X* and *Z* directions are given in Fig 3.10(b).

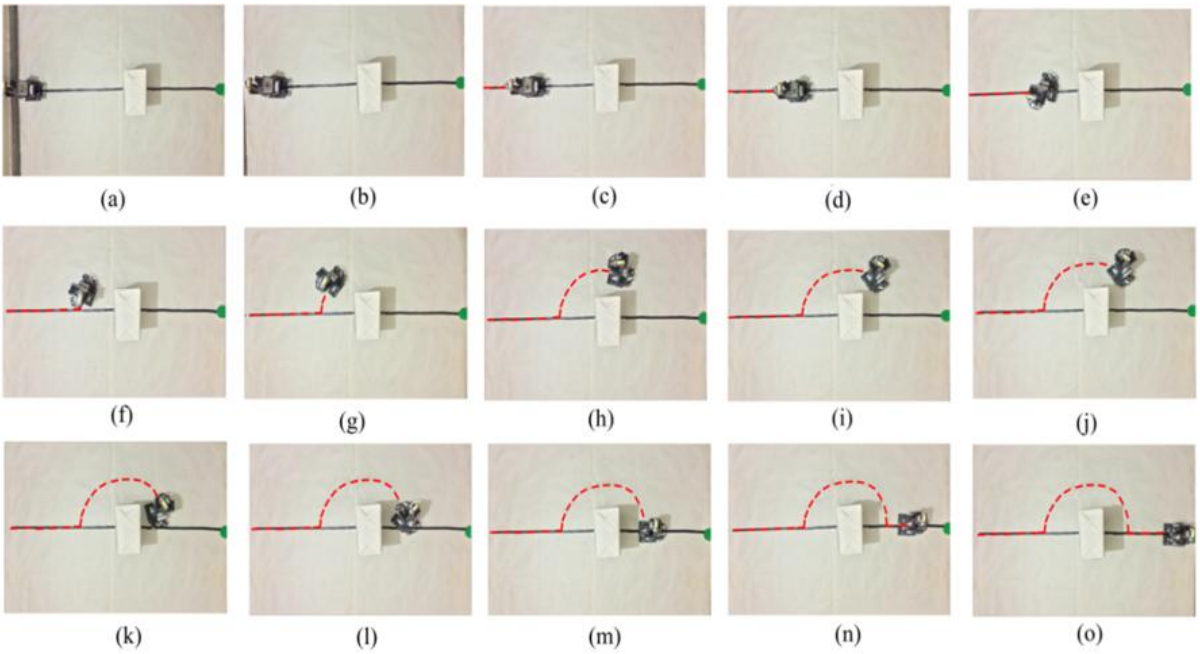
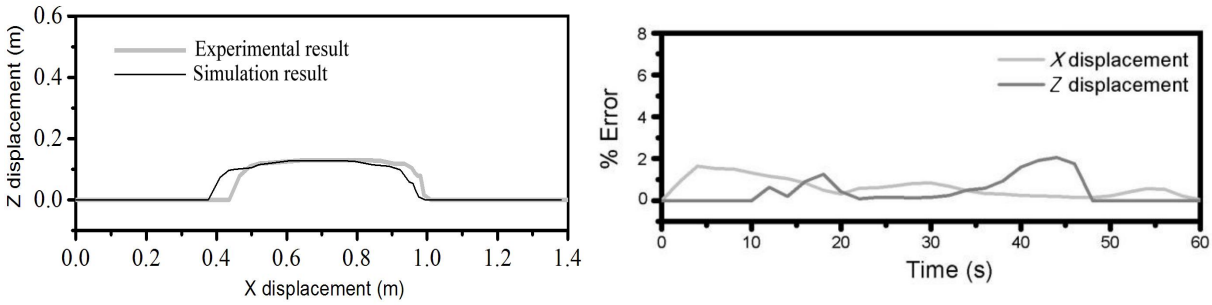


Fig. 3.9 Different poses of the robot to avoid a static obstacle in the experimental work



(a) (b)

Fig. 3.10(a) Validation of simulation and experimental results and (b) Error in *X*-direction and *Z*-direction

3.3 Dynamic Obstacle Avoidance for Four-Wheel Mobile Robot using Fuzzy Logic

The fuzzy logic based dynamic and static OAA of a four-wheel mobile robot is discussed in this section. One or more than one obstacles are considered to validate the obstacle avoidance algorithm performance. The bond graph (BG) model of the four-wheel mobile robot is developed. This bond graph model is then converted into the Simulink block to develop the controller in the *Matlab* environment. Then, navigation and obstacle avoidance controller based on the fuzzy logic was developed. The *Mamdani fuzzy model* is used to develop the fuzzy logic controller. The input membership functions are the distance of the robot from the target, angle and distance of the robot from the obstacles. Two output membership functions are the voltage fed to the motors of right and left wheel. Three ultrasonic sensors are used to detect and avoid the obstacles. The simulation has been done for static and dynamic obstacles.

This work was organised in the following sections: initially, the model was explained with the help of block diagram. Then, fuzzy logic input and output membership functions were presented. After that, fuzzy rules for the models were developed. Thereafter, bond graph models of the sub-systems and *Matlab-Simulink* model were presented. Finally, simulation results of the model for static, dynamic and both static-dynamic obstacles were shown.

3.3.1 Model Description

Firstly, the block diagram for navigation and obstacle avoidance of four-wheel mobile robot model is developed. Also, the fuzzy membership functions for input and output is discussed and developed. The fuzzy rules for controlling the motion of the robot to avoid the obstacles are developed. Then, the BG model of the four-wheel mobile robot is converted into the Simulink model. The fuzzy logic based controller to avoid obstacles is developed in *Matlab-Simulink* platform.

3.3.1.1 Block Diagram

A fuzzy logic based model for navigation and obstacle avoidance of four-wheel robot is developed. The present work deals with the development of navigation and obstacle avoidance controller by considering only the distance of the robot from the target, angle

and the distance of the robot from the obstacles. Fig. 3.11 shows the block diagram for the navigation and obstacle avoidance of four-wheel mobile robot.

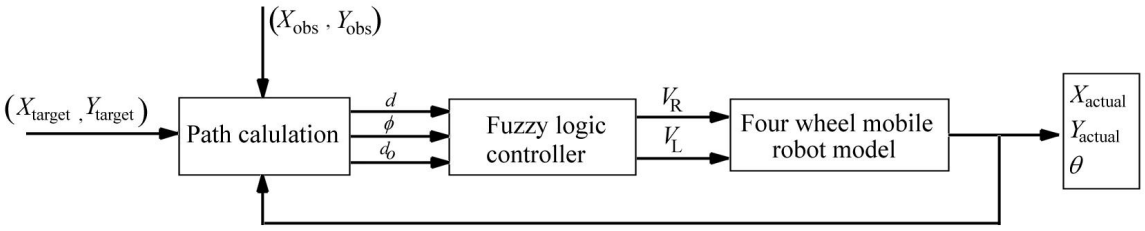


Fig. 3.11 Block diagram for navigation and obstacle avoidance of four-wheel mobile robot

It is seen from the block diagram in Fig. 3.11 that the distance of the robot from the target (d), angle (φ) and distance of the robot from the sensors (d_0) are computed based on the current position of the robot, position of obstacle and the target and these are fed to the fuzzy logic controller. This controller decides the voltages of the motors of left and right wheels for the absolute position of the four-wheel mobile robot in X and Y-direction of the robot. θ is the heading angle of the robot. This angle is not compared with any reference angle and current heading angle is required to calculate the angle (φ) of the robot. The current heading angle is obtained from Simulink model of four-wheel mobile robot.

3.3.1.2 Fuzzy Membership Functions

The model was developed for controlling the speed of motor of the four-wheel mobile robot. The three inputs i.e. distance of the robot from the target, angle and distance of the robot from the sensors are fed to the controller and two outputs i.e. velocities of right and left wheel are obtained. The input membership functions are shown in Fig. 3.12. The distance (d) of the robot from the target is the shortest or straight line path (crow fly distance) and is represented by the input membership functions. The actual path length covered by the robot (if curved path) is never used for any calculation. The actual position (X_c ,Y_c) of the robot with respect to the inertial frame is required to compute the distance (d) of the robot from the target and the angle (φ) of the robot. This actual

position of the robot at any instant is the shortest distance between the current and previous position. Zero membership function indicates that the distance of the robot from the target is zero or less than zero (robot crossed the target). For small membership function, robot distance from the target is greater than zero and less than 42 m. In the medium membership function, the distance is considered between 42 m to 122 m from the target. For the large membership function, the robot's distance from the target is greater than 119 m. Similarly, the zero membership function of the angle lies between -2 rad to 2 rad. For the negative membership function, the angle is zero or less than zero. Similarly, for positive membership function, the angle from the target is zero or greater than zero. There are always some overlaps between these three membership functions. Similarly, membership functions near, medium and far are the distance of the robot from the sensors for less than 5 m, between 5 m to 11 m and 8 m or greater, respectively.

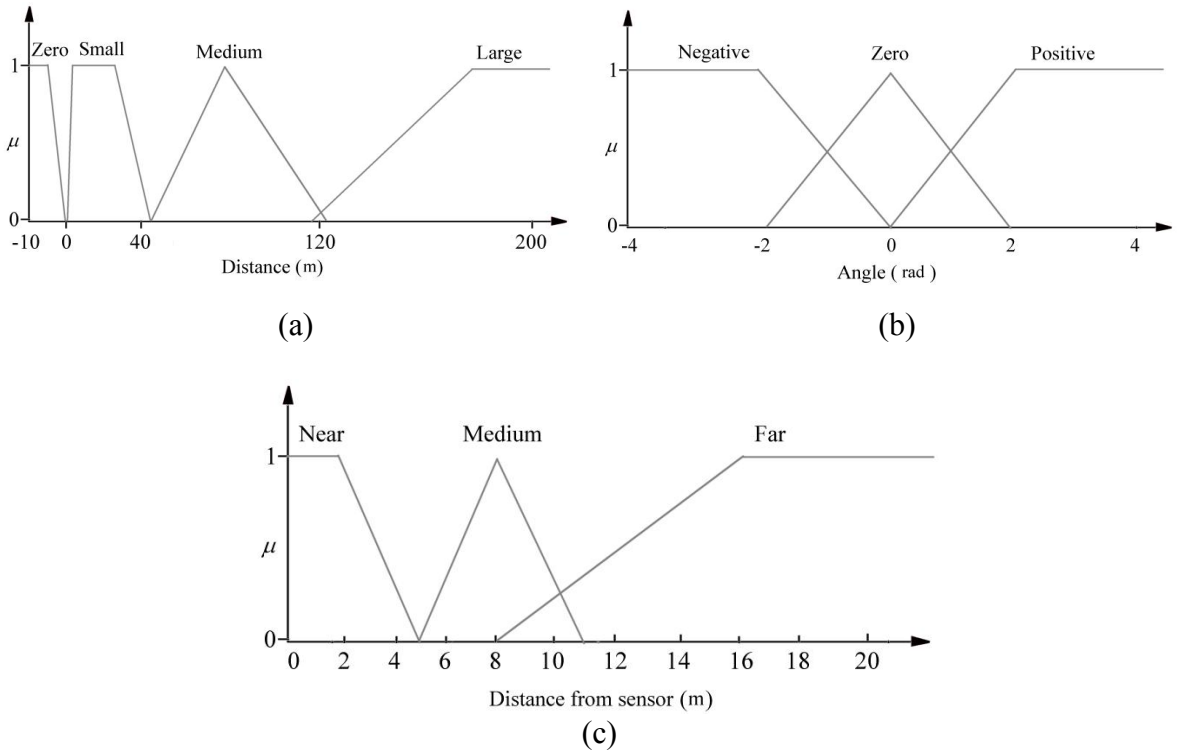
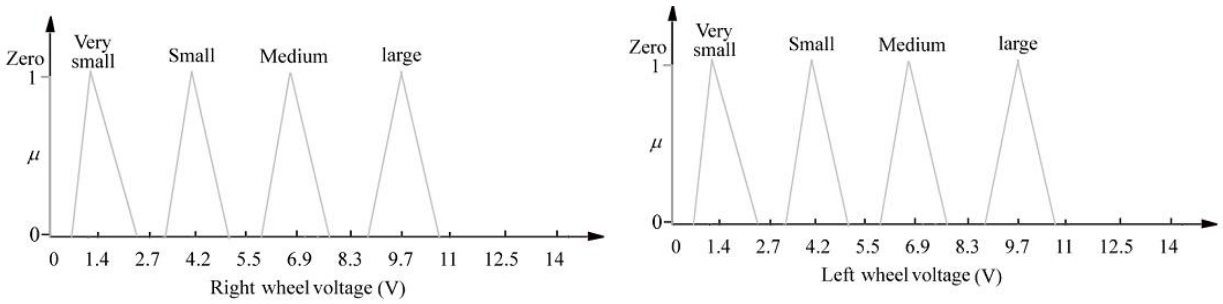


Fig. 3.12 Input membership functions: (a) Distance of the robot from the target, (b) Angle of the robot from target and (c) Distance of the robot from the sensors

The output membership functions are given in Fig. 3.13. Zero membership function means that voltage of the wheel is zero. For the very small membership function, the voltage is greater than zero and less than 2.5 V. For small membership function, the voltage lies between 3–4.8 V. For the medium membership function, the voltage is given to the motor varies between 5.8–7.8 V. For larger membership function, the voltage lies between 8.6–10.8 V. Similarly, zero, very small, small, medium and large membership functions are used for left wheel of the robot as depicted in Fig. 3.13(b).



(a) (b)

Fig. 3.13 Output membership function of velocity for (a) Right wheel and (b) Left wheel

3.3.1.3 Fuzzy Rules Matrix

The rules for navigation of the four-wheel mobile robot to reach the target are given in Table 3.2. These rules are developed by setting the membership functions during simulations. The rules for avoiding obstacles (static and dynamic) are given in Table 3.3.

Table 3.2 Fuzzy logic rules for target reaching

Input		Output	
Distance from target	Angle	Left wheel voltage	Right wheel voltage
Zero	Negative	Zero	Zero
Zero	Zero	Zero	Zero
Zero	Positive	Zero	Zero
Small	Negative	Very small	Medium
Small	Zero	Very small	Very small
Small	Positive	Medium	Very small

Medium	Negative	Very small	Large
Medium	Zero	Medium	Medium
Medium	Positive	Large	Very small
Large	Negative	Small	Large
Large	Zero	Large	Large
Large	Positive	Large	Small

Table 3.3 Fuzzy rules for obstacle avoidance

Input					Output	
Distance from target	Angle	Distance from sensors to the obstacle			Left wheel voltage	Right wheel voltage
		Sensor 1	Sensor 2	Sensor 3		
Large	Zero	Far	Far	Far	Large	Large
Large	Positive	Far	Medium	Medium	Medium	Small
Large	Positive	Medium	Medium	Medium	Small	Medium
Large	Positive	Medium	Far	Medium	Small	Small
Large	Positive	Medium	Medium	Far	Small	Medium
Large	Negative	Far	Medium	Medium	Small	Medium
Large	Negative	Medium	Medium	Medium	Medium	Small
Large	Negative	Medium	Far	Medium	Small	Small
Large	Negative	Medium	Medium	Far	Medium	Small
Medium	Zero	Far	Far	Far	Medium	Medium
Medium	Zero	Far	Medium	Far	Small	Large
Medium	Zero	Medium	Medium	Medium	Small	Medium
Medium	Positive	Far	Far	Far	Small	Large
Medium	Negative	Far	Far	Far	Large	Small
Medium	Zero	Near	Near	Near	Large	Very small
Medium	Positive	Near	Near	Near	Very small	Large
Medium	Negative	Near	Near	Near	Large	Very small

Small	Negative	Far	Far	Far	Large	Medium
Small	Positive	Far	Far	Far	Small	Medium
Small	Positive	Medium	Medium	Medium	Very small	Medium
Small	Positive	Near	Near	Near	Very small	Large
Small	Positive	Far	Medium	Medium	Very small	Medium
Small	Positive	Far	Medium	Far	Small	Large

3.3.1.4 Bond Graph Model

In this work, the BG model of the four-wheel mobile robot is developed. The kinematic model of four wheel mobile robot for obstacle avoidance is sufficient if the robot is lighter in weight. As the heavy weight (1630×9.81 N) robot is considered in this work (Table 3.4), the dynamic modelling of the robot using bond graph (multi-physics) approach is adopted for better results. It is not just the weight, but the speed also matters. At high speed, the inertial forces, especially the lateral forces, can be large and cause significant wheel slip. In such cases, dynamic models are necessary. However, the system becomes non-holonomic with wheel slip.The schema of the four-wheel mobile robot along with target and obstacle is shown in Fig. 3.14. (X_c, Y_c) is the current position of the centre of gravity of the mobile robot and the position of the target is (X_t, Y_t). The distance from the robot to the target is denoted by d. (X_o, Y_o) is the location of the obstacle. The distance of the robot from the obstacle is denoted by d_o. The heading angle of the robot is denoted by θ. The velocities of the right and left wheel are represented by V_r and V_l, respectively. Three ultrasonic sensors are used to detect the obstacles. The width of the mobile robot is shown by W and φ is the angle between the target and heading angle of the robot. The BG model of the four-wheel mobile robot is shown in Fig. 3.15. The output signal of the fuzzy logic controller modulates the voltages fed into the right and left wheel of the mobile robot. The reactive force (effort) from the ground is considered in the wheel in terms of Se: F_b. Similarly, Source of effort Se: F_f is a reactive force (effort) in the front wheel. The velocities of the robot in x and y-direction are denoted by

1_u and 1_v junctions, respectively. The heading angle of the robot is considered at 1_ω junction. The position of the robot is transformed into the inertial frame using transformer elements (TF). The displacement and heading angle are the inputs to the fuzzy logic (FL) controller.

3.3.1.5 Matlab-Simulink Model

The BG model of the four-wheel mobile robot is already developed in Section 3.3.1.4; now, this model is converted into the Simulink block using command *generate Malab S-function* from bond graph (*SYMBOLS Shakti*) software. The FL controller block is attached to the four-wheel mobile robot as shown in Fig. 3.16. The input to the FL controller i.e. distance of the robot from the target (d) is calculated from the current position of robot and the position of the target to be reached as given in Eq. (3.6). Similarly, the angle of the robot (φ) is the difference of heading angle (θ) and target angle (θ_t) as given in Eq. (3.7). The calculation of the distance of robot from obstacles (d_0) as per Eq. (3.8) is given for one sensor and this is same for other two sensors. The distance (d_o) of the robot from the obstacle is measured using three ultrasonic sensors mounted on the front chassis of the robot. The actual distance of the robot from the obstacle is summation of d_o and half of the wheel base. In this calculation, the wheel base is not considered as well as width of the mobile robot. The *Mamdani* algorithm is used in FL controller.

$$d = \sqrt{(X_t - X_c)^2 + (Y_t - Y_c)^2} \tag{3.6}$$

$$\phi = \theta_t - \theta \tag{3.7}$$

where $\theta = \tan^{-1}\left(\frac{(Y_t - Y_c)}{(X_t - X_c)}\right)$

$$d_0 = \sqrt{(X_o - X_c)^2 + (Y_o - Y_c)^2} \tag{3.8}$$

where (X_t, Y_t), (X_c, Y_c) and (X_o, Y_o) are the target position, the current position of the robot and obstacle position, respectively.

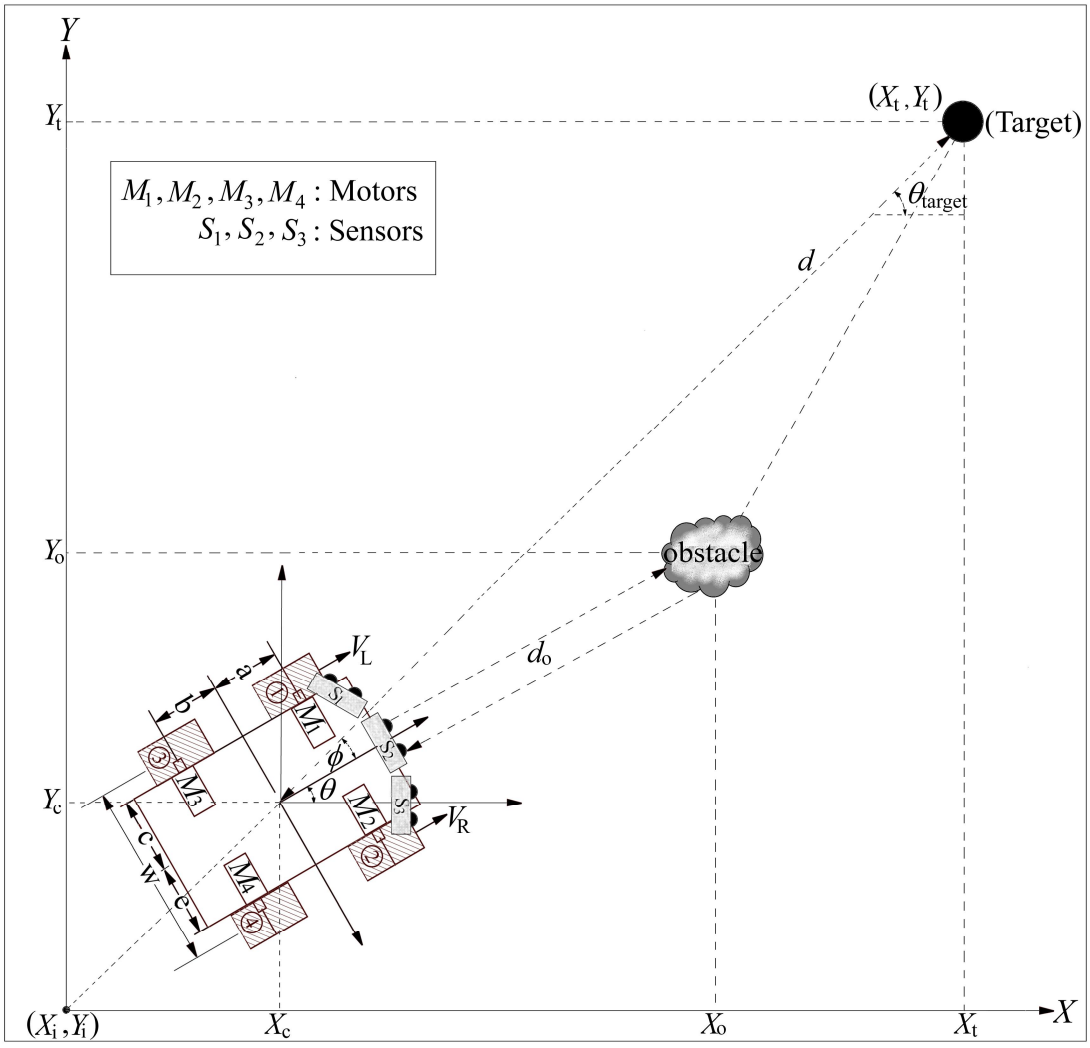


Fig 3.14 Schema of four wheel mobile robot along with obstacle and target

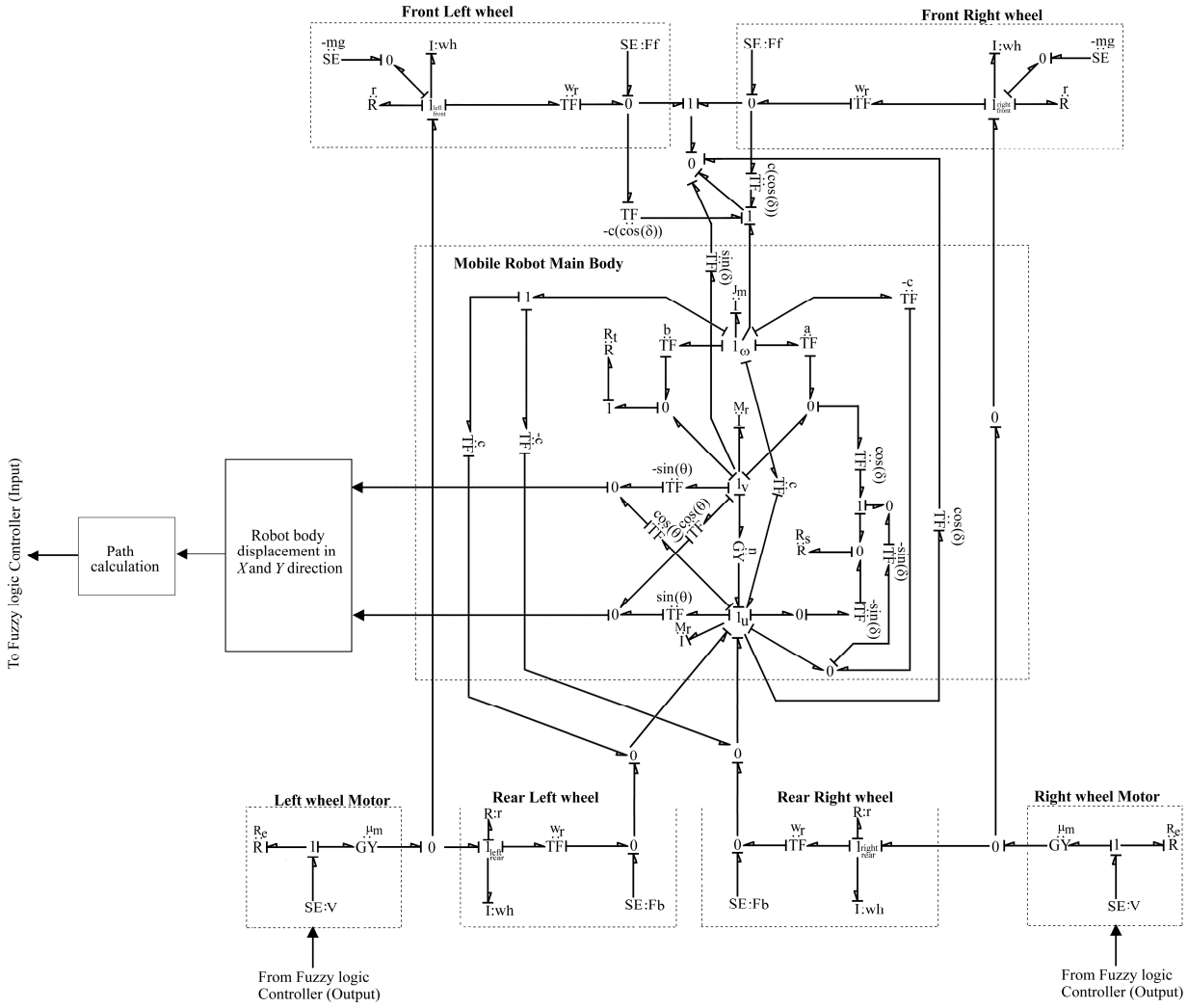


Fig 3.15 Bond graph model of four-wheel mobile robot

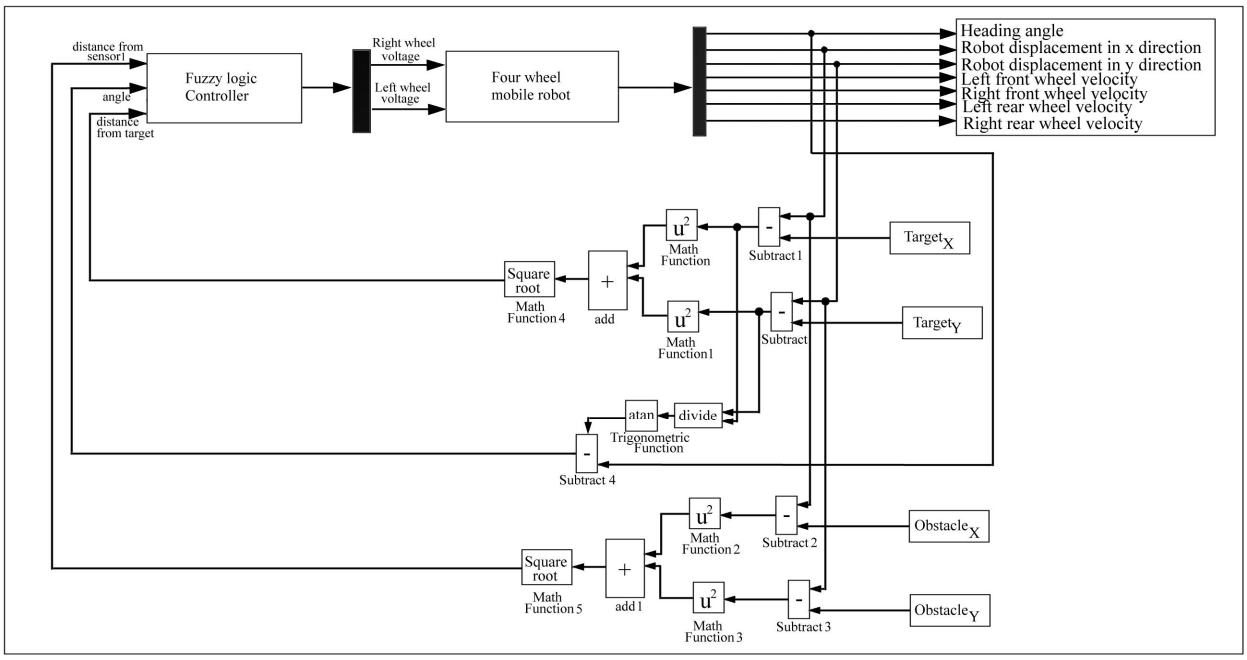


Fig. 3.16 Simulink model of the fuzzy control four-wheel mobile robot

3.3.2 Case Studies

3.3.2.1 Target Reaching of Robot without Obstacles

The parameter values are given in Table 3.4.

Table 3.4 Parameters used in modelling

Parameter	Unit	Value
Length of front axle from the centroid of the robot (a)	m	0.9
Length of rear axle from centroid of the robot (b)	m	1.5
Wheel radius (W_r)	m	0.3
Mass of robot (M_r)	kg	1630
Motor constant (μ_m)	N m/A	7
Mass moment of inertia of flywheel (J_m)	kg m^2	1110
Armature resistance (R_e)	Ω	0.1
Half of track width (c)	m	0.5
Mass moment of inertia of wheel (wh)	kg m^2	0.2

In the first scenario, the mobile robot starts from the origin heading towards right side i.e. zero heading angle and reaches the target without obstacles (static or dynamic). Figure 3.17 shows the result of the path traced by the robot to reach the goal. As depicted in Fig. 3.17(a), the robot starts from the origin (0, 0) and reaches the target (100, 100) by selecting a minimum distance to travel. In Fig. 3.17(b), the heading angle of the robot varies from 0 to 0.9 rad in the time duration of 11 min approximately and after that, the heading angle remains constant at 0.9 rad or 51.57°. It is seen from the Fig. 3.17(c–d) that robot starts to turn in the left direction as the right wheels have greater velocities (15 m/min) than the left wheels (13.5 m/min). After 20 min, the velocities of all the wheels become about 3 m/min and as soon as the robot comes close to the target, velocities of the wheels slow down to zero.

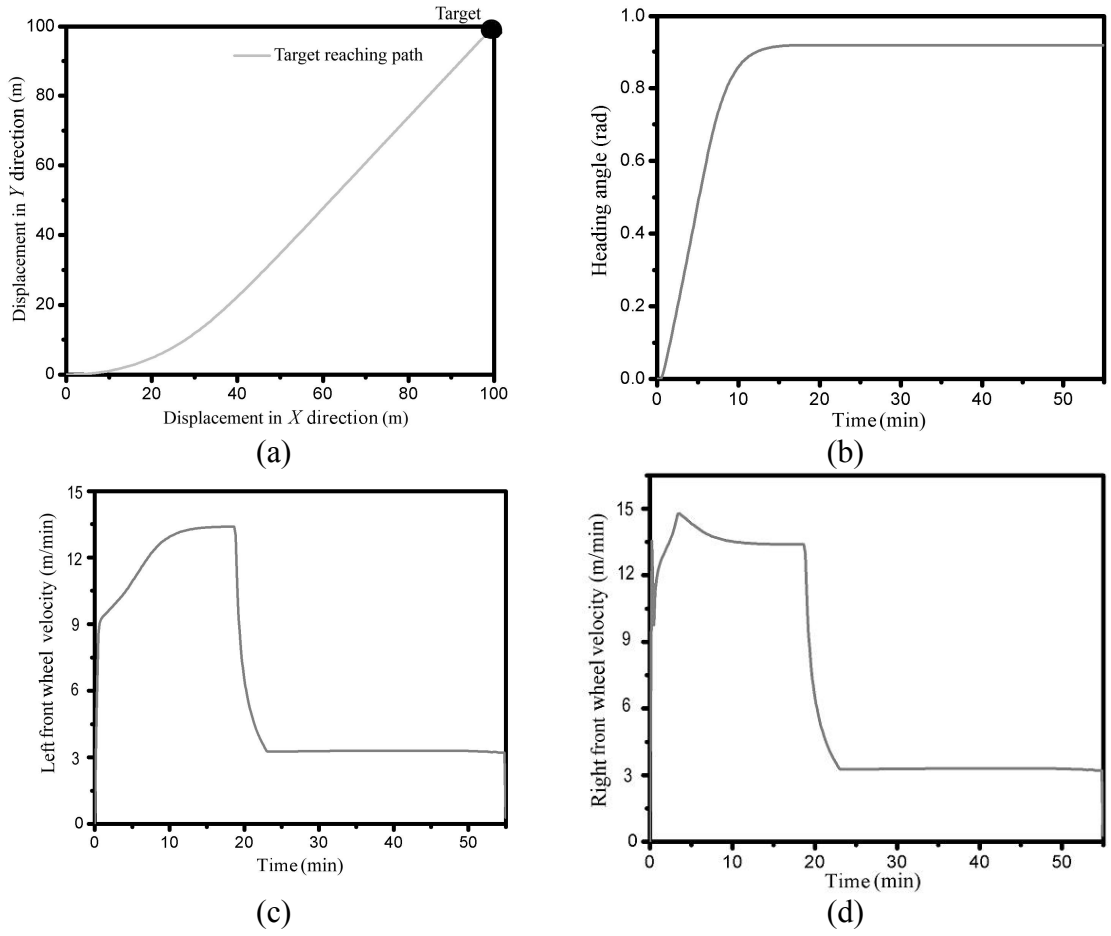


Fig. 3.17 (a) Path traced, (b) Heading angle of the robot, (c) Left front wheel velocity and (d) Right front wheel velocity of the robot for target reaching without obstacle

3.3.2.2 Target Reaching of Robot with Single Static Obstacle

In this case, the mobile robot reaches to the target by avoiding one static obstacle. The results are shown in Fig. 3.18. One static obstacle is located at (40, 22) on the path traced by the robot without obstacle. The path travelled by the robot while avoiding the static obstacle is illustrated in Fig. 3.18(a). The heading angle of the robot is shown in Fig. 3.18(b); robot starts to move in right direction and moves away from obstacle when it is detected by the robot. The maximum heading angle for single static obstacle (0.98 rad) is greater than the maximum heading angle without obstacle (0.9 rad). Figure 3.18 (c, d) shows the velocities of the four wheels of the robot. The maximum velocities of right and left wheels are 15 m/min and 12.9 m/min, respectively during turning of the robot towards left.

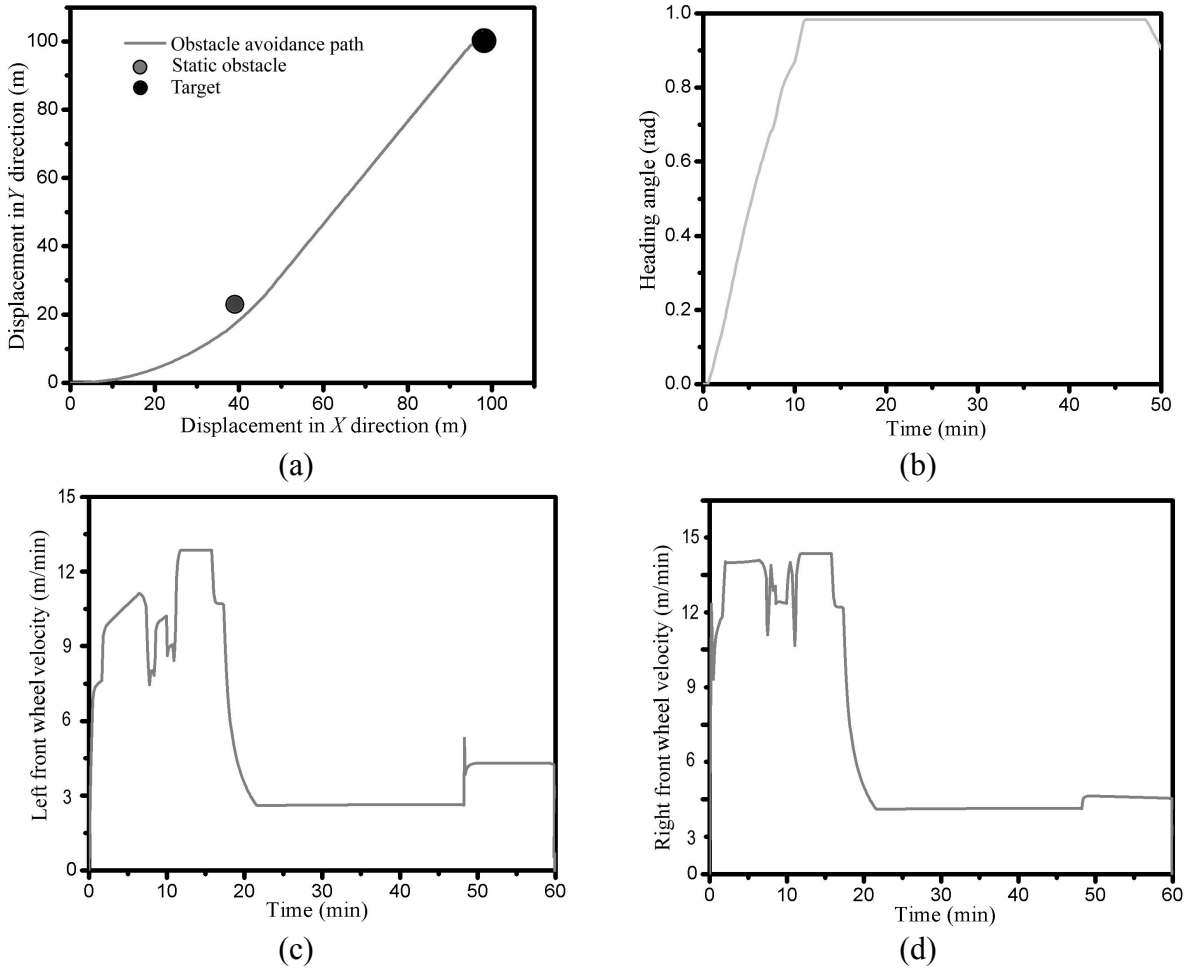


Fig. 3.18 (a) Path traced, (b) Heading angle, (c) velocity of left front wheel and (d) Velocity of right front wheel of the robot for target reaching with a single obstacle

3.3.2.3 Target Reaching of Robot with Two Static Obstacles

In this situation, the mobile robot reaches the target by avoiding two static obstacles. The path traced by the mobile robot and the original path are shown in Fig. 3.19(a). The first static obstacle is located at (40, 22) and the second static obstacle is located at (75, 69.02). Both the obstacles lie on the path traced by the robot without obstacle. During the left turn, the heading angle varies from 0 to 1.2 rad and for a right turn, the heading angle varies from 1.2 to 0.7 rad as illustrated in Fig. 3.19(b). For rest of the period, the heading angle remains constant at 0.7 rad. Figure 3.19(c, d) show the velocities of wheels of the mobile robot.

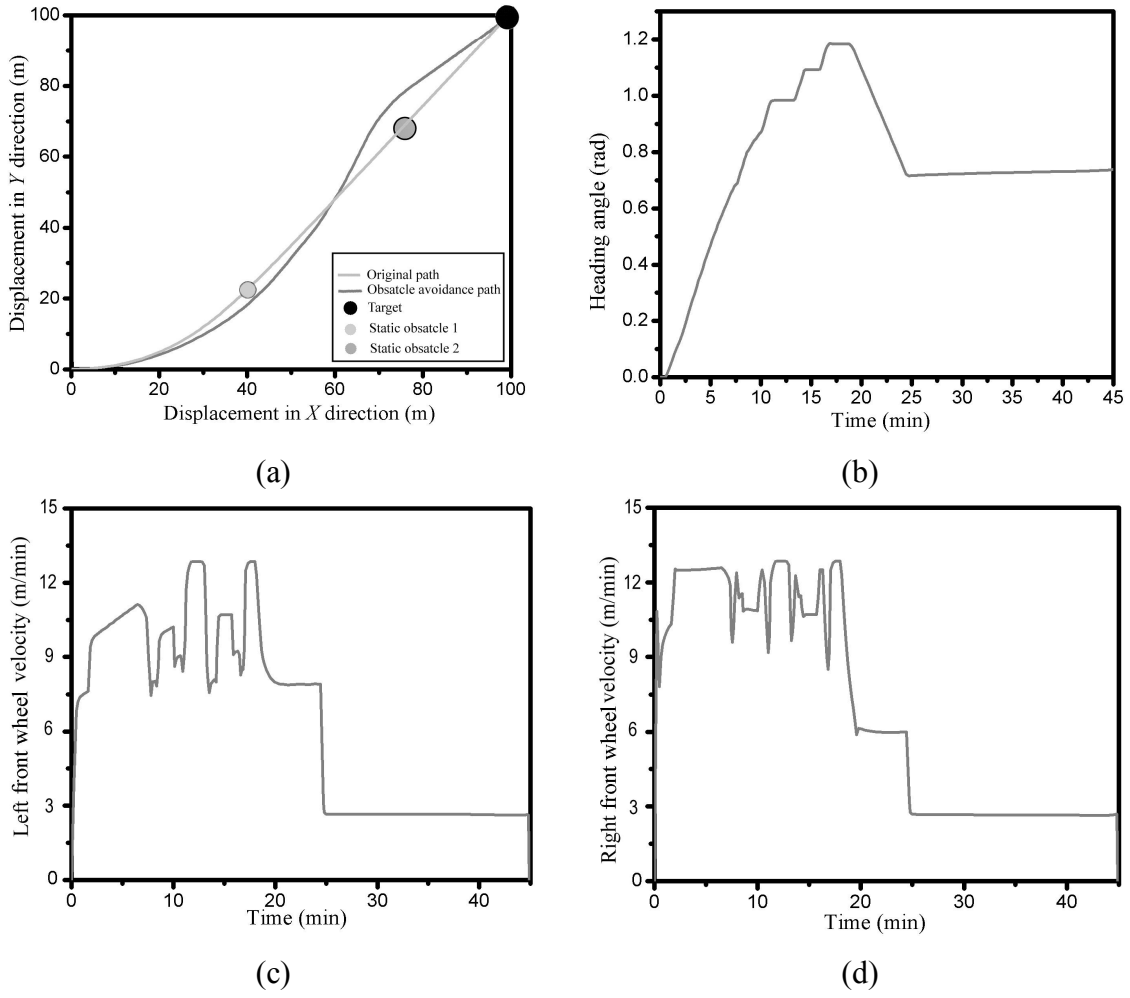


Fig. 3.19 (a) Path traced, (b) Heading angle, (c) Velocity of left front wheel and (d) Velocity of right front wheel of the robot during target reaching with two obstacles

3.3.2.4 Target Reaching of Robot with Three Static Obstacles

In this case, the mobile robot reaches the target by avoiding three static obstacles. The results are shown in Fig. 3.20. The first static obstacle is located at (40, 22), the second static obstacle is located at (75, 69.02) and the third static obstacle is located at (89.0, 90.10). All the obstacles lie on the path traced by the robot without obstacle. The path travelled by the robot while avoiding three static obstacles is shown in Fig. 3.20(a). Also in Fig. 3.20(a), the path traced by the robot without obstacle is shown. The heading angle of the robot is shown in Fig. 3.20(b). This heading angle varies from 0 to 1.2 rad for a left turn, for right turn, angle varies from 1.2 to 0.59 rad and then again for the last left turn, the heading angle changes from 0.59 to 0.98 rad. Figure 3.20(c, d) show the velocities of all four wheels of the mobile robot.

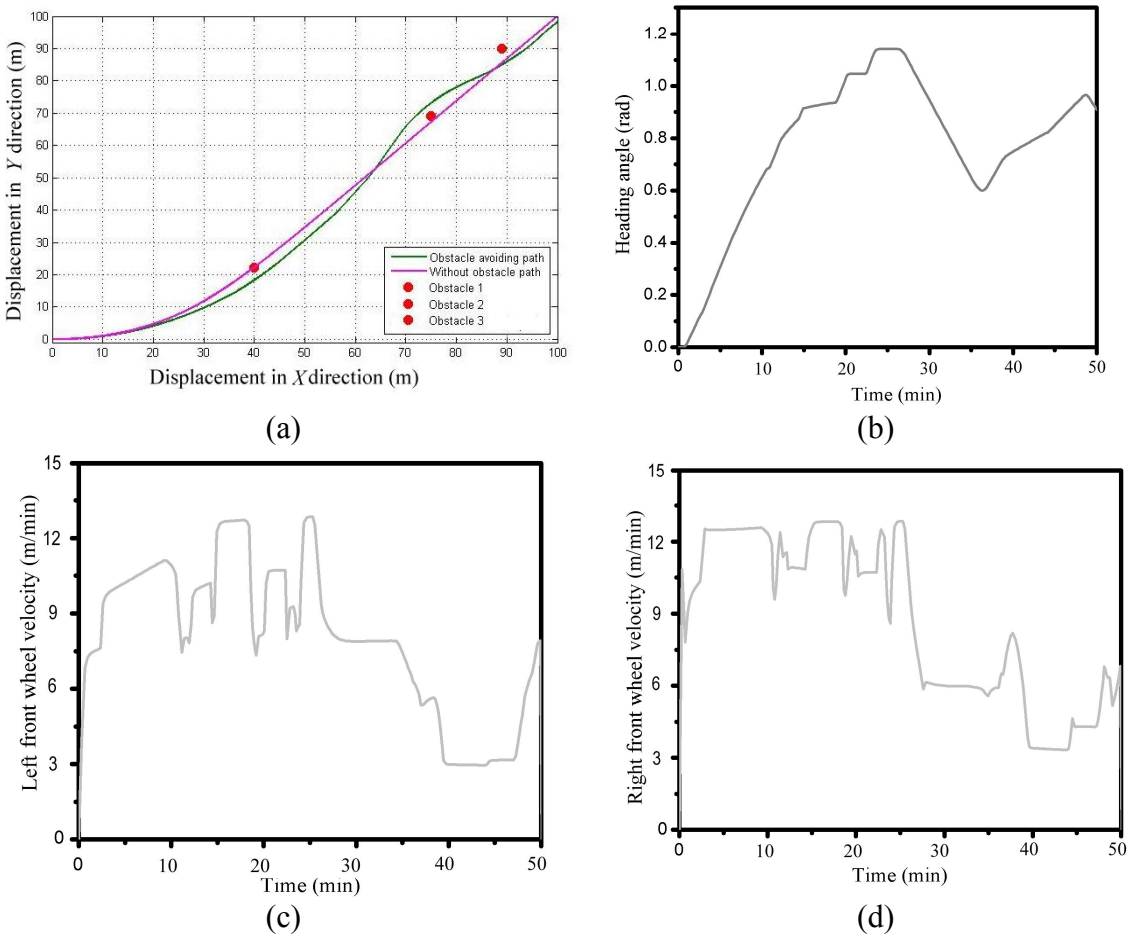


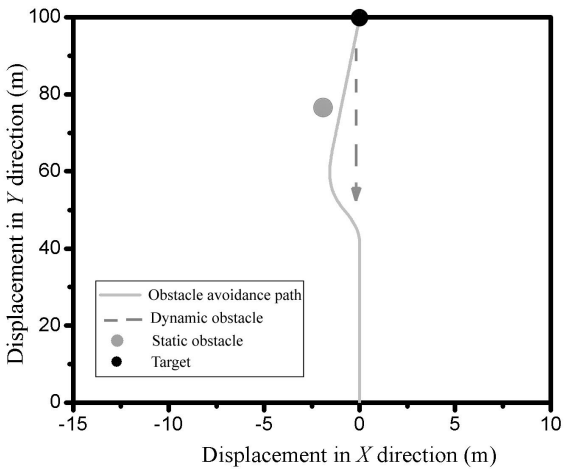
Fig. 3.20 (a) Path traced, (b) Heading angle, (c) Velocity of left front wheel and (d) Velocity of right front wheel of robot reaching the target with three obstacles

3.3.2.5 Lane Changing

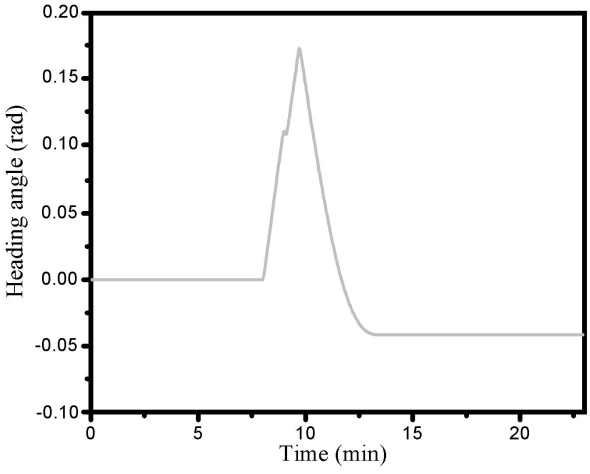
In this case, the dynamic obstacle is considered along with static obstacle and this is purely a lane change case. This situation is further divided into two generic cases.

- *Lane changing by avoiding a dynamic obstacle and then a static obstacle*

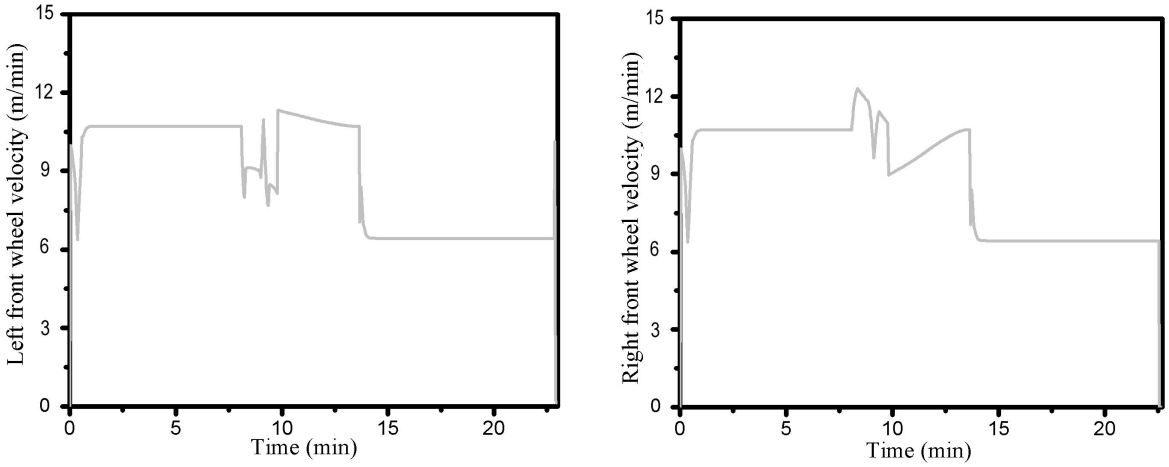
In this lane changing problem, one dynamic obstacle (maybe another robot) comes towards the mobile robot that starts from the origin. The robot may avoid the dynamic obstacle by changing the lane towards left. In the left lane, there is a static obstacle in the path of the robot; again, the robot changes the path towards the right by avoiding the static object and finally, moves toward the goal. The results are shown in Fig. 3.21. The lane changing of the robot is shown in Fig. 3.21(a), where the static obstacle is considered after the dynamic obstacle. Figure 3.21(b) shows the heading angle of the robot during changing the lane. The sensors are mounted on the robot to detect the position of the dynamic obstacle that comes towards the robot. The range is decided by the sensors membership function of FL controller after a number of simulations. Figure 3.21 (c, d) shows the velocities of the four wheels of the robot during lane changing.



(a)



(b)



(c) (d)

Fig. 3.21 (a) Path traced, (b) Heading angle, (c) Velocity of left front wheel and (d) Velocity of right front wheel during lane change considering dynamic and static obstacle

- *Lane changing by avoiding dynamic obstacle after static obstacle*

In this case, one static obstacle and a dynamic obstacle are considered. The difference with the earlier section is that robot initially avoids the static obstacle by changing the lane towards left. In the left lane, a dynamic obstacle comes towards the robot; the robot changes the lane towards the right and finally, moves towards the target. The results are shown in Fig. 3.22. The path traced by the robot is shown in Fig. 3.22(a), where the static obstacle is considered before the dynamic obstacle. Figure 3.22(b) shows the heading angle of the robot while changing the lane. Figure 3.22 (c, d) shows the velocities of the wheels of the robot during lane changing.

The effects of the uncertainties on modelling and external disturbances (contact efforts) cannot be considered to evaluate the robustness of the hybrid and fuzzy logic based obstacle avoidance algorithms, which have been proposed initially in the Thesis. This problem can be resolved by using overwhelming controller based obstacle avoidance algorithm. For this reason, third algorithm i.e. overwhelming control based obstacle avoidance is developed in Section 3.4 of the Thesis to avoid all external disturbances. While overwhelming control ensures robustness against modelling and parametric uncertainties, and external disturbances, the measurement uncertainties

(sensor noise, bias, etc.) cannot be handled by it. This would require implementation of appropriate filters or estimators, such as a Kalman filter. This can be a future research area. In our application, sensors are relatively noise-free and we have not faced problems due to measurement uncertainty.

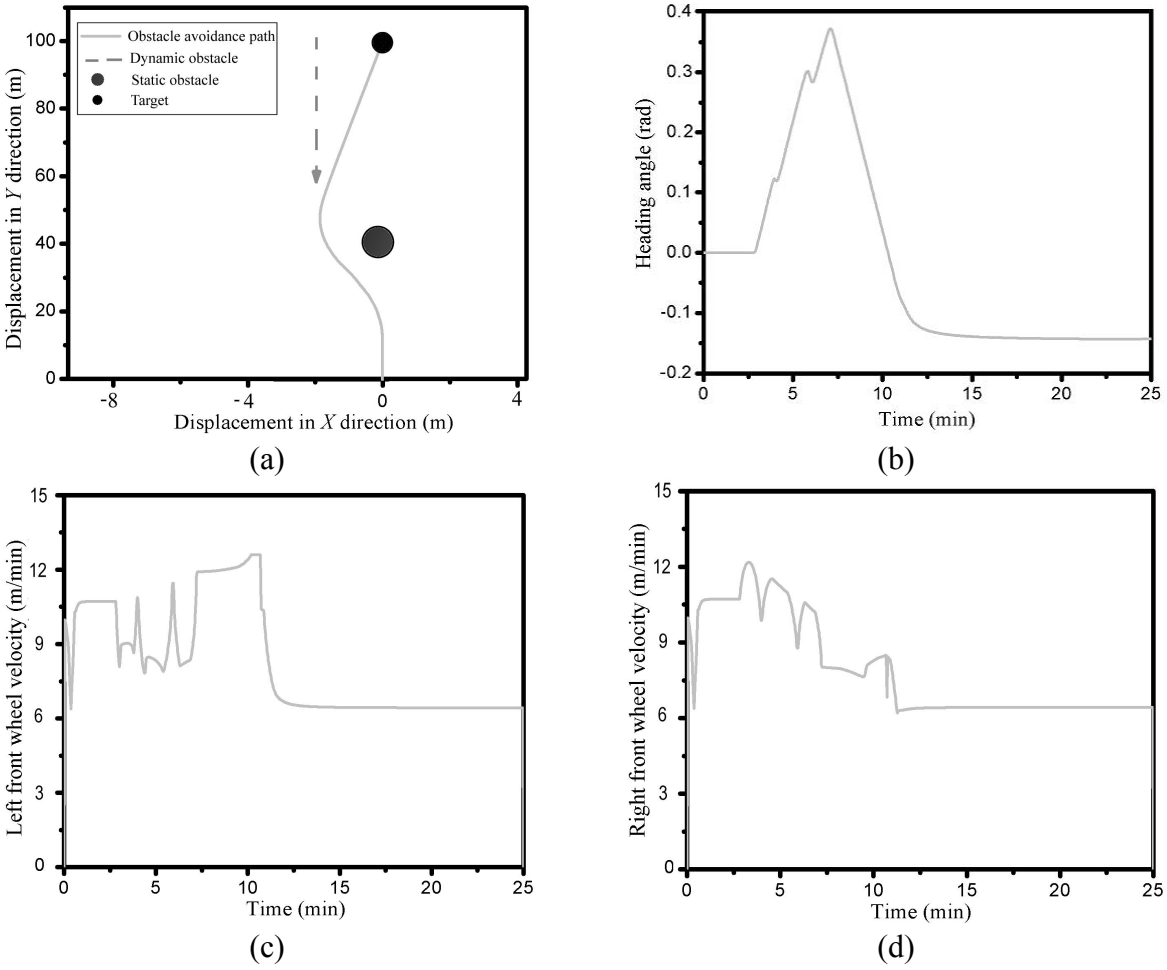


Fig. 3.22 (a) Path traced, (b) Heading angle, (c) Velocity of left front wheel and (d) Velocity of right front wheel during lane change considering static and dynamic obstacle

3.4 Obstacle Avoidance of Mobile Robot using Overwhelming Controller

Day by day, researches on the mobile robots are drastically increasing. The mobile robots are able to move on even or un-even surfaces as per the applications. The lot of research is active on the task of reaching the target by avoiding obstacles. The trajectory tracking is one of the feasible solutions if positions of the obstacle and the target are predefined.

In this work, the hybrid obstacle avoidance algorithm on two-wheel mobile robot is implemented. The trajectory is obtained from line following, tangent bug and wall following algorithms and this obstacle free trajectory is fed to the overwhelming controller to control the path of the robot. The effects of the uncertainties on modelling and external disturbances, if any, are removed by the overwhelming controller.

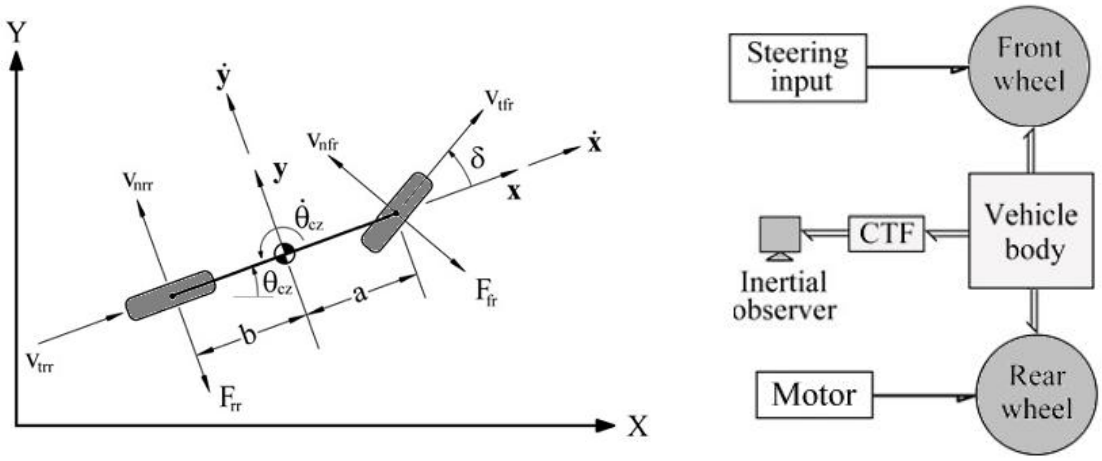
The work is structured as follows: in the first step, the forward model of bicycle-vehicle model is developed from the kinematic relations. Then, the inverse bicycle-vehicle controller is developed using overwhelming control scheme. A BG based overwhelming controller for the bicycle-vehicle model is discussed and developed. The obstacle avoidance algorithm is discussed and simulated for various cases depending upon the number, position and geometry of obstacles. These results are fed into the overwhelming controller. Finally, the simulation results for the trajectory tracking avoiding obstacles are shown and discussed.

3.4.1 Bicycle Vehicle Model

A bicycle model of the vehicle in the horizontal plane was used for the initial study of two-wheel mobile robot. The assumptions to this model are that roll, pitch and heave motions and the suspension dynamics are not considered. Therefore, the load transfer during yaw motion cannot be considered in this model. The schema of the mobile robot is illustrated in Fig. 3.23(a) and its word bond graph is depicted in Fig. 3.23(b). The flat surface and planar motion are considered for the mobile robot.

3.4.1.1 Forward Dynamics

The forward dynamics deals with to calculate the kinematic parameters (position, velocity and acceleration) of the body subjected to given set of forces. In robot forward model, the torque from motor shaft is given to the rear wheel and steering angle is provided to the front wheel. Therefore, robot's position is calculated along x and y-direction.



(a) (b)

Fig. 3.23 (a) Representation of the bicycle-vehicle model and (b) Word bond graph

3.4.1.2 Kinematic Relations

The kinematic relations are used to construct the system. Thereafter, the wheel rotations, longitudinal and lateral slip calculations are fed into the system. The front wheel is steered (by steering angle δ) and power is given to rear wheel by traction motor (Se). Normal and tangential velocities of the front wheel are

$$V_{\text{nfr}} = \left(\dot{y} + \dot{\theta}_{\text{CZ}}\, a\right)\cos\delta - \dot{x}\sin\delta \tag{3.9}$$

$$V_{\text{tfr}} = \left(\dot{y} + \dot{\theta}_{\text{CZ}}\, a\right)\sin\delta + \dot{x}\cos\delta \tag{3.10}$$

Likewise, normal and tangential velocities of the rear wheel are

$$V_{\text{nrr}} = \left(\dot{y} - \dot{\theta}_{\text{CZ}}\, b\right), \quad V_{\text{trr}} = \dot{x} \tag{3.11}$$

From Newton-Euler equations with $z = \dot{\theta}_{cx} = \dot{\theta}_{cy} = 0$, one obtains

$$m_{\text{v}}\ddot{x} = m_{\text{V}}\dot{\theta}_{\text{CZ}}\dot{y} + \Sigma F_x \tag{3.12}$$

$$m_{\text{v}}\ddot{y} = -m_{\text{V}}\dot{\theta}_{\text{CZ}}\dot{x} + \Sigma F_y \tag{3.13}$$

$$J_{\text{v}}\ddot{\theta}_{cz} = \sum M_z \tag{3.14}$$

3.4.1.3 Bond Graph Model

In the word BG of the bicycle-vehicle model illustrated in Fig. 3.23(b), a CTF block denotes coordinate transformations (Ersal *et al*., 2009). As per word bond graph model, the full bond graph model (Fig. 3.24) is drawn using Eqs. (3.9–3.11). Figure 3.24 shows the forward model which is used for trajectory tracking. The terms $m_v \dot{\theta}_{cz} \dot{y}$ and $-m_v \dot{\theta}_{cz} \dot{x}$ in Eqs. (3.12, 3.13) are conservative pseudo-forces which are implemented by a gyrator (GY) element in the bond graph. The robot inertias are represented in the moving frame and the pseudo-forces transform them into the inertial frame. The rotary inertia (J_v) is connected at 1-junction. The values of transformer elements (TF) are based on the Eqs. (3.9–3.11). The flow detectors (Df) are used to generate the x and y positions of the robot's centre of mass in the inertial frame. The MR (modulated resistance) element is used to implement Pacekja's magic formula which gives rolling resistance to the system. The steering angle calculated as $\delta = \tan^{-1}\left(\frac{\dot{y}_{cz} + a\dot{\theta}_{cz}}{\dot{x}_{cz}}\right)$ considering small values of δ. The modulated transformers (MTF) are modulated by velocities of the robot ($\dot{x}, \dot{y}$ and $\dot{\theta}_{cz}$) from the inverse controller.

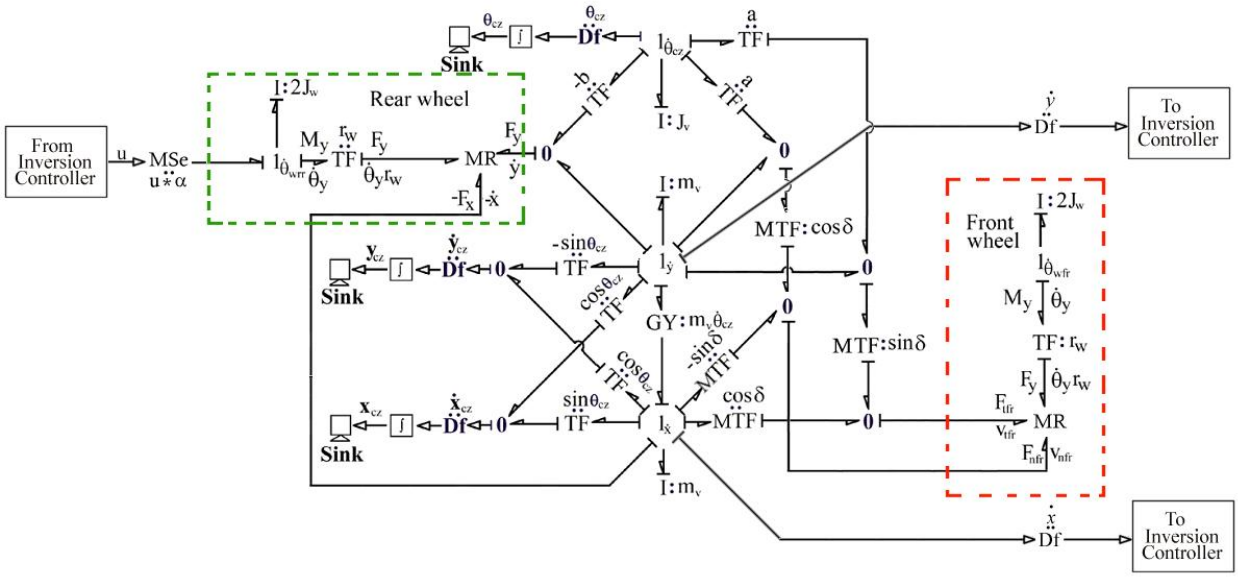


Fig. 3.24 Bond graph model of bicycle-vehicle model used for system inversion

3.4.2 Inversion of the System through an Overwhelming Controller

The plant dynamics is overwhelmed by the controller and this controller is referred to as the overwhelming controller. The controller forces the plant to follow the trajectory by neglecting the unaccounted dynamics if at all exists. This concept has been used in the past for developing controllers for serial and parallel manipulators of the robot and other applications. The concept of the overwhelming controller is shown in Fig. 3.25(a) as an example to explain the basics of robust overwhelming control.

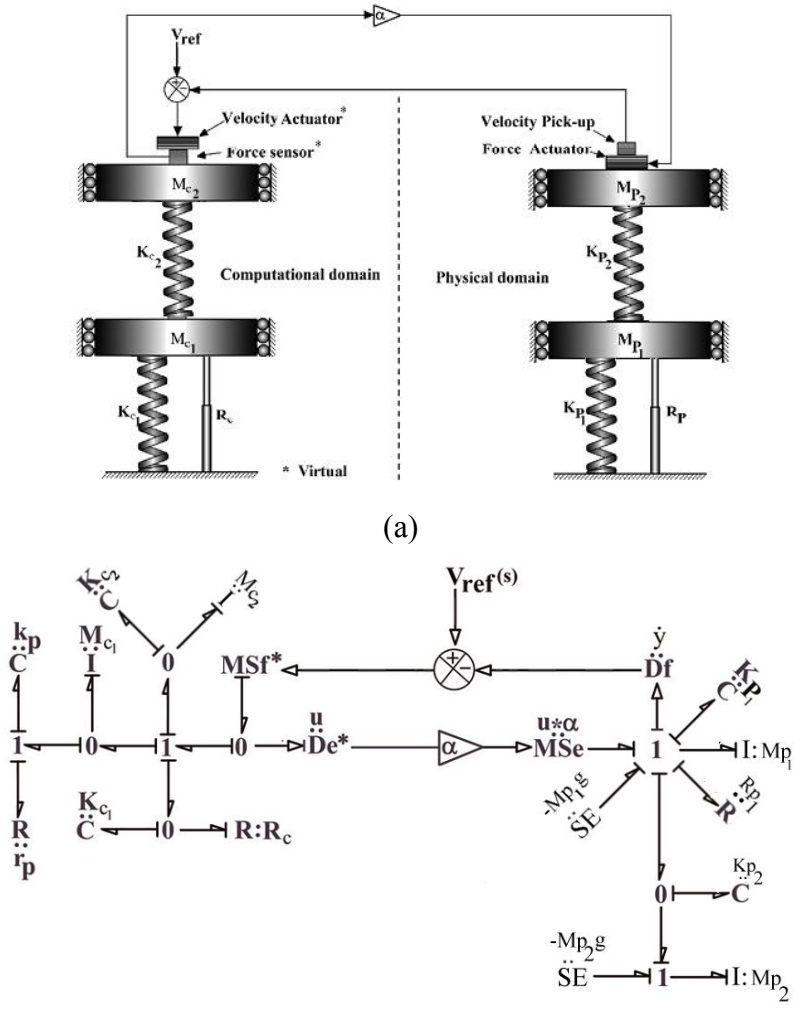


Fig. 3.25 (a) Concept of the overwhelming controller and (b) Bond graph model of the system

In this method, the mirror image of the actual system is generated in virtual domain for deriving control laws. The velocity pick-up element picks the velocity of the plant and then this velocity is compared with the reference velocity. The virtual system computes the reactive force. The virtual force sensor senses this force and amplifies it with a gain of α. Then a real force actuator applies this amplified force to the real system. The causalled BG model of the integrated system is illustrated in Fig. 3.25(b). To avoid the differential causalities, a pad or flexibility is provided at some required locations. The signal flow graph of the bond graph is illustrated in Fig. 3.26.

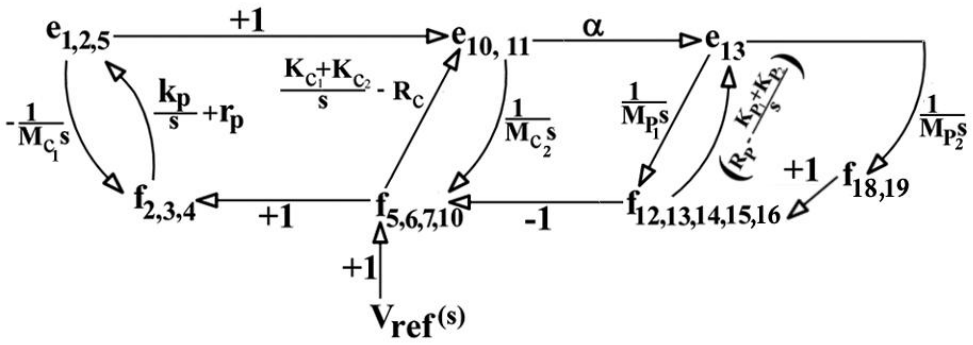


Fig. 3.26 Signal flow graph of the system

The transfer function between the velocity of the plant mass and the reference velocity is obtained as follows:

$$G(s) = \frac{f_{13}(s)}{V_{\text{ref}}(s)} = \frac{\alpha H_2(s)}{H_1(s) + \alpha H_2(s)} \tag{3.15}$$

Divide Eq. (3.15) by α and the following equation is obtained as

$$G(s) = \frac{H_2(s)}{\frac{H_1(s)}{\alpha} + H_2(s)} \tag{3.16}$$

Hence,
$$\lim_{\alpha \to \infty} G(s) \cong \frac{H_2(s)}{H_2(s)} = 1 \tag{3.17}$$

Therefore, the command is followed by the plant when the amplifier gain is very large. Hence, the overwhelming controller is applied to study the inverse dynamics of the bicycle-vehicle model. The problems of using overwhelming controller are the same as those faced with a high-gain controller. Most important is the noise sensitivity which

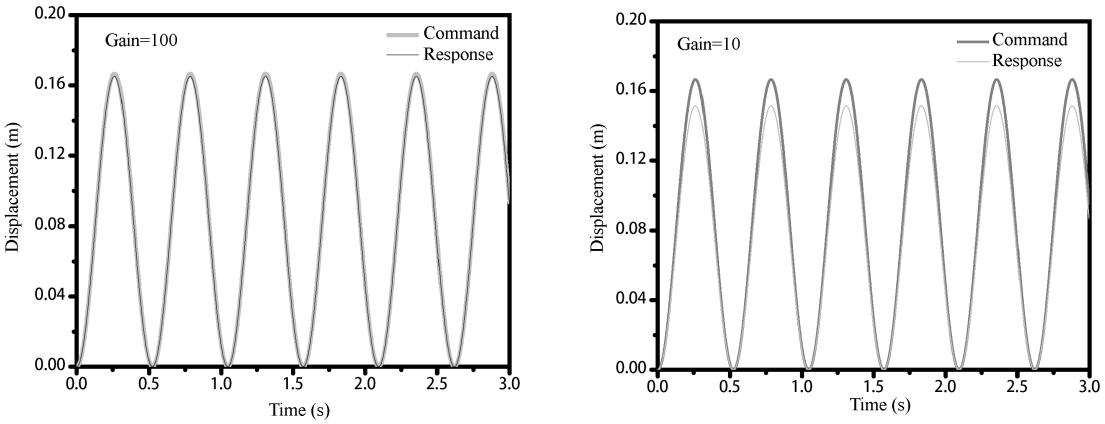
requires the use of appropriate filters for the sensor signals. In addition, the actuator inputs and slew rates may be outside the actuator capacity and hence, gain scheduling may be necessary. Therefore, although theoretically a high gain is needed in the controller, the gain values are not so high in real-time implementation. Furthermore, the overwhelming controller implementation requires solution of differential equations within the controller model and hence, the computational load on the controller side (computer or microprocessor device) is high. Thus, there is a need for fast processor and reduced order model. Otherwise, delayed feedback can cause instability. The simulation parameters values are shown in Table 3.5.

The inverse problem is to calculate the forces that should be given to the body to retain required kinematic parameters of the body in motion. As per Eq. (3.9), the gain (α) must have a very high value for the minimal error between the command and the response of the plant. It is observed from Fig. 3.27 that the value of the error between command and response decreases if the value of gain is increased from 10 to 100. The plant mass is displaced by 0.17 m from the initial condition. The reference displacement command is a symmetric sine wave with velocity amplitude of 1 m/s and circular frequency of 12 rad/s. In Fig. 3.27(a) and (b), the response of the plant is shown for a higher value gain (α =100) and for a smaller gain value (α=10), respectively. The comparison between displacement error with a gain value of 100 and 10 for the overwhelming controller is illustrated in Fig. 3.28. The average error with gain of 100 is 0.98% which is quite acceptable than 10% error. Therefore, if the value of gain increases, the percentage of error reduces.

Table 3.5 Parameter values for overwhelming controller model

Parameter	Description	Value
M_{P_1}	Mass of first weight of the plant	1 kg
M_{P_2}	Mass of second weight of the plant	1 kg
K_{P_1}	The stiffness of first spring of plant	10^8 N/m
K_{P_2}	The stiffness of second spring of plant	10^8 N/m

R_{P}	The stiffness of damper of plant	10 Ns/m
$M_{\mathrm{C_1}}$	Mass of first weight of the inverse system	1 kg
$M_{\mathrm{C_2}}$	Mass of second weight of the inverse system	1 kg
$K_{\mathrm{C_1}}$	Stiffness of first spring of inverse system	10^8 N/m
$K_{\mathrm{C_2}}$	Stiffness of 2nd spring of inverse system	10^8 N/m
R_{C}	Stiffness of damper of the inverse system	10 Ns/m
K_{Pad}	Stiffness of pad	10^8 N/m
R_{Pad}	Damping of pad	10 Ns/m



(a) (b)

Fig. 3.27 Performance of the overwhelming controller for gain value of (a) 100 (b) 10

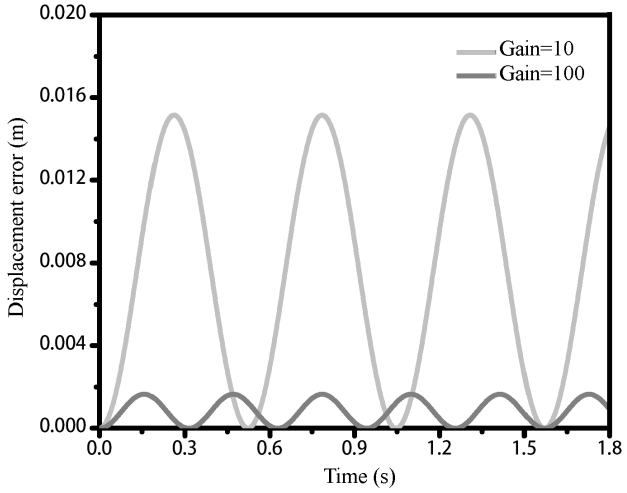


Fig. 3.28 Comparison between displacement errors for controller with different gains

3.4.2.1 Inverse Bicycle Vehicle Model

In the inverse model as depicted in Fig. 3.29, the position is given as input; required torque and steering angle are determined using control law. The inverse dynamic methodology is used in many research and engineering areas, such as optical devices, geophysical inverse methods, applications to spacecraft dynamics. Modern ground mobile robot dynamics have been proposed based on the forward dynamics methodology that shows the well understanding of the kinematic response of robot subjected to applied forces. The control of robot performance can be attained using inverse approach. In inverse dynamics system, the output of the actual robot is feedback to the inverse model (shown in Fig. 3.29) which is generally a controller whose input is predetermined. The robot's velocities are compared with the desired velocities and the error signal is given to the controller to produce the desired output.

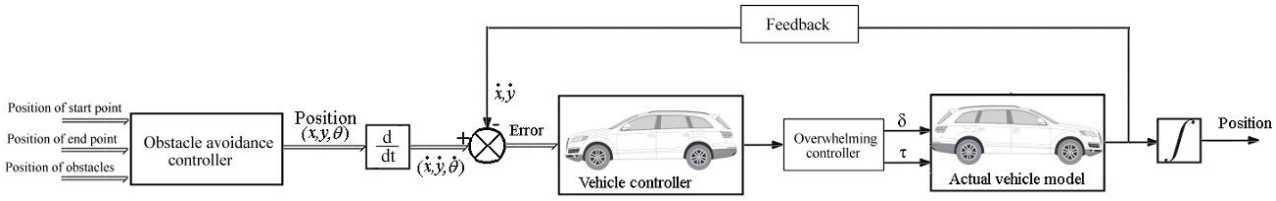


Fig. 3.29 Inverse vehicle dynamics control system

The inverse dynamics of the controller model for the bicycle-vehicle model is illustrated in Fig. 3.30. This model is the mirror image of the forward bicycle-vehicle model with few changes in power directions. The plant model in Fig. 3.24 is inverted as the input given here is velocity in x and y-direction to the centre of gravity of the vehicle and the output is torque and steering angle to the forward model. Some extra pad elements are added in the model to remove the differential causality of few I-elements. The final bond graph model is very easily computed and developed. The forward and transverse velocities of the robot are fed back to the inverse model.

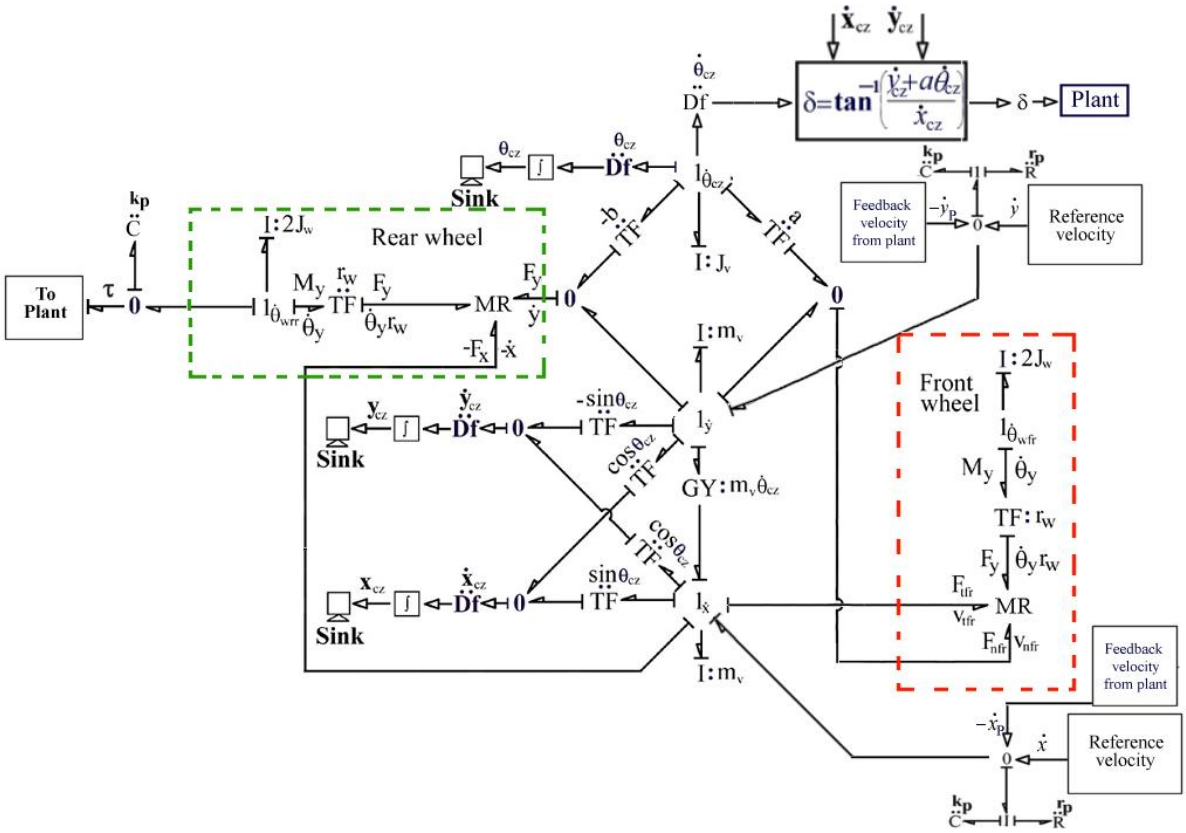


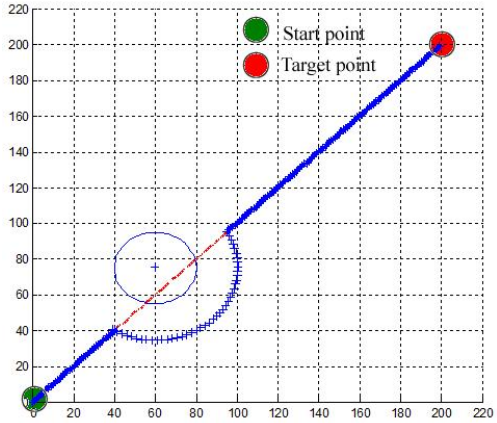
Fig. 3.30 Inverse bicycle-vehicle model

3.4.3 Obstacle Avoidance Algorithm

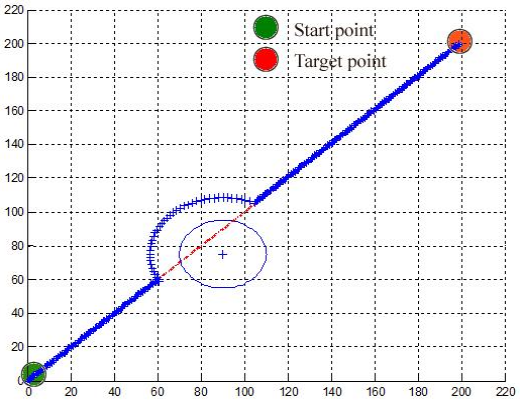
An obstacle avoidance algorithm is already discussed in Section 3.2.2. This algorithm is used to generate obstacle free trajectory which is fed into the overwhelming controller to control the path of the robot. The assumption to this model is that the path and obstacle geometry are predefined. The nature of the obstacle is static. The coding of the obstacle avoidance algorithm is done using *Matlab* and the test run of the algorithm is done for various cases depending upon the number, position and geometry of the obstacles. The cases are shown in Table 3.6. According to the scenario given in Table 3.6, the snapshot of simulation results of the algorithm is shown in Fig. 3.31 and this validates the feasibility of the obstacle avoidance for any situations.

Table 3.6 Various cases for a test run of obstacle avoidance algorithm

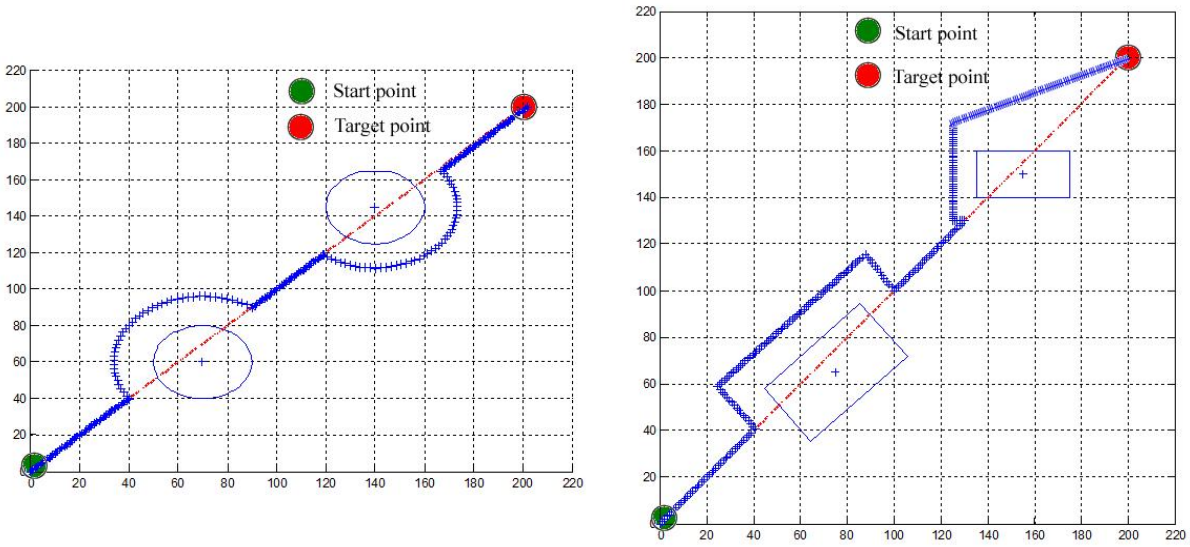
Cases	No. of obstacle	Position of obstacle's centroid		Geometry of obstacle
Case-I	Single	Left of centre line		Circular shape
Case-II	Single	Right of centre line		Circular shape
Case-III	Two	Left of centre line	Right of centre line	Circular shape
Case-IV	Two	Right of centre line		Rectangular shape



(a)



(b)



(c) (d)

Fig 3.31 The simulation results of obstacle avoidance algorithm for (a) Single circular obstacle left to the centre line, (b) Single circular obstacle right to the centre line, (c) Two circular obstacles (left and right to the centre line, respectively) and (d) Two rectangular obstacles both right to the centre line

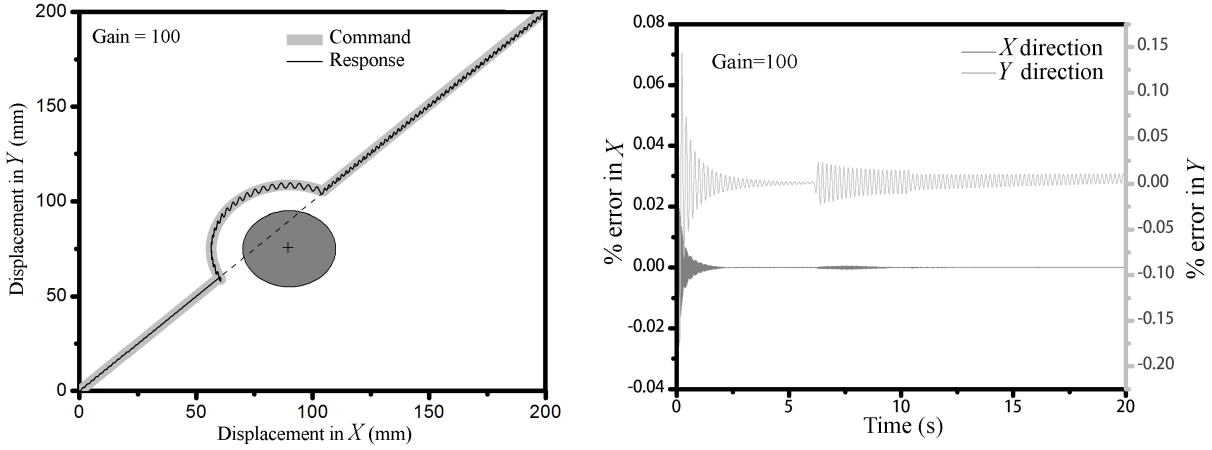
3.4.4 Simulation Results

The inverse model is developed using overwhelming controller model. The detail of the overwhelming controller model is discussed in the Section 3.4.2. In Section 3.4.3, the obstacle avoidance algorithm is developed and tested for various scenarios. The results from Section 3.4.3 are used in inverse bicycle-vehicle model that dictates the movement of forwarding bicycle-vehicle model. The starting and ending location of the robot is shown by green and red filled circles, respectively. The shortest path is shown by a red dashed line. The blue line (result from *Matlab* coding following obstacle avoidance algorithm) is the path to be traced by the robot. This path is fed to the inverse model as input. The various parameter values for the simulation using bond graph are shown in Table 3.7.

Table 3.7 Parameters for simulation

Parameter	Description	Value
Actual model		
r_w	Wheel radius	0.254 m
m_v	Mass of robot	924 kg
A	Distance between CG and front wheel	1.31 m
B	Distance between CG and rear wheel	0.62 m
Inverse model		
r_w	Wheel radius	0.254 m
m_v	Mass of robot	924 kg
k_p	Stiffness of pad	10^8 N/m
r_p	Damping of pad	10^8 Ns/m

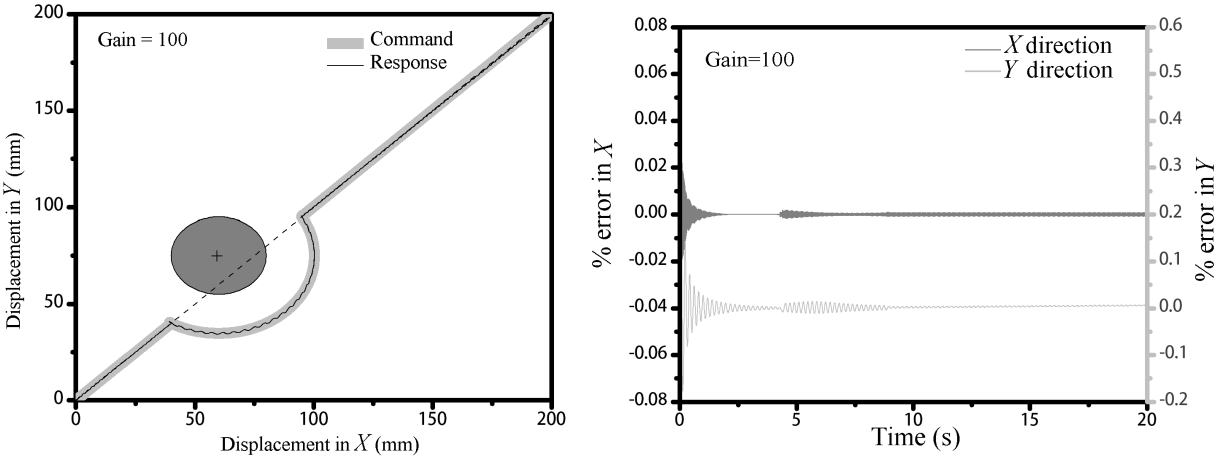
Figure 3.32(a) shows the Y vs X position of the robot for a single circular obstacle which is offset right to the shortest path. The response follows the command. The percentage errors in X and Y directions are shown in Fig. 3.32(b). The maximum percentage errors for X and Y directions are 0.01 and 0.05%, respectively for the overwhelming gain of 100.



(a) (b)

Fig. 3.32 Simulation results for single circular obstacle offset right to the centre line: (a) Comparison between command and response and (b) Percentage error in X and Y displacement

Figure 3.33(a) shows the Y vs X position of the robot for a single circular obstacle which is offset left to the shortest path. The response follows the command. The percentage errors in X and Y directions are shown in Fig. 3.33(b). The maximum percentage errors for X and Y directions are 0.02 and 0.1%, respectively for the overwhelming gain of 100.



(a) (b)

Fig. 3.33 Simulation results for single circular obstacle offset left to the centre line: (a) Comparison between command and response and (b) Percentage error in X and Y displacement

Figure 3.34(a) and (c) show the Y vs. X position of the robot for two circular obstacles (first one is offset to the right and the second one is offset left to the shortest path) with an overwhelming gain of 10 and 100, respectively. The response follows the command but the error is higher for the gain of 10. The percentage errors in X and Y directions are illustrated in Fig. 3.34(b) and (d). The maximum percentage errors for Y directions are 0.05 and 0.1%, for the overwhelming gain of 100 and 10, respectively.

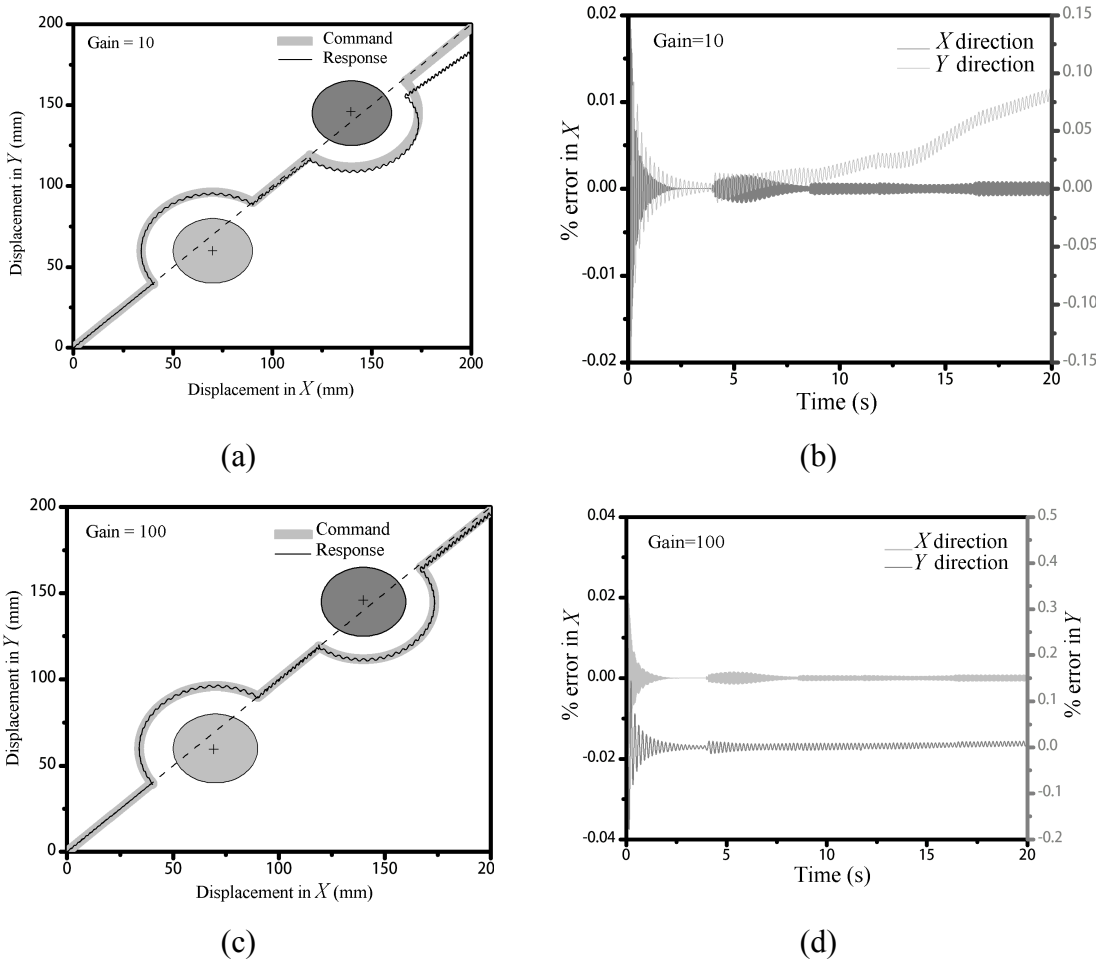
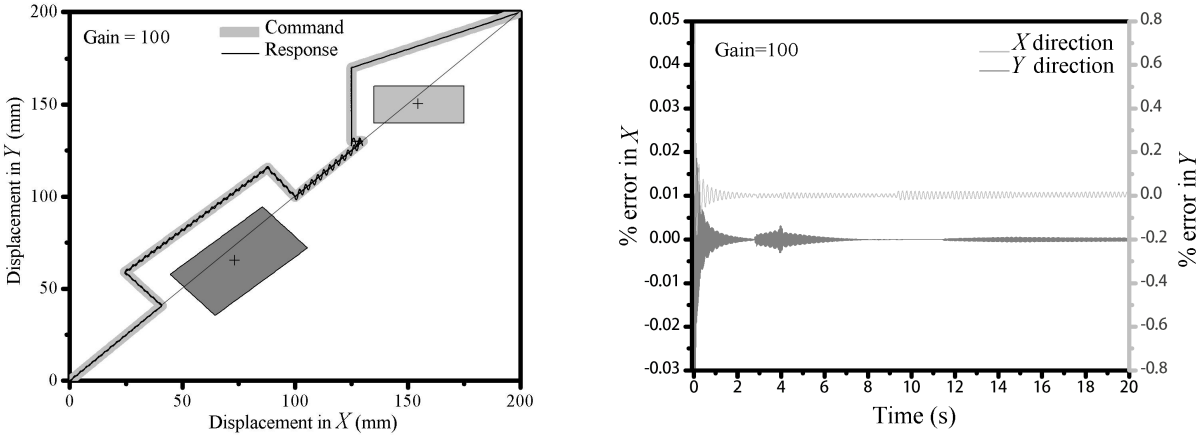


Fig. 3.34 Simulation results for two circular obstacles offset to the right and left of centre line: (a) Comparison between command and response trajectory tracking, (b) Percentage error in X and Y displacement with gain of 10, (c) Comparison between command and response trajectory tracking and (d) Percentage error in X and Y displacement with gain of 100

Figure 3.35(a) shows the Y vs X position of the robot for two rectangular obstacles (both is offset right to the shortest path but two sides of one obstacle are parallel to the centre line) with an overwhelming gain of 100. The percentage errors along X and Y directions are illustrated in Fig. 3.35(b). The maximum percentage errors for X and Y directions are 0.02 and 0.1%, respectively for the overwhelming gain of 100.



(a) (b)

Fig. 3.35 Simulation results for two rectangular obstacles both offset to the right of the centre line: (a) Comparison between command and response with gain of 100 and (b) Percentage error in *X* and *Y* displacement

3.5 Conclusions

Initially, a legged mobile robot was developed and simulated for the indoor environment having an obstacle. A hybrid algorithm including line following, wall following and tangent bug algorithm was developed to avoid the obstacles. Each algorithm was simple to design and easy to update according to the given environment. The obstacle avoidance algorithm for a legged robot was developed and simulated using *Matlab* and *Labview* software. The important characteristics of this architecture were its simplicity and flexibility. The application of such mobile robot was to carry objects from one point to other in a predefined path. The results of simulation and experimental work were compared and discussed. The errors of displacement along *X* and *Z* direction were within the acceptable limits. This algorithm was found to have adopted the shortest possible path based on the geometry of the obstacle by following its circumferential boundary.

This work was extended to construct the four-wheel mobile robot model having three ultrasonic sensors for detection and avoidance of obstacles with a fuzzy logic controller. The obstacle avoidance and navigation controller based on fuzzy logic were developed. Three input membership functions and two output membership functions were created for fuzzy controller. Then, the bond graph model of the four-wheel mobile robot was developed using BG. This model was then converted into the *Simulink* block. The

different cases were studied for the validation of the proposed control algorithm. The static and dynamic obstacles were considered in the simulation. The results show the effectiveness of the fuzzy logic controller for navigation and obstacle avoidance performances. This algorithm is useful in identifying not only the position of the mobile robot but also it is capable to determine the orientation of the robot w.r.t. the target.

At last, the overwhelming controller was used for obstacle avoidance of mobile robot. The inverse model was used to run the actual model of the robot for the desired trajectory. The feasibility of obstacle avoidance algorithm was checked for static obstacles depending upon various factors like number of obstacles, the position of obstacle's centroid and geometry of obstacle etc. The simulation of obstacle avoidance algorithm was done in *Matlab* environment. Then, the trajectory data was fed to the inverse controller of the robot along with the overwhelming controller that controls the actual system to follow the desired trajectory. The results show that the performance of the controller depends upon the value of the gain; the response of the actual system was better for higher gain values. This algorithm allows the mobile robot to follow any obstacle free path with very high accuracy in order to reach the target. The overwhelming controller may be used along with fuzzy logic controller for obstacle avoidance in the presence of external disturbances. This can be considered as future work. The *SimMechanics Simulink* model of the quadruped robot would be developed and validated for obstacle avoidance application in closed environment in Chapter 4. Then, the model reduction of four-wheel model through Eigen value separation and sensitivity methods would be developed. Thereafter, bond graph model of two wheel mobile robot (Segway) would be developed. In Chapter 5, the bond graph model of the quadruped robot would be developed to validate its motion with various gaits. Thereafter the BG model of the legged robot would be developed and simulated. The approach of fault detection and isolation through reconfiguration for the legged robot would be developed by using bond graph approach and would be validated for fault occurrence in leg actuator during turning phase. At last, the PI controller based modelling of the quadruped robot would be proposed. The validation of the controller is done under side ramp walking scenario.

Chapter 4

Dynamic Analysis for Mobile Robots

4.1 Introduction

The dynamic model is a set of equations that are solved to evaluate the dynamic characteristics of the robot during motion. It is a very important field of study for robotic engineers as it allows them to predict safety, performance and comfort characteristics of the mobile robots. There are different types of mobile robots i.e. two wheel, four wheel, legged robot etc. The dynamics plays a key role during locomotion of these robots, specially, in case of legged robots, because legged robots have only one or two contact surfaces at a time whereas wheeled mobile robots (WMR) have in general two or four contact surfaces with the ground. During obstacle avoidance application, the dynamics of the robots play significant role.

The legged mobile robots as compared to WMR have greater flexibilities in terms of agility on rough and irregular terrains. The serial kinematic structure is used to develop two legged robots with 6 degrees-of-freedom (DOF) [Kaneko *et al*., 2009; Yu *et al*., 2014; Rocchi *et al*., 2015]. Based on linkage mechanisms, some other walking machines are developed [Ceccarelli *et al*., 2015]. The energy efficient SpaceClimber-1 robot was developed for space surface investigation [Bartsch *et al*., 2012]. Now-a-days, numerous quadruped robots have been developed. MIT has developed Cheetah Robot. Cheetah and WildCat [Sangok *et al*., 2013] have been developed by Boston Dynamics Lab and these robots can replicate walking characteristics of the animals. Many researchers have worked on the several characteristics of Jansen mechanism. The additional up-down motion of Jansen linkage for generating new gait cycles for avoiding obstacles has been introduced by Komoda *et al*. [2011]. An attempt has been made for physical modelling of the quadruped robot based on Jansen mechanism using *SolidWorks* and *Matlab/SimMechanics* toolbox in this work. The motion of the quadruped leg is controlled by separated DC motor. Also, the prototype model of the robot using Jansen mechanism is developed here. The proposed quadruped robot is used for obstacle avoidance. The fuzzy logic based controller is proposed to avoid the obstacle in a closed

environment (arena). The simulation result of path traced by the robot is validated with the experimental result.

Due to the application of various new forms of control techniques and algorithms, the need of simulation does not end with the design process only. The simulation allows the algorithms to predict and hence, to control the system in a better manner with more accuracy. Most of the control algorithms have mathematical equations which can only be solved numerically and not analytically. As numerical methods are iterative algorithms, a complex system increases the simulation time drastically. Hence, a complex system is difficult to handle for controller design, optimization, and computation and storage requirements. In order to reduce high computation time and cost, a reduced model of the system without compromising on the accuracy is required. By considering the above advantages of model reduction, eigenvalue sensitivity and separation method is proposed for model reduction of bicycle vehicle and four-wheel model of mobile robot.

Now-a-days, stabilization of inverted pendulum system based two-wheel mobile robot (segway) is another area of research. The focus of this work is to develop the dynamic model and to simulate the two-wheel mobile robot using non-model based control system design. Initially, the BG model of inverted pendulum system is developed. The design of a nonlinear heuristic controller for swing up control and the proportional integral (PI) controller as a stabilization control are also proposed in this Thesis. In order to validate the bond graph simulation results, the experimental set-up of GLIP is used and the results are found in close agreement. Finally, the bond graph model of segway with PI control is developed. The simulation results for the forward and backward motion of segway are presented. For the turning of the robot, controller based on Ackermann steering mechanism is adopted to modulate the voltages of two motors. The results of turning motion of the segway are also presented.

The Chapter 3 deals with development of the obstacle avoidance algorithms and their implementation on various mobile robots. In Chapter 4, the dynamic analysis of the mobile robots is considered. Model reduction technique is one of the vital parts for dynamic analysis of the systems as computational time (related to controller delay) is reduced without significantly affecting system performance. The model order reduction

will enhance the performance of the controller of wheeled mobile robot during navigation in presence of static and dynamic obstacles, especially during high-speed manoeuvring. This work can be considered as future work.

The organization of this chapter is as follows: At first, the *SimMechanics* model of the quadruped robot based on Jansen mechanism is developed and validated for obstacle avoidance application in a closed environment. Also, prototype model of the robot is developed. The fuzzy logic based controller is developed to avoid the obstacles and to control the robot's motion. The controller's performance is analyzed using the simulation as well as experimental environment. After that, a model reduction technique is discussed. The procedure of eigenvalue separation and sensitivity methods are implemented on some application areas of vehicle dynamics. With the help of model reduction techniques, some elements in the bond graph model of vehicles may be eliminated without altering the performance of the system. At last, modelling of the two-wheel mobile robot (segway) is done and discussed. The heuristic and PI based controller is developed to control the motion of the inverted pendulum. The inverted pendulum based segway modelling is done using bond graph approach and results such as forward/backward motion, response with a stabilization controller and yaw response of segway are discussed.

4.2 Modelling of Quadruped Robot based on Jansen Mechanism

4.2.1 Physical Modelling of Quadruped Robot

A schema of the Jansen linkage is presented in Fig. 4.1. As depicted in Fig. 4.1, the Jansen linkage consists of 8 links where Link 1 is the driving element. A pair of two links is connected using revolute joint. All the 8 links are connected as shown in CAD model (Fig. 4.1). The entire mechanism is connected to the fixed frame that acts as chassis of the robot.

The input motion is given to the Link 1 and the path traced by the various joints is mentioned in Fig. 4.2(a). Figure 4.2(a) depicts the motion analysis of the CAD model while Link 1 is rotating. CAD software is used to develop the Jansen linkage mechanism and motion analysis of the model. Figure 4.2(b) shows the prototype model of the Jansen

linkage and the path traced by the various joints. The path traced by the prototype model is quite in similar fashion as path traced by CAD model. Similarly, other legs of the robot are developed. Table 4.1 shows the link lengths of the Jansen mechanism. Figure 4.3 depicts the CAD model of the quadruped robot based on the Jansen mechanism.

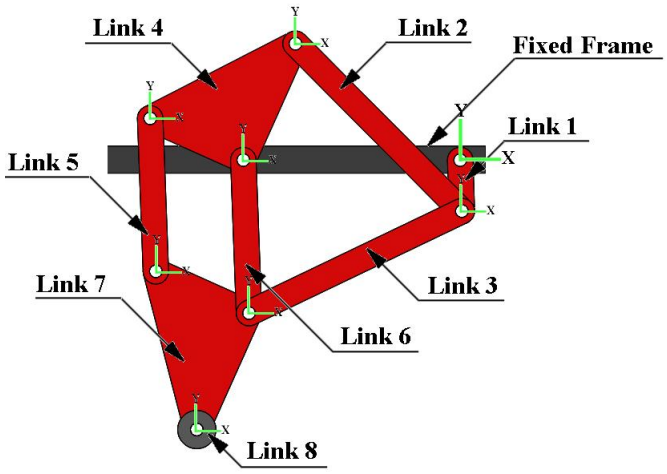
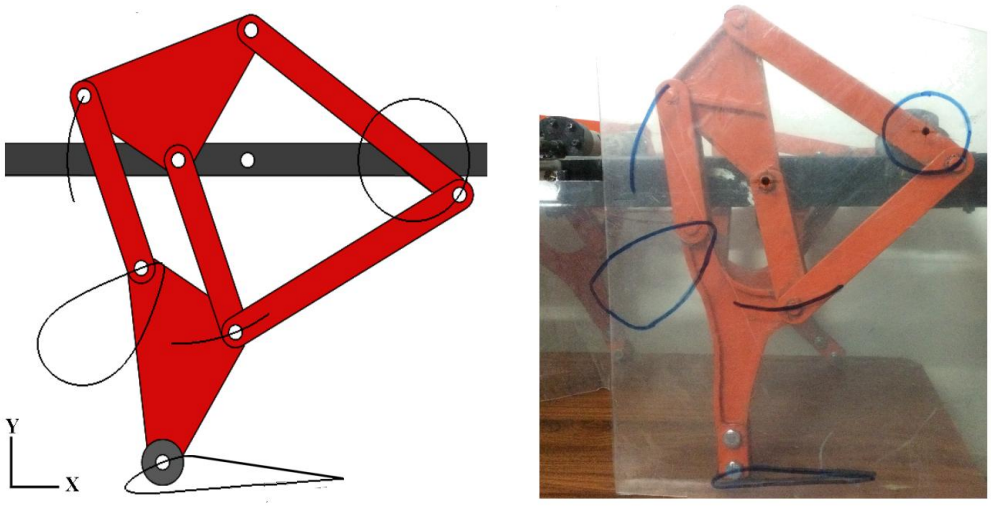


Fig. 4.1 CAD model of Jansen linkage

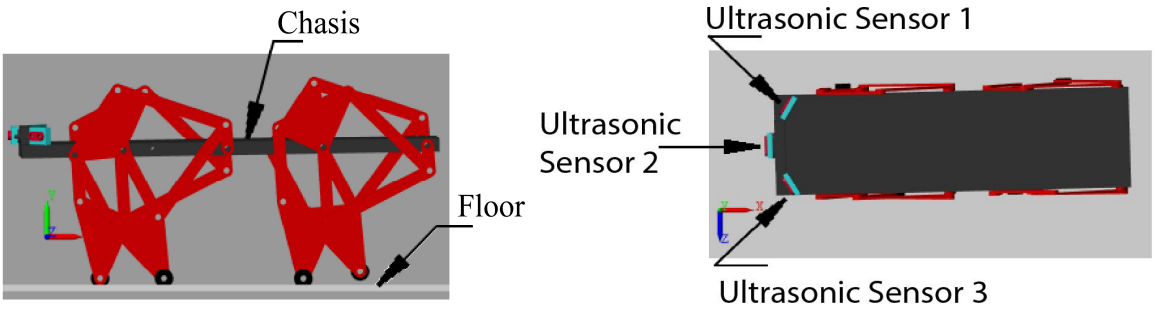
Table 4.1 Link lengths used in Jansen mechanism

Links	Length (m)
Link 1 (crank)	0.05
Link 2	0.21
Link 3	0.18
Link 4 (tertiary link)	[0.09 0.11 0.13]
Link 5	0.12
Link 6	0.12
Link7 (Tertiary link)	[0.13 0.11 0.09]
Link 8 (Stationary)	

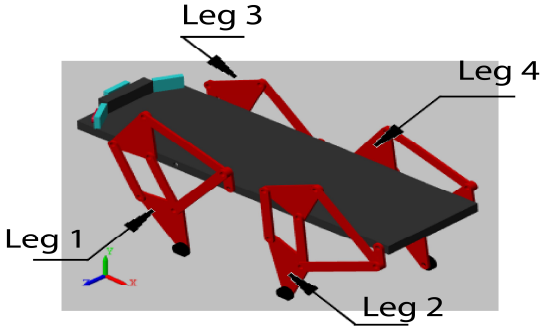


(a) (b)

Fig. 4.2 Path traced by various joints of Jansen mechanism: (a) CAD model and (b) Prototype model



(a) (b)



(c)

Fig. 4.3 CAD model of the quadruped robot

The CAD model is imported to the *SimMechanics* toolbox. The *Simulink* model of a single leg is developed. Figure 4.4 shows the detailed *Simulink* model of the single leg system. The solver configuration, world frame and mechanism configuration blocks are

used to build the robot model under *Simscape* environment. The solver parameters which are required by the model are taken care by the solver configuration block. The global frame is same as the inertial frame. The axes of the frame are orthogonal as per right-hand rule. This block provides mechanical and simulation parameters to a mechanism, i.e., a self-contained group of interconnected *Simscape*™ Multibody™ blocks. In this model, the parameters for the mechanism configuration is [0, 0, -0.98] m/s^2. The signal builder is used to feed the rotational angle of the shaft of the DC motor. Further, the Link 1 is connected to the motor shaft. The revolute 1 joint is used to connect the Link 2 and 3. The triangular shaped Link 4 has three revolute joints (revolute 2, 3 and 4). Link 5 and 6 are the connector between Link 4 and Link 7 by means of joints 5 and 6. The Link 8 (stationary) is joined to Link 7. The Link 8 acts as foot which is movable on the ground surface.

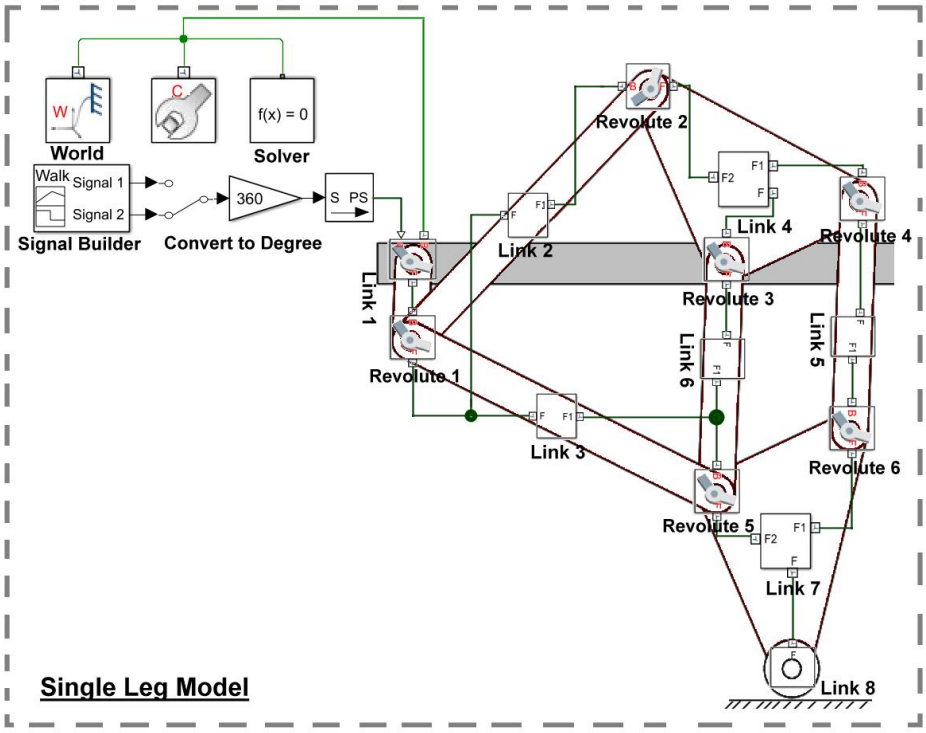
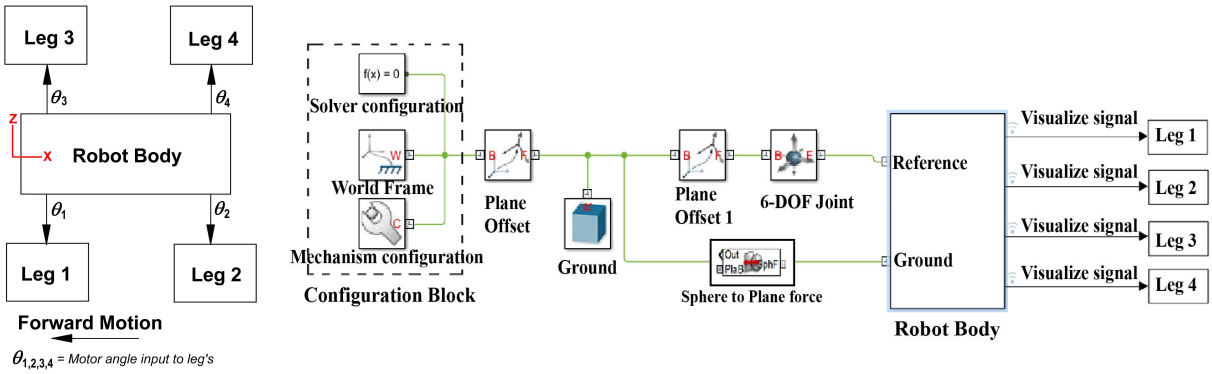


Fig. 4.4 *Simulink* model of the single leg

The complete quadruped robot model is illustrated in Fig. 4.5. The schematic arrangement of the legs with the robot body is shown in Fig. 4.5(a). The forward motion

of the robot is considered along *X*-direction and lateral movement along *Z*-direction. The complete *Simulink* quadruped model is illustrated in Fig. 4.5(b). The ground surface is connected to the solver configuration, world frame and mechanism configuration blocks. 6-DOF block is connected to the robot model. This joint enables the robot to move in 3-D space.

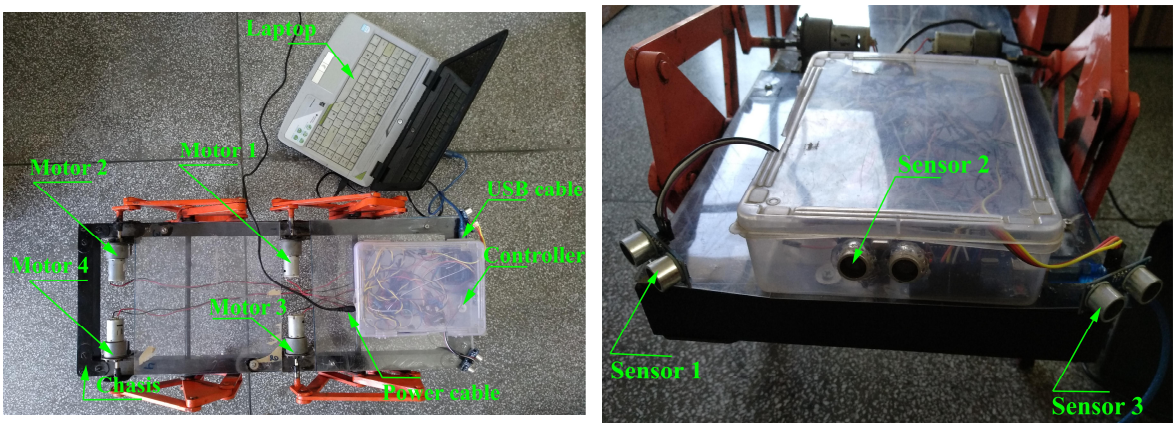


(a) (b)

Fig. 4.5 (a) Schematic diagram of complete robot and (b) *Simulink* model

4.2.2 Prototype Model of Quadruped Robot

The development of prototype of the quadruped robot is discussed in this section. Figure 4.6 shows the prototype of the quadruped. Aluminium is used to construct the legs of the robot. The square section channel is used to build the chassis of the robot. Four-gear DC Johnson motor of 60 rpm is used to drive the leg mechanism. The Arduino mega controller is used to control the motion of the robot. The motor driver L298 is used to control the motors. Three ultrasonic HC-SR04 sensors are mounted on the front top of the robot and are used to detect the obstacles and to map the surroundings. A power supply of 12 V, 3 A is used to drive the motors. The laptop is connected to the controller via USB cables.



(a) (b)

Fig. 4.6 Prototype of the quadruped: (a) Top view and (b) Front view

4.2.3 Obstacle Avoidance Control

The proposed quadruped robot model is used for application of the obstacle avoidance. The fuzzy logic controller for obstacle detection and avoidance of obstacles is proposed in this section.

4.2.3.1 Block Diagram

The block diagram of the fuzzy obstacle avoidance controller is illustrated in Fig. 4.7. The inputs to the controller are the distances of the robot from the target and the obstacles. Based on the rules, the controller decides the angular motion of each motor shaft. The predefined target (*X*, *Y*) position is known to the system.

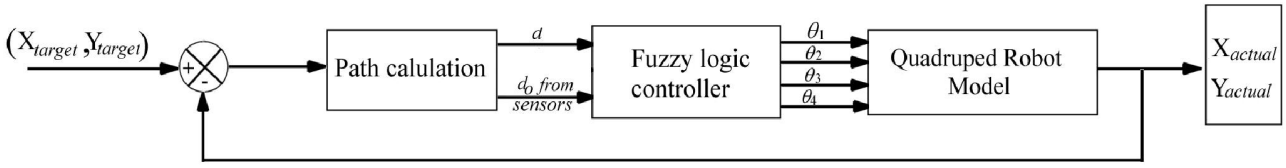
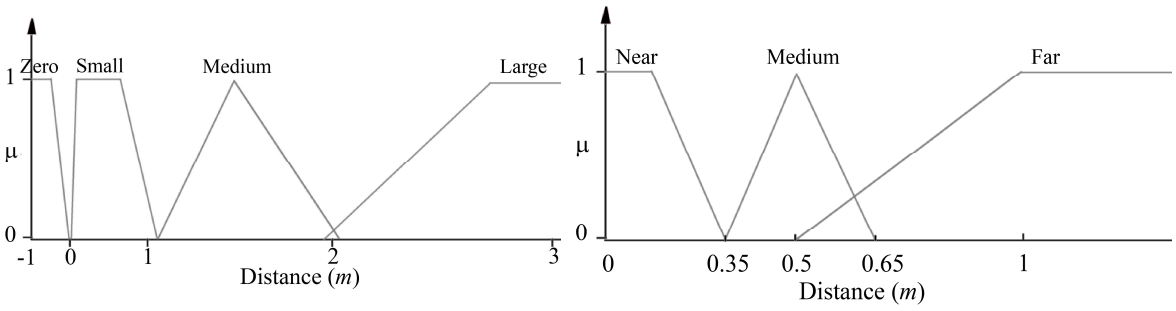


Fig. 4.7 Block diagram for navigation and obstacle avoidance

4.2.3.2 Fuzzy Membership Functions

Two input and one output membership functions are developed for the fuzzy controller. The input membership functions are the distance of the robot from the target and the distance of the robot from the obstacles. The ultrasonic sensors are used to compute the distance of the robot from the obstacles. The output membership function is the angular movement of the motor shaft. Figure 4.8 depicts the input membership functions. A set of 23 rules are developed to control the performance of the controller. The rules are mentioned in Table 4.2.



(a) (b)

Fig. 4.8 Fuzzy membership functions: Distance from (a) Target and (b) Obstacle

4.2.3.3 Obstacle Avoidance in Closed boundary environment

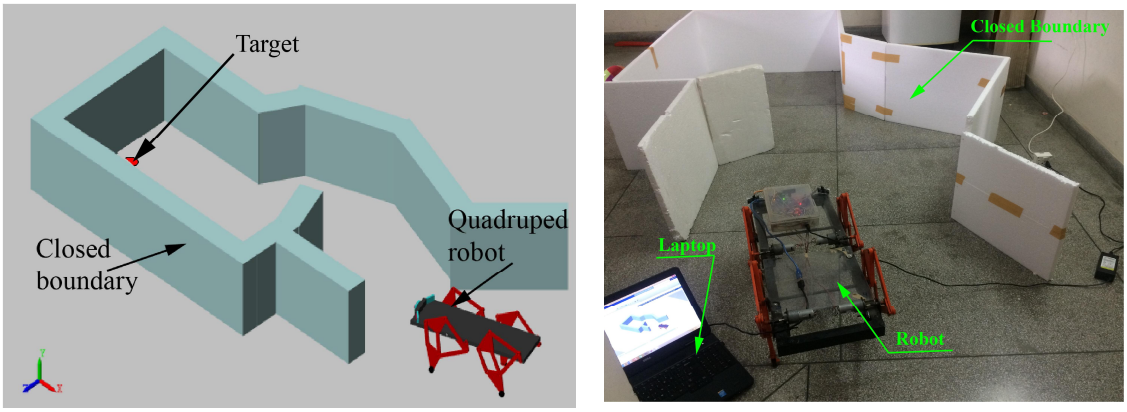
The closed boundary environment is considered for validation of proposed fuzzy based obstacle avoidance controller of quadruped robot. Figure 4.9 shows the setup for obstacle avoidance application. Figure 4.9(a) shows the simulation-environment model having closed boundary in *SimMechanics Simulink* software and robot. Figure 4.9(b) shows the prototype model of the setup for experimental work. In this scenario, the robot has to enter into a closed boundary and to arrive at the goal by avoiding the walls of the boundary. The fuzzy based controller for the robot is used for avoiding walls of the boundary.

Table 4.2 Fuzzy rules for obstacle avoidance

Input				Output
Distance from target	Distance from sensors to obstacle			Angular motion of motor shaft
	Sensor 1	Sensor 2	Sensor 3	
Large	Far	Far	Far	Large
Large	Far	Medium	Medium	Medium
Large	Medium	Medium	Medium	Small
Large	Medium	Far	Medium	Small
Large	Medium	Medium	Far	Small
Large	Far	Medium	Medium	Small
Large	Medium	Medium	Medium	Medium
Large	Medium	Far	Medium	Small
Large	Medium	Medium	Far	Medium
Medium	Far	Far	Far	Medium
Medium	Far	Medium	Far	Small
Medium	Medium	Medium	Medium	Small
Medium	Far	Far	Far	Small
Medium	Far	Far	Far	Large
Medium	Near	Near	Near	Large
Medium	Near	Near	Near	Very small
Medium	Near	Near	Near	Large
Small	Far	Far	Far	Large
Small	Far	Far	Far	Small
Small	Medium	Medium	Medium	Very small
Small	Near	Near	Near	Very small
Small	Far	Medium	Medium	Very small
Small	Far	Medium	Far	Small

The simulation parameters are mentioned in Table 4.3. The wheels (rollers) are connected with the end points of the legs as shown in Fig. 4.2. But, these rollers are fixed with leg's end points. As these rollers have no rolling motion, the values of coefficient friction are comparatively higher due to their sliding motions along the ground surfaces.

Figure 4.10(a–c) depicts the results of the robot displacement along *X, Y* and *Z* directions. As depicted from Fig. 4.10(a), it is seen that the robot travels the distance of 1.6 m approximately along negative *X*-direction. Negative *X*-direction is considered as the forward direction of the robot. The vertical movement of the robot body is illustrated in Fig. 4.10(b). Figure 4.10(b) depicts that vertical movement of the robot is quite less for the simulation time of 11s. During manoeuvring of the robot body, vertical movement is significant due to the changes in velocity of the robot legs. The lateral movement of the robot is presented in Fig. 4.10(c). Figure 4.10(d) shows the comparisons of the simulation and experimental result of the path traced by the robot during obstacle avoidance in a closed boundary environment. Figure 4.10(e) depicts that robot follow the yaw angle from -0.3 to 0.3 rad during whole obstacle avoidance process.

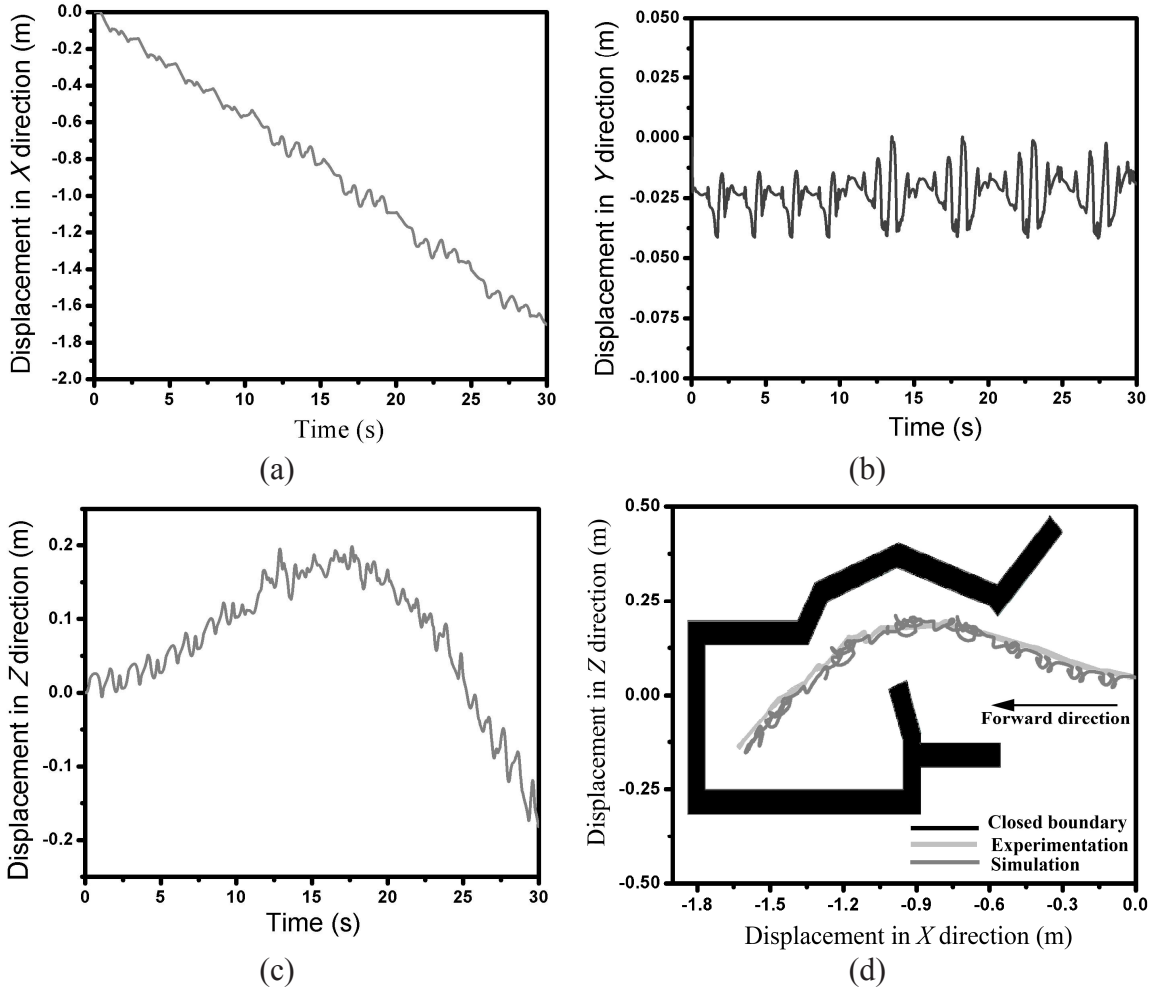


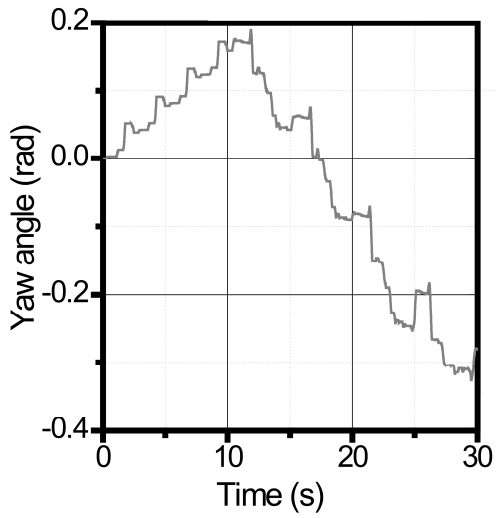
(a) (b)

Fig. 4.9 Setup for obstacle avoidance application

Table 4.3 Simulation parameters

Parameter	Description	Value
m_r	Mass of robot	4.2 kg
K_c	Contact stiffness	10^5 N/m
R_c	Contact damping	1000 Ns/m
m_k	Co-efficient of kinetic friction	0.92
m_s	Co-efficient of static friction	0.89





(e)

Fig. 4.10 Displacement in (a) *X* direction, (b) *Y* direction, (c) *Z* direction, (d) Path travelled by the robot in a closed boundary and (e) Yaw angle

4.3 Physical Model Reduction Technique

In the last two decades, several researchers have worked on the model reduction methods. Many techniques are identified for model reduction by Orbak *et al*. [2004]. These methods are not applied for modelling and reducing the system in physical domain because of no information of internal structure of system. Orbak *et al*. [2003] proposed an approach to find the reduced model by eliminating several physical elements from the full model. All elements of the system do not contribute to the system behaviour. The method used to reduce the system model by removing the elements which have less power flow from the full model is proposed by Louca *et al*. [1997]. The singular perturbation based physical reduction of system is discussed by Sueur & Dauphin-Tanguy [1991]. Some other methods like eigenvalue separation method and sensitivity method are discussed in [Orbak, 2010].

The main advantages for using the reduced order models are:

- The physical dynamics of the original system becomes simple to understand using lower order model as these focus on the most important component of the system
- The time required for simulation of reduced system is lesser than that of actual system

- The controller is more efficient in numerical terms
- The computational complexity is reduced and it is more accurate than the actual system
- As the control laws become simpler, the hardware and software requirements for the controller are reduced. Therefore, the controller becomes cost effective with the reduced model

A mathematical model describing the dynamic characteristics of the vehicle needs to be solved for predicting the future position and orientation of vehicle relative to the changing environment. By reducing the consumed time to solve the complex model, the cycle time for the overall process can be decreased. The work done in this section, attempts to reduce the computational time by mathematically finding the redundant elements, which do not add to the major dynamics of the system but increase the calculation time by increasing the system complexity. Here, in this work, eigenvalue separation and eigenvalue sensitivity methods have been applied to the different bond graph models of bicycle vehicle with suspension system, antilock braking system, steering system and four-wheel vehicle model. The reduced system behaviour is similar as that of actual bond graph model. This method is applicable to linear systems as well as for the non-linear systems after certain modifications [Orbak, 2010].

4.3.1 Eigenvalue Separation Method

The eigenvalues play a crucial role in the system dynamics. The eigenvalues are actually the roots of the characteristic equation of dynamic system; hence, eigenvalues contain all the information about the behaviour of the system. These eigenvalues of the system can be dissociated into two or more sets giving different subsystems and having different behaviour. The eigenvalue separation method is based on the comparison of energy exchange rates between two different elements. This rate of energy exchange is given by the loop gain of the causal loop containing these elements. The elements which have very high exchange rates contribute to the faster subsystem.

4.3.1.1 Procedure

A method for recognizing high and low damped systems was proposed by Rosenberg & Zhou [1988]. For identification, calculation of the local damping ratios with loop gains for I and C is required. The local damping ratios are specified as $G_{RC}/\sqrt{G_{IC}}$ and $G_{RI}/\sqrt{G_{IC}}$ for all compliance and inertia elements, respectively for each causally connected I-C pair in the system where R is causally coupled to either inertia element or compliance element or both. In formula, G_{IC} is the I-C loop gain, G_{RC} is the addition of the loop gains of R-C pairs and G_{IR} represents the addition of the loop gains of I-R pairs. This gives the damping ratios of the system.

- **Heavily Damped Sub-system**

- All the compliance elements are replaced by flow sources of zero values. The remaining resistive and inertia pairs are identified. These R-I elements and the involved junctions are denoted as part of a set called H. Step 1 identifies the R and I elements that are responsible for the heavily damped modes, given that they affect the dynamics even if all of the capacitances are disabled.
- The C elements are restored. If $\sqrt{G_{IC} >> G_{IR}}$, compliance elements are replaced by flow sources with zero called as part of the subsystem H.
- Inertia elements are identified and are called as part of H which is dependent on the causalities forced by the above sources. The inertia elements are identified in 2nd and 3rd steps which are concerned with the heavily damped modes by the power transmission through inertia and compliance loops
- A similar procedure for steps 1–3 is repeated by exchanging I with C and flow source with effort source and vice versa. Step 4 repeats the same method for resistive and compliance elements.
- The resistances concerned with heavily damped local loops are identified. These resistive elements and the concerned inertia and compliance pairs denoted as part of H system. Step 5 includes the over damped subsystems.
- The compliance elements are not concerned with above step, but are causally linked directly to the above inertia elements, are identified. If $\sqrt{G_{IC} >> G_{IR}}$ the compliance elements are replaced by zero flow sources and these flow sources are called as part of the H subsystem.
- The step for inertia elements is repeated. In Steps 6 and 7, the inertia or compliance elements which affect the heavily damped modes are spotted by power transmission during inertia-compliance loops.
- Elements not present in H are removed. This gives heavily damped system.

- **Lowly Damped Sub-system**
  - The inertia-compliance pairs concerned in lightly damped local loops are identified and are denoted as part of a system named L.
  - Those Resistive elements which are not concerned in step 1 but are causally related directly to the above inertia or compliance element are identified. If $\sqrt{G_{IC} >> G_{IR}}$ the resistive elements are replaced by zero flow sources and resistive elements are replaced by zero effort sources and denote as a part of L system.
  - The C and I elements that depend because of the casualty function imposed are identified and denoted in L.
  - Elements not present in L are removed. This gives lowly damped system.

In this method, step 1 detect the lightly damped sub components. Steps 2 and 3 spot the energy storage elements involved in the lightly damped modes considering the power transmission during the other I-R loops.

4.3.1.2 Generalized Procedure

The local damping ratios of the system are found. If all the local damping rations are very small, the system can be treated as I-C system and decoupling can be done to find the high and low frequency subsystems. Similarly, if all the local damping ratios are very high, the system can be divided into fast and slow dynamic components. But, if in a system some of the local damping ratios are high, while the others are very low, the system must first be divided into heavily damped subsystem and lowly damped subsystem. Heavily damped subsystems can be decoupled into fast and slow subsystems while the lowly damped subsystem can be further subdivided into high and low frequency dynamics. The detail generalized procedure for eigenvalue separation method is shown in Fig. 4.11.

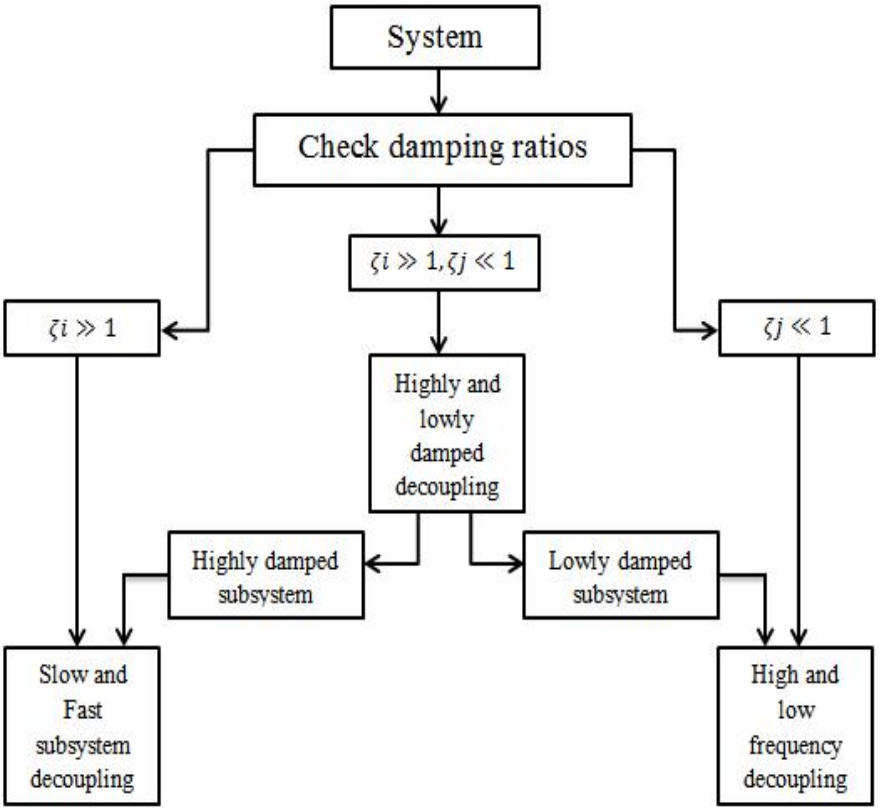


Fig. 4.11 Generalized procedure for eigenvalue separation method

4.3.1.3 Application

- **Anti-lock Braking System Model**

The anti-lock braking system (ABS) is used in automobiles for an improved performance and better control over the vehicle for different road scenarios. The objective of ABS is to retain a required slip ratio. The mechanical equivalent of ABS model shown in Fig. 4.12(a) is used for simulation. The slip ratio can be found out. Slip ratio during acceleration and braking is given by

$$\text{Slip ratio (during braking)} = \frac{\dot{x}_w - \dot{\theta}_w r}{\dot{x}_w} \tag{4.1}$$

$$\text{Slip ratio (during acceleration)} = \frac{\dot{\theta}_w r - \dot{x}_w}{\dot{\theta}_w r} \tag{4.2}$$

The operating logic is summarized in Fig. 4.12(b). The BG model of ABS system is presented in Fig. 4.12(c). The torque obtained from the motor is shown as Se-element. The rod magnifies the motion of the shaft and hence is shown by a transformer element. The losses at the end of the rod are shown by R-element. The stiffness of the cable connected at the end of the rod is modelled by C-element. The other end of the cable is attached to a spring shown in the bond graph with another C-element. As this spring is fixed at one end, a flow source of zero value is connected to it. The flow at the end of the cable is transferred to the wheel through a rod. The inertia element (I) represents the mass moment of inertia of the wheel and the flow transformation due to rod is expressed with a transformer element. Slow subsystem is illustrated in Fig. 4.12(d).

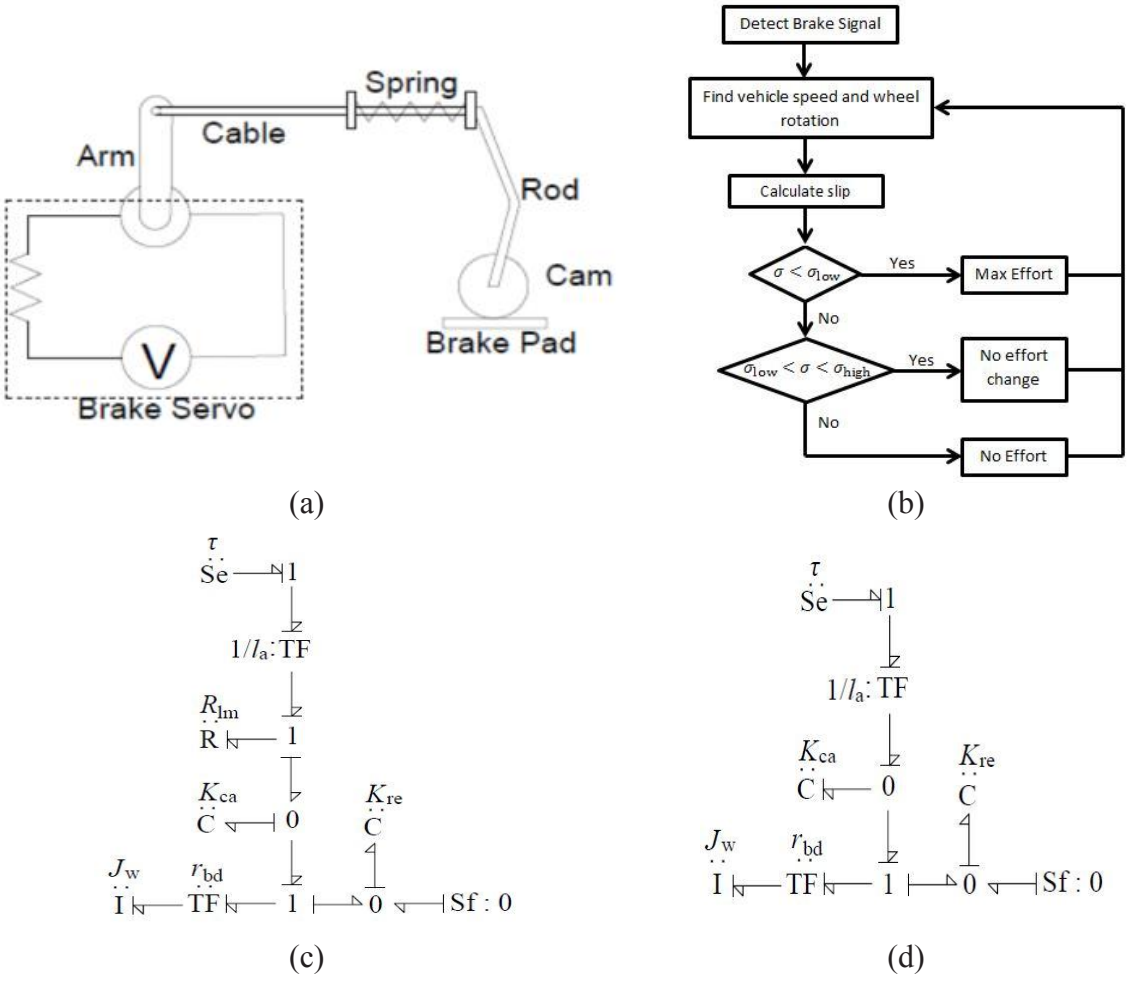


Fig. 4.12 (a) Schematic diagram, (b) Operating logic [Lalonde *et al*., 1992], (c) Bond graph model and (d) Reduced bond graph model of ABS

The physical interpretation is that the exchange of energy between R_{lm} and K_{ca} is too high and dissipates before any observable effect, hence slow system can be used as approximation for all times. It is to be noted that for the slow subsystem, the C-element has a differential causality. Hence, a boundary condition is provided for solving the system i.e., rate of change of spring stiffness is zero. The ABS model is simulated to operate as per the parameters value shown in Table 4.4. The full system and reduced system were compared by observing the torque supplied to the wheel. The difference between results of the models was plotted and is shown in Fig. 4.13(a). The reduced system gives very close results as compared to the full system.

Table 4.4 The values of the simulation parameters

Parameters	Description	Values
T	Torque	100 Nm
I_w	Polar moment of inertia	15 kg m^2
l_a	Length of arm	1 m
R_{1m}	Motor damping	0.04 Ns/m
K_{ca}	Stiffness of cable	10^4 N/m
K_{re}	Stiffness of return	10^6 N/m
r_{bd}	Effective radius of brake drum	0.15 m

Figure 4.13(a) shows the comparison of braking torque achieved for the full and reduced ABS models. The results are very close and can be considered within the error limits of computation. But from the Fig. 4.13(c), it is noted that the percentage error of the system follows a cyclic variation and reaches very high values, but such an error is not observed when comparing the torque directly. The error percentage increases at times when the torque on the wheels is close to zero. So, a percentage error compared to such a small base term may give a high percentage of error, even if the error is within acceptable limits in absolute terms. This can be observed from Fig. 4.13(b), which shows the absolute error in torque observed at the wheel between full and reduced model.

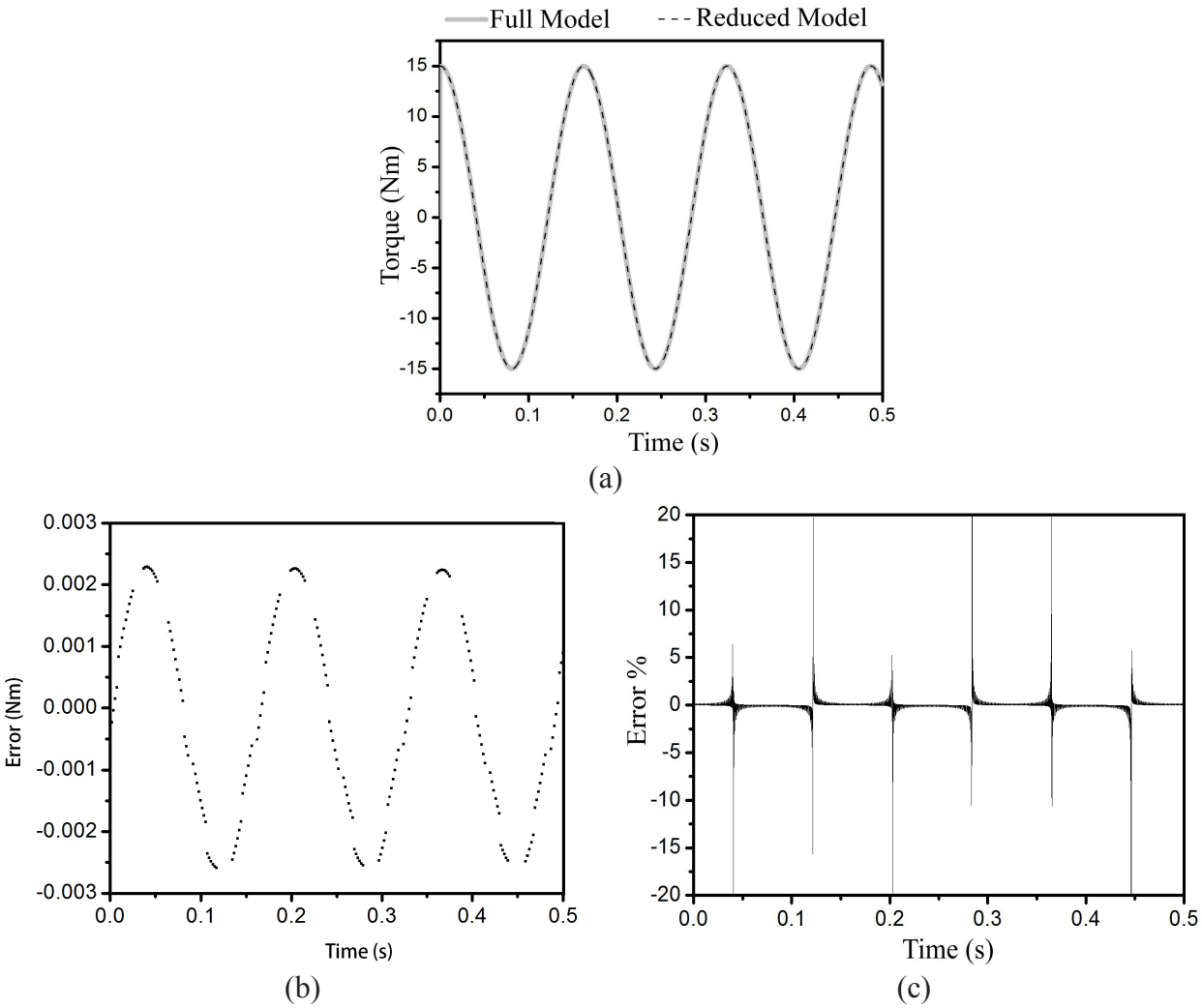


Fig. 4.13 (a) Comparison of torque supplied at wheel, (b) Absolute error and (c) Percentage error between torques for full and reduced model

- **Bicycle vehicle model without and with steering**

Bicycle model, is an approximation of a four-wheel vehicle under the assumption that there is no lateral load transfer. The bicycle model used here, lies in the horizontal plane and does not consider the pitch, roll and heave motions of the vehicle. The dynamics due to suspensions are also neglected. These assumptions do not allow any kind of lateral load transfer in the vehicle during the manoeuvring. Figure 4.14(a) depicts the schematic diagram of the model and Fig. 4.14(b) presents the word bond graph of the model. CTF represents the necessary coordinate transformation. Kinematic equations used for creating the bond graph are as follows:

$$v_{\text{nfr}} = \left(\dot{y} + \dot{\theta}_{\text{cz}} a\right)\cos d - \dot{x}\sin d \tag{4.3}$$

$$v_{\text{tfr}} = \left(\dot{y} + \dot{\theta}_{\text{cz}} a\right)\sin d + \dot{x}\cos d \tag{4.4}$$

$$v_{\text{nrr}} = \left(\dot{y} - \dot{\theta}_{\text{cz}} b\right) \tag{4.5}$$

$$v_{\text{trr}} = \dot{x} \tag{4.6}$$

As the model is in the horizontal plane, one translation and two rotational constraints are added. Hence, the Newton-Euler equations are reduced to

$$m_v \ddot{x} = m_v \theta_{cz} \dot{y} + \sum F_X \tag{4.7}$$

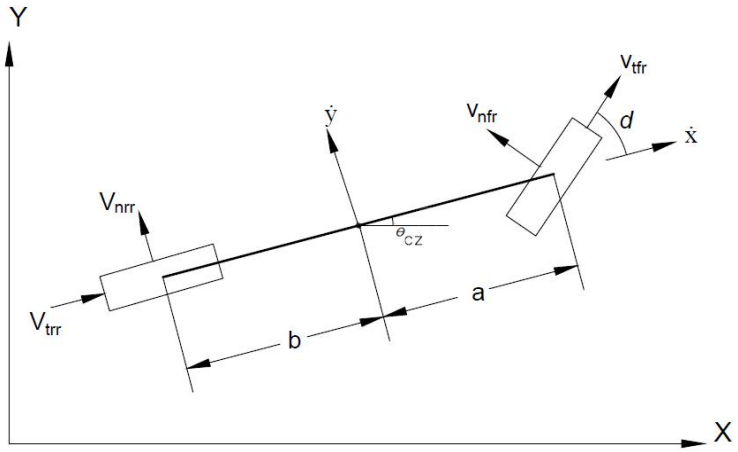
$$m_v \ddot{y} = -m_v \theta_{cz} \dot{x} + \sum F_Y \tag{4.8}$$

Figure 4.14(c) shows the bond graph model of the system. Eigenvalue separation technique cannot be applied for non-linear models; therefore, a reduced model can be obtained only for case in which the system behavior is linear i.e., when there is no steering. The bond graph of the reduced model without steering is shown in Fig. 4.14(d). After a straight line motion during 10s, the vehicle was steered for two seconds with a steering angle of 0.1 rad. The magnitudes of loop gains were compared during steering. The duration of steering was divided into various periods during which various elements were removed depending on the relative magnitudes of the loop gains. It was assumed that the loop gain could be neglected if the relative magnitude was less than 0.15. Table 4.5 shows the state of the various elements during the whole trajectory. Figure 4.15(a) shows the variation in the relative magnitudes of various causal loops for time of 10–12 s, for which the vehicle was steered. Brake force was also applied on the wheels during steering to check reduce wheel slip. The full and reduced systems were simulated using parameters in Table 4.6. for steering motion. Results from the full and the reduced model were compared on the basis of the longitudinal vehicle velocity. Figure 4.15(b–d) show the variation in longitudinal velocity in full and reduced system model, the absolute error and percentage error in the vehicle velocity for the initial straight line motion, respectively. The maximum percentage error is 0.0021%. This error is within the acceptable limit. Figure 4.15(e–g) show the comparison of full and reduced model during steering with braking. The reduced model has a maximum error of 25% during stopping at the end. This can be attributed to the fact that longitudinal velocity of vehicle was

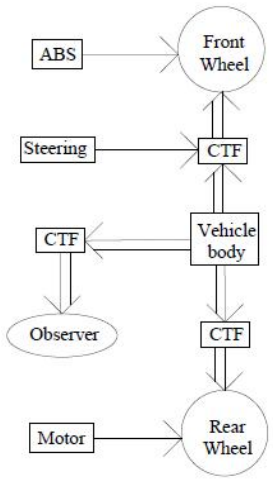
nearly low. As the absolute error is within limits, it can be neglected.

Table 4.5 Role of various elements during trajectory

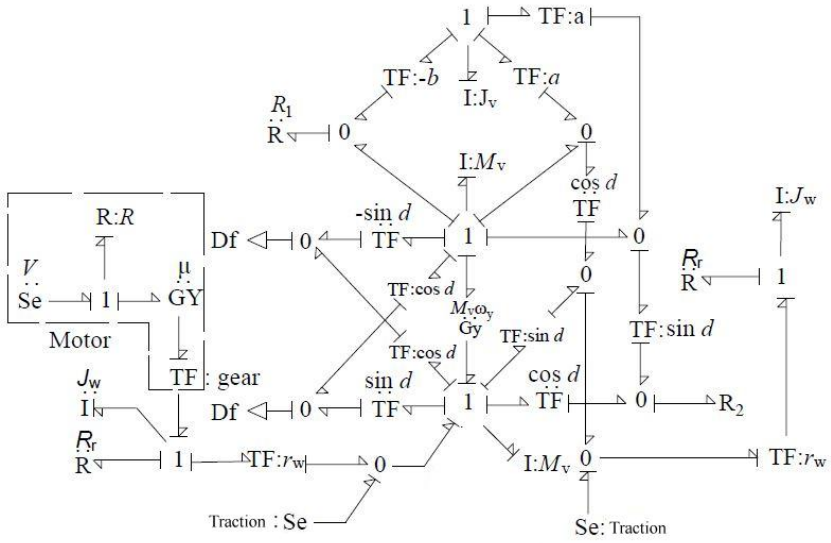
Element	Time				
	0–10 s	10–10.4 s	10.4–10.7 s	10.7–10.9 s	10.9–12 s
R_1	Removed	Removed	Used	Used	Used
R_2	Removed	Removed	Removed	Used	Used
R_r	Used	Used	Used	Used	Removed



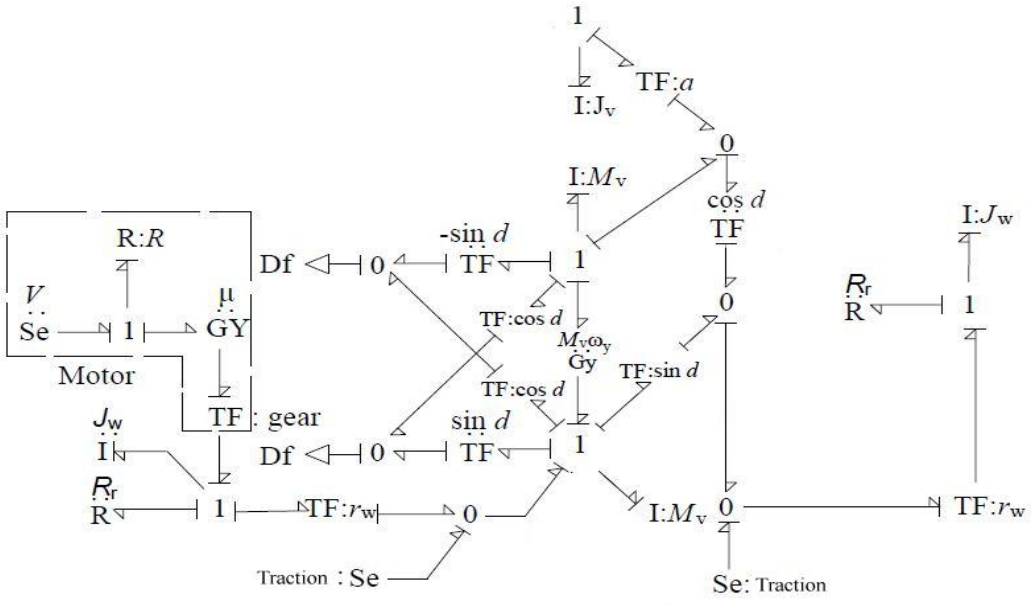
(a)



(b)

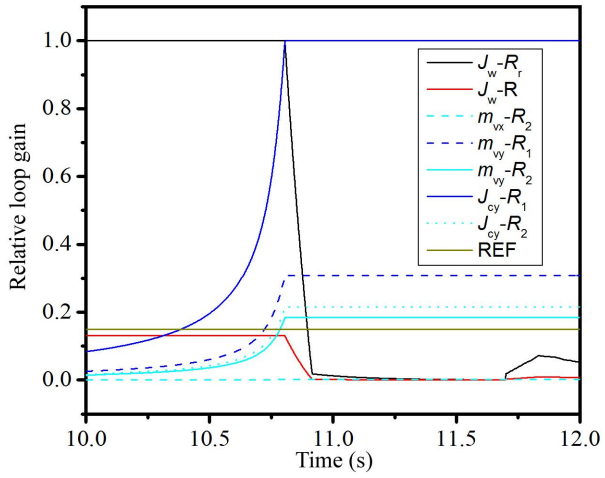


(c)

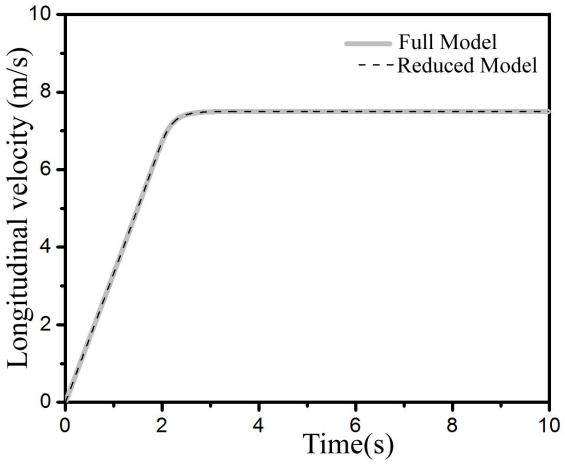


(d)

Fig. 4.14 (a) Word bond graph, (b) Schematic diagram with steering [Xihang, 1987], (c) Bond graph model and (d) Reduced bond graph of bicycle model without steering



(a)



(b)

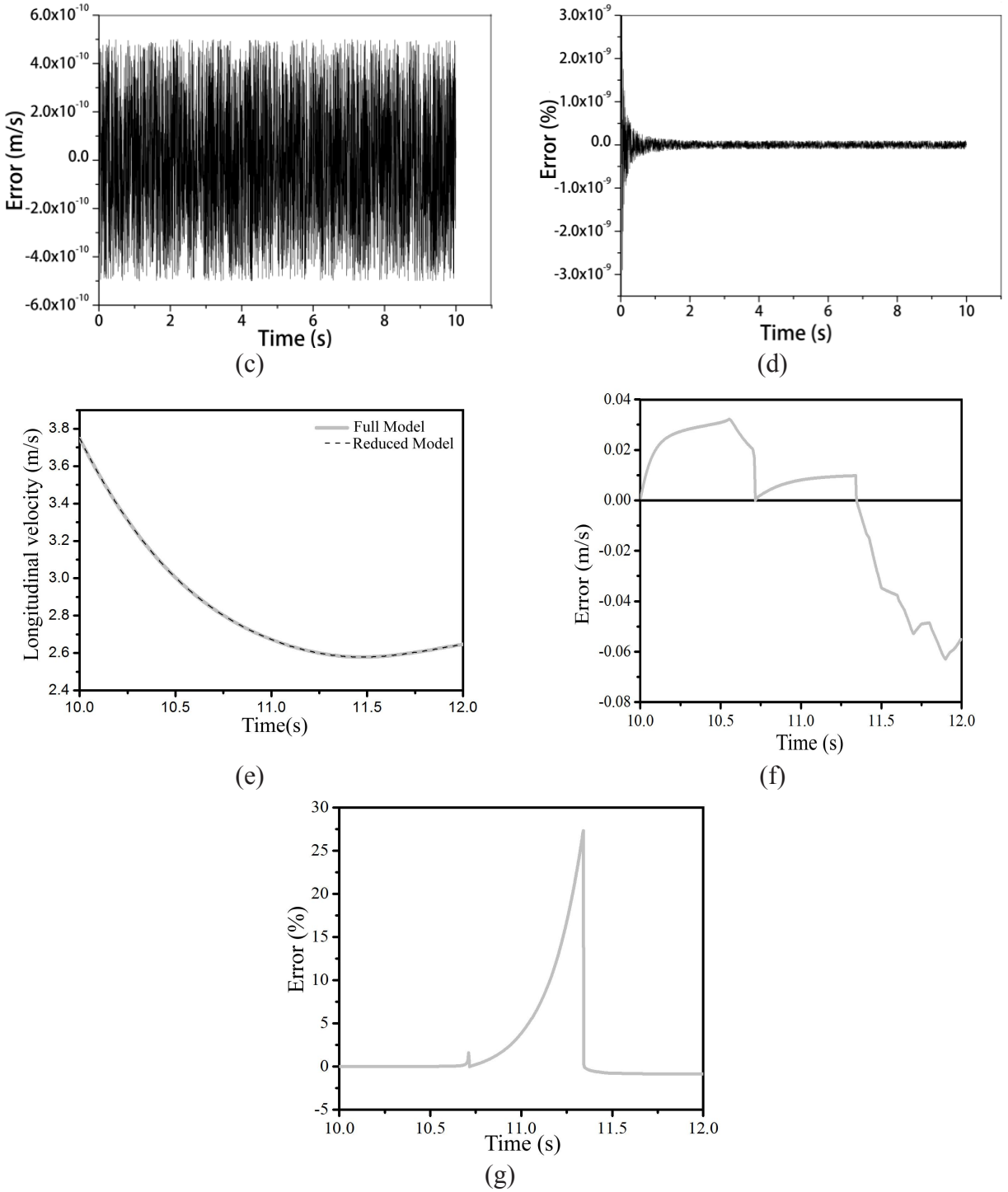


Fig. 4.15 (a) Variation of relative magnitudes of causal loops during steering, (b) Comparison of longitudinal velocity, (c) Absolute error, (d) Percentage error in full and reduced bicycle model without steering, (e) Comparison of longitudinal velocity, (f) Absolute error and (g) Percentage error in full and reduced bicycle model during steering

Table 4.6 Parameter values of bicycle model with steering and suspension

Vehicle body			
M_v	1600 kg	J_{cv}	1110 kg m^2
A	0.9 m	B	1.5 m
Wheel			
M_w	15 kg	J_{wy}	0.2 kgm^2
r_w	0.3 m		
Motor			
R	0.1 Ω	μ	2 Nm/A
V	175 V		
Steering			
Δ	0.1 rad		
Tyre			
R_t	200 N/m	K_t	305 KN/m
Suspension			
K_{sx}	107 N/m	R_{sx}	2000 N/m
K_{sz}	80000 N/m	R_{sz}	500 N/m

4.3.2 Eigenvalue Sensitivity Method

According to the eigenvalue sensitivity method, first the eigenvalues of the system are found out and then the sensitivity of eigenvalues for all the energy storing and energy dissipating elements is calculated. This provides the relative importance of every element for any given eigenvalue. As I, C and R elements have different scales of magnitude; they must be weighed in separate groups. Knowing the exact location of the eigenvalues gives a better control over the process; this is a major advantage over the eigenvalue separation technique.

4.3.2.1 Procedure

Procedure as discussed in [Rosenberg & Zhou, 1988] is as follows:

Find the matrix S and L such that $Z = S_x$

Find the connectivity matrices

Find matrix A such that $A = \left[J_{ss} + J_{SL}L(1 - J_{LL})^{-1}J_{LS}\right]S$

Calculate the eigenvalues of state matrix A.

Calculate U and V i.e., right and left eigenvector matrices for state matrix A.

For each eigenvalue, find their sensitivity with all elements.

Find the absolute value $\frac{\partial \lambda_i}{\partial \lambda_j}$ and $\frac{\partial \lambda_i}{\partial l_j}$,Then arrange them in the form of matrices

4.3.2.2 Applications

- **Bicycle Model with Suspension**

Bicycle model is an approximate model which is assumed to be symmetric about the longitudinal mid-plane. The model considers three degrees-of-freedom (DOF), with 2 translational DOF and 1 rotational DOF. The vehicle body, the front wheel and the rear wheel have three DOF while the front and rear suspensions have only 2 DOF i.e., 2 translational DOF. The vehicle is a rear wheel drive. The schematic diagram of the model and its BG model are illustrated in Fig. 4.16(a, b), respectively. The relation between velocities at centre and at point above wheel is given by

$$\dot{x}_p = \dot{x}_c + \alpha\dot{\theta}_c \tag{4.9}$$

$$\dot{z}_p = \dot{z}_c + \beta\dot{\theta}_c \tag{4.10}$$

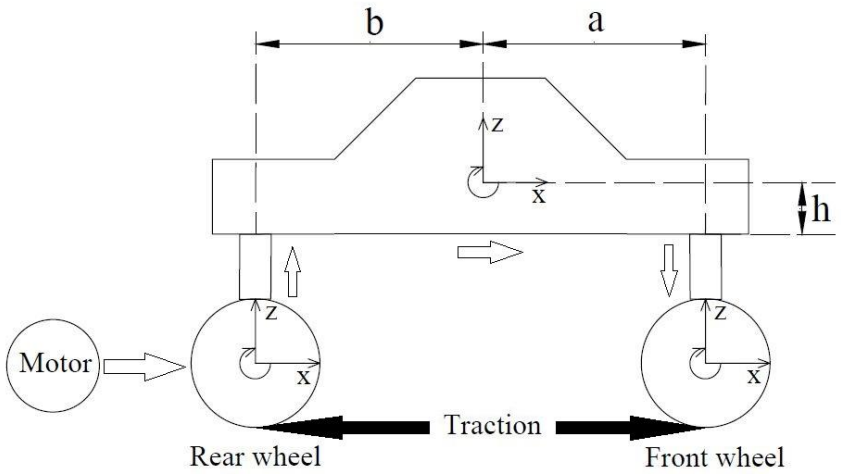
Using the procedure already discussed, the various coefficient matrices and connectivity matrices were calculated.

$$Ef_I = \begin{bmatrix} 0.0005 & 1 & 0.0005 & 0.0018 & 0.0263 & 0 & 0.7091 & 0 & 0 \\ 0.0005 & 1 & 0.0005 & 0.0018 & 0.0263 & 0 & 0.7091 & 0 & 0 \\ 0 & 0.7091 & 0 & 0.2525 & 0.0014 & 0 & 1 & 0 & 0 \\ 0 & 0.7091 & 0 & 0.2525 & 0.0014 & 0 & 1 & 0 & 0 \\ 0.0001 & 0 & 0.0001 & 1 & 0.0063 & 0 & 0 & 0 & 0 \\ 0.0001 & 0 & 0.0001 & 1 & 0.0063 & 0 & 0 & 0 & 0 \\ 0.0179 & 0 & 0.0179 & 0.0719 & 1 & 0 & 0 & 0 & 0 \\ 0.0179 & 0 & 0.0179 & 0.0719 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

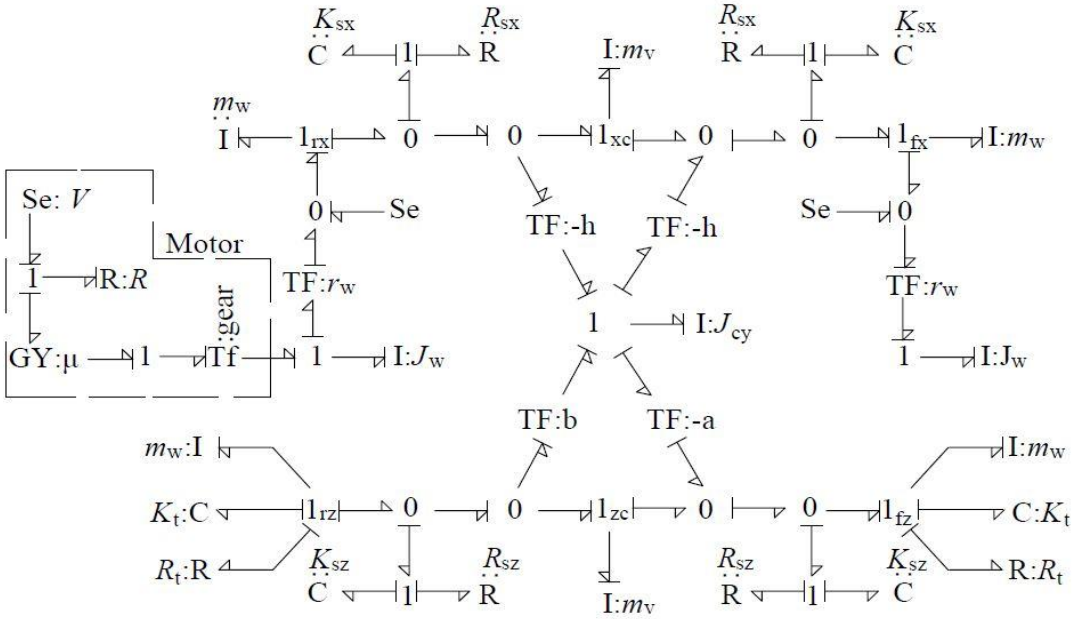
$$Ef_R = \begin{bmatrix} 0 & 1 & 0 & 0.7091 & 0.7068 & 0.9968 & 0 \\ 0 & 1 & 0 & 0.7091 & 0.7068 & 0.9968 & 0 \\ 0 & 0.7091 & 0 & 1 & 0.9913 & 0.7029 & 0 \\ 0 & 0.7091 & 0 & 1 & 0.9913 & 0.7029 & 0 \\ 0 & 0.7007 & 0 & 1 & 0.6940 & 0.0486 & 0 \\ 0 & 0.7007 & 0 & 1 & 0.6940 & 0.0486 & 0 \\ 0 & 1 & 0 & 0.7007 & 0.0483 & 0.0690 & 0 \\ 0 & 1 & 0 & 0.7007 & 0.0483 & 0.0690 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$Ef_C = \begin{bmatrix} 0 & 1 & 0 & 0.7091 & 0.7068 & 0.9968 \\ 0 & 1 & 0 & 0.7091 & 0.7068 & 0.9968 \\ 0 & 0.7091 & 0 & 1 & 0.9913 & 0.7029 \\ 0 & 0.7091 & 0 & 1 & 0.9913 & 0.7029 \\ 0 & 0.7007 & 0 & 1 & 0.6940 & 0.0486 \\ 0 & 0.7007 & 0 & 1 & 0.6940 & 0.0486 \\ 0 & 1 & 0 & 0.7007 & 0.0483 & 0.0690 \\ 0 & 1 & 0 & 0.7007 & 0.0483 & 0.0690 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

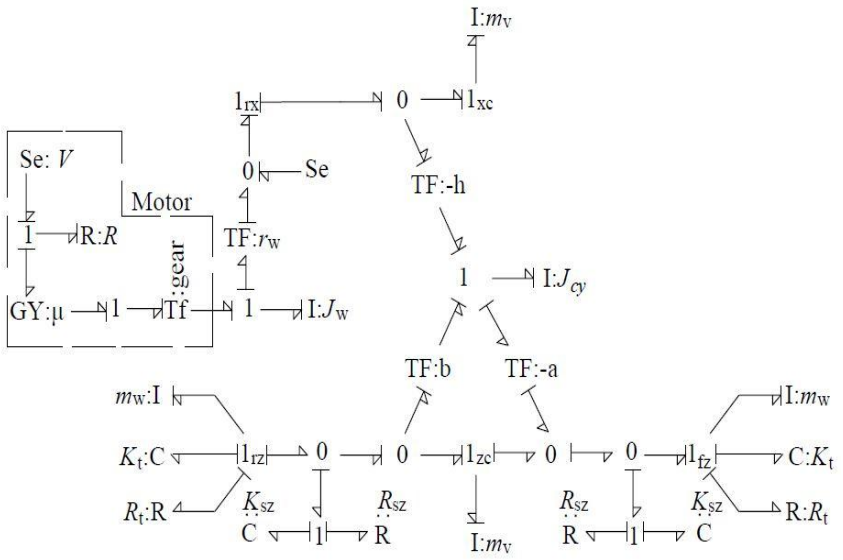
Ef_I, Ef_C and Ef_R show the relative importance of I, C and R elements, respectively for the selected eigenvalues. By comparison, it is found that some columns have low magnitude. The elements corresponding to those columns are neglected and a reduced model is developed. The bond graph approach is used to model the reduced system model and the results are compared. From matrix E_{fI}, it is observed that first and third columns have all the elements of very low magnitude. This means that these elements contribute very less in comparison to other selected eigenvalues; hence, they can be neglected and removed from the model. Similarly, the first and third columns of E_{fC} and E_{fR} also have very low magnitude; hence, these elements can also be neglected. The reduced model of the system is obtained by eliminating these elements from the system. The BG model of the reduced system is illustrated in Fig. 4.16(c).


b
a
z
x
h
z
x
z
x
Motor
Traction
Rear wheel
Front wheel

(a)


Ksx
C
1
Rsx
R
I:mv
Rsx
R
1
Ksx
C
mw
I
1rx
0
0
1xc
0
0
1fx
I:mw
0
Se
TF:-h
TF:-h
Se
0
Se: V
Motor
1
R:R
TF:rw
TF:rw
gear
GY:μ
1
Tf
1
I:Jw
1
I:Jcy
1
I:Jw
TF:b
TF:-a
mw:I
I:mw
Kt:C
1rz
0
0
1zc
0
0
1fz
C:Kt
Rt:R
Ksz
C
1
Rsz
R
I:mv
Rsz
R
1
Ksz
C
R:Rt

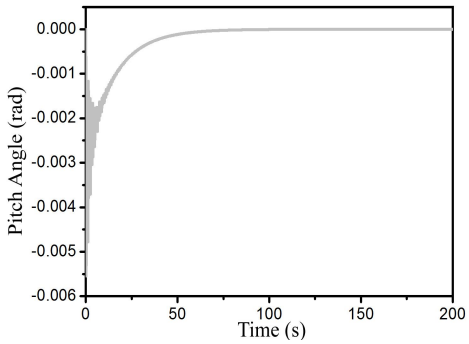
(b)



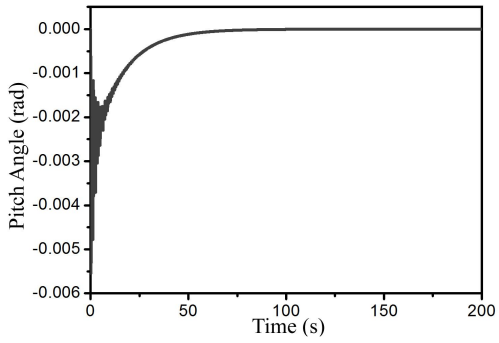
(c)

Fig. 4.16 (a) Schematic diagram of bicycle model, (b) Bond graph of bicycle model and (c) Reduced bond graph of bicycle model with suspension

Figure 4.17(a–d) show the comparison of full and reduced model by comparing the pitch angle of vehicle body centre. It is observed that reduced model gives results very close to the full model. The percentage error becomes high if pitch angle becomes close to zero, but this high percentage error can be neglected as the absolute error is negligible. Figure 4.17(e–g) compare the full and reduced model by considering the vehicle velocity. Figure 4.17(g) shows the variation in percentage error of the reduced system. The maximum percentage error of nearly 2% is observed. Hence, the reduced model conserves the longitudinal dynamics of the system. In reduced model, the velocity, pitch angle etc. reach to the steady state 1.374 sec earlier than the full model.



(a) Full model



(b) Reduced model

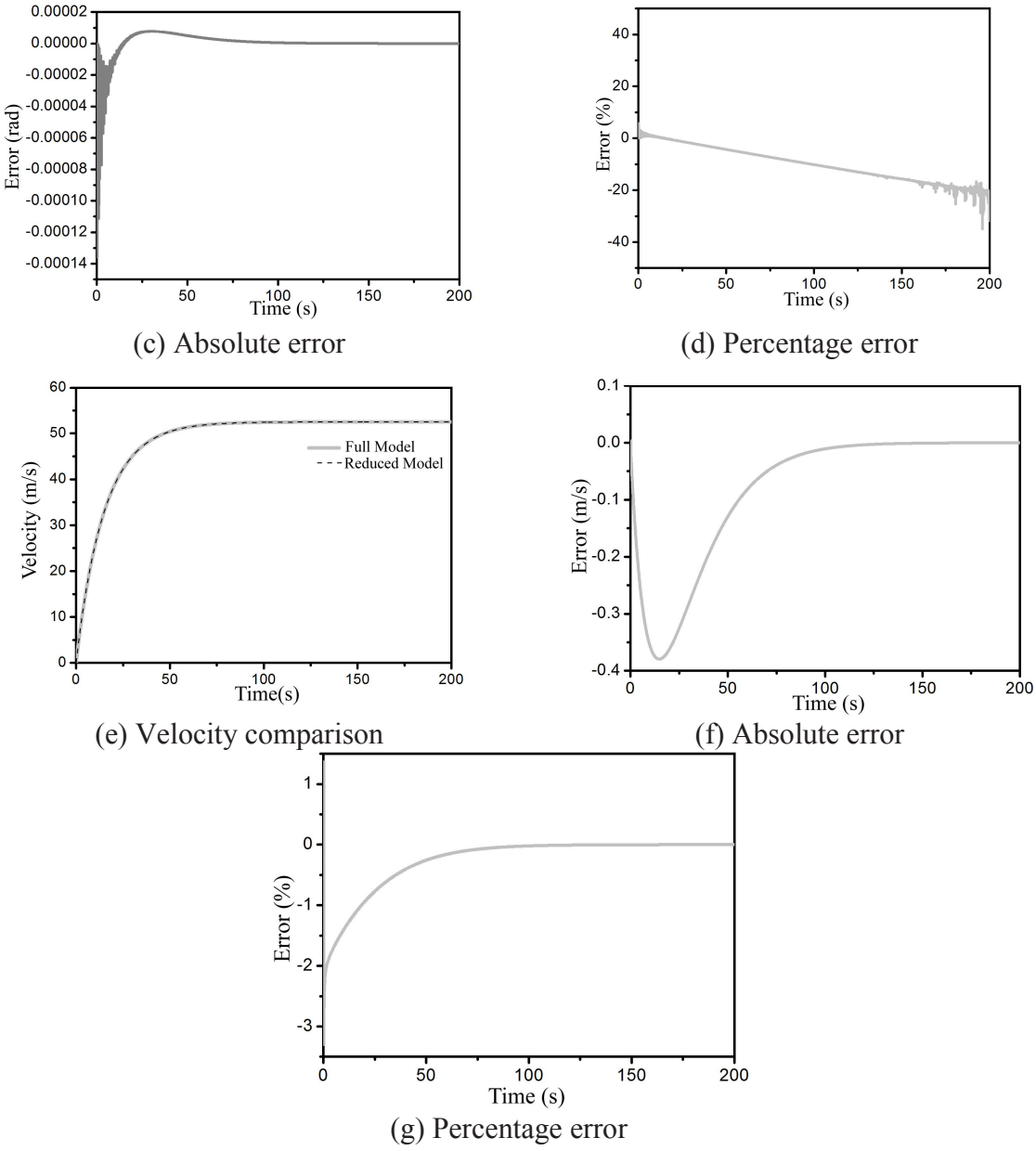


Fig. 4.17 (a) Pitch angle of full model, (b) Pitch angle of reduced model, (c) Absolute error of pitch angle of full and reduced model, (d) Percentage error in value of pitch angle of full and reduced model, (e) Velocity comparison, (f) Absolute error and (g) Percentage error

- **Four-Wheel Vehicle Model**

The four-wheel dynamic model consists of six basic components i.e., the vehicle body, suspensions, wheels, differential, steering and ABS. The word bond graph of a four-wheel vehicle model is shown in Fig. 4.18.

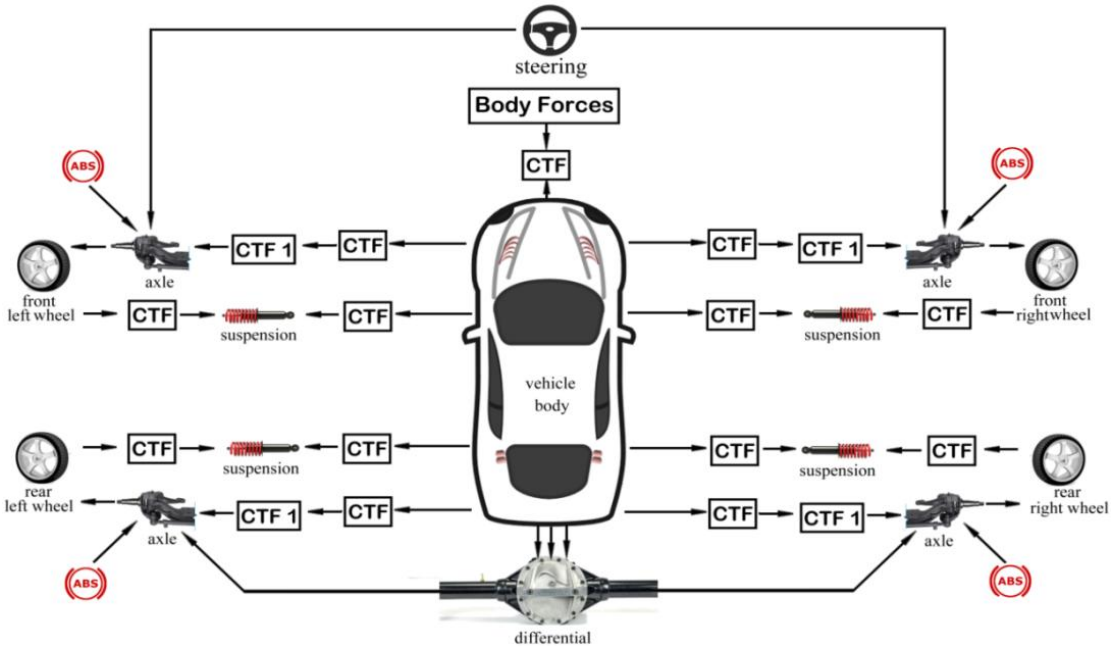
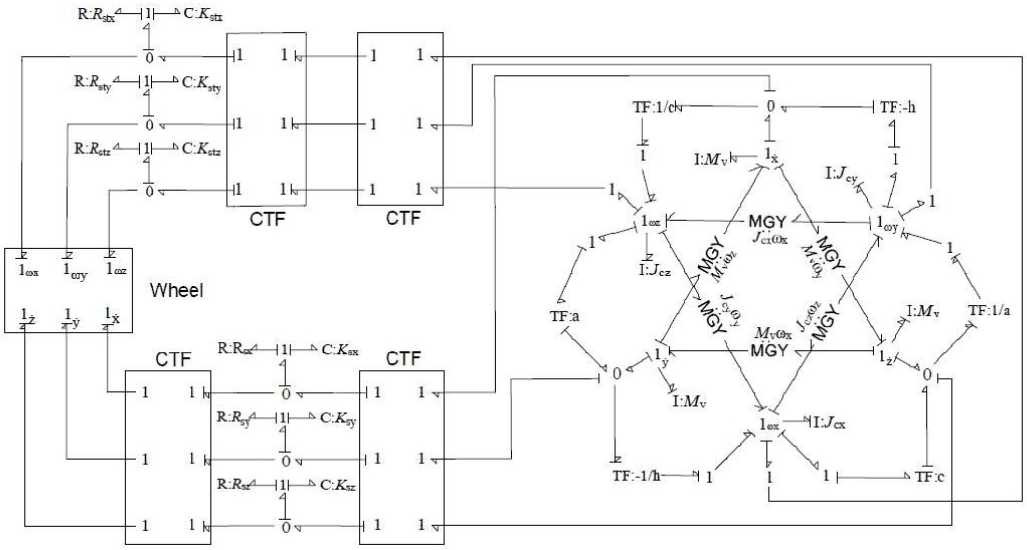
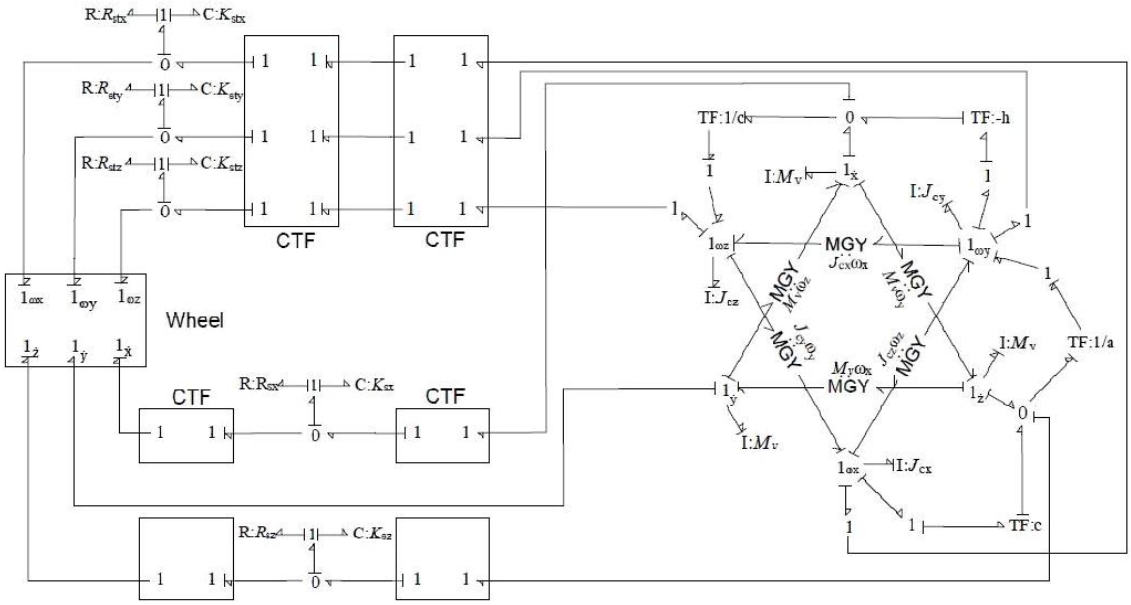


Fig. 4.18 Word bond graph model of four-wheel vehicle model

The vehicle body and the wheels are considered to be in non-inertial frame, while the suspensions are considered in the inertial frame. So, the components are connected to each other through various coordinate transformations. The bond graph model and the reduced bond graph showing connection between vehicle body and the front left wheel are illustrated in Fig. 4.19(a, b), respectively.



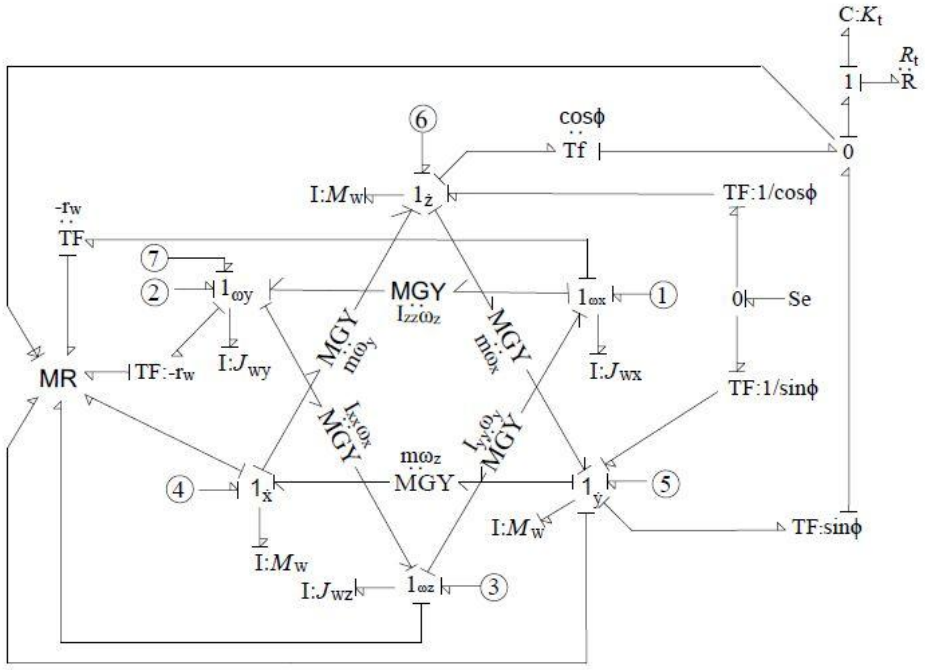
(a)



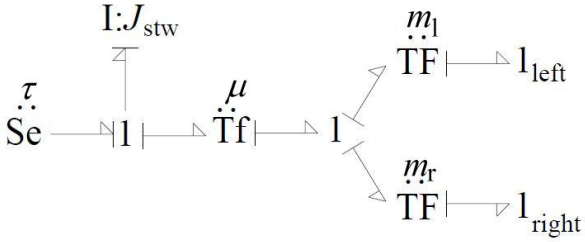
(b)

Fig. 4.19 (a) Full bond graph model and (b) Reduced bond graph model of vehicle body

The BG model of wheel and steering system are shown in Fig. 4.20(a, b). Torque (represented as effort source) is applied by the operator on the steering wheel. An inertial element J_{stw} represents the moment of inertia of the steering wheel. The transformer modulus μ is a mechanical transformation representing the ratio between the rotation of the steering wheel and actual steering rotation. The modulated transformer moduli m_l and m_r are calculated from the Ackerman steering geometry.



(a)



(b)

Fig. 4.20 Bond graph of (a) Wheel and (b) Steering system used in four-wheel model

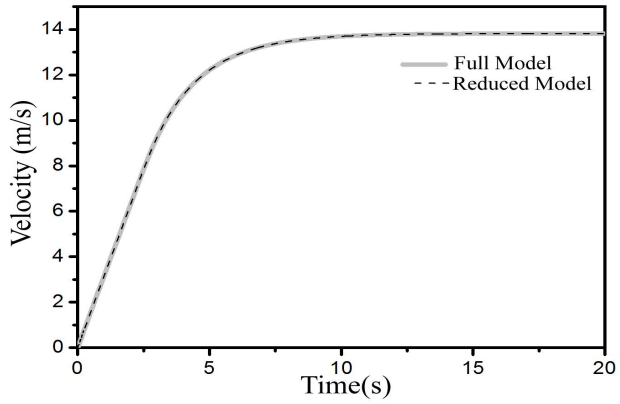
The eigenvalue sensitivity method was applied for the three cases in four-wheel model. For all the cases, the eigenvalues with magnitudes greater than 10^4 and those less than unity were neglected. Figure 4.19(b) shows the reduced bond graph of the vehicle body and the connection between the vehicle body and front left wheel. The suspension elements K_{sy} and R_{sy} were removed. The parameter values used to simulate the complete system model and the reduced system model are listed in Table 4.7. The full model and reduced model were simulated using the above mentioned parameters. As per the assumption on no steering condition, the lateral dynamics does not have much of an effect, so the models were compared in terms of their longitudinal dynamics. Initially, the vehicle was at rest and it gradually accelerated to a constant velocity.

Table 4.7 Parameters used for simulation of four-wheel model

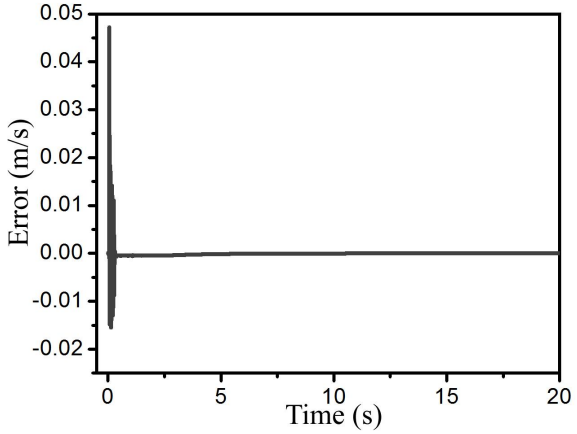
Vehicle body			
m_v	1600 kg	J_{cv}	1110 kg m^2
a	0.9 m	b	1.5 m
J_{cx}	260 kg m^2	J_{cz}	1370 kg m^2
C	0.7 m	H	0.1 m
Wheel			
m_w	15 kg	J_{wy}	0.2 kg m^2
r_w	0.3 m	J_{wx}	0.1 kg m^2
J_{wz}	0.1 kg m^2		
Tyre			
R_t	200 N/m	K_t	305 KN/m
C_1	1.029	C_2	17.16
C_3	0.523	C_4	0.03
Suspension			
K_{sx}	107 N/m	R_{sx}	2000 N/m
K_{sz}	80000 N/m	Rsz	500 N/m
K_{sy}	107 N/m	R_{sy}	2000 N/m
K_{stx}	107 Nm/rad	R_{stx}	2000 Nms/rad
K_{sty}	0 Nm/rad	R_{sty}	0 Nms/rad
Kstz	106 Nm/rad	R_{stz}	360 Nms/rad
Brake			
σ_{low}	0.2	σ_{high}	0.25
s_g	0.01	Kg	250 Nm
r_{bd}	0.15 m	R_{1m}	0.04 Ns/m
K_{ca}	104 N/m	K_{re}	106 N/m
l_a	1 m		
Steering			
J_{stw}	1 kg m^2	Δ	0 rad

The longitudinal velocity of the vehicle and pitch angle of the vehicle body, were calculated and compared in both the models. Figure 4.21(a–c) compare the longitudinal velocities of the models. Figure 4.21(c) shows that higher percentage error is present only at the start, when velocity is very low, but absolute error is within limits. A similar error trend is observed in Fig. 4.21(d–g) which compare the pitch angle in the vehicle body. The distance (a) between the centroid of the vehicle and front axle is less than the distance (b) between the centroid of the vehicle and rear axle as mentioned in Table 4.7.

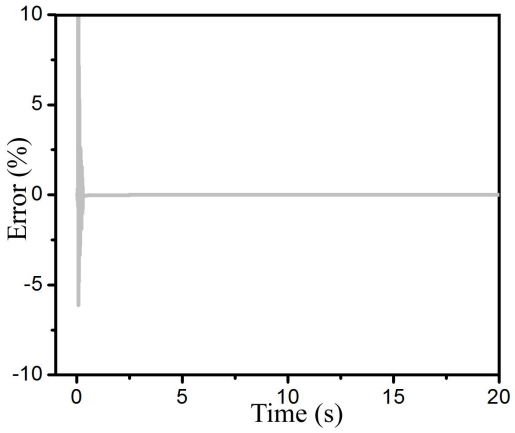
The reactive force at the contact point between front wheel and surface is more than the reactive force for rear wheel. Therefore, the vehicle rotates in the counter clockwise direction through an angle of 0.6° (pitch angle) when it comes to the steady state.



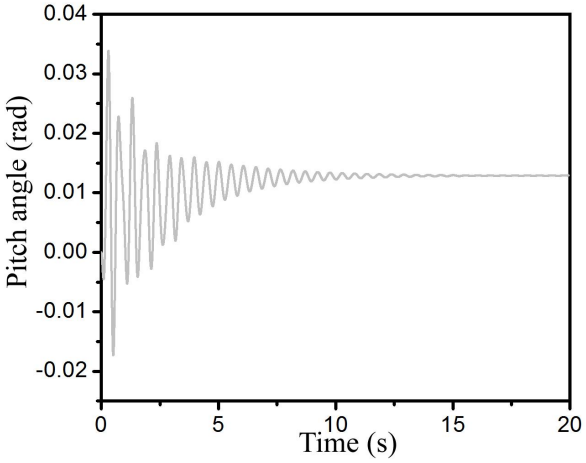
(a)



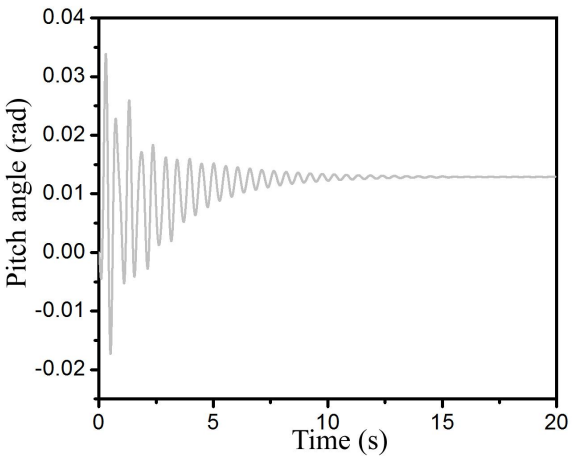
(b) Absolute error



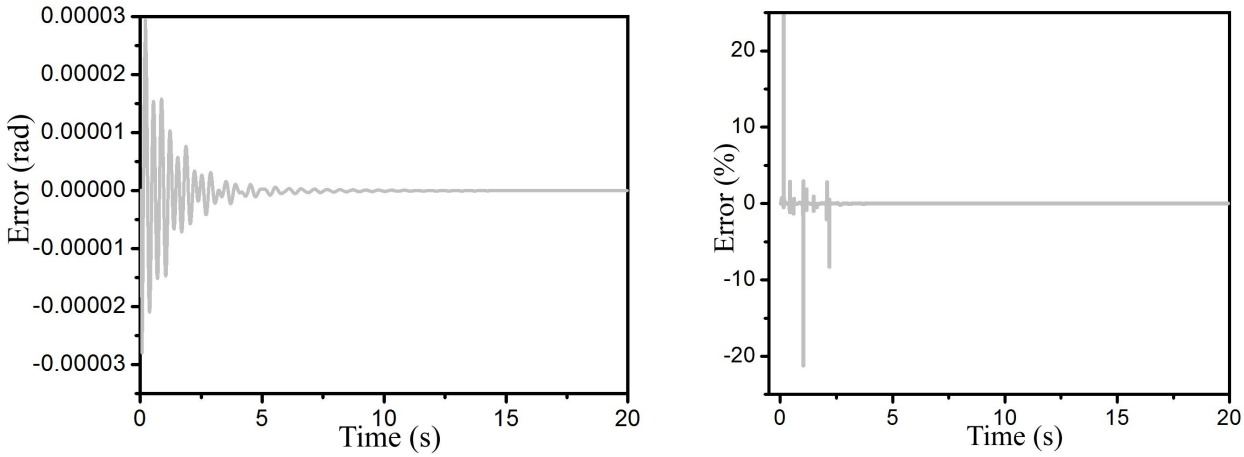
(c) Percentage error



(d) Full model



(e) Reduce model



(f) Absolute error (g) Percentage error

Fig. 4.21 (a) Comparison of longitudinal velocity in full and reduced model, (b)Absolute error in longitudinal velocity,(c) Percentage error in longitudinal velocity, (d) Pitch angle in full model, (e) Pitch angle in reduced model, (f) Absolute error in pitch angle and (g) Percentage error in pitch angle

4.4 Modelling and Control of a Two-Wheel Mobile Robot

The most relevant example of inverted pendulum is a human being because a human with upright body have to make constant adjustments to maintain the balance while walking or running. Suppose, if a person is in standing position and then he suddenly leans into forward direction. The person will be out of balance at that stage and he will probably fall on his face. At that time, brain anticipates that the person is going out of balance and then brain commands a signal to step forward and to stop the falling down. The principle of inverted pendulum system has been employed into two-wheel mobile robot (segway) which is a self-balancing human transporter and looks like a high technological scooter as shown in Fig. 4.22.

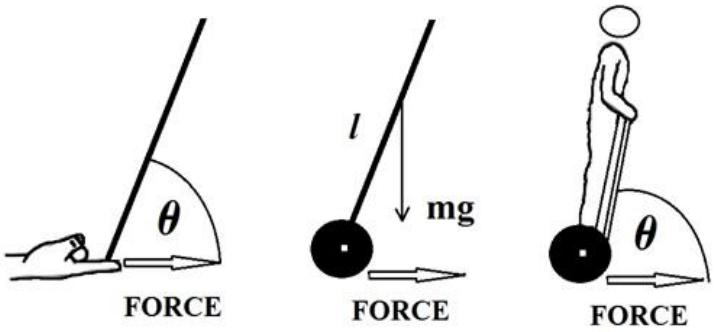


Fig. 4.22 Concept of balancing a broomstick on index finger applied to segway

Pendulum system is controlled for two specific stages. The first one is the swinging up of the pendulum rod to the upright unstable equilibrium position from the downward equilibrium position with the help of a heuristic controller. As the pendulum is reached near to the unstable equilibrium, the second task is the stabilization of the pendulum near the inverted position with the help of state feedback controller (PI controller). The stabilization concept of inverted pendulum system has been employed in this work to form a segway (two wheeled inverted pendulum system). The simulation results for straight line motion as well as turning motion are also presented. The segway moves forward if the handle bar is tilted in the front and moves backward if the handle bar is tilted in the rear. For the left turn, the bar is turned towards left and for the right turn, the bar is tilted towards right. The motivation of this work is to validate through modelling and simulation of a non-model based control system design i.e. energy based control technique is used.

Modelling and simulation play a vital role in the domain of design process. The accurate mathematical depiction of any system gives an easy way to understand the system in a very quick and accurate manner. The data and information about any system can be obtained from modelling and simulation before implementing it in the real life situation. Bond graph model is an efficient tool for modelling engineering systems. Bond graph modelling technique is a way to depict the dynamic behaviour of a physical system. The technique is very similar to the block diagram and signal flow graph but the major difference is that the bonds in the bond graph represent bi-directional energy exchange [Karnopp *et al*., 2000; Mukherjee *et al*., 2006]. The Segway is the type of two-wheel mobile robot. Basically, it is inverted pendulum based robot. The Segway is complex in nature during maneuvering. Therefore, dynamics play an important role. The bond graph technique is used to develop the dynamic model of the Segway.

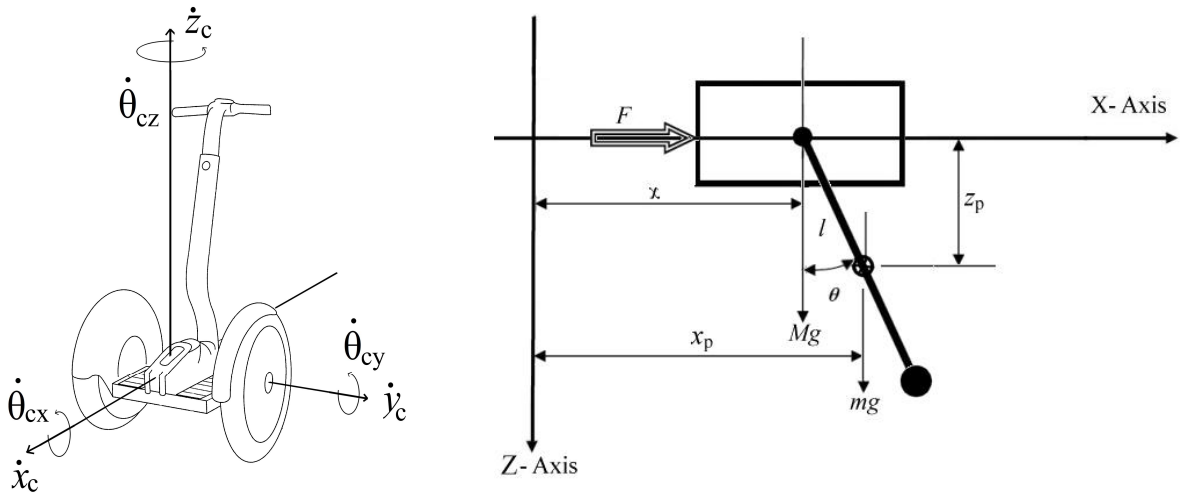
4.4.1 Bond Graph Model of Inverted Pendulum System

The two-wheel mobile robot is composed of a swinging bar (inverted pendulum) attached to the vehicle body which is connected to the two wheels through an axle. The wheel-bushing connects the wheel axle and mounting point. The bushing system is modelled as a parallel spring damper system. The principle behind the working of this two wheeled

scooter, segway (Fig. 4.23(a)) is the stabilization control principle of the inverted pendulum system. Initially, the modelling of inverted pendulum is discussed.

4.4.1.1 Mathematical Model

The mathematical model of inverted pendulum system (Fig. 4.23(b)) is obtained using Euler-Lagrangian (EL) approach to compare the system equations obtained from bond graph model. Only to compare the system equations, EL method is adopted along with bond graph technique. As the equations that would be obtained are same for both the modelling technique, the results would be similar. Hence, results of EL method are not shown. The inverted pendulum system comprises of a cart and a pendulum rod, mounted on the cart. The air resistance and other friction factors are not taken into consideration while deriving the equations of motions.



(a) (b)

Fig. 4.23 Schema of (a) Segway and (b) A cart with an inverted pendulum

Since, this system has two degrees-of-freedom, two generalized coordinates (x, θ) are required to represent the position of the system in space at some particular instant. The coordinates of mass centre of pendulum rod are given as

$$X = \begin{bmatrix} x_p \\ z_p \end{bmatrix} = \begin{bmatrix} x + l\sin\theta \\ l\cos\theta \end{bmatrix} \tag{4.11}$$

where l is the distance from rod's mass centre to the cart's mass centre. The kinetic energy of the pendulum rod and the cart is given as

$$KE_{\text{total}} = \frac{1}{2}(M+m)\dot{x}^2 + \frac{1}{2}\left(J + ml^2\right)\dot{\theta}^2 + ml\dot{x}\dot{\theta}\cos\theta \tag{4.12}$$

where M, m, J are the mass of the cart, pendulum and rotary inertia of rod about the hinge, respectively. The potential energy of the complete system is given as

$$PE_{\text{total}} = mgl\cos\theta \tag{4.13}$$

Now, the Lagrangian is given by

$$L = \frac{1}{2}(M+m)\dot{x}^2 + \frac{1}{2}\left(J + ml^2\right)\dot{\theta}^2 + ml\dot{x}\dot{\theta}\cos\theta - mgl\cos\theta \tag{4.14}$$

To derive the damping force of the system, Rayleigh dissipation function along x-direction is given as

$$R = \frac{1}{2}R_c\dot{x}^2 \tag{4.15}$$

where R_c is the viscous damping of cart. Therefore, equations of motion for the system are given as

$$(M+m)\ddot{x} + ml\ddot{\theta}\cos\theta - ml\dot{\theta}^2\sin\theta + R_c\dot{x} = F \tag{4.16}$$

$$\left(J + ml^2\right)\ddot{\theta} - mgl\sin\theta + ml\ddot{x}\cos\theta = 0 \tag{4.17}$$

4.4.1.2 Bond Graph Model

The complete model of the inverted pendulum system comprises of three sub-systems which are cart-pendulum, DC motor to control the wheel of the cart and the controller part for swinging up and stabilization control. The cart moves along the horizontal track with the help of control input voltage applied to the DC motor. The input voltage is modulated by the control output. The cart has only one velocity component which is in the horizontal direction. On the other hand, the pendulum rod has angular motion about the point where it is pinned to the cart. The bond graph model (Fig. 4.24) of the inverted pendulum system is developed. The half arrow line is the power bonds and the complete arrow line represents the information bonds (sensor) measuring either rate of flow or effort. The junction $1_{\dot{x}}$ in the bond graph denoted the cart's linear velocity component in

the horizontal direction. The cart's mass (*M*), viscous damping of the cart (R_C) and a flow detector element (Df) are connected to the linear velocity component of the cart. $1_{\dot{x}_P}$ and $1_{\dot{z}_P}$ denote the horizontal and vertical component of velocities of rod's mass centre, respectively. The pendulum mass (*m*) is connected to these both velocity components. The weight of the pendulum (*mg*) which is acting in the vertical downward direction and coupling capacitor (K_P) is connected to the vertical velocity component of pendulum's mass centre to provide flexibility. $1_{\dot{\theta}}$ denotes the pendulum's angular velocity. The rotary inertia of rod (*J*), stiffness of revolute joint (K_S), damping (R_P) and a flow detector (Df) which gives the angular position of the inverted pendulum are connected to the angular velocity component ($1_{\dot{\theta}}$-junction). The cart and pendulum are connected via the two modulated transformer elements to form the equations of motion. $u_1 = l\cos\theta$, $u_2 = l\sin\theta$ are modulus of the transformer. The motor electrical resistance (R_m), motor inductance (*L*), motor torque constant (μ_m), flywheel inertia of motor (J_m), gear ratio (*G*), viscous damping (R_d) are the parameters of the motor shown in the bond graph model (Fig. 4.24). The flow detector element (Df) is used for the measurement of gear output shaft's angular velocity (ω_m). The motor shaft's angular velocity is converted into linear velocity with the help of the transformer element having modulus *'r'* which is the radius of the belt. The belt stiffness (K_b), belt damping (R_b) are represented in the model.

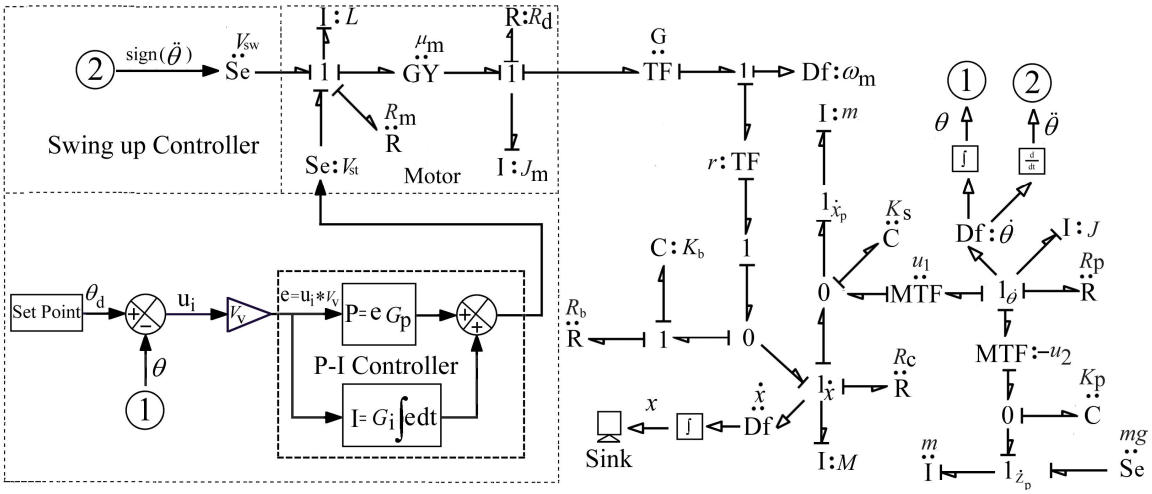


Fig. 4.24 Bond graph model of inverted pendulum system

4.4.1.3 Controller Design

The control scheme used for swing up is heuristic controller and for stabilizing control, the controller used is proportional-integral (PI) control. By default, the swing up controller is in action. The stabilizing controller is in action if the pendulum makes an angle less than ±20° from vertical z-axis (from the inverted position). In this way, the switching between swing up and stabilizing controller is done.

- **Heuristic Controller**

Heuristic controller is a type of logic based non-linear controller that determines the direction and time instant at which the cart should move in the forward or backward direction. The main objective here is to take the pendulum to the upright position and maintain the pendulum over there despite of the small disturbances. The appropriate amount of energy must be injected to the system so that the sum of the potential energy and the rotary kinetic energy of pendulum become equal to the potential energy of the pendulum at the upright position [Maeba *et al.*, 2010] considering no loss of energy. This means that initially the rotational kinetic energy of the pendulum system must be increased whatever may be the value of the translational kinetic energy. Let H denote the sum of the potential energy and the rotational kinetic energy and it is given as

$$H = mgl\cos\theta + \frac{1}{2}\left(J + ml^2\right)\dot{\theta}^2 \tag{4.18}$$

where θ is the angular displacement of the rod from vertical axis. Now, taking the

derivative of the H with respect to time we get

$$\frac{dH}{dt} = \left(J + ml^2\right)\dot{\theta}\ddot{\theta} - mgl\dot{\theta}\sin\theta \tag{4.19}$$

The angular acceleration is calculated from Eq. (4.17) and is given by

$$\ddot{\theta} = \frac{mgl\sin\theta - ml\ddot{x}\cos\theta}{J + ml^2} \tag{4.20}$$

Putting the value of $\ddot{\theta}$ from the Eq. (4.20) in Eq. (4.19), we get

$$\frac{dH}{dt} = -ml\ddot{x}\dot{\theta}\cos\theta \tag{4.21}$$

In order to increase energy, the value of the derivative must be greater than or equal to zero. The above equation signifies that in order to increase the energy function, the acceleration of the cart ($\ddot{x}$) and the pendulum's angular velocity ($\dot{\theta}$) must be of opposite sign. The cart will not be able to add more energy to the pendulum system when the pendulum's angular position becomes horizontal. In any case, it has to induce more energy to the pendulum. If such condition arises, the cart's movement will be stopped and the pendulum will be allowed to simply return to the downward position. When the pendulum crosses the downward position once again, the logic based controller will again be able to add more energy to the pendulum by the cart's movement until the pendulum approaches to the upright position. There is a direct relation between the magnitude of the voltage gain and the time taken by the pendulum to swing up. The sign of voltage (V) is changed when the acceleration of the pendulum is maximum at the mid position. This swing up controller is shown in the left-top part of Fig. 4.25. This controller is in action if the pendulum makes an angle of more than ±20° from vertical z-axis (from the inverted position).

- **PI Controller**

The PI controller is used here as a stabilization controller to catch and hold the inverted pendulum at the inverted position. The mathematical expression for PI controller is given as

$$u(t) = G_P e(t) + G_i \int_0^t e(t)dt \tag{4.22}$$

where G_p and G_i are the proportional and integral gains, respectively. The block diagram for PI controller (stabilization controller) is shown in the left part of Fig. 4.25. Two signals that are added to the junction, out of which, one signal is the desired value that is the inverted angular position (θ_d) and the second one is the negative value of actual angular displacement (θ) of pendulum measured by the sensor. The error $e(t)$ between these signals is amplified with a gain of V_v. V_v is the gain of the signal conditioner that converts angle error to voltage error. The output of the PI controller modulates the stabilization voltage. This controller would be active when the inverted pendulum reaches near to the upright position (±20° from vertical z-axis).

4.4.1.4 Simulation Parameters

The bond graph simulation results of the cart inverted pendulum system are presented in Fig. 4.27. The parameters required for bond graph simulation are given in Table 4.8.

Table 4.8 Parameter values

Symbol	Parameters	Parameter values
G	Gear ratio	1
G_p	Proportional gain	0.121
G_i	Integral gain	1.9
J	Rotary inertia of rod	3.4×10^{-4} kg m^2
J_m	Motor inertia	1.4×10^{-4} kg m^2
K_b	Belt stiffness	10^6 N/m
K_p	Coupling capacitor	10^6 N/m
K_s	Stiffness of revolute joint	10^6 Nm/rad
L	Distance from rod mass centre to the cart mass centre	0.125 m
L	Motor inductance	1.0 H
M	Mass of pendulum	0.109 kg
M	Mass of the cart	1.096 kg
R	Belt radius	0.0195 m
R_b	Belt damping	0.1 Ns/m
R_c	Viscous damping of cart	0.1 Ns/m
R_d	Viscous damping of motor	0.03 Nm.s/rad
R_m	Motor resistance	2.5 Ω
R_p	Damping	0 Nm.s/rad
V_v	Angular displacement to voltage convertor	33 V/rad
μ_m	Motor constant	0.1 Nm/A

4.4.2 Experimentation Setup

The linear inverted pendulum system developed by Googol Tech, Hong Kong is used for the experimentation. The overall design is composed of three sub-systems i.e. mechanical system, feedback system and controller interface. Mechanical system consists of cart pendulum system, track and mechanism used to drive the cart. The drive mechanism is a motor with a toothed wheel mounted onto shaft that is used to pull a chain to which the cart is connected. Feedback network consists of sensor and encoders to read position of cart and pendulum.

4.4.2.1 Block Diagram of Experimental Setup

The block diagram of inverted pendulum system is illustrated in Fig. 4.25. The nature of the system is a single input and multiple output (SIMO). The input is voltage and output is the linear cart displacements and angular position of pendulum.

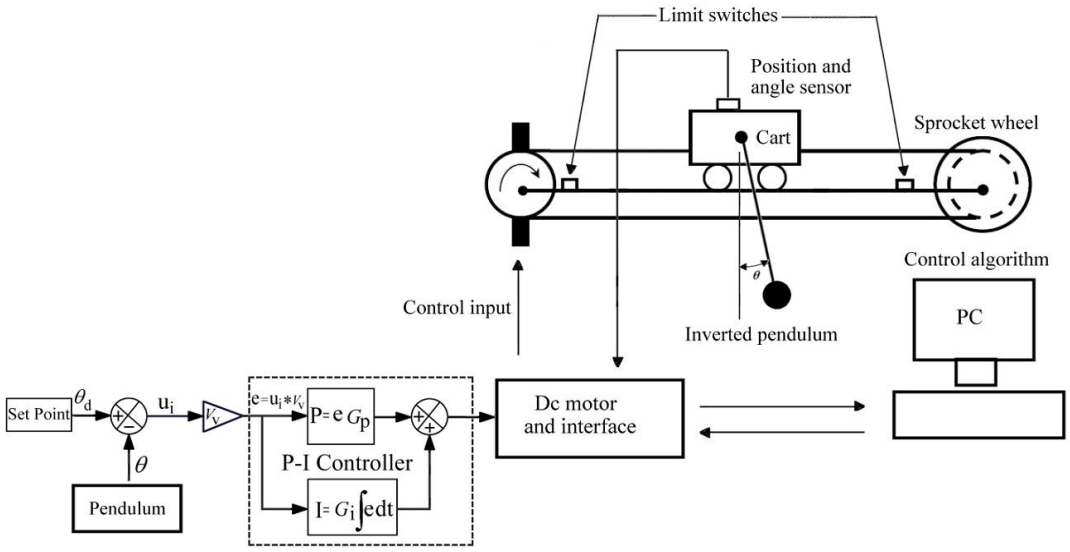


Fig. 4.25 Block diagram of inverted pendulum system

4.4.2.2 Technical Specifications

MATLAB Windows2000 with *Simulink* is used as a user interface between the hardware and software. The experimental setup is shown in Fig. 4.26. Limit switches, motor, anti-

collision buffer device have unique structural design for safety and suitability. The hardware platform of the setup is based on DSP-based motion controller.

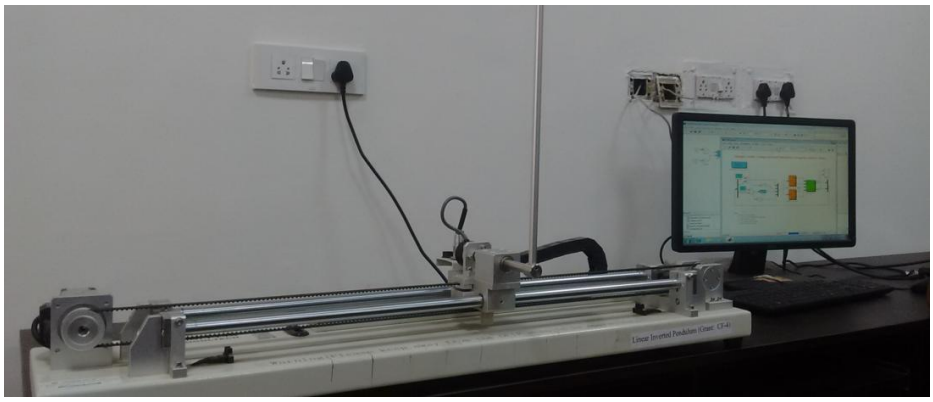


Fig. 4.26 Experimental setup of GLIP

4.4.2.3 Validation of Results

The validation of results for inverted pendulum system is given by comparing the bond graph simulation results (dashed line) with the experimental results (firm line). The inverted pendulum is initially in the vertically downward direction and the angular position (Θ) is measured from this position as shown in Fig. 4.23(b). The first plot is for the pendulum angle with time as shown in the Fig. 4.27(a). Figure 4.27(b) depicts the control input voltage and the third plot is for the cart position with time as illustrated in Fig. 4.27(c). In both the results, the pendulum angle is reaching to the inverted position in six increasing swings. The stabilization time is almost the same for the both results which is approximately about 8.5s. The control input voltage plot with time from both the experimentation as well as bond graph simulation is also the same for the swing up control. For the swing up control, the control input voltage varies between -7.5 V to 7.5 V in both the cases. Figure 4.27(b), soft grey colour shows the experimental oscillations which occur due to the switching of the controller from nonlinear (energy based swing-up controller) mode to the linear mode (i.e. the linearized stabilization controller). The switching takes place at 8.5s when the pendulum follows the linear band ($\theta < 20^{o}$) around the unstable equilibrium point. The transients are shown in Fig. 4.27 (a, b & c) and it comes into the picture after 8.5 sec when stabilization controller takes control of the system. The prime reason of these oscillations is the un-modelled friction dynamics,

sensitivity of the setup due to temperature variation. In the plot of cart position with time, the results from the bond graph simulation and experimentation are not exactly matching with each other but approximately both the results are following the same pattern. In the real time experimentation, there are various resisting factors which are not considered in modelling the behaviour of the system in bond graph.

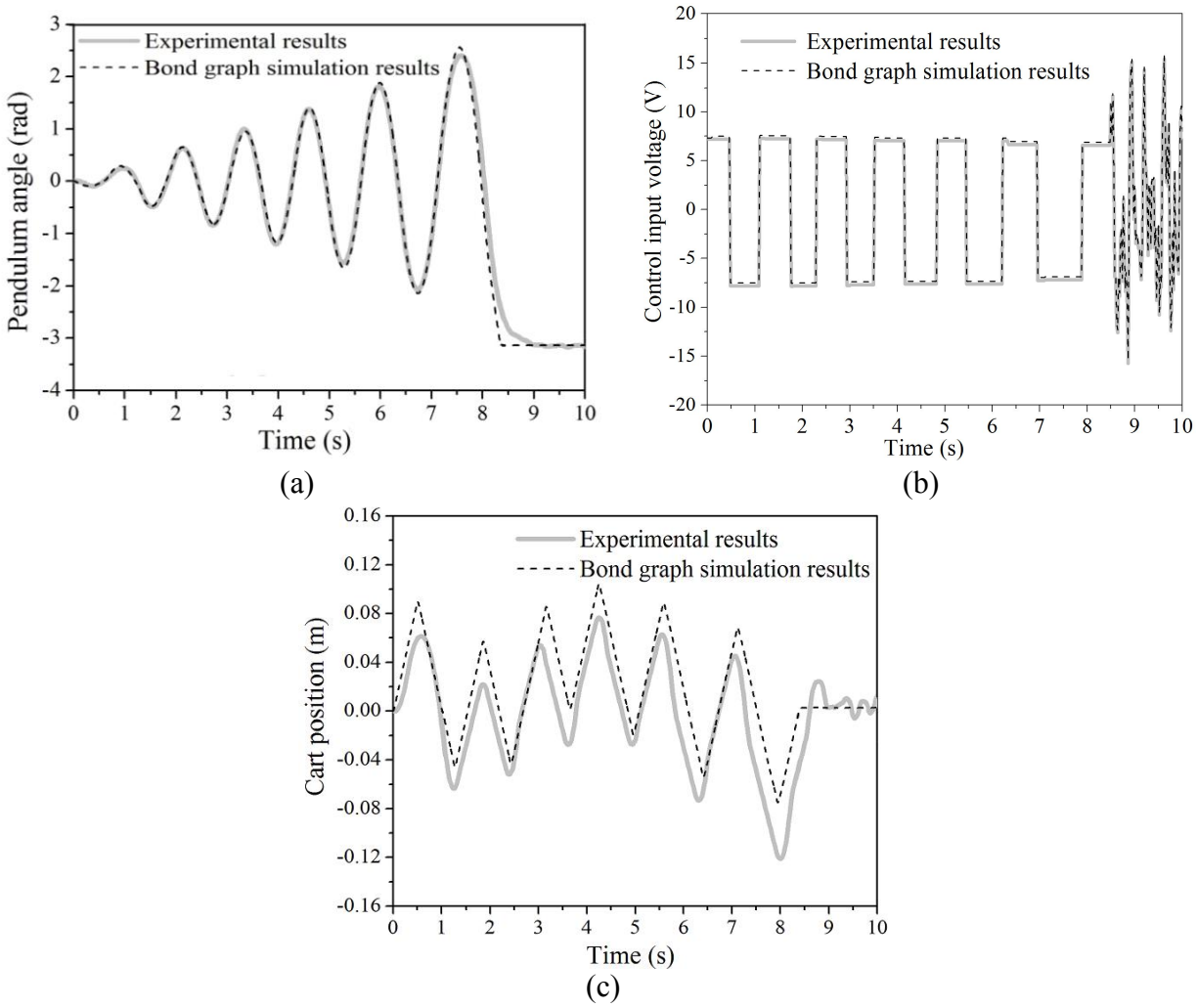


Fig. 4.27 Comparison between experimental result and simulation result of (a) Pendulum angle, (b) Control input voltage and (c) Cart position

4.4.3 Bond Graph Modelling, Control and Simulation of Segway

The modelling of segway is done by the combination of five sub models i.e. vehicle body, wheel, inverted pendulum, electric motor and wheel-bushing system. Figure 4.28

represents the whole system, segway in word bond graph form where the bonds with two parallel lines denote multi bonds between two modules. At the interface of these subsystems (Fig. 4.28), flow variables are only highlighted. The generalized effort variables (forces and torques for linear and angular velocity, respectively) are not mentioned in Fig. 4.28 for maintaining the clarity of it. The bushing system is used to connect the left and right wheels to the main body of the vehicle. The electric motors are connected to the main body of the vehicle and the two wheels with the help of scalar bonds. Likewise, the inverted pendulum is also connected to the main vehicle body via scalar bonds. The coordinate transformation (CTF) is provided in the proper places of the word bond graph. These are used either to convert body fixed velocities into inertial velocities or inertial velocities into body fixed velocities.

4.4.3.1 Vehicle Body

The model of vehicle body is illustrated in Fig. 4.29. The vehicle is assumed to be symmetrical about the longitudinal axis. The bushing is taken as a simple linear spring damper system. The main vehicle body's motion can be expressed as the 3 linear displacements about the three body fixed coordinate axis and rotational motion of the body is given as yaw, pitch and roll motion. The Newton-Euler (NE) equations are used to model the main body of the vehicle. The NE equations are explained in detail in Appendix A. These equations contain attached body fixed axis which are aligned to the inertia principal axis. The Newton-Euler equations are given in Bera *et al.*, [2011], [2012].

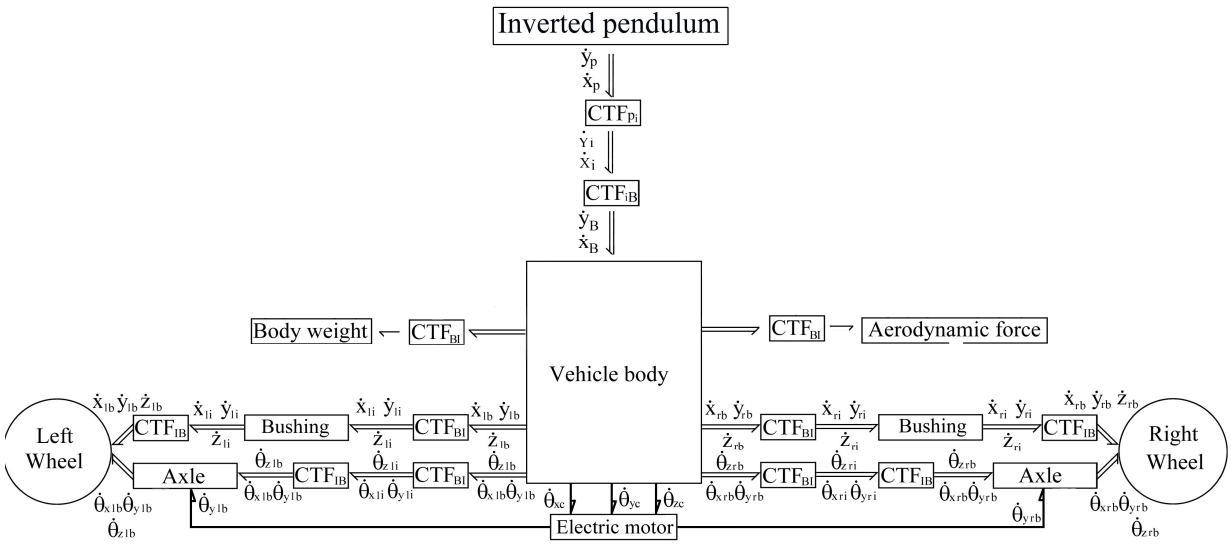


Fig. 4.28 Word bond graph model of segway

Now, further step in the modelling is to compute the velocities at different points on the rigid body, transform them to the inertial coordinate system and physical constraint implementations at chosen points. For linear velocities of left bushing reference point (Fig. 4.29), the equations can be written as

$$\dot{x}_1 = \dot{x}_c + z_1\dot{\theta}_{cy} - y_1\dot{\theta}_{cz} \tag{4.23}$$

$$\dot{y}_1 = \dot{y}_c + x_1\dot{\theta}_{cz} - z_1\dot{\theta}_{cx} \tag{4.24}$$

$$\dot{z}_1 = \dot{z}_c + y_1\dot{\theta}_{cx} - x_1\dot{\theta}_{cy} \tag{4.25}$$

Same type of expressions can be written for the right-side bushing. The transformation of velocity from the moving frame to the inertial frame is given in detail in the appendix and Euler angle rates from the body fixed angular velocities are given in [Bera *et al.*, 2011]. Figure 4.30 represents the models of main vehicle body and transformation of velocities (linear to angular) to the bushing reference points. Three moments and forces (three set each) act upon the body. The forces on both sides of the body are to be added to obtain the total bushing force at the vehicle body's centre of gravity. The bond graph coordinate transformations block (CTF) is used in [Bera *et al.*, 2011] and also explained in the appendix. As the weight of the vehicle body and the aerodynamic forces act in the inertial frame, these forces are to be converted to body

fixed forces through coordinate transformation before being applied to the vehicle body. Here, in this simulation, the aerodynamic forces are neglected as the speed of the segway is not too high. The handle bar which is treated as inverted pendulum is connected with the vehicle body through double coordinate transformations i.e. CTF for body fixed (pendulum frame) to inertial frame and CTF for inertial frame to another body fixed frame.

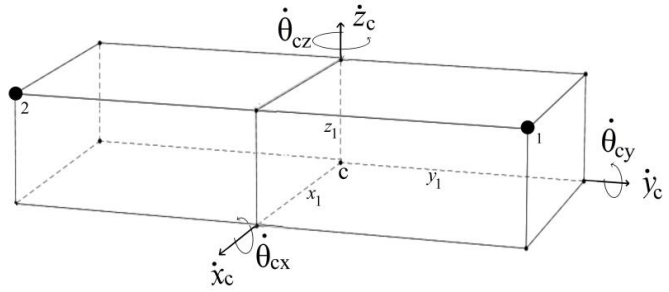


Fig. 4.29 Sketch for illustration of velocities in the moving frame

The bond graph modelling for inverted pendulum system which is attached to the vehicle main body is shown in the Fig. 4.31. The inverted pendulum system is connected to the main body of the vehicle through the two coordinate transformation blocks. The angle θ shown in the bond graph of inverted pendulum in Fig. 4.31 represents the pitch angle which is to be controlled by the PI controller.

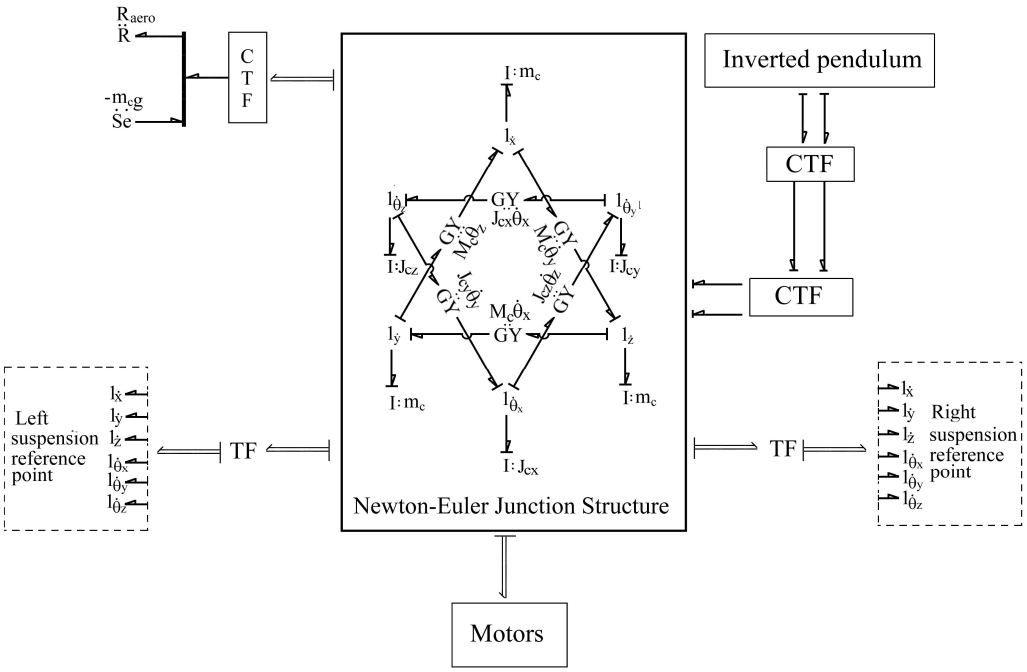


Fig. 4.30 Bond graph modelling of vehicle body and attached components

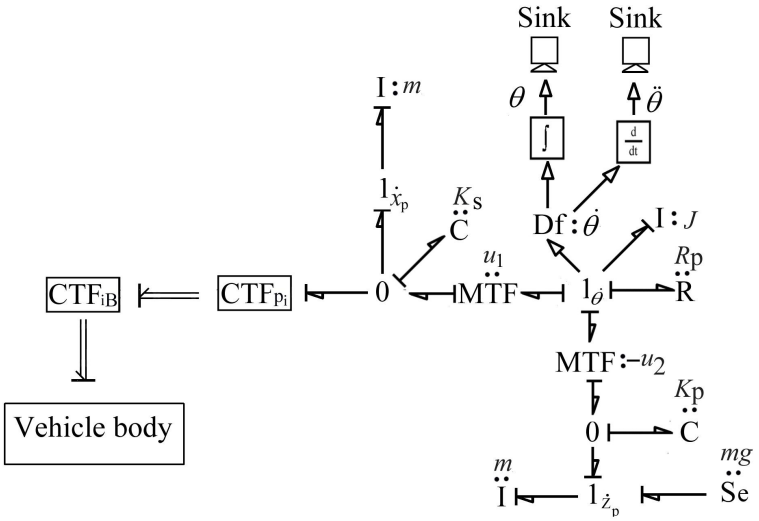


Fig. 4.31 Bond graph model of inverted pendulum attached with segway

4.4.3.2 Model of Wheel

The schematic diagram of wheel is shown in Fig. 4.32(a). The bond graph model for the wheel is illustrated in Fig. 4.32(b). It is considered as a rigid body having five DOF and the rolling about *x*-axis is neglected here. There are six input ports in the bond graph modelling of wheel. Three effort inputs are coming for the bushing and torque is supplied

from the differential. The contact point between the wheel and road changes as per the wheel rotation about the axle. The wheel is rotated about the axle axis and it does not change its orientation in the inertial frame. The gravity force and the normal contact force between the wheel and road always act in the inertial direction. The weight of the wheel is acting in inertial z-direction which can be denoted by source of effort $(\mathrm{Se}: w_{\mathrm{m}} g)$. The mass of the wheel is represented as $(\mathrm{I}: w_{\mathrm{m}})$ and r_{w} is the radius of the wheel. The tractive force which is used to generate the motion between the wheel and road surface is given by $\mathrm{Se}: F_{\mathrm{t}}$. The mass moments of inertias are represented by I_{wy} and I_{wz} in y and z directions, respectively.

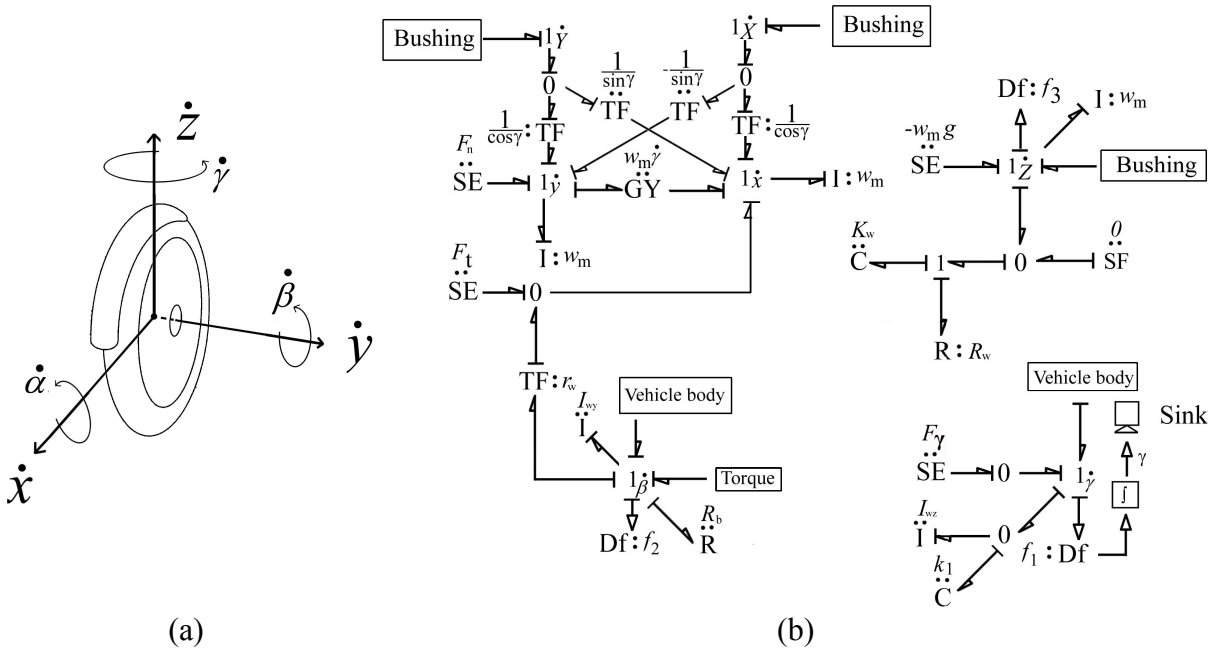


Fig. 4.32 (a) Schematic diagram and (b) Bond graph model of wheel

4.4.3.3 Model of Wheel Bushing

The schematic diagram of wheel bushing is illustrated in Fig. 4.33(a). The BG model of wheel bushing that is located between the wheel axle and mounting point is shown in Fig. 4.33(b). The bushing is modelled as a parallel spring damper system. In the BG model, the stiffness $(\mathrm{C}: K_{\mathrm{S}})$, damping $(\mathrm{R}: R_{\mathrm{S}})$ in x, y directions and stiffness $(\mathrm{C}: K_{\mathrm{z}})$ and

damping $(\mathrm{R}:R_z)$ in inertial z-direction are connected to 0-junction between the flow input from the vehicle body (mounting point) and effort output to the wheel.

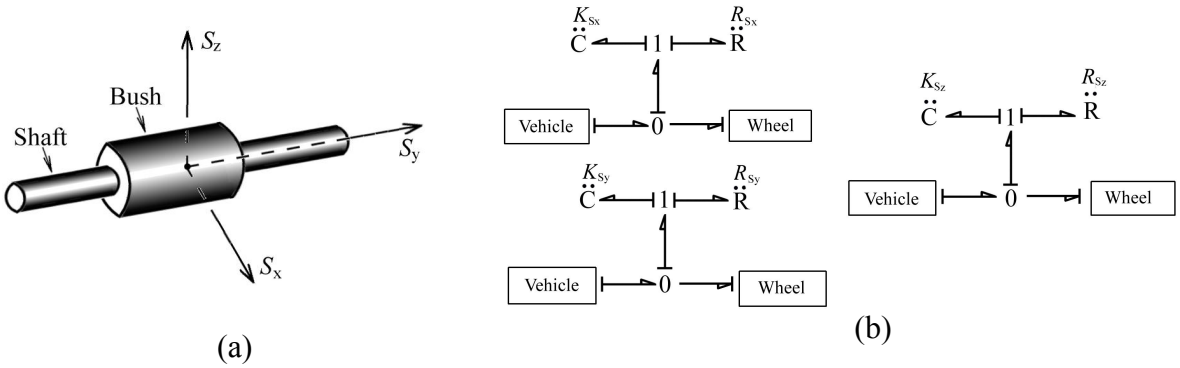


Fig. 4.33 (a) Schematic diagram and (b) Bond graph model of wheel bushing

4.4.3.4 Model of Electrical Motor with PI Control

The bond graph modelling for electrical motor with PI controller is shown in Fig. 4.34(a). Two separate electrical DC motors are used for the motion of the segway. The PI controller is used as a stabilization controller for balancing the pitch angle θ of inverted pendulum attached to the main vehicle body of segway. In case of PI controller, the desired value (θ_{d}) is the inverted position of the pendulum and feedback value (θ) is the value of the pitch angle measured by the sensor. The error between these flows is converted to the voltage error signal with the help of gain V_{v}. Now, according to the voltage error signal, a control signal is generated by the PI controller to modulate the voltage applied to the motor. In this block diagram representation, G_{P} is the proportional gain and G_{I} is the integral gain.

In the bond graph modelling of motors (Fig. 4.34(a)), there are two output ports from 0-junction providing the equal torque to both the left/right wheel and vehicle body. There are three input ports from the vehicle body to the motors and these inputs are transformed with the help of coordinate transformation block CTF from moving frame to inertial frame.

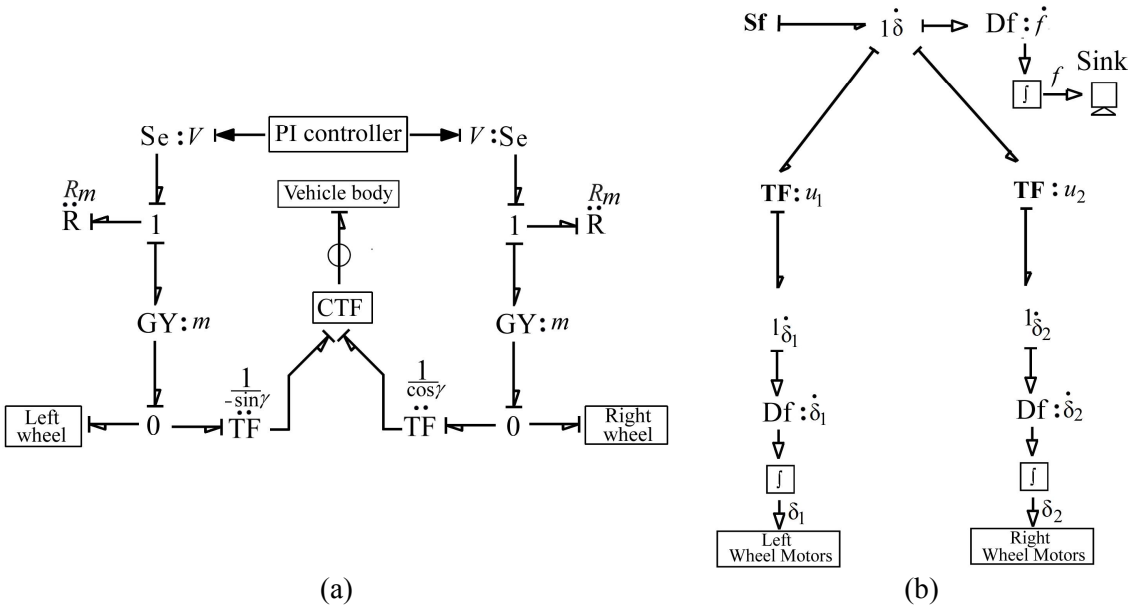


Fig. 4.34 Bond graph model of (a) Motors and (b) Steering controller

4.4.3.5 Model of Steering Control

The turning of vehicle is done by using steering motion. The bond graph modelling for steering controller is given in Fig. 4.34(b). Steering is basically a moment which has applied to the axles of the two wheels about Z-axis causing the change in the yaw angle of the wheel (also the relative torque on the body of the vehicle. The steering is provided by changing the rotational speed of the motors. The speeds of the motors are modulated by δ_1 and δ_2 where δ_1 and δ_2 are the angular position of the wheel about z-axis. These angular position is determined by steering controller. The modulus of two transformers μ_1 and μ_2 as shown in Fig. 4.34(b) are determined by the Ackermann's formulae [Bera *et al*., 2011] in the controller domain and these are given as

$$\dot{\delta}_i = \left[\frac{a\cos^2\theta_l + c\tan\theta_l\cos^2\theta_l}{a\cos^2\theta_{st} - c\tan\theta_{st}\cos^2\theta_{st}}\right]\dot{\delta} \tag{4.26}$$

$$\dot{\delta}_o = \left[\frac{a\cos^2\theta_r - c\tan\theta_r\cos^2\theta_r}{a\cos^2\theta_{st} + c\tan\theta_{st}\cos^2\theta_{st}}\right]\dot{\delta} \tag{4.27}$$

4.4.3.6 Controller for Single Wheel Model of Segway

Initially, single wheel model of segway without controller (Fig. 4.35) is developed in the similar line of the Fig. 4.31. M_2, M_6, M_{35}, M_{16}, M_{20}, M_{23} and M_{30} are inductance of motor, flywheel along with wheel mass moment of inertia, mass of wheel, mass of vehicle (segway), pendulum mass, mass moment of inertia of pendulum rod and pendulum mass, respectively. The signal flow graph of the single wheel model is shown in Fig. 4.36.

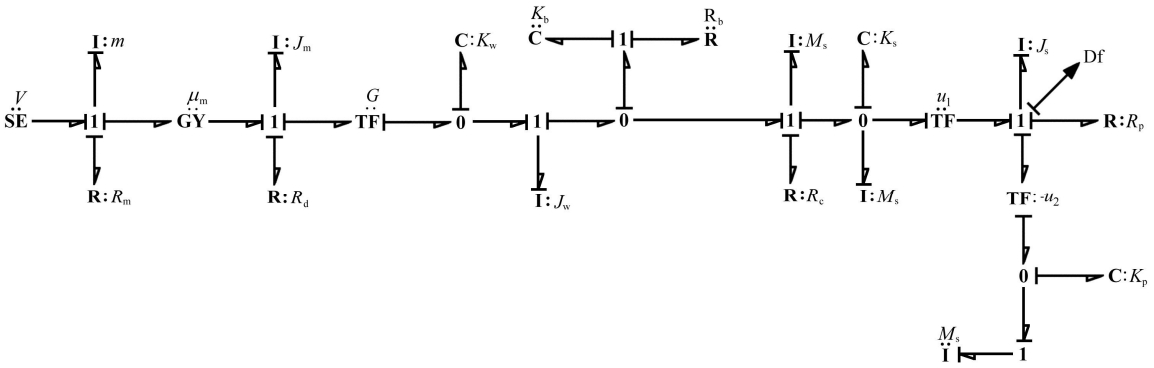


Fig. 4.35 Single wheel model of segway without controller

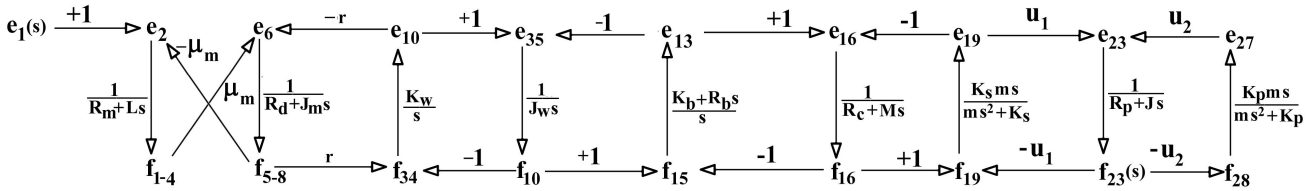


Fig. 4.36 Signal flow graph of single wheel model of segway without controller

The transfer function between the angular velocity of Segway arm and the voltage applied to the motor when linearized about the vertical position is

$$\frac{f_{23}(s)}{e_1(s)} = \frac{3.1\times10^{9}s^{4} + 3.1\times10^{16}s^{3} + 6.3\times10^{14}s^{2} + 6.3\times10^{21}s + 5.8\times10^{8}}{s^{11} + 3.9s^{10} + 1.2\times10^{7}s^{9} + 4.2\times10^{7}s^{8} + 1.3\times10^{13}s^{7} + 4.2\times10^{13}s^{6} + 4.4\times10^{18}s^{5} + 1.4\times10^{19}s^{4} +}$$

$$4.8\times10^{23}s^{3} + 1.4\times10^{24}s^{2} + 5.5\times10^{23}s + 3.3\times10^{21}$$

(4.28)

From the open-loop transfer function of the system as given in Eq. (4.28), it is found that there is a double pole at the origin, which makes the system unstable. Rest all the poles are either at imaginary axis or in the left-half plane. Further, the controllability of the open-loop system is calculated and it is found that all the 11 states are controllable. The bond graph model of single wheel model of segway with PI controller is shown in Fig. 4.37. The transfer function between the angular velocity of Segway arm and the voltage applied to the motor with PI controller is

$$\frac{f_{23}(s)}{e_{37}(s)} = \frac{2.6\times10^{10}s^5 + 2.6\times10^{17}s^4 + 2.2\times10^{16}s^3 + 5.2\times10^{22}s^2 + 3.3\times10^{21}s - 3.9\times10^{16}}{s^{12} + 3.9s^{11} + 1.2\times10^{7}s^{10} + 4.2\times10^{7}s^{9} + 1.3\times10^{13}s^{8} + 4.2\times10^{13}s^{7} + 4.4\times10^{18}s^{6} + 1.4\times10^{19}s^{5} +}$$

$$4.8\times10^{23}s^4 + 1.4\times10^{24}s^3 + 1.1\times10^{24}s^2 + 2.2\times10^{23}s + 3.3\times10^{20} \qquad (4.29)$$

After implementing the PI controller, all the closed-loop poles are lying in the left half plane, which ensure the stability of the system. When PI controller is added in the forward loop, there is a new pole lies at the negative real axis (-0.51+0. i) due to the use of integral term and a new zero is also located at (-0.64+ 0. i) on the same axis.

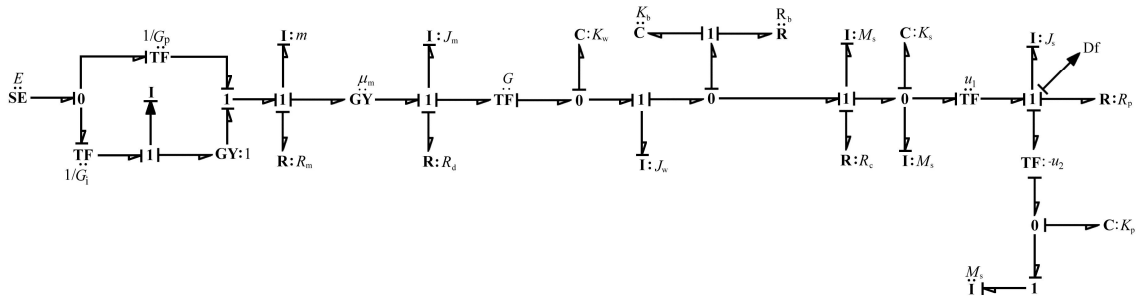


Fig. 4.37 Single wheel model of segway with PI controller

4.4.3.7 Bond Graph Simulation Results of Segway

The parameter values for simulating segway for forward or backward motion or for turning are given in Table 4.9.

Table 4.9 Simulation parameters for segway

Nomenclature	Description	Values
a	Distance of the axle form the vehicle CG	0.05 m
G_p	Proportional gain	1.0
G_i	Integral gain	1.21
h	Height of the vehicle CG from bushing reference point	0.05 m
J_{cx}, J_{cy}, J_{cz}	Moment of inertia of vehicle about x, y and z directions	26, 111, 137 kg m^2
J	Mass moment of inertia of pendulum about z direction	1 kg m^2
J_m	Mass moment of inertia of flywheel	0.1 kg m^2
J_{wy}	Mass moment of inertia of wheel about y direction	2 kg m^2
J_{wz}	Mass moment of inertia of wheel about z direction	2 kg m^2
K	Stiffness	10^5 N/m
K_{sx}, K_{sy}	Bushing stiffness in x and y direction	10^5 N/m
K_{sz}	Bushing stiffness in z-direction	10^9 N/m
K_w	Wheel stiffness in Z-direction	10^6 N/m
l	Length of pendulum	0.5 m
L	Inductance of motor	1.0 H
m	Mass of the pendulum	5 kg
m_c	Mass of the vehicle body	160 kg
r_w	Radius of wheel	0.3 m
R_d	Viscous damping of motor	0.03 Ns/m
R_m	Resistance of motor	2.5 Ω
R_{sx}, R_{sy}	Bushing damping in x and y direction	1.6×10^5 Ns/m
R_{sz}	Bushing damping in z-direction	2×10^5 Ns/m
R_w	Damping in z-direction	10^3 Ns/m
Vv	Voltage to angular displacement conversion gain	118 V/rad
w_m	Mass of wheel	15 kg
θ_{st}	Steering input	0.1 rad/s
θ_{max}	Maximum angle rotation	0.25 rad
μ	Coefficient of friction between tyre and road	0.98
μ_m	Motor torque constant	1 Nm/A

The bond graph simulation results for the Segway are given here by simulating the various conditions.

Case 1: Forward/backward motion of the segway without the application of force at the handle

The results are simulated by providing a constant positive voltage of 70 V (without control) to the motors which produce torque to the wheels and vehicle starts to move in forward direction. If the voltage given is -70 V with the help of inverting amplifier, the vehicle starts to move in the backward direction. The plot for forward linear speed of the vehicle is shown in Fig. 4.38(a) and it indicates that the vehicle accelerates and attains a constant speed of 5.2 m/s within 5s. For the backward motion, the magnitude of speed remains same. It will continue to move until the constant voltage is removed from it. The plot for the angular speed of wheel is shown in Fig. 4.38(b) which shows that the wheel attains a constant angular velocity of 17.3 rad/s. The direction of rotation of wheel will be opposite if negative voltage is provided. Figure 4.38(c) shows the behaviour of the pitch angle with time when the constant positive voltage from the motor is provided to the wheels. Since, there is no stabilization control for pitch angle in this case, the vehicle accelerates; there are some oscillations in the pitch angle about the reference position (inverted position in case of segway only) and when the vehicle attains a constant speed in the forward direction, the pitch angle comes to the fixed value of 1.5 rad from the reference position in the backward or forward direction due to the inertia depending on forward or backward motion of the vehicle.

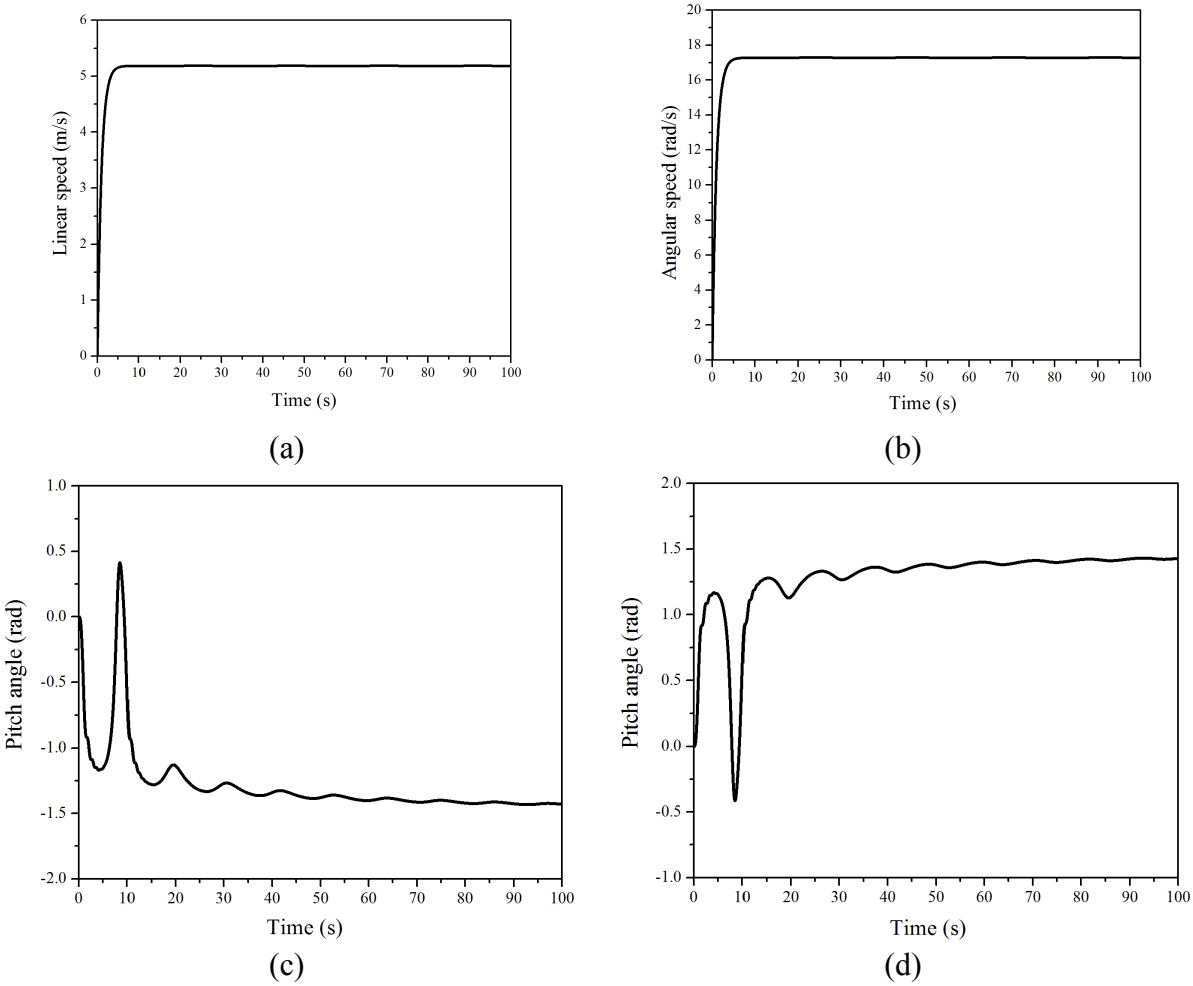
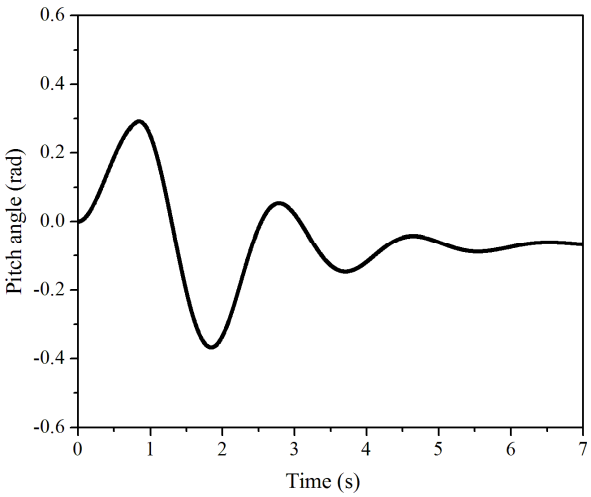


Fig. 4.38 Plot for (a) Linear speed of the vehicle, (b) Angular speed of the wheels, (c) Pitch angle in forward direction and (d) Pitch angle in backward direction

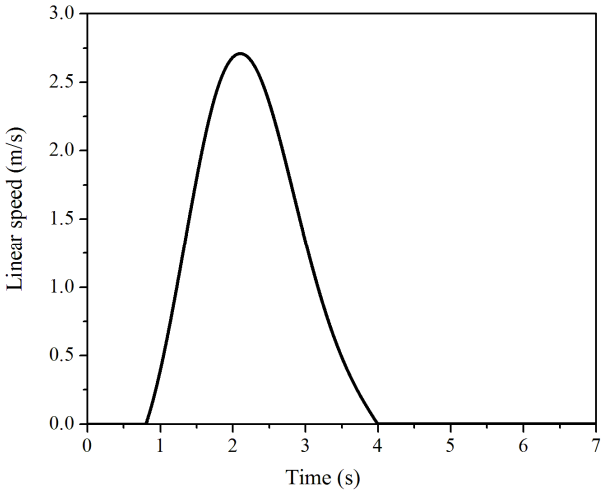
Case 2: Response of the segway with a stabilization controller

The bond graph simulation results for the pitch angle of handle bar/pendulum and speed response of segway are presented in Fig. 4.39, when a constant force is provided to the handle in the forward direction. A constant force of 11 N is provided in the forward direction to the handle (inverted pendulum) of the segway. As the force is provided to the handle in the forward direction, the pitch angle is displaced for its reference position (inverted position). For the vehicle, the control is designed in such a manner that the vehicle will not move in the forward or backward direction to balance the force on the vehicle due to change in the pitch angle until the pitch angle is displaced through 15° from its reference position. Up to the time equal to 0.8s, the vehicle will not move and the

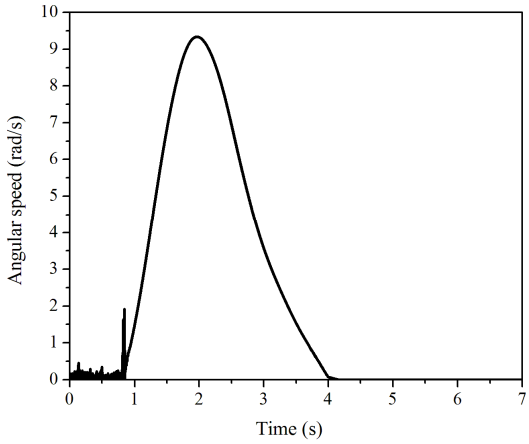
linear and angular velocities of the vehicle will be zero. When the pitch angle is displaced through 15° in the forward direction from the reference position, the stabilization control part (PI controller) will be activated. The voltage signal is applied to the motor to minimize the change in the pitch angle. As a result of this, torque is applied to the two wheels and vehicle starts moving in the forward direction. As the force acts in the forward direction, the vehicle will continue to move forward. Now, at 2s, the force from the handle is removed and PI Controller will try to take the handle to the reference position (inverted position). But due to inertia, the pitch angle is displaced in backward direction due to forward motion of segway. Then due to this change in the pitch angle, segway moves in the backward direction to minimize the change in pitch angle and finally vehicle comes to the rest after 4s and handle will come to its reference position after a little bit oscillations about the reference position in 7s. The plot for pitch angle with time (Fig. 4.39(a)), plot for linear speed of the vehicle with time (Fig. 4.39(b)) and plot for angular speed of the wheel with time (Fig. 4.39(c)) are shown.



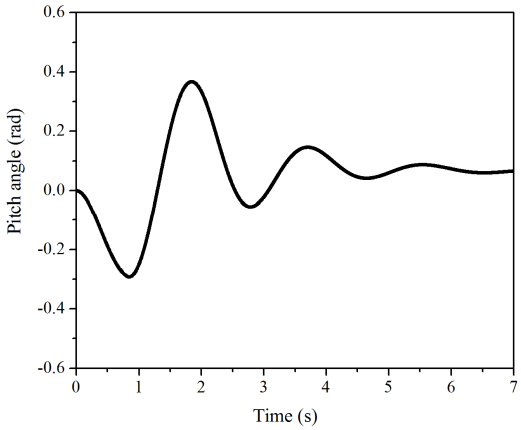
(a)



(b)



(c)



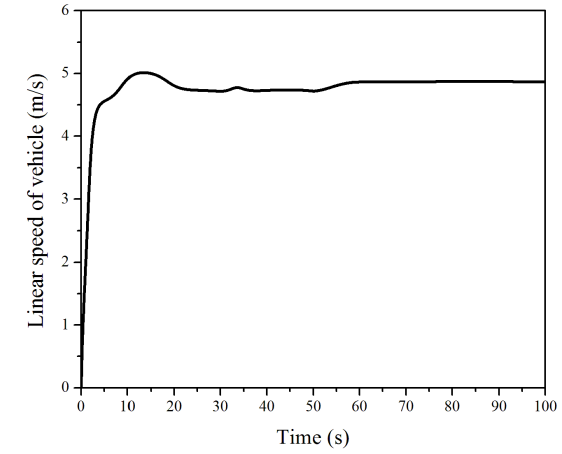
(d)

Fig. 4.39 Plot for (a) Pitch angle, (b) linear speed of the vehicle, (c) Angular speed of the wheel for positive force applied to the handle and (d) Pitch angle for backward force

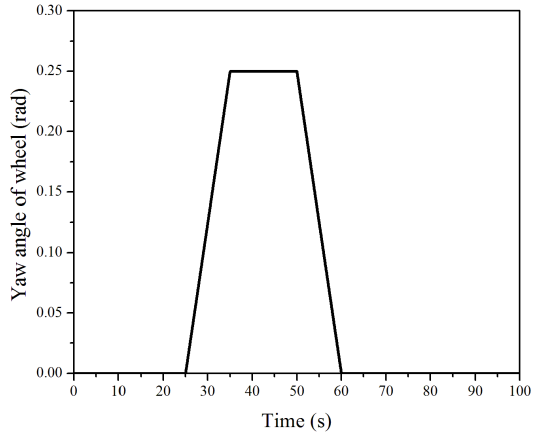
Case 3: Yaw response of segway

The simulation results are shown here in which a constant positive voltage of 68 V is provided to the motors and torque is given to the wheels of the vehicle and vehicle moves in the forward direction with a linear speed of about 5 m/s. Then after 25s, the voltage to the left motor of the wheel is reduced to 62 V unless yaw angle of the wheel reaches to 0.25 rad about the z axis of the wheel. The maximum angle of rotation is limited to the 0.25 rad. Figure 4.40(a) shows the linear speed of the vehicle with time. This plot shows that after 25s, there is decrease in the vehicle speed as the turning motion starts and also there is change in the yaw angle of the wheel (Fig. 4.40(b)). As the yaw angle of the wheel reaches to the 0.25 rad, wheels stops turning further and the vehicle moving with a speed of 4.8 m/s constantly. After 50s, the voltage applied to the left motor is increased from 62 V to 68 V for the return-time. As the steer is returned back, the speed of the vehicle increased up to the earlier speed of 5 m/s. The yaw angle of the wheel also comes to the initial position before steering. The variation of the yaw angle of the vehicle is

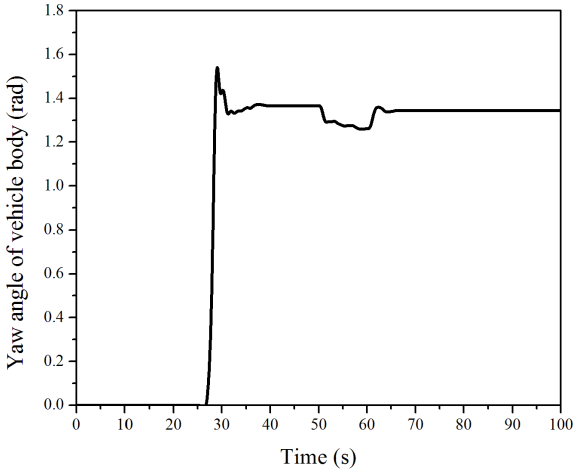
depicted in the Fig. 4.40(c). There are very small changes in the pitch angle during the steering motion as shown in Fig. 4.40(d). The angular speeds of left and right wheels are shown in Fig. 4.40(e, f), respectively. The speed of the left wheel is lower than that of right wheel's speed during turning motion.



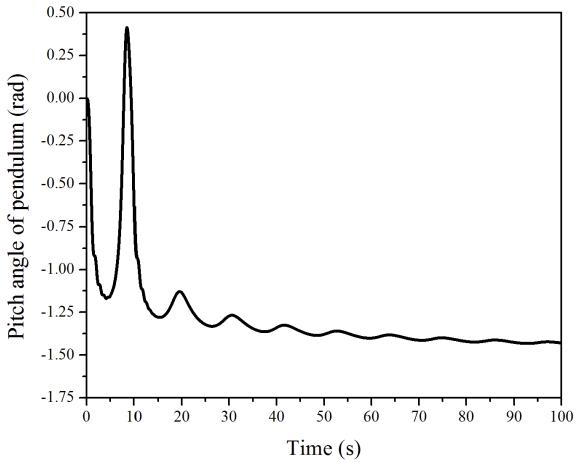
(a)



(b)



c)



(d)

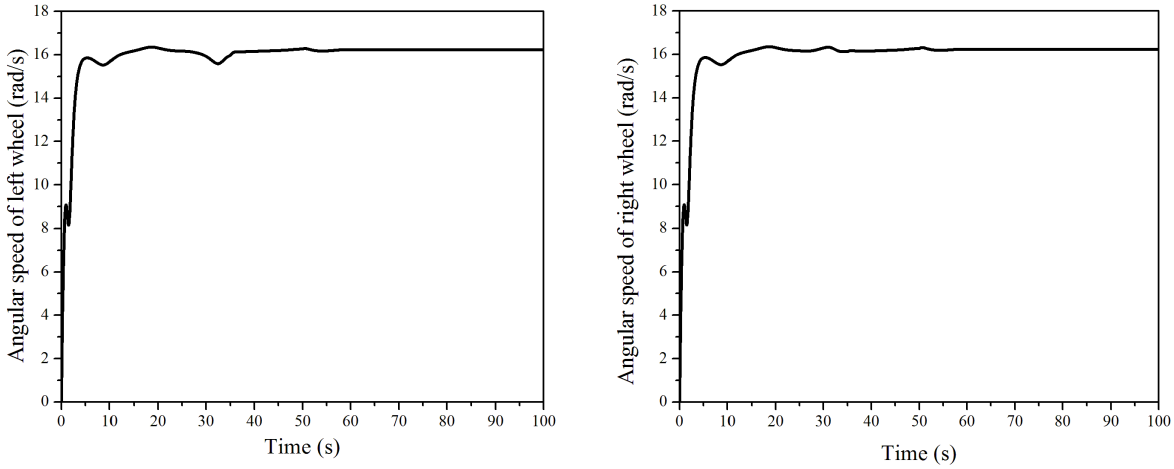


Fig. 4.40 Plot for (a) Linear speed of vehicle, (b) Yaw angle at the centre, (c) Yaw angle of the vehicle body, (d) pitch angle of vehicle, (e) Angular speed of left wheel and (f) Angular speed of right wheel during steering

4.5 Conclusions

In this chapter, the modelling of quadruped robot based on Jansen walking mechanism was done. The CAD model of the quadruped robot is developed and imported into *SimMechanics* model. The path traced by the joints of the Jansen mechanism based leg model is validated with the prototype model of the robot leg. The prototype model of the complete Jansen linkage based quadruped robot is fabricated. Further, the proposed quadruped model is used for the obstacle avoidance application. The fuzzy logic controller is developed for the robot to avoid the obstacles (preferable static in nature). The proposed controller is implemented on the prototype model. The closed boundary environment is considered for validation of the obstacle avoidance controller. The path travelled by the robot in simulation is compared with the experimental work and this is quite satisfactory. This fuzzy logic controller is suitable for the distances of the mobile robot either from the target or from the obstacle ranging from small, medium to large. The future scope of the work includes the dynamic obstacle avoidance along with trajectory tracking for the Jansen mechanism based quadruped robot.

Then, the eigenvalue sensitivity method based model reduction in physical domain is presented and applied on a bicycle vehicle and four-wheel vehicle model. The energy

storage elements are identified from two effect matrices. The eigenvalue sensitivity matrix provides a reduced model for any time scale based on selection of eigenvalues. The four-wheel vehicle model was reduced using the eigenvalue sensitivity method. Non-linear components were linearized by assuming a straight line motion. The process led to reduction of four I-elements, four C-elements and four R-elements, reducing the model order by eight. The simulation time and the number of iterations are reduced using the reduced model even though the system dynamic remains exactly unchanged. Two-wheel and four-wheel mobile robot with model order reduction can be done for less time requirement during controller implementation which eventually avoids too much delayed feedback. This will be helpful for wheeled mobile robot during navigation in presence of static and dynamic obstacles. This work can be considered as future work.

The logic based Heuristic controller has been shown to be capable of swinging up the pendulum from the downward equilibrium position to the upright, unstable equilibrium position by the use of voltage as a control input. The PI control has been demonstrated as an effective control technique for stabilizing the pendulum rod at the inverted unstable equilibrium position. The bond graph simulation results for the swinging up and stabilizing control of cart pendulum system has been validated through the experimental results. After the successful modelling and control of linear inverted pendulum system through bond graph, the application of the inverted pendulum system i.e. Segway has also been modelled in bond graph domain and control system has been designed on the basis of the stabilizing control of inverted pendulum system. The PI control is well demonstrated to be capable of measuring the change in the pitch angle from the reference position (inverted position) and generates a control input signal for forward and backward motion of the segway to minimize that change.

Chapter 5

Walking Mechanism of Mobile Robot

5.1 Introduction

The legged robots have become quite popular because of their ability to walk in terrains that are inaccessible to humans and wheeled robots. Their ability to walk like animals makes it easy for them to walk on difficult terrains. While designing the walking robot, one needs to develop a model that provides the desired locomotion to the robot. The benefit of these robots are that the robots don't roll on surface instead they use footsteps to move and this helps them to overcome obstacles easily, as compared to the wheeled robots that traverse according to terrain. Further, legs can bend like knees at their joints and this helps the body of the robot to move up and down. In addition, these robots can move in every direction i.e. they can move ahead, left, right and can turn also.

The structure of this chapter is as follows: first, the walking mechanism for quadruped robot and bipedal robot with fault detection and isolation is proposed. The BG based model of quadruped is developed and simulated with different gaits. Later, the fault detection and isolation for walking mechanism of bipedal robot is developed. The fault accommodation by reconfiguration is considered for biped-legged robot if it faces an actuator failure fault. Then an on-board real-time fault diagnosis for locomotion of legged robot is considered. The results for fault detection, isolation and reconfiguration from experimentation are compared with the simulation results. At last, PI based controller is created to control the walking mechanism of the quadruped robot and is also validated for application of side ramp walking.

5.2 Bond Graph Modelling and Simulation of Quadruped Robot

The design of the legged robots is a very complex task. A legged robot has large DOF that increase its complexity and as a result of this, weight as well as price increase. In addition, it has to deal with its varying dynamics during up and down motion of the legs and foot's contact with the ground.

5.2.1 Model of Quadruped Robot

Leg model is developed for quadruped robot. The modelling of the legs needs special attention as they control the forward motion of the robot as well as different kinds of manoeuvring motion. Different gaits are also achieved by coordinating the movement of the legs in a certain manner. The legs considered here have two DOF (single DOF for each link) i.e. rotation of each link about y-direction (β). The first link (connected with the body) transfers its rotational motion to the second link. Further, the second link (link with ground contact) has its own rotation and the traction force acts on it due to its contact with the ground. In this section, the modelling of links, modelling of the complete leg and final modelling of quadruple robot were done. The schematic representation of the quadruped robot is illustrated in Fig. 5.1.

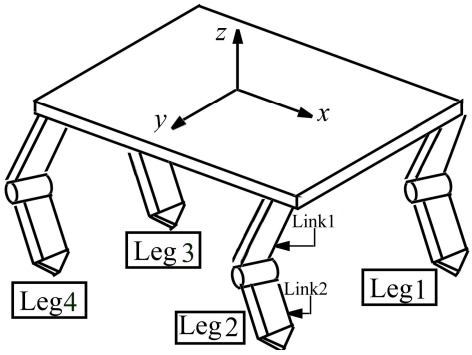


Fig. 5.1 Schematic representation of quadruped robot

5.2.1.1 Modelling of Leg

For modelling of legs, the links of the leg are first modelled. Each link gets its power supply from a DC motor connected to each link. The current supply to the motor is modulated in such a way that it gives the desired locomotion to the legs. The most important aspect is that the modulated power is supplied to the link in the desired form for the accurate movement of the links. BG models for both the links are same as shown in Fig. 5.2. Similarly, the rotational speed is also differentiated before being applied to the link's mass moment of inertia.

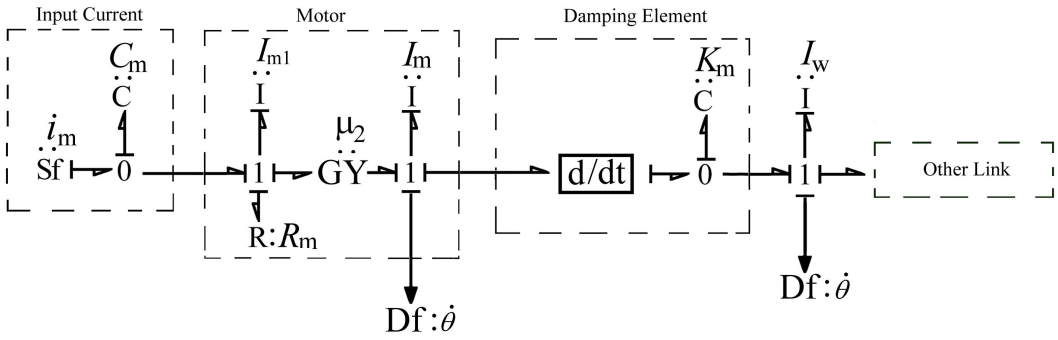


Fig. 5.2 Bond graph model of a single link of leg

The source of flow (Sf) element here represents the input current i_m to the motor for rotation of the legs. The effort (voltage) from 0-junction is then passed to the inductance connected at 1-junction with a resistance R_m that is in series with the gyrator element GY having modulus of μ_2. The gyrator element (GY) converts the flow into the effort i.e. the current to torque. 1-junction to which I-element and flow sensor D_f is connected represents rotor speed of the motor. Here, I_m is the mass moment of inertia of the motor. The sensor D_f senses the angular velocity $\dot{\theta}$ of the rotor and then integrates to give angular position. To convert the angular velocity into the effort input for a link a 0-junction is added with compliance element K_m. But, since K_m is an integrator element, there is need to differentiate the angular velocity. Thus, differentiator is added before K_m. The differentiator along with the stiffness connected at 0-junction is equivalent to mechanical damping. Thus, the effort (torque) received goes to 1-junction to which an inertial element and a flow sensor are attached. I_m is mass moment of inertia of the leg and the flow sensor gives angular velocity of the leg. Once both the links are modelled, they are joined together by a joint stiffness to frame the complete model of the leg Fig. 5.3(a). The bond graph for the full leg model is shown in Fig. 5.3(b).

5.2.1.2 Kinematics Relations

Kinematics where the motion is illustrated in the form of velocity and acceleration of its components is the review of the components of a mechanical system. The components can be joined through various types of joints, thus constraining movement of the components with respect to each other. Each component of the mechanical system is

assumed as a rigid body. The body fixed frame moves with the body i.e. it will move relative to inertial frame whereas the inertial frame is fixed. This is generally a simple problem that the coordinates chosen for defining the system are function of a variable.

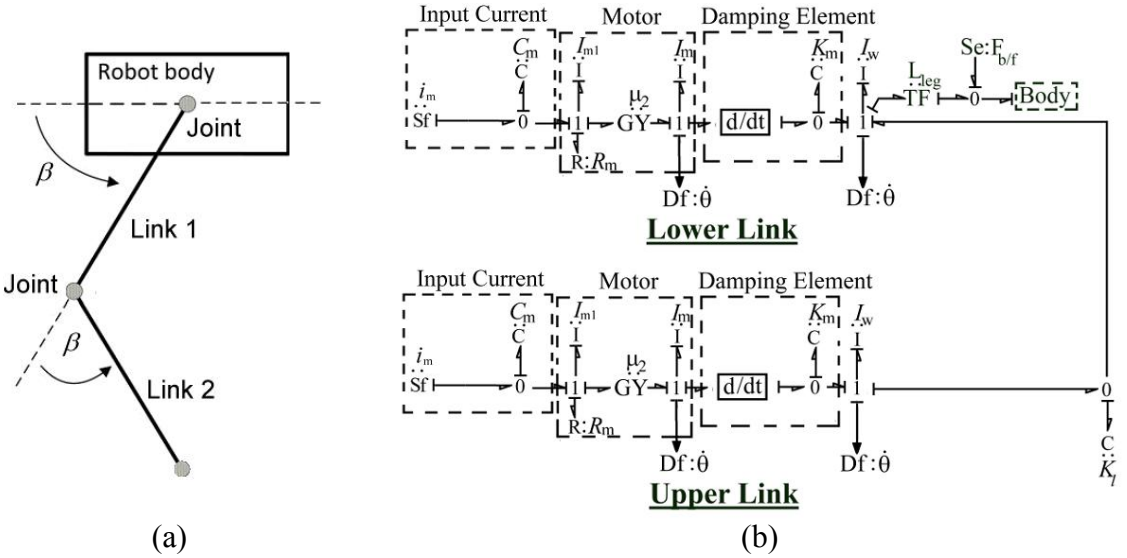


Fig. 5.3 (a) Schematic diagram of legs arrangement and (b) BG model of complete leg

The kinematic relations discussed here are similar to that of vehicle model. From these relations, the motion of the centre of gravity (CG) of body [Bera *et al.*, 2011] is deduced when the robot moves with its four legs. The front leg's velocities that are normal and tangential to the rotation axis are

$$v_{1n} = (v + \omega l_1)\cos\delta - u\sin\delta \tag{5.1}$$

$$v_{1t_L} = (v + \omega l_1)\sin\delta + u\cos\delta - c\omega \tag{5.2}$$

$$v_{1t_R} = (v + \omega l_1)\sin\delta + u\cos\delta + c\omega \tag{5.3}$$

where v_{1t_L} = Leg1 (front left leg), v_{1t_R} = Leg2 (front right leg), c = distance of legs from CG of body in y-direction and δ is the steering angle. The rear leg's velocities that are normal and tangential to the rotation axis are

$$v_{2n} = (v - \omega l_2) \tag{5.4}$$

$$v_{2t_L} = u - c\omega \tag{5.5}$$

$$v_{2t_R} = u + c\omega \tag{5.6}$$

where v_{2t_L} = Leg3 (rear left leg), v_{2t_R} = Leg2 (rear right leg). From Newton-Euler equations

$$m_v \dot{u} = m_v \omega v + \sum F_x \tag{5.7}$$

$$m_v \dot{v} = -m_v \omega u + \sum F_y \tag{5.8}$$

5.2.1.3 Bond Graph Modelling

Based on the kinematic relations deduced in the previous section, the bond graph model has been made as shown in Fig. 5.4. The ports described by the circles (1, 2, 3, and 4) represent the legs of the robot. The BG for the same has been shown in Fig. 5.3. The sensors D_f are attached to 1_u, 1_v and 1_ω to measure the velocities as well as position of the centre of the body.

5.2.2 Joint Rotation for Development of Motion

The joint rotation of both the links is an important aspect of modelling of legs. It has been observed in locomotion of animals that the first link which is connected to the body always moves forward, whereas the lower link of leg first goes backward and then moves forward. Both the links reach their maximum angular position with the same time and then they together move down. For achieving the same, mathematical formulations and programming are required and this is explained in the forthcoming sections:

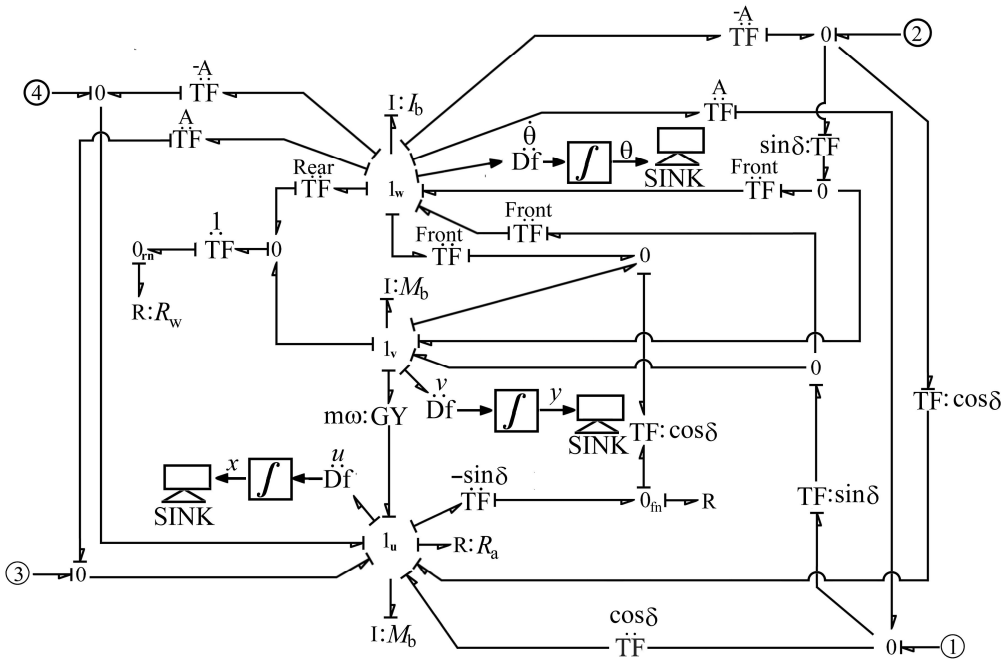
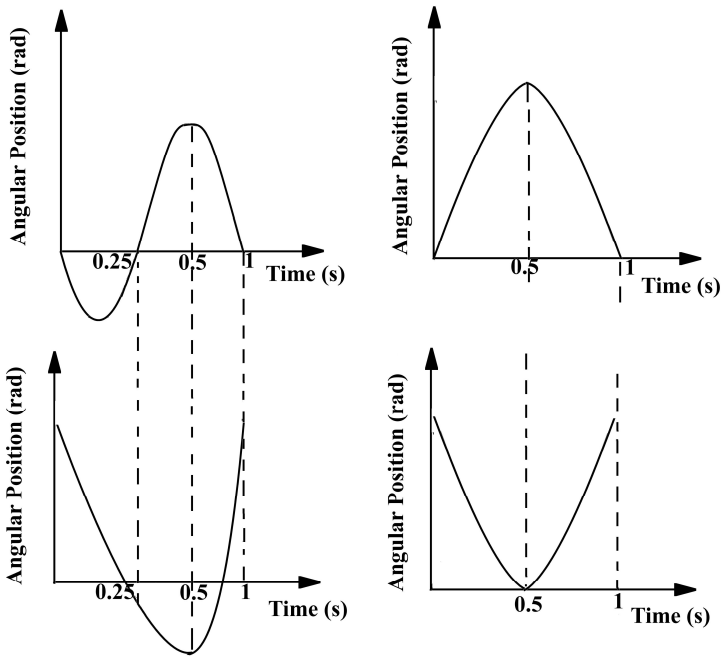


Fig. 5.4 Bond graph model of quadruped robot

5.2.2.1 Mathematical Formulation for Desired Motion of Robot

A mathematical formulation is required to achieve the desired motion for both the links. Four different types of motions of legs at a given time are required while deriving the mathematical formulation. The reason for the same is that there will be a phase difference between the legs, when one leg will come down the other will start rising from the ground to keep the movement on-going. The desired motion of the links is graphically represented in Fig. 5.5. Figure 5.5(a) shows the phase difference between first and second steps of the lower link. It clearly depicts that when the lower link of the first leg is at its peak angular position, the lower link of the second leg is at the ground. Similarly, Fig. 5.5(b) shows the phase difference between first step and second steps of the upper link. A four-degree polynomial has been derived to achieve the desired results

$$\theta(t) = a_1 t^4 + a_2 t^3 + a_3 t^2 + a_4 t + a_5 \tag{5.9}$$



(a) (b)

Fig. 5.5 Motion of (a) Lower link and (b) Upper link for first and second step

Thus, it is assumed that $\theta(0)=\theta_0, \theta(0.25)=\theta_1, \theta(0.5)=\theta_2, \theta(0.75)=\theta_3, \theta(1)=\theta_4$ and putting above values in Eq. (5.9), representation in matrix form is

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0.25^4 & 0.25^3 & 0.25^2 & 0.25 & 1 \\ 0.5^4 & 0.5^3 & 0.5^2 & 0.5 & 1 \\ 0.75^4 & 0.75^3 & 0.75^2 & 0.75 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}^{-1} \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \end{bmatrix} \tag{5.10}$$

Using the values from Table 5.1 and solving the Eq. (5.9) in *Matlab* software, the values of a_1, a_2, a_3, a_4, a_5 are obtained as

$$\text{SF1} = -4.1045(t)^4 + 6.3509(t)^3 - 3.1115(t)^2 + 0.8651(t) + 0.1047 \tag{5.11}$$

$$\text{SF2} = 1.8198(t)^4 - 16.2177(t)^3 + 20.0207(t)^2 - 5.6228(t) + 0.1047 \tag{5.12}$$

$$SF3 = 4.4651(t)^4 - 6.8821(t)^3 + 3.2073(t)^2 - 0.7903(t) + 0.3839 \quad (5.13)$$

$$SF4 = 21.7451(t)^4 - 36.2261(t)^3 + 19.3993(t)^2 - 4.9183(t) + 0.8832 \quad (5.14)$$

where SF1 and SF3 are the current supply to the motors of lower link for the first and second steps, respectively; SF2 and SF4 are the current supply to the motor for upper link for the first and second step, respectively.

Table 5.1 Joint rotation (rad) for motion

Time period	Link 1 first step Angles (θ)	Link 2 first step Angles (θ)	Link 1 second step Angles (θ)	Link 2 second step Angles (θ)
$t = 0$	0.1047	0.1047	0.3839	0.8832
$0 < t \leq 0.25$	0.2097	-0.2960	0.2967	0.3850
$0.25 < t \leq 0.5$	0.2967	0.3850	0.2094	0.1047
$0.5 < t \leq 0.75$	0.3839	0.8832	0.1047	-0.2960
$0.75 < t \leq 1$	0.1047	0.1047	0.3839	0.8832

5.2.2.2 Development of Program for Continuous Motion

The mathematical formulation for the motion of leg is already developed in the last section. The robot follows this motion during its total time period. As the above formulation has each step for 1s, it is desired that each step should repeat itself after completion of 1s. The motion of quadruped animals like horse has motivated to use the similar gait pattern for motion of quadruped robots. The walking speed of horse is approximately 1.8 km/h or 0.5 m/s. As it is seen from Table 5.3 that average speed of robot with bound and pace gait is approximately 0.5 m/s, the sampling time is considered as 1s for all the gait patterns used for motion of quadruped robots. Thus, after mathematical formulation, it is required to program the motor input so that it keeps on repeating itself until the supply is stopped. Let the time period for each step $= T$ and simulation time $= t_1$.

$$t = t_1 - T \times \text{floor}\left(\frac{t_1}{T}\right) \quad \forall \; t_1 \rangle T \quad (5.15)$$

$$t = t_1 \quad \forall \; t_1 \leq T \quad (5.16)$$

Thus, the polynomial (replacing Eq. (4.15) or Eq. (4.16) in Eq. (5.9)) will repeat after completion of each step. The values of a_1, a_2, a_3, a_4, a_5 already had been found in the above section.

5.2.3 Parameter Values and Simulation Results

This section discusses the parameters and the results for the quadruped model. The results presented are for different types of gaits.

5.2.3.1 Simulation Results for Trot Gait

In trot gait, the diagonally opposite legs follow the same fashion whereas the other diagonally opposite legs remain on ground as illustrated in Fig. 5.6(a). The trot gait is generally used by horses while racing because it brings the variation in speed using this gait. In the model, Leg1-4 and Leg2-3 are diagonal to each other. Thus, the Leg1-4 moves together during the first step and Leg2-3 move together during the second step. The parameters are shown in Table 5.2.

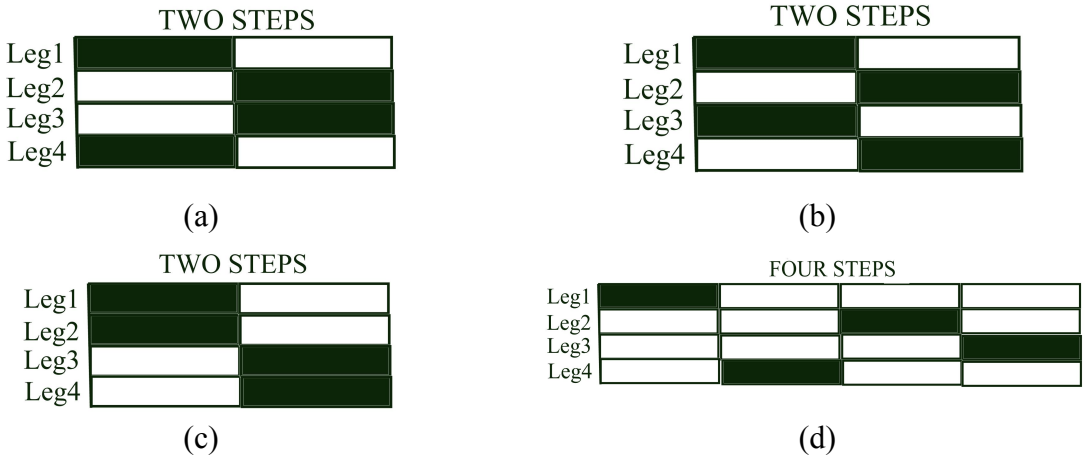


Fig. 5.6 (a) Trot gait, (b) Bound gait, (c) Pace gait and (d) Amble gait

The effort-control of leg by mathematical formulations is given to the links. The *x*-direction on positive side shows the forward direction. The time cycle for each step is 1s. The simulation was run for 10s. The result for trot gait is presented in Fig. 5.7.

Table 5.2 Parameter values for trot gait

Parameters	Description	Values
C_{m}	Capacitance of input circuit (F)	10
I_{b}	Moment of inertia of body (kg m^2)	10
I_{m}	Moment of inertia of flywheel (kg m^2)	1
I_{m1}	Inductance of motor (H)	1
I_{w}	Moment of inertia of leg (kg m^2)	5
$K_{\mathrm{m_{f1}}}, K_{\mathrm{m_{f2}}}, K_{\mathrm{m_{r1}}}, K_{\mathrm{m_{r2}}}$	Compliance element of leg $(\mathrm{N/m})$	10^8
K_{l}	Compliance element of leg $(\mathrm{N/m})$	10
l_{leg}	Length of complete leg (m)	0.6
l_1, l_2	Distance of front and rear leg from CG (m)	0.5
M_{b}	Mass of body (kg)	600
R_{m}	Resistance of motor (Ω)	0.1
μ_1	Motor constant for front leg's link 1	2.1
μ_2	Motor constant for front leg's link 2	0.7
μ_3	Motor constant for rear leg's link 1	1.95
μ_4	Motor constant for rear leg's link 2	0.64

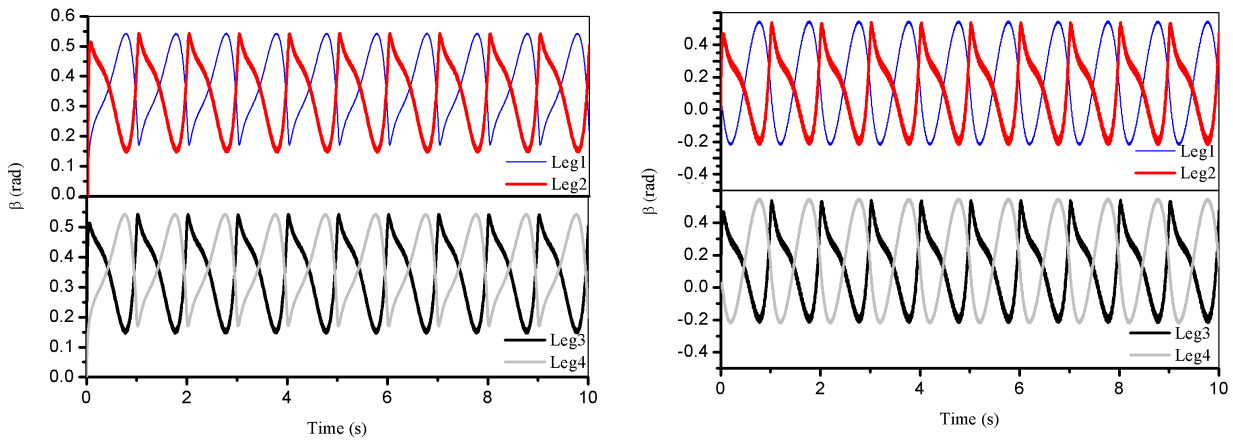


Fig. 5.7 (a) Rotation of link 1 and (b) Rotation of link 2

Figure 5.7(a) and (b) show the rotation of link 1 and link 2 for all the four legs. The graphs clearly show that Leg1-4 and Leg2-3 move together. The displacement in *x*-direction is 6.25 m and there is no displacement in *y*-direction and this depicts that the robot is moving in straight path. The longitudinal velocities of quadruped robot with different gaits are illustrated in Fig. 5.8.

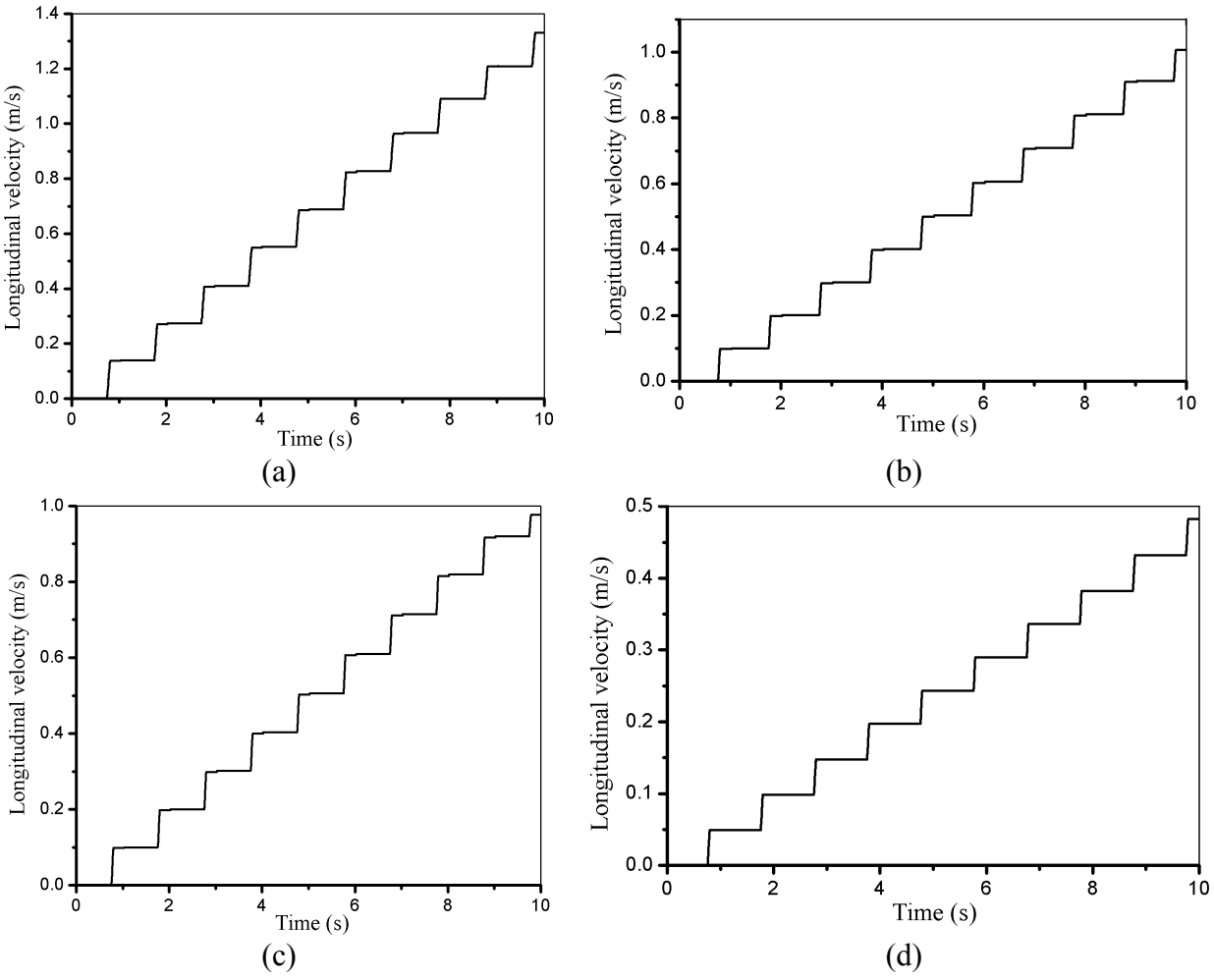


Fig. 5.8 Velocities for (a) Trot gait, (b) Pace gait, (c) Bound gait and (d) Amble gait

5.2.3.2 Simulation Results for Bound Gait

In bound gait, the front and rear right leg move together and front and rear left leg move forward together. The animal generally uses this type of gait when it has to balance its weight while moving forward. This gait is generally used by wide-bodied animals. In the

model, Leg1-3 and Leg2-4 are on left and right side, respectively with respect to the body frame. Thus, Leg1-3 move together taking first step and Leg2-4 move together during second step. The parameters are same as given in Table 5.2. The time cycle for each step is 1s. The simulation was run for 10s. The results for bound gait are shown in Fig. 5.9. The rotation of link 1 and link 2 of all the four legs are illustrated in Fig. 5.9(a) and (b), respectively. The graphs clearly show that legs 1-3 and legs 2-4 move together. The displacement in *x*-direction is about 4.78 m and there is no displacement in *y*-direction.

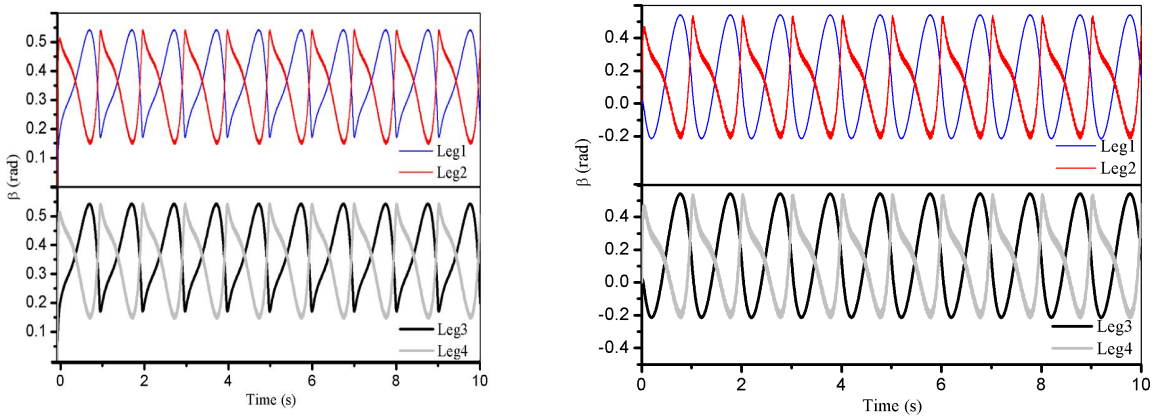


Fig. 5.9 (a) Rotation of link 1 and (b) Rotation of link 2

5.2.3.3 Results for Pace Gait

In pace gait, both the front legs move together and the rear legs move together. This gait is also called gallop gait. Rabbits and rodents generally use this type of gait. In this model, Leg1-2 and Leg3-4 are front and rear legs, respectively. Thus, the Leg1-2 move together during first step and Leg2-4 move together during second step. The parameters are same as given in Table 5.2. The effort-control of legs by mathematical formulation is given to the links. The time cycle for each step is 1s. The simulation was run for 10s. The results for pace gait are illustrated in Fig. 5.10. Figure 5.10(a) and (b) show the rotation of link 1 and link 2 of the all the four legs. The graphs clearly show that Leg1-2 and Leg3-4 move together. The displacement in *x*-direction is 4.76 m.

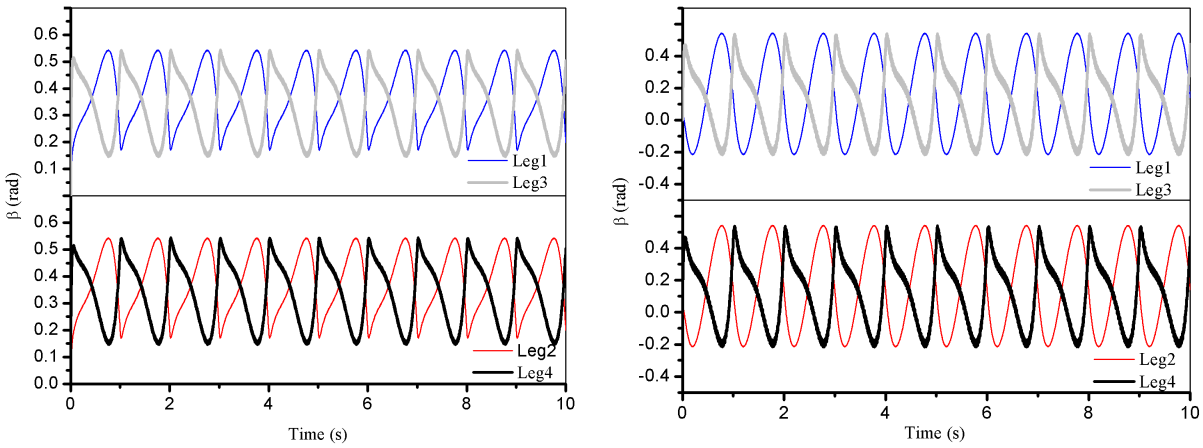


Fig. 5.10 (a) Rotation of link 1 and (b) Rotation of link 2

5.2.3.4 Results for Amble Gait

The statically balanced gait is also known as amble gait. In amble gait, only one leg moves forward and rest of the legs remain on the ground. The speed of this gait is slow. But, this is the most stable gait as the CG of the body always remains inside the triangle formed by the legs which are on the ground. The order of movement of legs can be anything in according to the requirement of terrain and situation. The amble gait following the order of 1-4-2-3 is shown in Fig. 5.6(d). The parameters are same as given in Table 5.2. The results for amble gait are shown in Fig. 5.11 and Fig. 5.12. Figure 5.11 and Fig. 5.12 show that Leg1 moves first and then comes down, meanwhile legs 2, 3 and 4 are on ground giving support to the robot. After 1s, Leg1 touches the ground; Leg4 starts its step, while Leg1, 2 and 3 stay on the ground. The Leg2 and Leg3 start at 2s and 3s, respectively. The above results clearly show that for each step taken by the robot, three legs always stay on the ground. The above result shows that there is a displacement of 2.28 m in *X*-direction.

Figure 5.13 shows the comparison of displacement achieved for four gaits. It can be clearly seen that trot is fastest, pace and bound gait are almost equal and amble gait is the slowest. These results completely validate the theory that states that dynamically stable gaits are faster than statically stable gait and also, the trot gait is the fastest. The comparative energy efficiencies for all the gaits in term of distance covered (%) with respect to trot gait are given in Table 5.3.

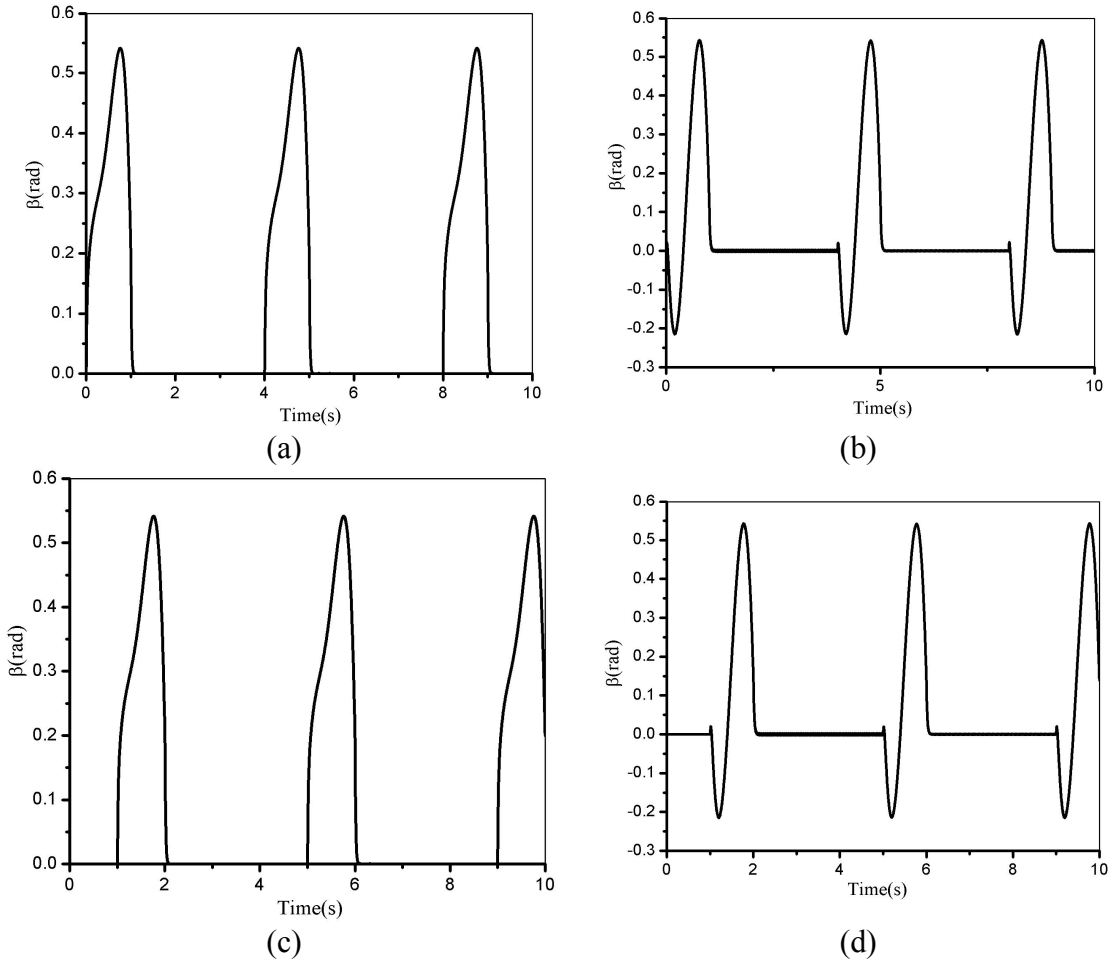
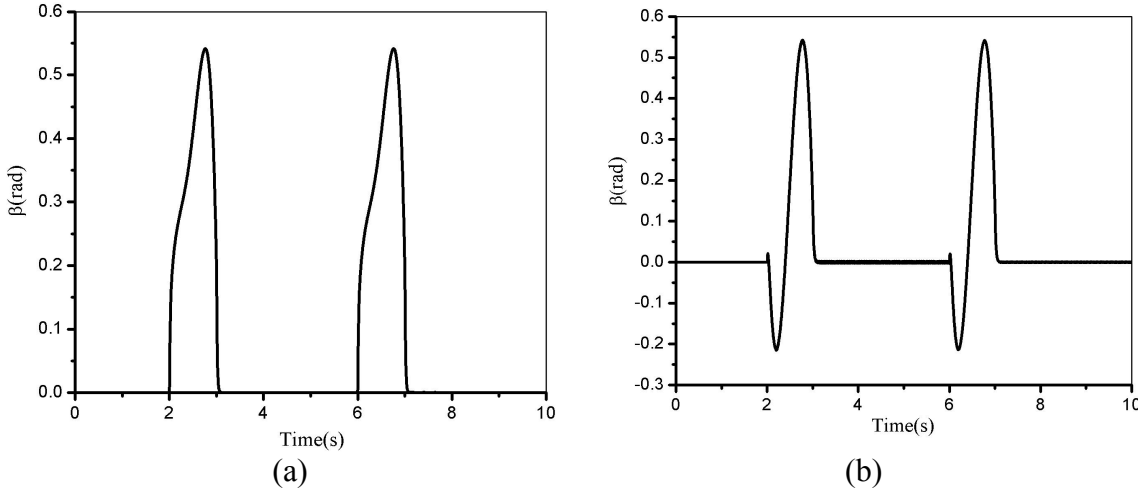


Fig. 5.11 (a) Rotation of link 1 of Leg1, (b) Rotation of link 2 of Leg1, (c) Rotation of link 1 of Leg4 and (d) Rotation of link 2 of Leg4



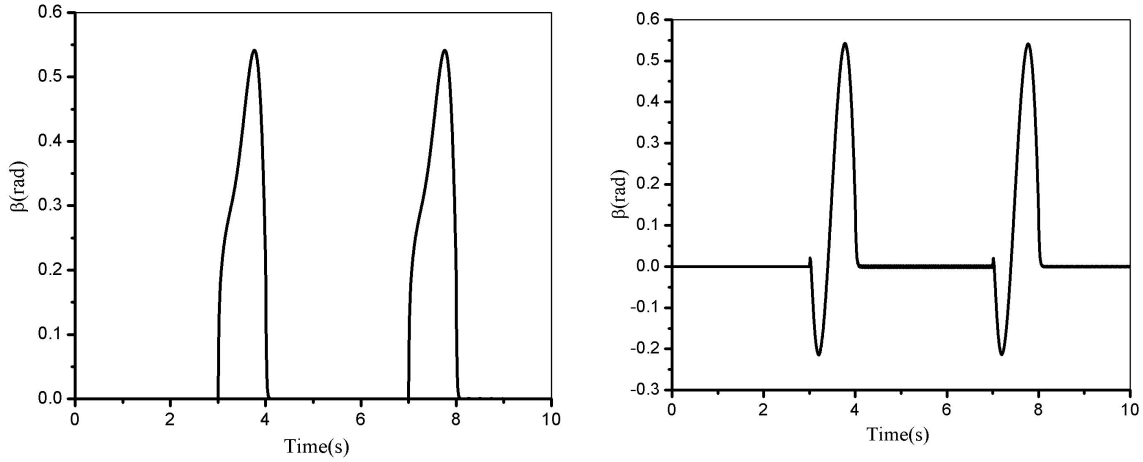


Fig. 5.12 (a) Rotation of link 1 of Leg2, (b) Rotation of link 2 of Leg2, (c) Rotation of link 1 of Leg3 and (d) Rotation of link 2 of Leg3

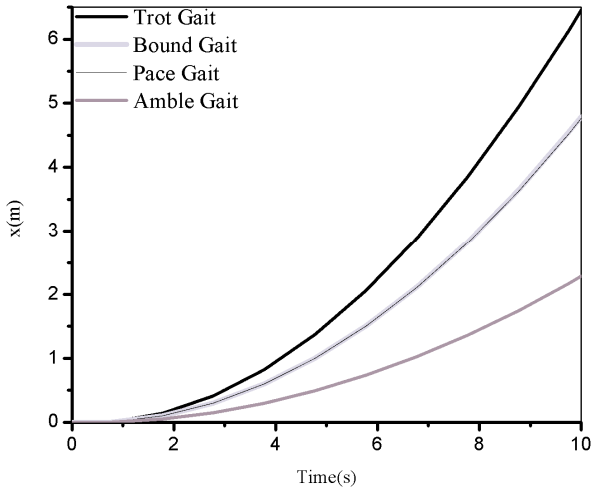


Fig. 5.13 Comparison of displacement of body during four gaits

5.3 Fault Detection, Isolation and Reconfiguration of a Bipedal-Legged Robot

This work deals with the fault detection, isolation (FDI) and reconfiguration of the locomotion of a bipedal robot. Initially, the planar model of the legged robot in the vertical plane is developed using bond graph (BG) approach. Then, the planar BG model of the legged robot is extended to the 3-D legged robot. Two individual motors are used to actuate the prismatic leg of the robot for the locomotion. The BG simulation results for

straight walking based on oscillating cylinder mechanism and turning motion of the legged robot are discussed. The prototype model of the legged robot is also developed and experimentation is done for straight and inclined plane applications. Finally, fault detection and isolation (FDI) technique for the 3-D model of legged robot is proposed for the generation of fault indicators i.e. analytical redundancy relations (ARR) in the presence of the system failure. The ARRs are derived from the BG model of the legged robot during the occurrences of the fault. The experimental results are validated with the simulation results for fault detection, isolation and reconfiguration when the robot manoeuvres in a U-shaped path. The real-time fault diagnosis and reconfiguration for locomotion of the legged robot is possible with this FDI approach.

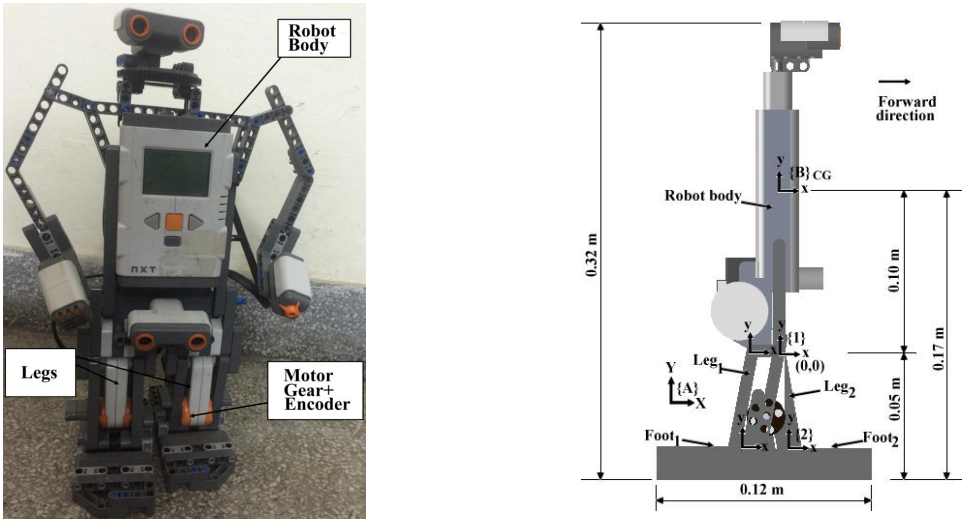
Table 5.3 Comparative energy efficiency in term of distance covered (%) w.r.t. trot gait

	Time, t (s)	Displacement, X (m)	Velocity, $\dot{X}$ (m/s)	Comparative energy efficiency in terms of distance covered (%) w.r.t. trot gait
Trot gait	1	0.0313	0.1376	100
	5	1.5275	0.6850	100
	10	6.4551	1.3306	100
Pace gait	1	0.0223	0.0986	71.2
	5	1.1135	0.5003	72.8
	10	4.7665	1.0064	73.7
Bound gait	1	0.0223	0.0986	71.2
	5	1.1135	0.5003	72.8
	10	4.7665	1.0064	73.7
Amble gait	1	0.0112	0.0492	35.7
	5	0.5476	0.2435	35.8
	10	2.2867	0.4824	35.4

5.3.1 Modelling of the Walking Robot

The dynamic modelling of the legged robot is essential to implement the walking of the robot. It is assumed that the robot is made to walk on a hard surface. The physical model of the legged robot is illustrated in Fig. 5.14(a). Figure 5.14(b) illustrates the schematic representation of it (inertial frame {A} and body frame {B} are connected to the CG of the robot). The frame {1} of the leg is attached to the robot and is modelled by considering as oscillating cylinder mechanism. The cylinder-end is joined with the robot

body and the piston end is kept free to move. The frame {2} is used to provide the rotation between the leg and the foot. The rotational inertia is defined at the CG of the leg.



(a) (b)

Fig. 5.14 (a) Physical model and (b) Schematic representation of legged robot

5.3.2 Modelling of Planar Robot

The forward motion of the robot is considered along *x*-direction. The walking phenomenon of the robot is based on the oscillating cylinder mechanism as shown in Fig. 5.15. The piston end is attached to the motor through the crank and the fixed end of the cylinder is attached to the body of the robot. The foot is attached to the end (point F) of the extended part of the connecting rod and the piston reciprocates within the oscillating cylinder (robot legs). The walking action takes place due to the sliding motion of the piston inside the cylinder. The motor controls the angular displacement of the crank element that induces the reciprocating motion to the piston rod inside the cylinder. The sliding motion of the piston that is one of the most complicated multi-body elements develops a non-linear behaviour. The BG approach is used to model such a multi-body system. A schematic diagram of prismatic-link is illustrated in Fig. 5.16 [Bera *et al.*, 2009].

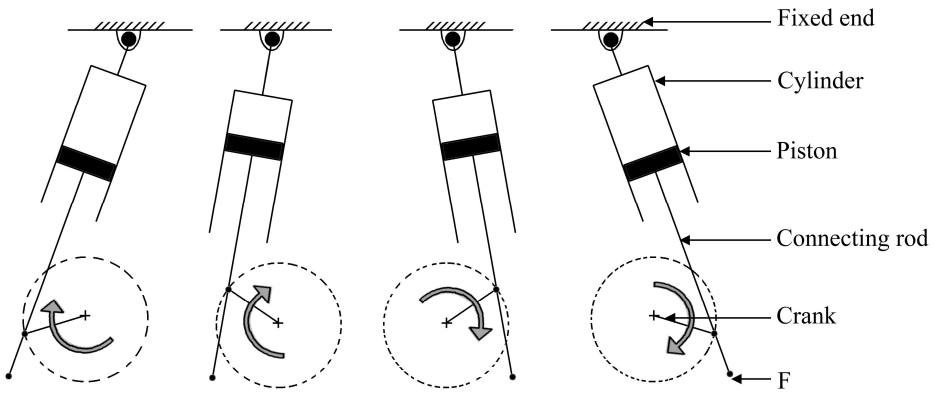


Fig. 5.15 Different positions of robot leg considered as oscillating cylinder mechanism

The local co-ordinate frame of the centroid of piston and cylinder are (x_p, y_p, z_p) and (x_c, y_c, z_c), respectively and the local coordinate frames are collinear with the principal axes. The instantaneous length l is the distance from piston endpoint to cylinder assembly. The CG of the cylinder and piston is placed at a distance of l_{cg}, l_{pg} from the fixed end respectively. d and l_p denote the thickness of the piston and distance of piston from the rod end, respectively.

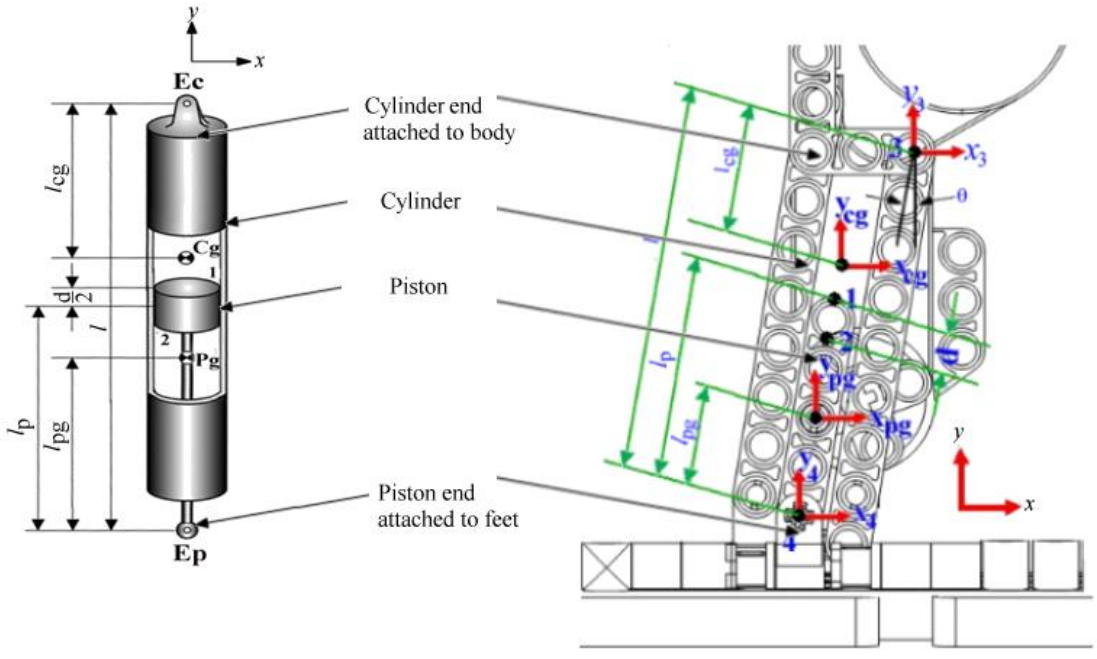


Fig. 5.16 Schematic diagram of the prismatic link [Bera *et al.*, 2009] and robot leg

5.3.2.1 Planar Leg Model

The planar prismatic leg model is developed in *x*-*y* vertical plane using BG and is simulated for walking along straight and inclined plane applications. First, the word BG is discussed and then complete BG model is developed. Figure 5.17 illustrates the word BG of the legged robot. In the word BG, the robot is divided into five sub-components i.e. robot body, left leg, right leg, motor for left leg and motor for right leg. The coordinate transformation (CTF) block is used to transform the coordinate from local to global frame. The international standard norms of the word bond graph are considered for presenting Fig. 5.17. The full-arrow bond (signal bond) is used to express the unidirectional signal and this is similar to the bond of the signal flow graph. The half-arrow bond (power bond) is used for bi-directional signal in bond graph lexicon i.e. effort signal in one direction and the flow signal in other direction. The half arrow bond with a circle indicates a vector bond i.e. bonds for different directions (*x*, *y*, *z*) is expressed by a single bond.

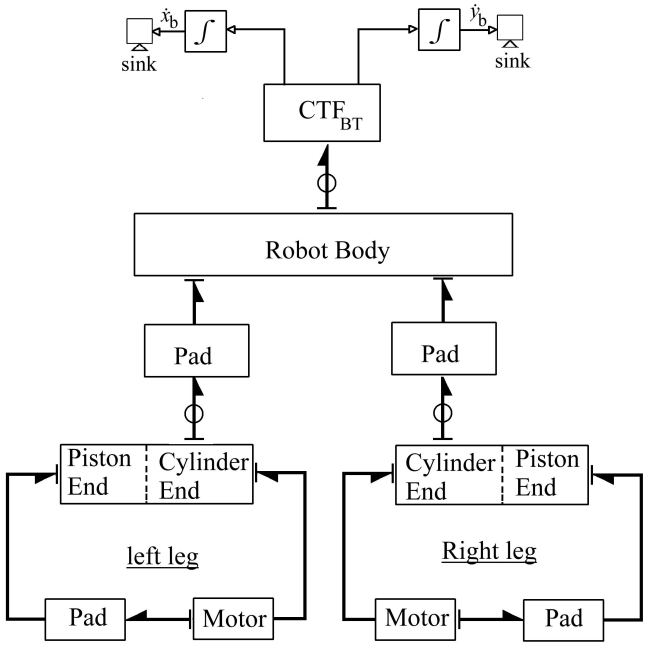


Fig. 5.17 Word bond graph of the legged robot model

The design of a robot leg is similar to the design of oscillatory cylinder mechanism. The cylinder end acts as a pivot joint and is joined to the robot body and other end (piston

rod end) is connected to the foot of the robot. The piston slides inside the cylinder and this causes the forward or backward motion of the cylinder. The motor generates the rotational motion and this rotational motion is converted into linear motion of the piston with the help of crank and connecting rod. Figure 5.16 illustrates a planar piston-cylinder model. The ends velocities, E_c (x_1, y_1) and E_P (x_2, y_2) of the piston and cylinder are described in terms of the velocities of the cylinder and piston, respectively and the rotational velocities about the z-axis. The BG model of any mechanism is developed based on the kinematic equations and the dynamic equations are automatically developed from the BG model. The velocity at any point of the mechanism can be expressed by the velocity of other point using transformer element (TF). The x-velocity $\dot{x}_1$ of cylinder endpoint can be represented by velocities $\left(\dot{\theta}_{cg}, \dot{x}_{cg}\right)$ of the centroid of the cylinder i.e. $\dot{x}_1 = l_{cg}\sin\theta_{cg}\dot{\theta}_{cg} + \dot{x}_{cg}$. Hence, the modulus of the transformer can be expressed as $\mu_1 = l_{cg}\sin\theta_{cg}$. Similarly, other 13 moduli of transformer elements can be obtained using different kinematic equations. The transformer moduli TF are

$$\mu_1 = l_{cg}\sin\theta_{cg}, \mu_2 = -l_{cg}\cos\theta_{cg}, \mu_3 = -\frac{1}{\sin\theta_{cg}}, \mu_4 = \frac{1}{\cos\theta_{cg}}, \mu_5 = \frac{1}{l - l_p - l_{cg} - \frac{d}{2}},$$

$$\mu_6 = \frac{1}{l - l_p - l_{cg} + \frac{d}{2}}, \mu_7 = \frac{x_1 - x_2}{l}, \mu_8 = \frac{y_1 - y_2}{l}, \mu_9 = -l_{Pg}\sin\theta_{Pg}, \mu_{10} = l_{Pg}\cos\theta_{Pg},$$

$$\mu_{11} = -\sin\theta_{Pg}, \mu_{12} = \cos\theta_{Pg}, \mu_{13} = -\left(l_p - l_{Pg} + \frac{d}{2}\right), \mu_{14} = -\left(l_p - l_{Pg} - \frac{d}{2}\right).$$

The mass moment of inertias (J_P, J_c) and masses (M_P, M_c) of piston and cylinder are represented by the I-elements connected to 1-junctions. The modulated transformers with moduli μ_7 and μ_8 are used to calculate the relative velocities of piston and cylinder, respectively at 0-junction. The friction is represented by R-element (r_f). The stiffness (C: k_b) and damping (R:r_p) are used to apply limits on velocities at junctions 0_1 and 0_2 that are placed between the contact points of cylinder and piston. The gravity along y-direction is introduced at 1-junction.

5.3.2.2 Bond Graph Model of Combined Body and Leg

The model of each leg is attached to the body of the legged robot. The pads that are simulated lumped flexibilities are used in BG model wherever differential causality occurs [Pathak *et al.*, 2005]. The linear velocities in directions *x* and *y* are represented by $\dot{x}_B$ and $\dot{y}_B$, respectively. The angular velocity of the robot body about z-axis is shown by $\dot{\theta}_B$. The BG model of robot leg is developed in *SYMBOLS* Shakti software and shown in Fig. 5.18. The complete BG model of legged robot is illustrated in the Fig. 5.19.

5.3.2.3 Simulation Results

The planar robot BG model is simulated for walking along a straight path and an inclined plane. The input parameters related to the simulation are given in Table 5.4. All the parameters other than stiffness and damping parameters are taken from the physical system. The initial conditions for simulation are mentioned in Table 5.5. These initial conditions are same as the experimentation of the legged robot. Figure 5.20(a) shows the displacement of both ends of the leg along *X* and *Y* directions, while Fig. 5.20 (b, c) show the displacement of both the legs along *Y*-direction. The maximum displacement along *Y*-direction is about 1 cm. Each leg moves alternately and this is in quite similar to the walking pattern of human. Figure 5.21 shows the simulation of each leg while walking on the inclined plane. Figure 5.21 (a, b) show the line diagram of the robot and ramp arena, respectively. The simulation results of end displacement in *Y*-direction of robot leg while moving on the ramp is shown in Fig. 5.21 (c).

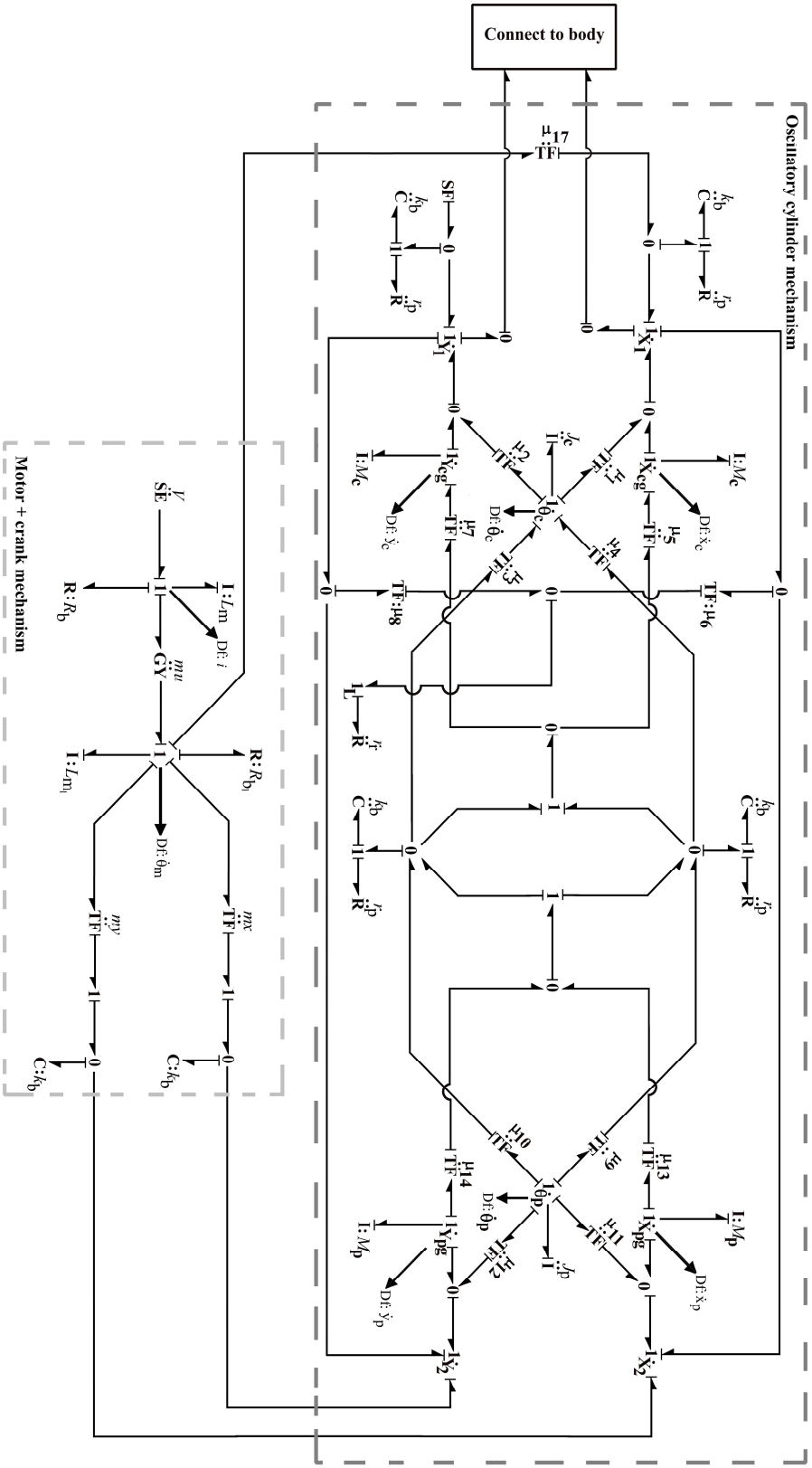


Fig. 5.18 Bond graph model of planar left leg (oscillatory cylinder mechanism)

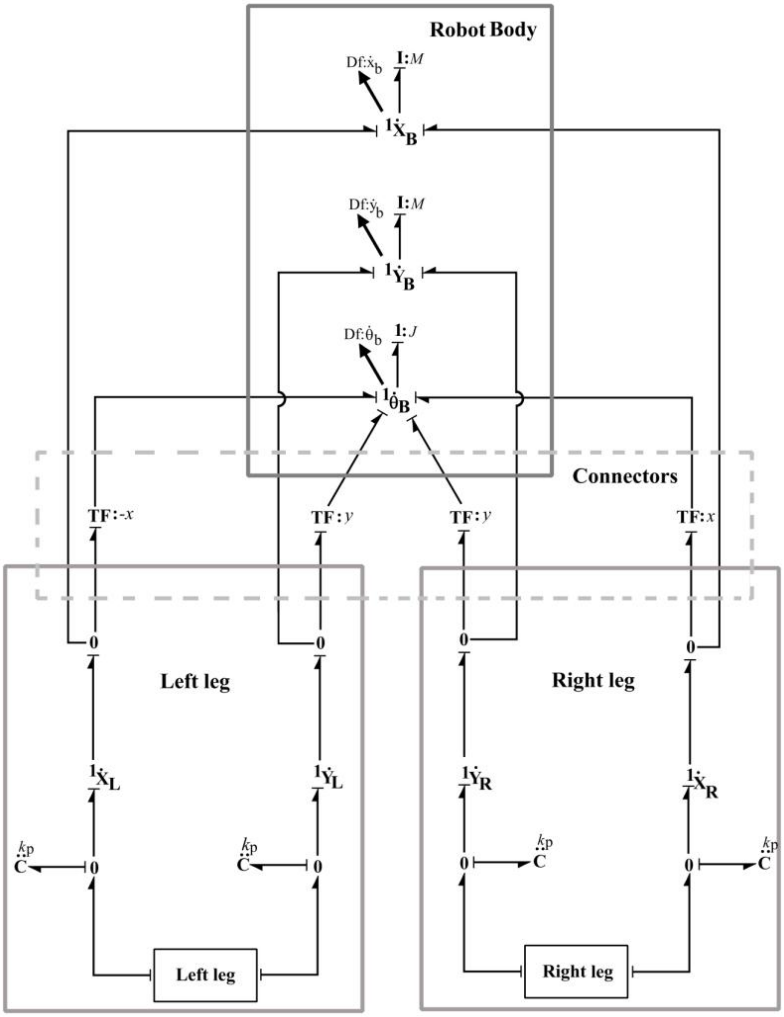


Fig. 5.19 Complete bond graph model of the planar legged robot

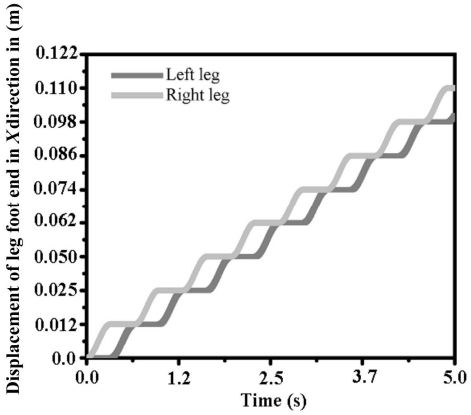
Table 5.4 Parameter values for simulations

Parameters	Values
Leg Parameters	
Cylinder mass of prismatic link (M_c)	0.015 kg
Piston mass of prismatic link (M_p)	0.01 kg
Cylinder inertia of prismatic link (J_c)	0.01 kg m^2
Piston inertial of prismatic link (J_p)	0.01 kg m^2
Stiffness at the piston cylinder of prismatic link (k_b)	10^8 N/m

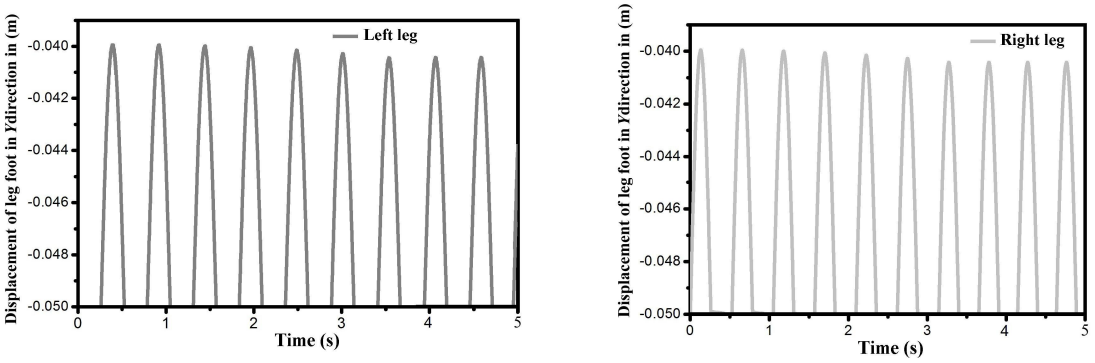
Resistance at the piston and cylinder of prismatic link (R_b)	10^4 N/m
Distance of cylinder CG from cylinder end frame of prismatic link (l_{cg})	0.015 m
Piston & piston rod mass (M_p)	0.01 kg
Cylinder mass (M_c)	0.015 kg
Width of piston (d)	0.01 m
Common Parameters	
Mass of body (M_b)	0.5 kg
Inertia of body (J_b)	0.01 kg m^2
Stiffness of body (Kp)	10^6 N/m
Actuator Parameter	
Motor constant (mu)	0.5 Nm/A
Motor armature resistance (R_b)	0.1 Ω
Motor inductance (L_m)	0.01 H
Experimentation Parameter	
Angle of the ramp (γ)	0.122 rad
Simulation time	5 s

Table 5.5 Initial conditions for the legged robot

Point	x (m)	y (m)
Start position of left leg's foot end	0	-0.05
Start position of right leg's foot end	0	-0.05
Coordinates of robot body's CG	0	0.1
Origin at leg and body joint	0	0

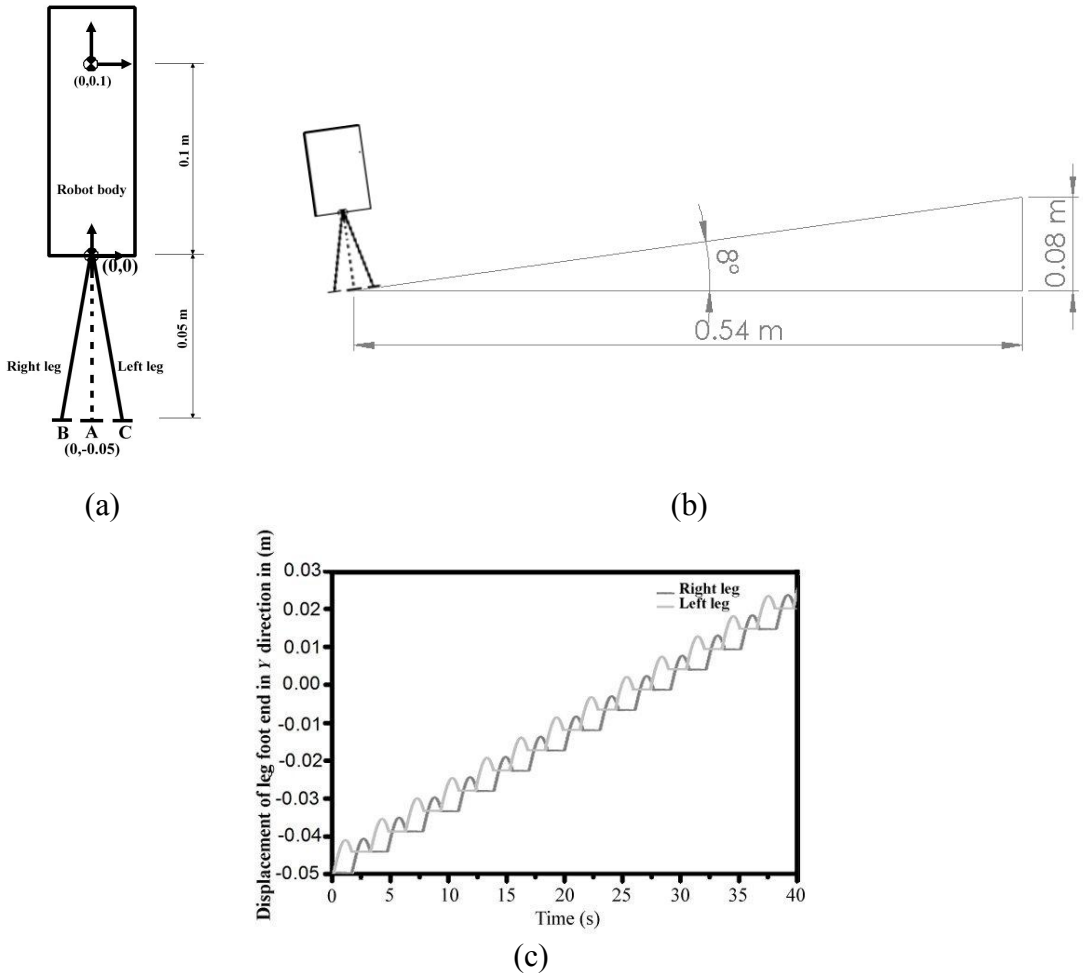


(a)



(b) (c)

Fig. 5.20 Simulation results of legged robot model during straight walking: (a) Leg's foot-end movement in *X*-direction, (b) Left leg's foot-end movement in *Y*-direction and (c) Right leg's foot-end movement in *Y*-direction



(a) (b)

(c)

Fig. 5.21 Simulation results of legged robot model while walking on ramp in *Y*-direction: (a) Initial position of robot, (b) Line diagram of robot and ramp and (c) Displacement of both leg's foot-end in *Y*-direction

Figure 5.22 illustrates the movement of the robot's CG along *X* and *Y* directions while walking on the straight path and inclined plane. The movement of CG along *X* and *Y* direction while walking on the flat straight path is depicted in Fig. 5.22 (a, b). The distance covered by the robot in *X* direction is approximately 10 cm in 5s for the straight path. The movement in the *Y*-direction is near to zero while walking on the straight path. Figure 5.22(c, d) show the movement of CG of the robot while walking on the inclined plane. The surface of the ramp is considered as a flat surface.

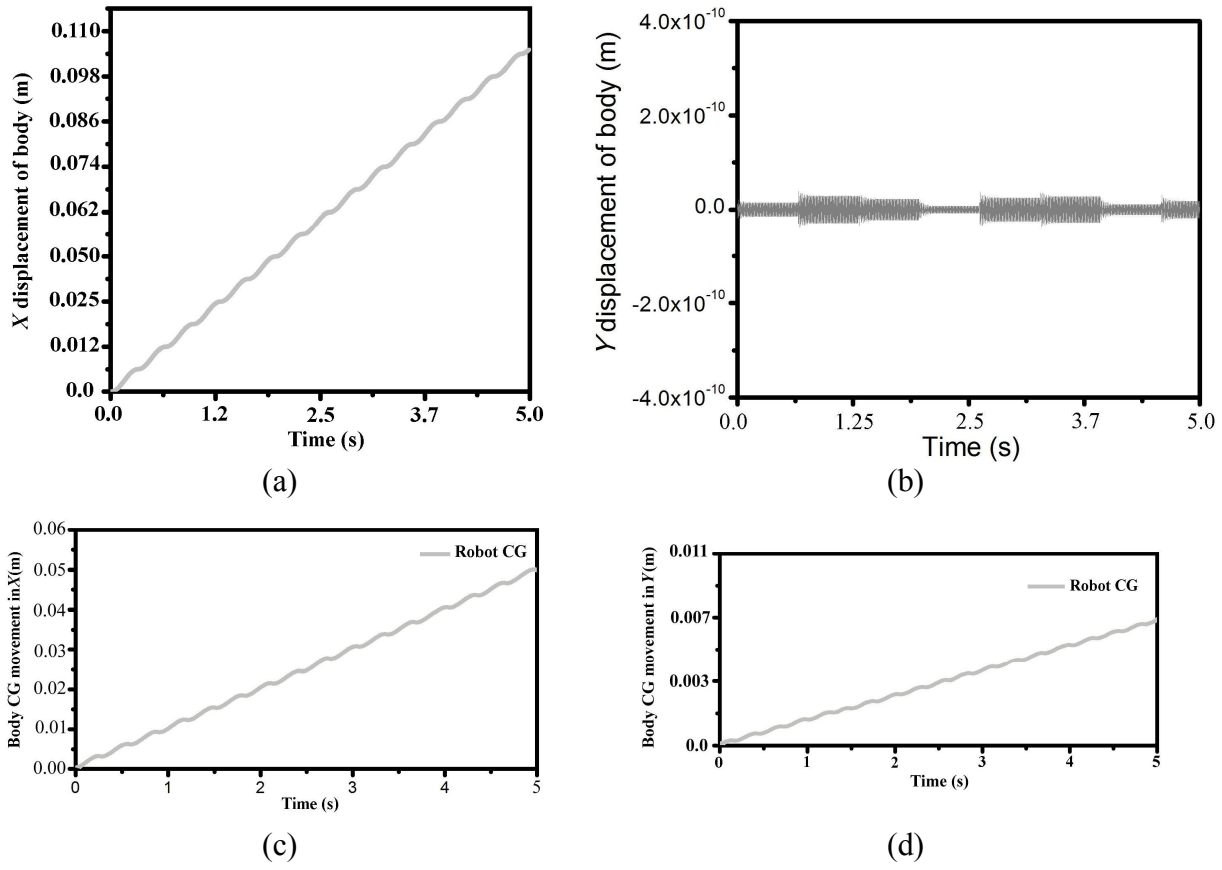


Fig. 5.22 CG movement of the body in (a) *X*-direction and (b) *Y*-direction during straight walking, (c) *X*-direction and (d) *Y*-direction during ramp walking

5.3.3 Bond Graph Model of 3D Legged Robot

The planar BG model of the legged robot is initially developed in vertical (*X*-*Y* plane) plane where *X*-direction is the longitudinal direction and *Y*-direction is the vertical direction. This model is suitable if the robot moves along a straight path i.e. along *X*-

direction. But, this model is not suitable for the movement of the robot in the horizontal *X*-*Z* plane. 3D model of the robot is basically developed for the FDI analysis of robot. Here, trajectory followed by the robot is in the horizontal plane (*X*-*Z* plane) and heave motion takes place along *Y*-direction. The planar robot is considered for developing 3-D legged robot model using BG. *X*, *Y* and *Z*-directions respectively are the longitudinal, heave and lateral motions of the robot. 3-D prismatic joint that is attached to the robot body is considered as a leg of the robot. Figure 5.23 shows the BG model of the 3-D prismatic joint. The cylinder ends are attached to the robot body and piston end tip acts as robot foot that freely moves on the ground surface. The co-ordinate transformation (CTF) function is used to convert the body fixed frame into the inertial frame. Two 3-D prismatic legs are joined with the robot body. The BG model of the complete 3-D model of the robot is illustrated in Fig. 5.24.

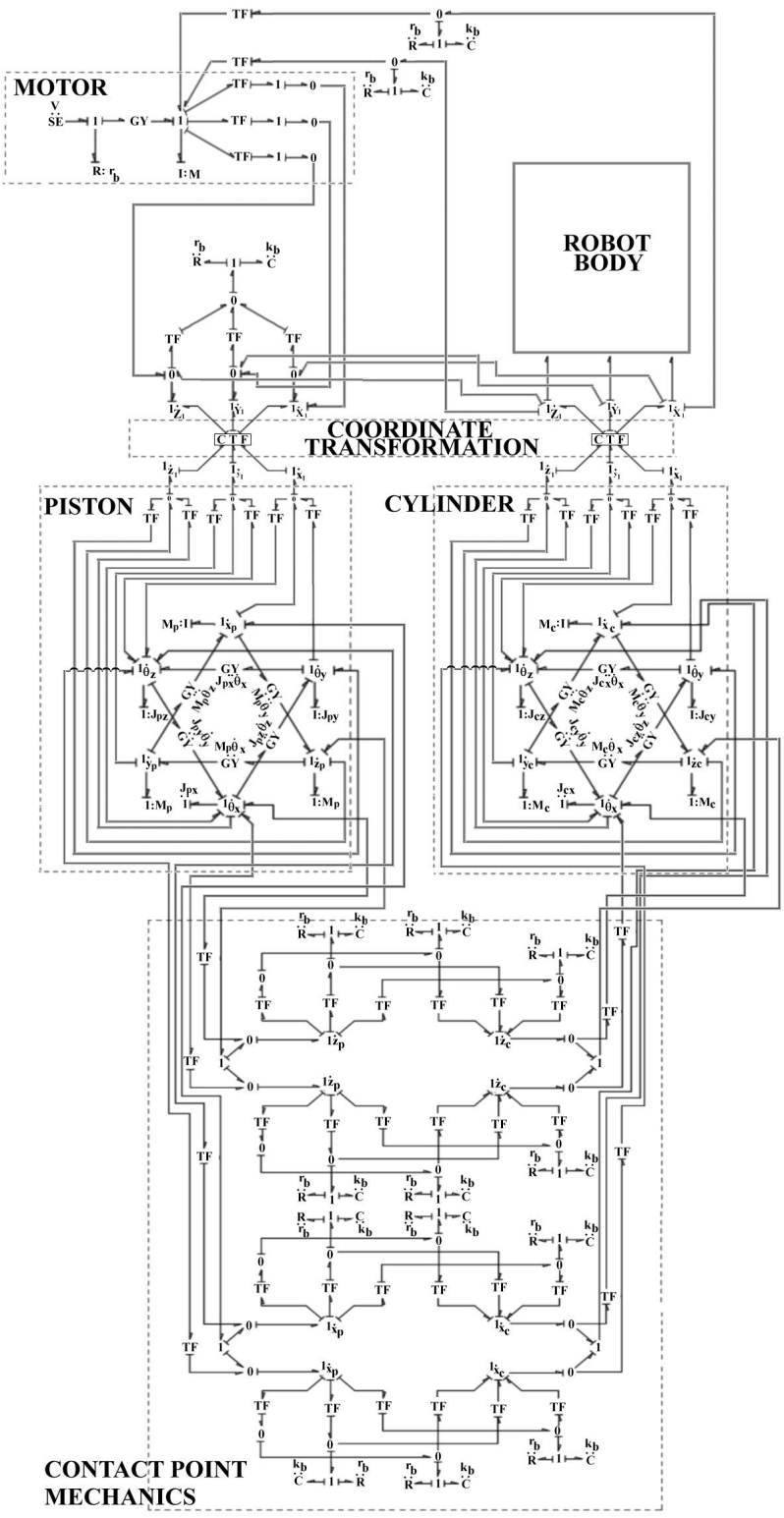


Fig. 5.23 Bond graph model of 3-D prismatic joint

The simulation results of the 3D-legged robot are shown in Fig. 5.25. Figure 5.25 (a–c) illustrate the displacements of the robot body along *X, Y* and *Z* directions. The displacement of each leg is shown in Fig. 5.25(d–f). The time required for simulation is 5s. The displacement of the robot body in *X*-direction is approximately 0.1 m i.e. velocity is 2 cm/s and the displacement along *Y* and *Z* directions are negligible. The maximum displacement of each leg along *Y*-direction is about 1 cm. The simulation is done for walking along a straight path and the simulation along the curved path is done later.

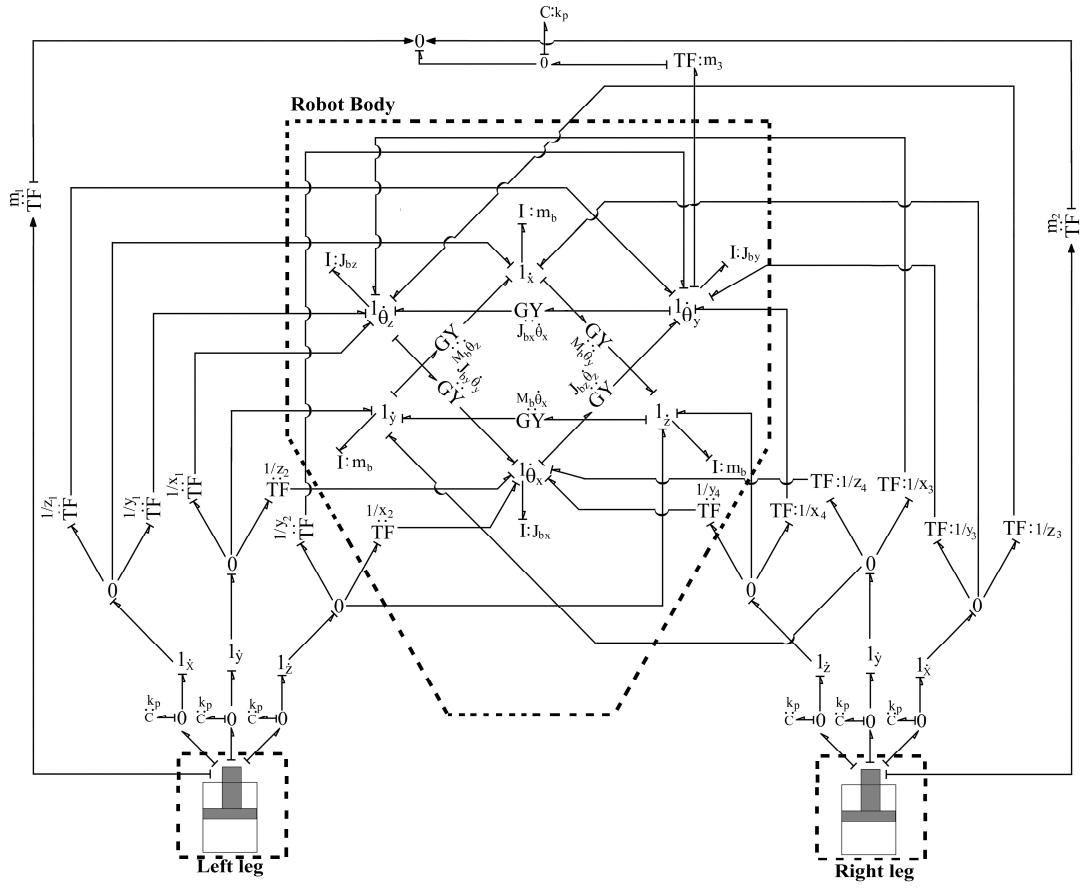


Fig. 5.24 Bond graph model of 3-D robot

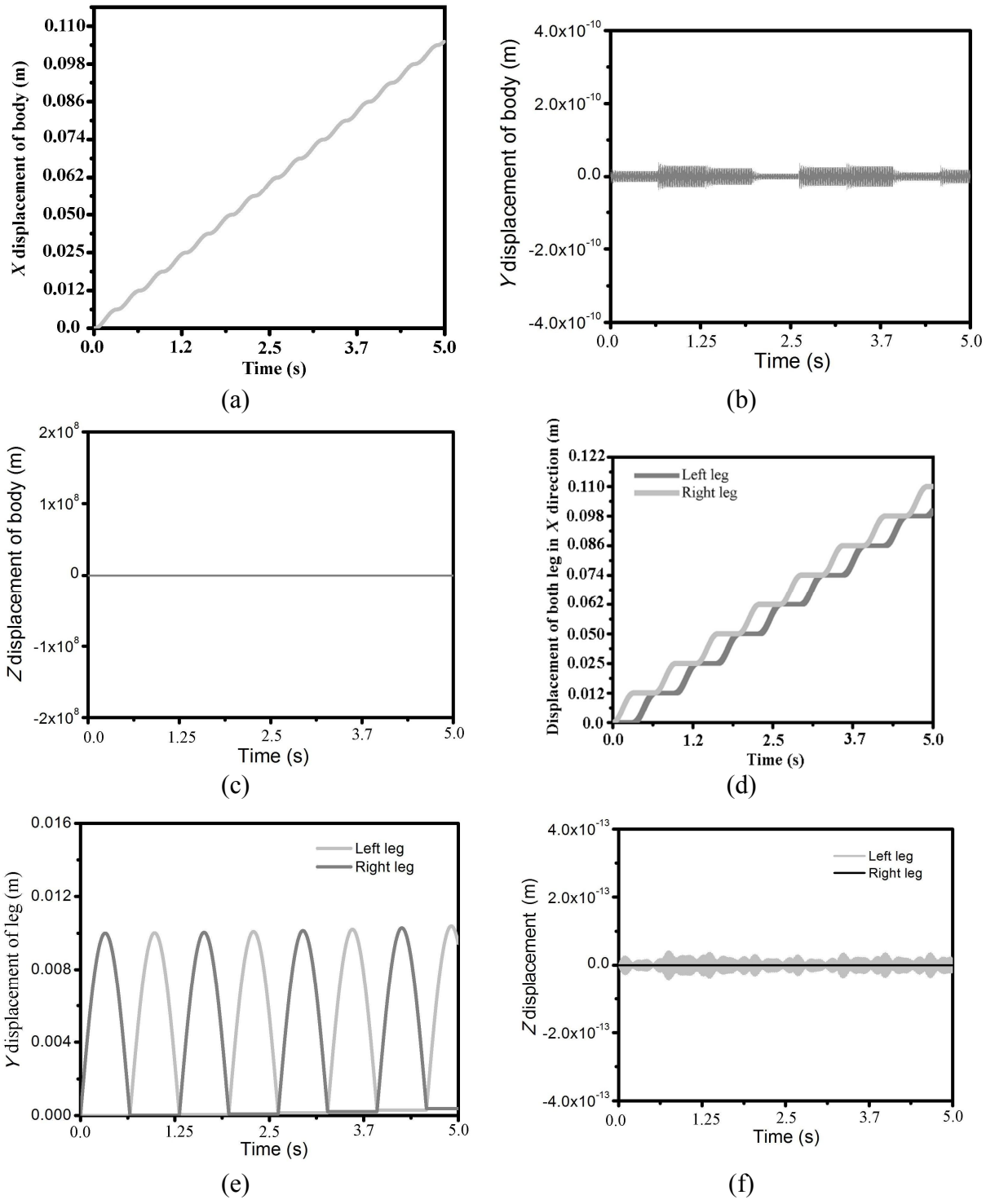
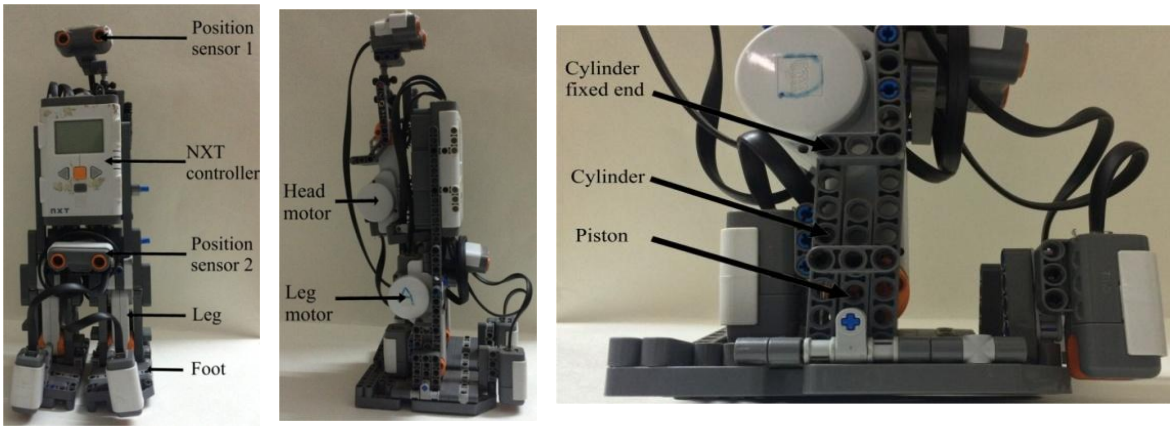


Fig. 5.25 (a) *X*-displacement, (b) *Y*-displacement and (c) *Z*-displacement of body (d) *X*-displacement, (e) *Y*-displacement and (f) *Z*-displacement of both legs

5.3.4 Experimental Work

Figure 5.26 shows the experimental setup for the legged robot. The prototype model of the biped robot used for experimental scenario is of low cost hardware available in the laboratory. The main focus of this work is to implement the proposed fault detection and isolation technique based on ARR using the biped robot for validation. For implementing in general cases, the actual bipedal robot required is expensive and that is currently not available with us. Hence, low cost, small and didactic bipedal robot is considered under this work. Initially, the planar motion of the robot walking along straight and inclined path is considered. Later, motion along *x*-*z* plane is considered.



(a) (b) (c)

Fig. 5.26 Actual experimentation robot setup (a) Front view, (b) Side view and (c) Leg mechanism

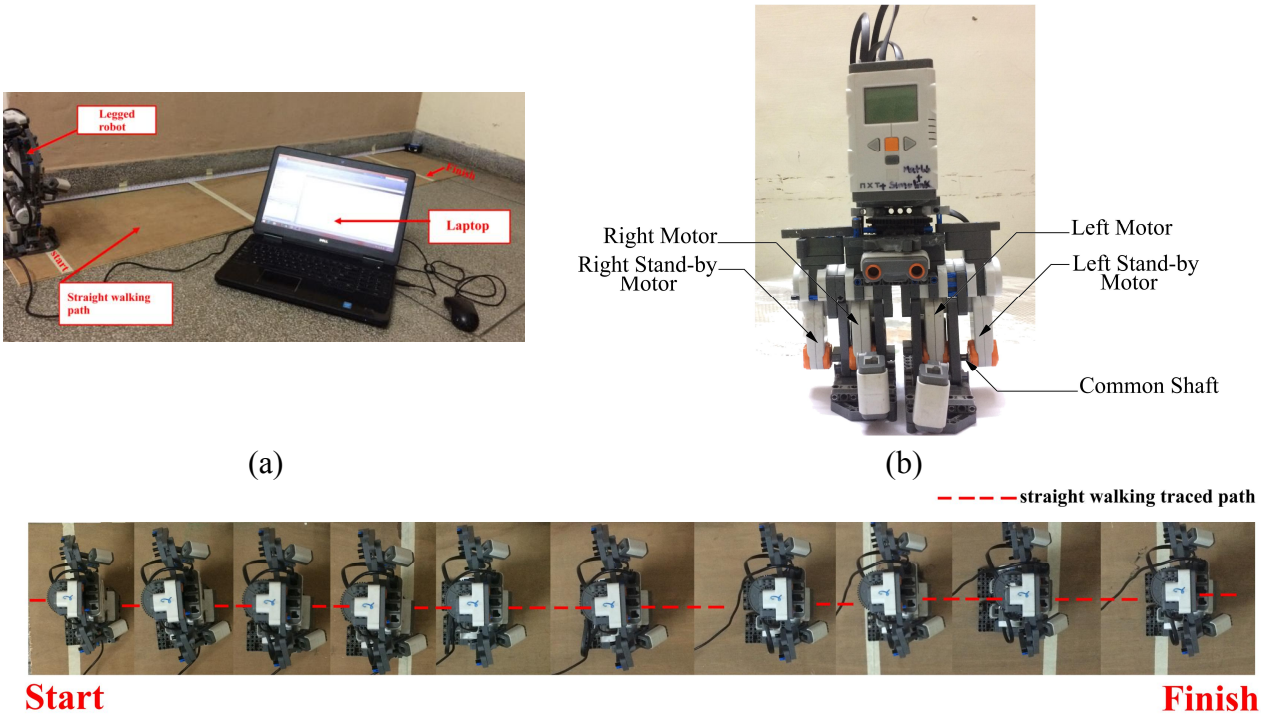
5.3.4.1 Straight Walking

The experimental setup for straight walking is shown in Fig. 5.27(a). The result of straight walking of legged robot is captured from overhead camera. Figure 5.27(b) depicts that robot walking in a straight path.

5.3.4.2 Ramp Walking

Figure 5.28(a) shows the experimental setup for ramp walking. The angle of ramp (γ) is 0.122 rad. The path is ramp up type. The simulation time is 5 seconds. The ramp walking

experimentation results are shown in Fig. 5.28(b). The images are captured in *x-y* frame by overhead camera.

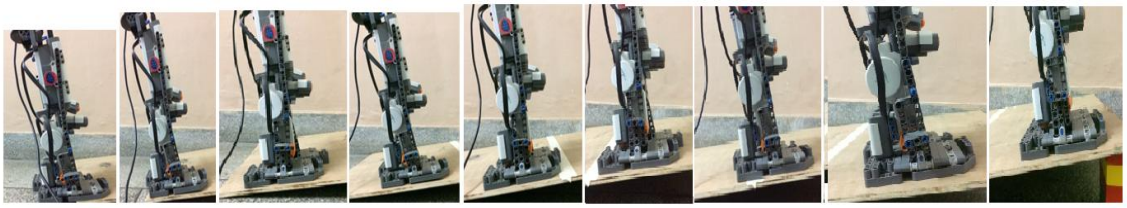


(a) (b)

(c)

Fig. 5.27 (a) Straight walking experimental setup, (b) Setup with standby motor and (c) Robot movement in *X*-direction capture by overhead camera

(a)



(b)

Fig. 5.28 (a) Ramp experimentation setup and (b) Robot movement capture by the camera in *x-y* vertical plane

5.3.5 Fault Diagnosis

The mathematical model is required to develop the FDI model of the system. The ARRs are used to check the relations between the actual performance and the forecast manners of the model. These errors are called residuals. The number of ARRs should be equal to the no's of sensors. A unique fault signature is allotted for an independent residual i.e. it is reactive to a certain fault and non-reactive to other failures. The value of error should be zero or negligible in the running process without fault [Samantaray & Ould Bouamama, 2008] and [Parkash *et al*., 2017]. The monitoring of the failure in a system is only performed if at least one error is affected. The faulty element can be secluded if its value varies from the fault signatures of other elements. The fault signature matrix (FSM) is a set of predefined fault signature. It can be generated from the BG model of the system.

For a single degree-of-freedom system with overwhelming controller, the transfer function between the responses of the system to the command velocity given to the controller can be written as

$$G(s)=\frac{\mu M_c\left(k_p+r_p s\right)}{M_P M_c s^2+M_P\left(k_p+r_p s\right)+\mu M_c\left(k_p+r_p s\right)} \tag{5.17}$$

All notations are already given in the nomenclature of the thesis. The characteristic equation for the system is given by

$$M_P M_c s^2+M_P\left(k_p+r_p s\right)+\mu M_c\left(k_p+r_p s\right)=0 \tag{5.18}$$

For the stability of the system, Routh's Hurwitz criterion may be used. The Routh's stability criterion is shown in Table 5.6.

Table 5.6 Routh's stability criterion

	s^2	s^1	s^0
s^2	$M_P M_c$	$M_P k_p+\mu M_c k_p$	0
s^1	$M_P r_p+\mu M_c r_p$	0	0
s^0	$M_P k_p+\mu M_c k_p$	0	0

From Routh's table, we can see that the system will be stable if the following conditions are satisfied:

(1) $M_P M_c > 0$

(2) $M_P r_p + \mu M_c r_p > 0 \quad \text{or} \quad \mu > -\frac{M_P}{M_c} \text{ if } r_p \neq 0$

(3) $M_P k_p + \mu M_c k_p > 0 \quad \text{or} \quad \mu > -\frac{M_P}{M_c} \text{ if } k_p \neq 0$

As we are considering only positive gains, these are trivially satisfied.

Though the gain of the overwhelming controller needs to be very large for trajectory tracking, the gain value in the actual system is chosen to be too large because in the presence of measurement noise, high gain feedback can lead to system instability. In this work, the measurement noise has been neglected. Another disadvantage of high-gain is that it causes the simulations to be stiff and hence, the computations to be performed in the overwhelming controller domain can give feedback delay and that can cause system instability.

The high-gain implicit system inversion concept is demonstrated in [Samantaray & Ould Bouamama, 2008]. If the trajectory to be followed is sharp, high-gain can give unwanted transients. Therefore, a low-pass filter is also used in the feedback loop. The overwhelming controller is similar to that where the functionality of the low-pass filter is implemented by the spring and damper in the overwhelming controller.

A system is passive if it cannot produce energy on its own and can only dissipate the energy that is stored in it initially. More generally, an I/O map is passive if, on average, increasing the output requires increasing the input. Most physical systems are passive. The Passivity Theorem holds that the negative-feedback interconnection of two strictly passive systems is passive and stable. As a result, it is desirable to enforce passivity of the controller for a passive system. In practice, passivity can easily be destroyed by the phase lags introduced by sensors, actuators and communication delays. These problems have led to extension of the Passivity Theorem that considers excesses or shortages of passivity, frequency-dependent measures of passivity and a mix of passivity and small-gain properties.

In this work, the phase lags introduced by sensors, actuators and communication delays are not considered. Thus, the passivity can be demonstrated by Nyquist plot of the system and the controller. A system is passive if the Nyquist plot lies entirely in the right half of the complex plane. For all positive values of system parameters M_p, k_p, r_p and controller parameters M_c, k_c and r_c, it can be shown that the system as well as the controller are passive. As a consequence, stability at any positive gain is guaranteed.

5.3.5.1 Operating Modes

The reconfiguration of the system means the redundant devices are in working order when one or more components are in failure state [Pathak *et al.*, 2008]. The equipment availability flowchart for a conventional biped robot with operating conditions [Ghoshal & Samantaray , 2009; Krishnan *et al.*, 2011] is shown in Fig. 5.29.

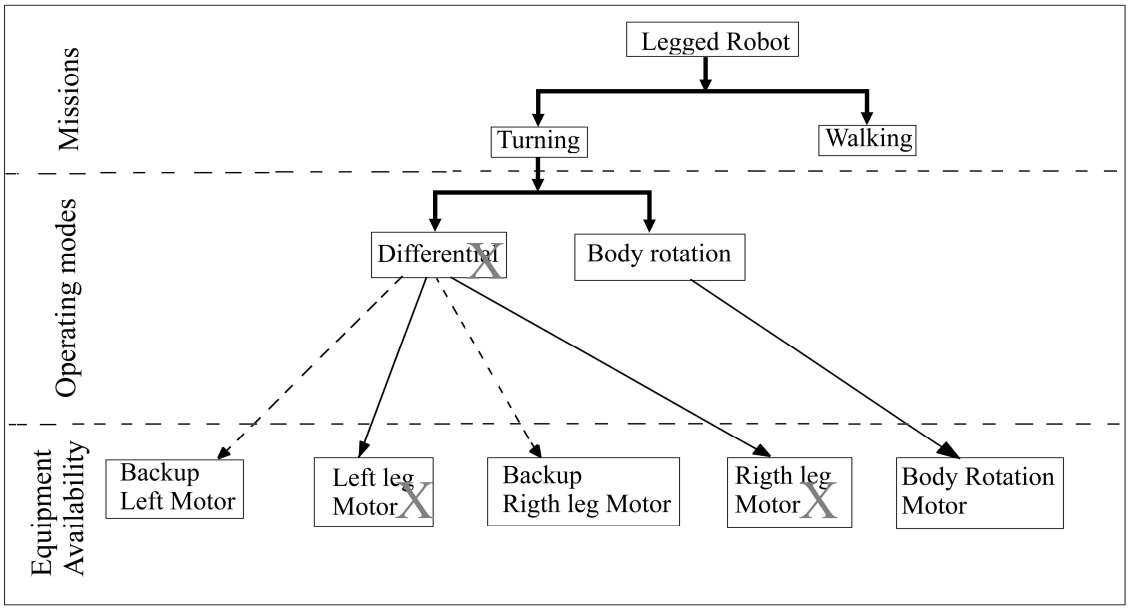


Fig. 5.29 Equipment availability chart with operating modes

The related components for operation of the basic role are expressed in a tree structure. Here, the redundant components are allocated apart from the required component for a particular task to be performed. If the failure in any component is detected, the connection linked with that device, is eliminated and the process can be reconfigured utilizing the available components e.g. if the left leg motor is under fault, the robot can be reconfigured during the manoeuvre by the backup left leg motor [Chaudhari

et al., 2016]. This conclusion is shown in Fig. 5.29 where a cross sign indicates the unavailability of the component. Now, if the left leg motor, right leg motor and its backup motors are not functioning, it is apparent that the robot cannot be operated during a turning operation; then motor for the rotation of the body may be used for the turning operation only and it cannot be used for walking operation mode. In the equipment availability chart with operating modes (Fig. 5.29), it is seen that turning of the legged robot can be done either by differential speeds or by rotation of the body. The differential speeds can be attained by changing the speeds of motors of left or right leg or by varying the speeds of backup motors from the same speeds. The rotation of the body can be attained using a separate motor for rotation of the body. These options are possible for a generalized situation during turning of the robot. However, in this work, separate motor for rotation of the body is not considered and the robot is made to turn due to the differential speeds of left and right leg motor.

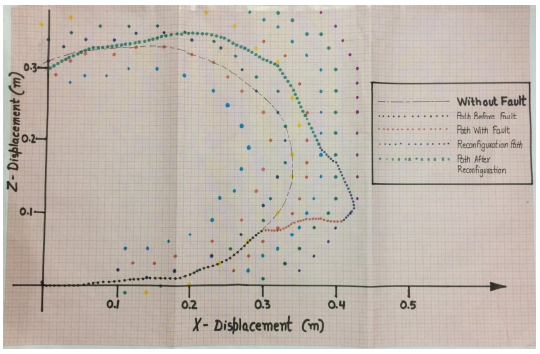
5.3.5.2 Fault Detection

The sensors used in the bond graph model illustrated in Fig. 5.18 and 5.19 are used for generation of the ARRs. The ARRs are generated based on the algorithms by Samantaray & Ould Bouamama [2008]. Eight sensors were used to measure traction motor current i, angular velocity of motor $\dot{\theta}$, piston velocity $\dot{X}_{pg}$, piston velocity $\dot{Y}_{pg}$, piston angular velocity $\dot{\theta}_{pg}$, cylinder velocity $\dot{X}_{cg}$, cylinder velocity $\dot{Y}_{cg}$ and cylinder angular velocity $\dot{\theta}_{cg}$ in each leg. The additional three ARRs are used for measuring the robot body's velocities $\dot{X}, \dot{Y}$ and $\dot{\theta}$. The equations for each ARR are derived from BG model and are given in Appendix B. The Eq. (B_1–B_2) show the ARRs for traction motor of left leg and Eq. (B_3–B_8) represent the ARRs for left leg's piston and cylinder end. Similarly, for right leg, ARRs are derived. The robot body's ARRs are shown in the Eq. (B_9–B_{11}).

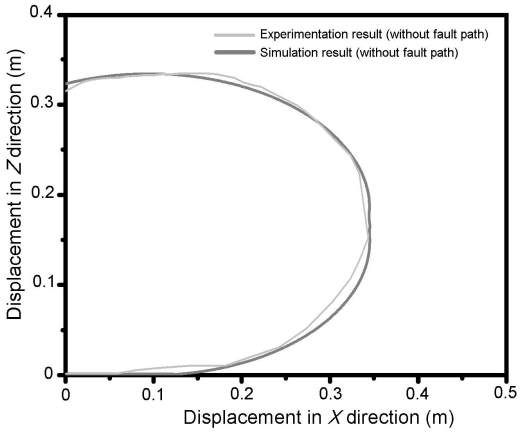
Two motors for the left and right legs (one motor each) are used for the movement of the robot. The speeds of the motors are same for the straight motion and the differential speeds are provided to the legs for turning of the robot. One additional backup motor for each leg is considered as redundant motor. The green indicators denote that there is no

fault situation of conventional motors of robot legs. The red indicators denote the fault situation of motors of robot legs and approximately, 10s is required for the fault detection, isolation and reconfiguration with the additional motor attached to the leg.

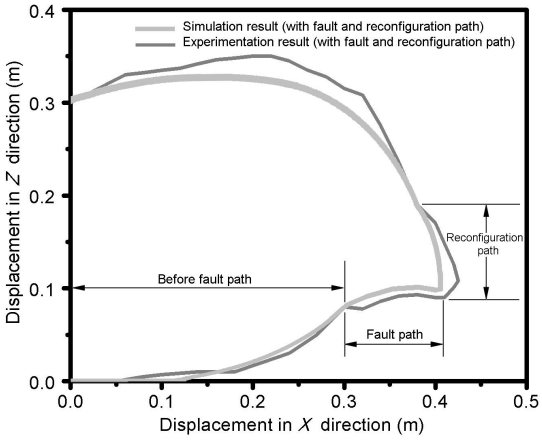
In this section, experiment and simulation are done for the movement of walking robot without and with faults along a U-shaped path and the corresponding ARRs are also determined. The robot starts from the point (0, 0) and after 14 cm of straight movement along *X*-direction, takes left turn using differential speeds of motors as shown in Fig. 5.30(b). The robot again takes left turn when it moves 19 cm along Z-direction and finally, it reaches to the final destination at the point (0, 30) cm. This is the first scenario without fault. The comparison of the experimental trajectory with the simulation profile without fault is shown in Fig. 5.30(b). In the second case, when the robot is at a position of (30, 8) cm during first turning, a fault occurs in the motor of the left leg and it continues up to the position of (40, 10) cm. The reconfiguration is started from (40, 10) cm and it continues until the end. The comparison between the experimental trajectory with the simulation profile with fault and reconfiguration is shown in Fig. 5.30(c). The experimental profiles with and without fault are shown in Fig. 5.30(a). As a location-pointer of the robot, the colour dots are created on the graph paper before movement of the robot. The experiment was conducted several times and based on the positions of the robot with respect to the location of colour dots (location-pointer), the path traced by the robot is drawn manually from the video file. The green indicators denote no-fault situation of motors of robot legs and the red indicators denote the faulty situation of the motors. The turning radius depends on the differential voltage between the outer and inner wheels. The turning radius decreases with the increase of this differential voltage and vice versa. The smaller radius imposes the restriction on the turning motion. In the experimentation and simulation of the robot, the turning radius is about 0.15 m. This radius can be decreased approximately up to 0.1 m for safe turning.



(a)



(b)



(c)

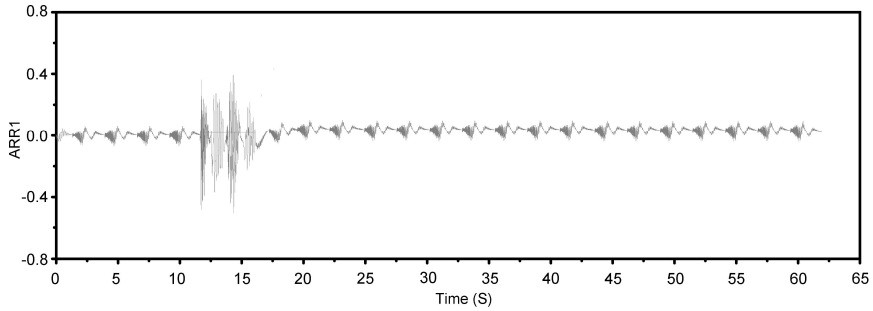
Fig. 5.30 (a) Experimental results, (b) Simulation vs. experimental results of robot during manoeuvring without fault and (c) With fault

Eight ARRs of left leg are shown in Fig. 5.31. In this scenario, the fault is generated from 12s to 17s and then the robot is reconfigured. These ARRs are generated for the left leg. Similarly, Fig. 5.32 shows the behaviour of the ARRs for right leg and in this case, no fault is introduced. Only four ARRs are shown and others are not provided as these are similar in nature due to no fault situation. The ARRs corresponding to different sensors are given in Table 5.6. The readings from the eight sensors during movement of the robot including fault detection to reconfiguration are used to calculate the ARRs. As the fault is generated in the left leg motor during 12s to 17s, the deviation in the residual values (ARRs) of the left leg motor is observed during this period as shown in Fig. 5.31. However, no deviation (shown in Fig. 5.32) in the residual values (ARRs) of the right leg motor is observed during this period as no fault is created in that motor. No deviation in

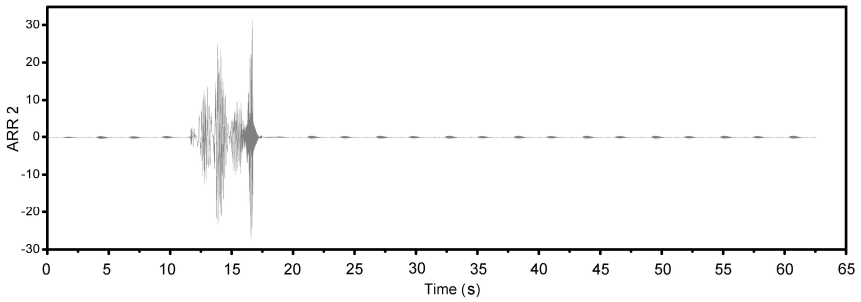
the ARRs is seen during no-fault situation or reconfiguration of backup motor for the left leg.

Table 5.7 ARRs corresponding to different sensors

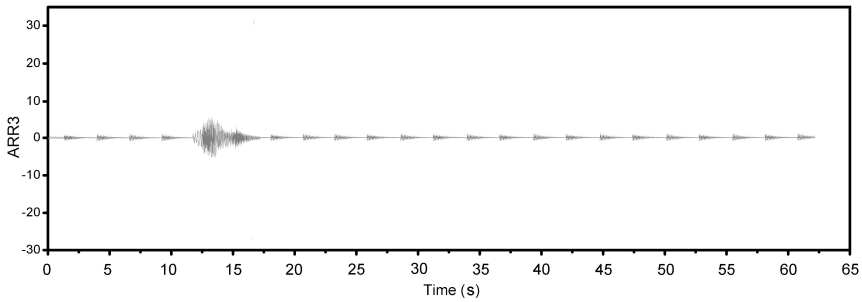
Sl. No.	Sensors	Concerned ARRs
1	Current sensor i	ARR1, ARR2
2	Angular velocity sensor of motor $\dot{\theta}$	ARR1, ARR2, ARR3, ARR4, ARR5, ARR6, ARR7, ARR8
3	Piston X-velocity sensor $\dot{X}_{\text{pg}}$	ARR2, ARR3, ARR4, ARR5, ARR6, ARR7, ARR8
4	Piston Y-velocity sensor $\dot{Y}_{\text{pg}}$	ARR2, ARR3, ARR4, ARR5, ARR6, ARR7, ARR8
5	Piston angular velocity sensor $\dot{\theta}_{\text{pg}}$	ARR2, ARR3, ARR4, ARR5, ARR6, ARR7, ARR8
6	Cylinder X-velocity sensor $\dot{X}_{\text{cg}}$	ARR2, ARR3, ARR4, ARR5, ARR6, ARR7, ARR8
7	Cylinder Y-velocity sensor $\dot{Y}_{\text{cg}}$	ARR3, ARR4, ARR5, ARR6, ARR7, ARR8
8	Cylinder angular sensor velocity $\dot{\theta}_{\text{cg}}$	ARR2, ARR3, ARR4, ARR5, ARR6, ARR7, ARR8



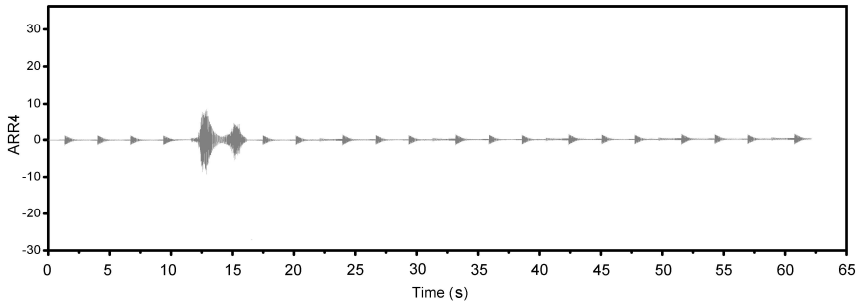
(a)



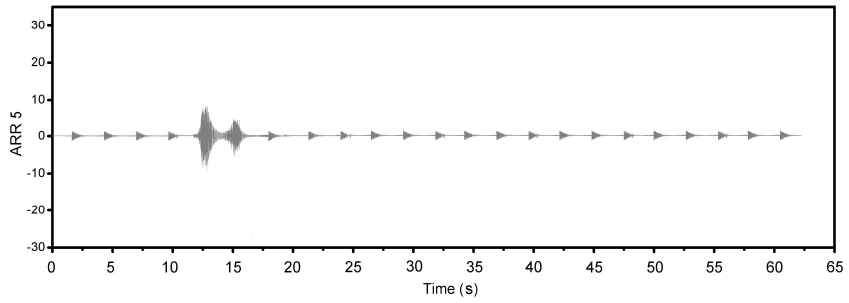
(b)



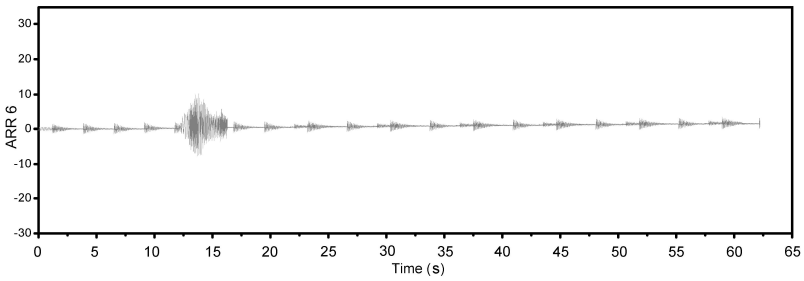
(c)



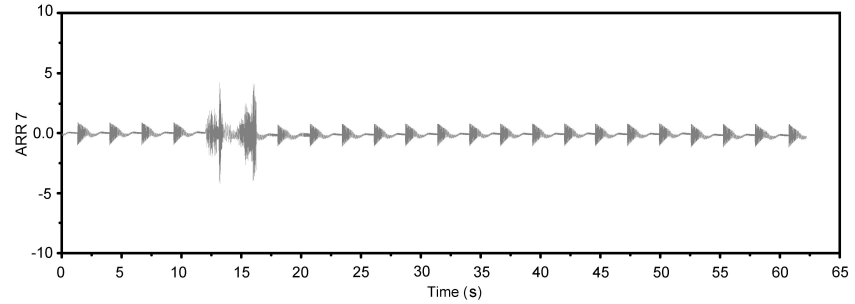
(d)



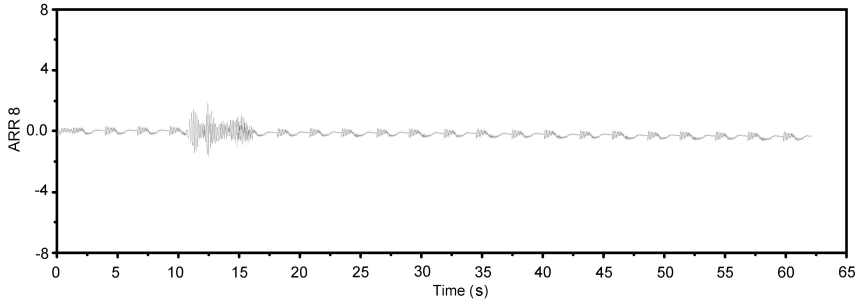
(e)



(f)

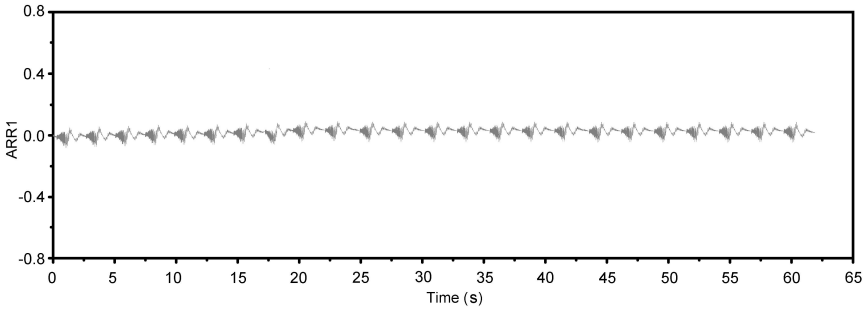


(g)

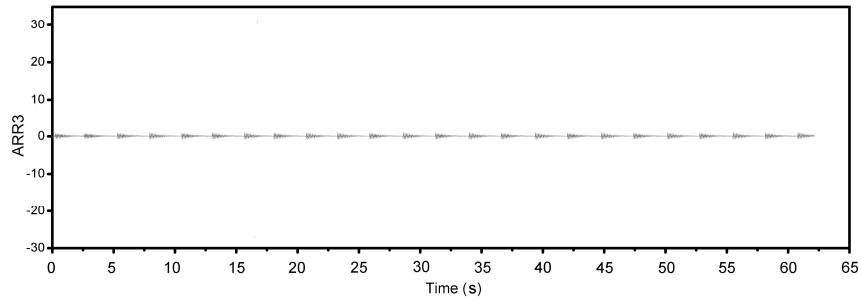


(h)

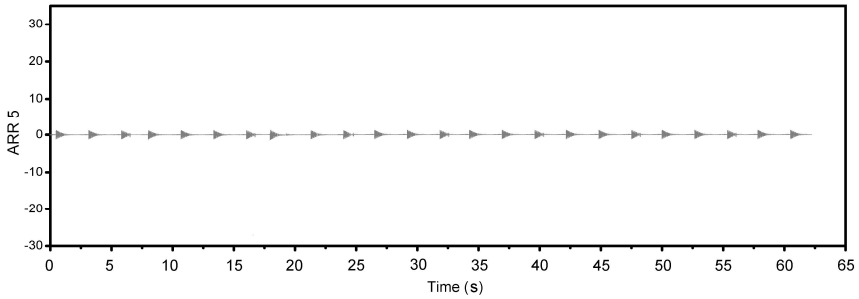
Fig. 5.31 Left leg's ARRs behaviour when a fault is generated during 12–17s



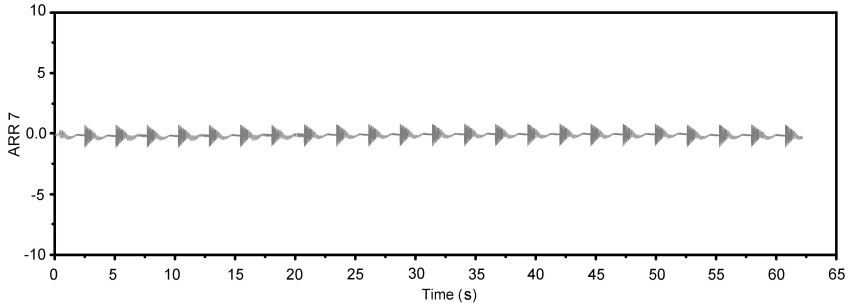
(a)



(b)



(c)



(d)

Fig. 5.32 Right leg's ARRs behaviour when no fault is generated

Table 5.7 shows the fault signature matrix. If one component of a row in the table is non-zero, the equivalent device can be controlled. If there is no fault, the residuals (left hand side of ARRs) are ideally zero. The residuals deviate from their nominal values if there is any error in the component on which the residuals are dependent. The ARRs are derived with the help of FDI Pad toolbox of *SYMBOLS* software. The residuals of the legged robot are evaluated using *Matlab*. The deviation and no-deviation of residuals from nominal values are denoted by binary numbers 1 and 0, respectively. The residuals are calculated at every instant of time during movement of the legged robot and if there is any non-zero term in any row of FSM, the corresponding component can be isolated and the robot can be reconfigured with the help of back up or redundant component.

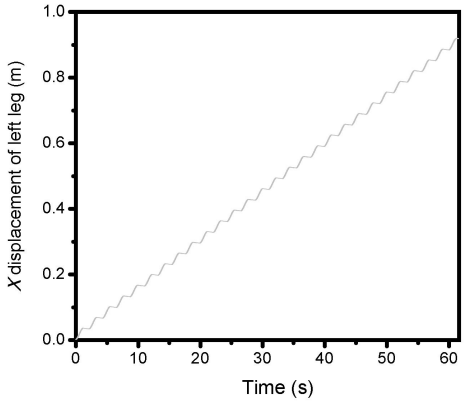
Table 5.8 Theoretical FSM

Components	Motor (L)	Motor (R)	Piston (L)	Piston (R)	Cylinder (L)	Cylinder (R)	Body		
	Residuals								
	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13	R14	R15	R16	R17	R18	R19	M_0	I_0
$Piston_L$	0	0	0	0	1	1	1	0	0	0	1	1	1	0	0	0	0	0	0	1	1
$Piston_R$	0	0	0	0	0	0	0	1	1	1	0	0	0	1	1	1	0	0	0	1	1
$Cylinder_L$	0	0	0	0	1	1	1	0	0	0	1	1	1	0	0	0	1	1	1	1	1
$Cylinder_R$	0	0	0	0	0	0	0	1	1	1	0	0	0	1	1	1	1	1	1	1	1
$Motor_L$	1	1	0	0	1	1	1	0	0	0	1	0	0	0	0	0	0	0	0	1	1
$Motor_R$	0	0	1	1	0	0	0	1	1	1	0	0	0	1	0	0	0	0	0	1	1
Body	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1

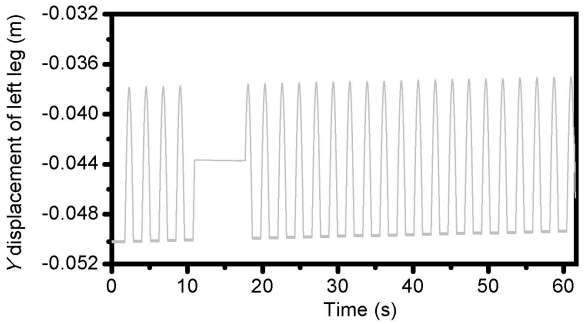
M_o: monitorable I_o: isolable

5.3.6 Simulation Results

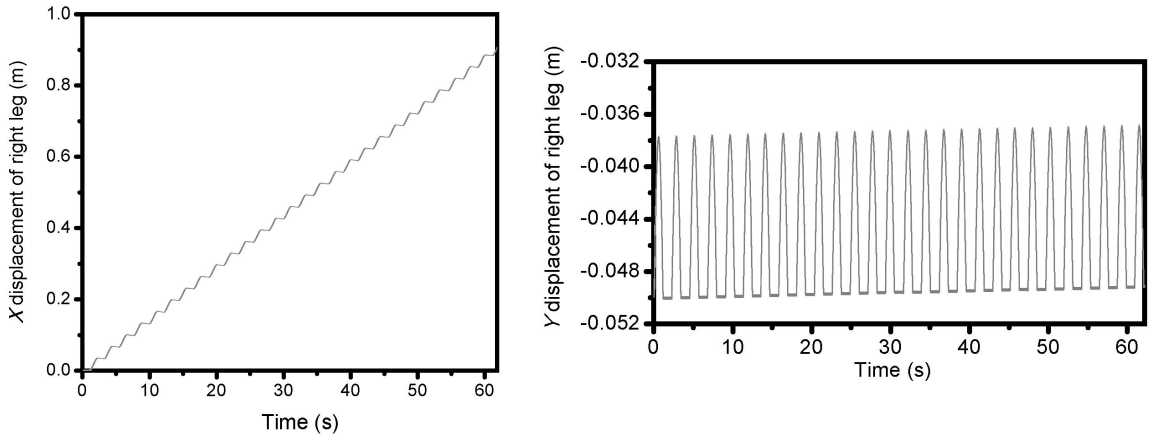
The parameters for the simulation of the legged robot are mentioned in Table 5.1. The robot moves with constant speed along the straight line using both the motors. The motor of the left leg is disconnected for fault-simulation when the position of the robot is at the point (30, 8) cm. As a result, the displacement along *Y*-direction of robot's left leg is disturbed as illustrated in Fig. 5.33(b) and the reconfiguration state occurs starting the left leg's backup motor. Similarly, no-fault is given to the right leg motor during movement of the robot. The effect of the same is presented in Fig. 5.33(d).



(a)



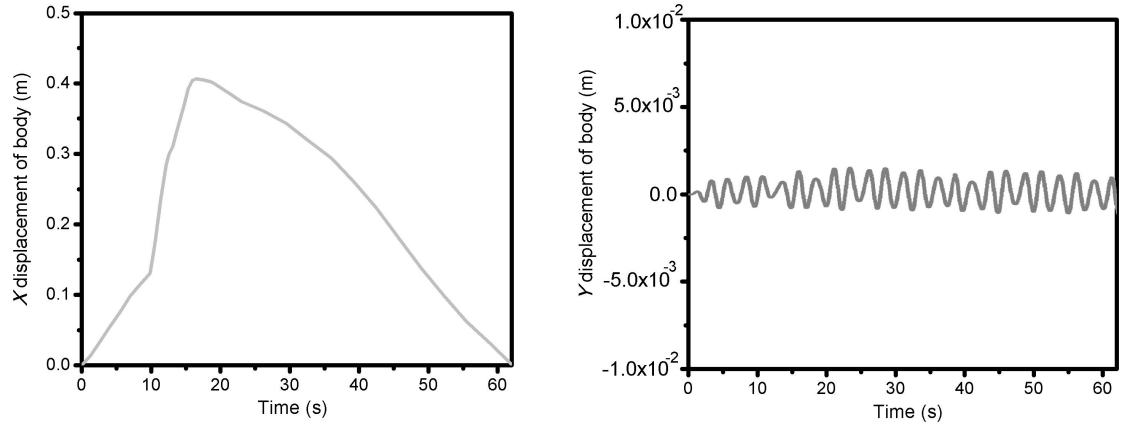
(b)



(c) (d)

Fig. 5.33 Simulation results when a fault is generated and reconfigured in the left leg and no fault situation in the right leg

In this scenario, the fault is given in the left motor when the robot takes a left turn. The displacements along *X* and *Y*-direction of the robot body are shown in Fig. 5.34(a, b). The system is reconfigured during turning operation by operating backup motors when the robot is at a position of (40, 10) cm. The robot takes a turn of 2.8 rad approximately during faulty situation as shown in Fig. 5.34(d). The path traced by the robot is illustrated in Fig. 5.34(e).



(a) (b)

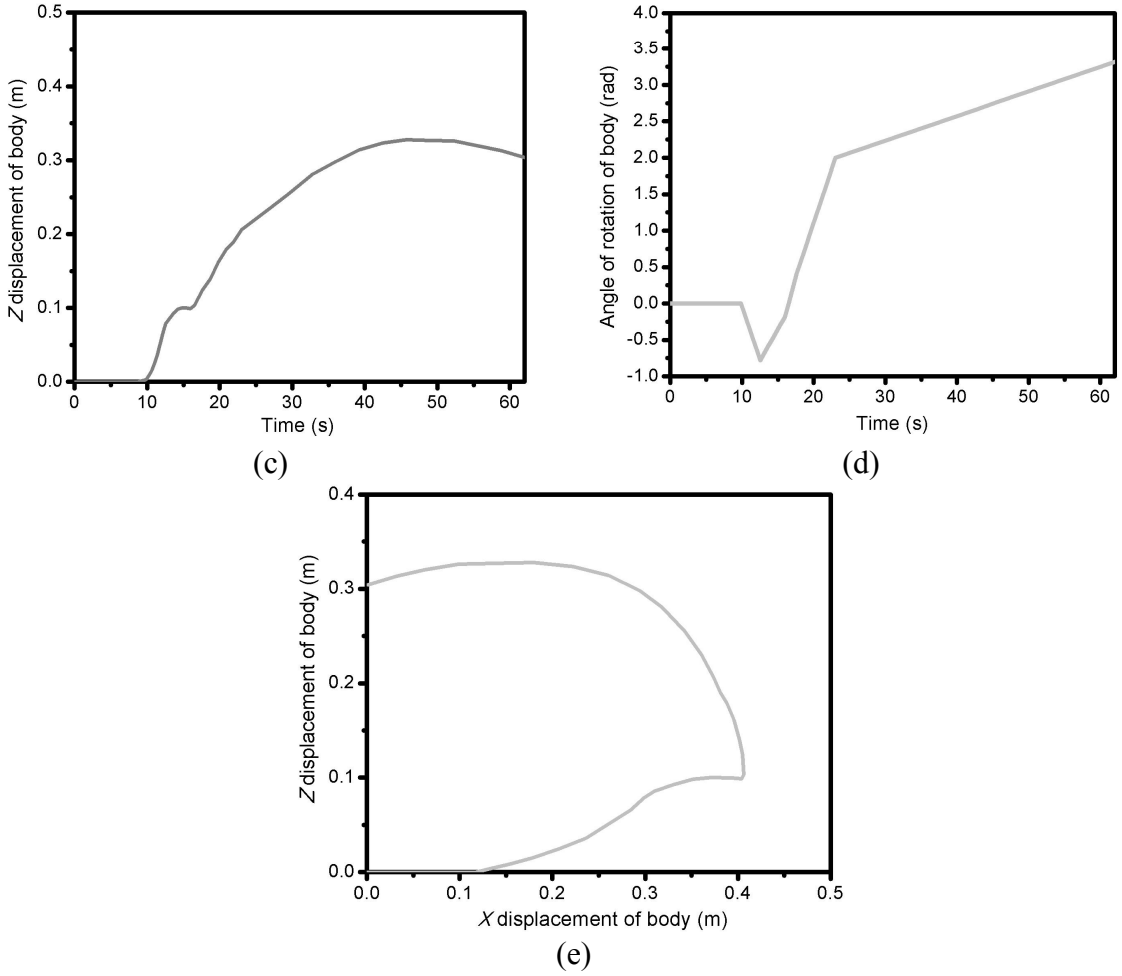


Fig. 5.34 Simulation results of the robot during manoeuvring

5.4 Walking Mechanism of Quadruped Robot on a Side Ramp using PI Controller

The modelling of the legs of the quadruped robot is the key consideration during forward and different kinds of manoeuvres of robot. The different gaits are also accomplished by coordinating the movement of legs in a certain manner. Two-DOF legs are considered in this work (single DOF for each link) i.e. rotation of each link about y-axis. In this section, modelling of links, complete leg and quadruped robot using both bond graph and *SimMechanics*™ techniques were done for a side ramp application.

5.4.1 Bond Graph Modelling of Quadruped Robot

Initially, the links of the leg are modelled. DC motor supplies power for rotation of the links. The current supplied to the motor is modulated for the desired locomotion of legs.

The schematic diagram of legs arrangement is shown in Fig. 5.3(a). BG model for a single link is done. The rotational speed is differentiated before being applied to the link mass moment of inertia. Once both the links are modelled, they are joined together by a joint stiffness to frame the complete model of a leg as shown in Fig. 5.3(b). The kinematic relations (leg's velocities that are normal and tangential to the rotation axis) considered here are similar to that of vehicle model. From these relations, the motion of centre of gravity of body is deduced when the robot moves with its four legs [Bera *et al*. 2011]. The kinematic relations of the quadruped robot have already been discussed in Section *5.2.1.2*.

The bond graph model of the quadruped robot is presented in Fig. 5.4. This model is already discussed. The desired motion of links is graphically represented in Fig. 5.5 and this shows the phase difference between first and second step of lower and upper link.

5.4.2 SimMechanics Modelling

For simulation of multibody dynamics models such as vehicle suspensions, robots and landing gear of planes etc. *SimMechanics* is used. The computer model of the considered bipedal robot is built using *SimMechanics* tools to conduct overall movement of the robot [Zhao *et al.,* 2010]. The virtual platform for the biped robot Silo 2 is developed using *SimMechanics* [Ponticelli *et al.,* 2006].

The robot considered in this work consists of 4 legs, each having two revolute joints (i.e. j = {1, 2} = {upper leg, lower leg}) as illustrated in Fig. 5.1. CAD model of the quadruped is modelled. The model of the contact with the ground is an important part of the robot model. When robot's leg end point touches the ground, it exerts a definite force. The *Simscape* multibody contact forces library block "Sphere to Plane Contact Force (3D)" is used to develop the foot ground contact model of robot. Figure 5.35 shows *Simulink* model of quadruped robot developed using *SimMechanics* blocks. The Solver configuration, world frame and mechanism configuration blocks are used to build the robot model under *Simscape* environment. The solver configuration block identifies the parameters that are required by this model before beginning of simulation. In this model, the parameters for mechanism configuration is [0, 0, -0.98] m/s^2. The angular motion to

the robot Leg 2 and 4 is given with delay (gait period /2.0) as compared to motion in Leg 1 and 3. The Legs 1-3 start first and then with delay, the Legs 2-4 start.

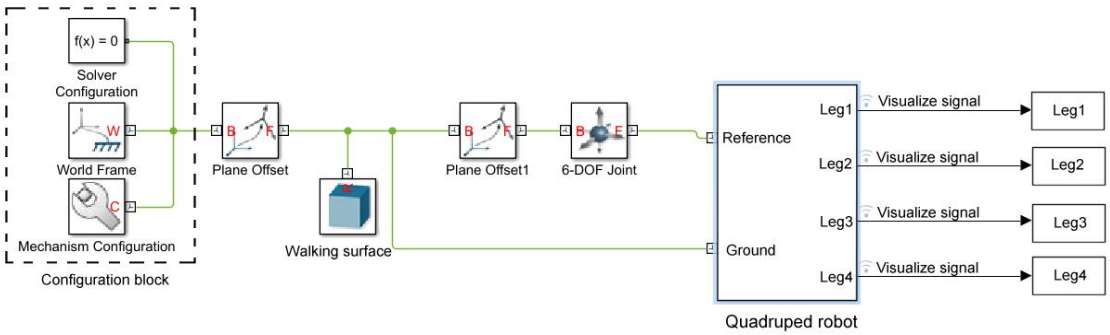


Fig. 5.35 *Simulink* model of quadruped robot

5.4.3 Motion with Trot Gait

In this work, the walking mechanism of a quadruped robot is considered. Therefore, trot gait is considered for balanced walking of the robot. In trot gait, the diagonally opposite legs move together whereas the other diagonally opposite legs remain on the ground.

5.4.3.1 Motion Control Input

The angular position of the actuator joints is the input to the model. From trial and error methods and after lot of run, the data points of these positions were generated. Figure 5.36 shows the angular position of Link 1 for gait period of 1s.

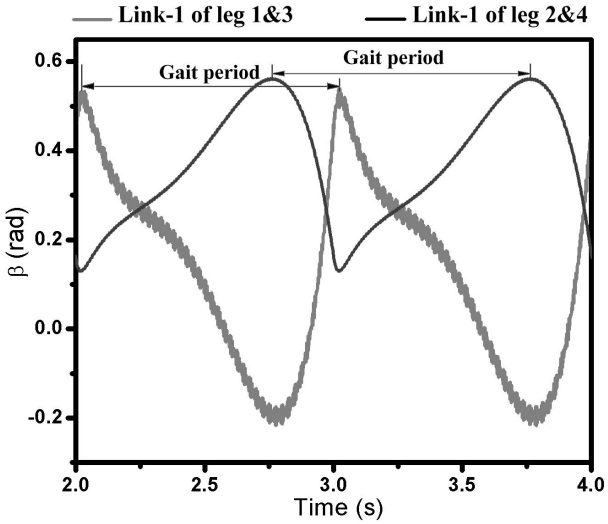


Fig. 5.36 Angular motion of Link 1 of all the legs

5.4.3.2 Results from BG Model

The parameter values used for simulation are mentioned in Table 5.8.

Table 5.9 Parameter values for trot gait

Parameters	Description	Values
C_{m}	Capacitance of input circuit (F)	10
I_{b}	Moment of inertia of body (kg m^2)	10
I_{m}	Moment of inertia of flywheel (kg m^2)	1
I_{m1}	Inductance of motor (H)	1
I_{w}	Moment of inertia of leg (kg m^2)	5
$K_{\mathrm{m_{f1}}}, K_{\mathrm{m_{f2}}}, K_{\mathrm{m_{r1}}}, K_{\mathrm{m_{r2}}}$	Compliance element of leg $(\mathrm{N/m})$	10^8
K_{l}	Compliance element of leg's $(\mathrm{N/m})$	10
L_{leg}	Length of complete leg (m)	0.6
l_1, l_2	Distance of front and rear leg from CG (m)	0.5
M_{b}	Mass of body (kg)	3.0
R_{m}	Resistance of motor (Ω)	0.1
μ_1	Motor constant for front Leg's Link1	2.1
μ_2	Motor constant for front Leg's Link2	0.7
μ_3	Motor constant for rear Leg's Link1	1.95
μ_4	Motor constant for rear Leg's Link2	0.64

The effort-control of leg by mathematical formulations is given to the links. The simulation was run for 10s for the straight movement of the robot. The angular rotation of the Link 1 of any leg is shown in Fig. 5.37.

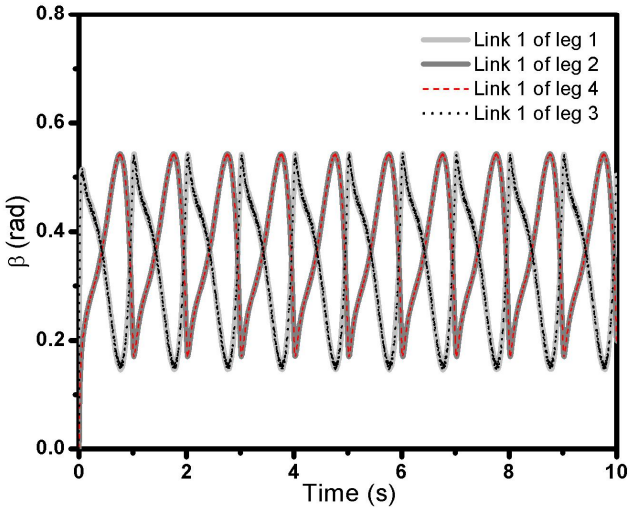


Fig. 5.37 Rotation of link 1

5.4.4 PI Control Scheme

PI controller is used here as a stabilization controller to catch and hold the quadruped robot legs (link) position. Linear PI controllers are also useful for non linear systems provided operating points do not change considerably and if the external disturbances are minimal. As in present work, the slope on which quadruped robot moves is constant (10°) throughout its travel and maximum speed of the robot does not exceed 1m/s, PI controller can be used without much affecting the results. There is no undulation along the path of travel of the quadruped robot as well as contact mechanism between slope (path) and leg is not considered in this work. So, PI controller provides better results. If there is a large variation in operating points then the model can be linearized piece-wise and gain can be scheduled for specific operating zones. At least, this is the common practice followed in various industrial system automations where non-linear control algorithms have not yet made much of an inroad. The mathematical expression for PI controller is given as

$$u(t) = G_p e(t) + G_i \int_0^t e(t)dt \tag{5.20}$$

where G_p and G_i are the proportional and integral gain, respectively.

5.4.4.1 Bond Graph Model with PI Controller

PI controller scheme is discussed in Section *5.4.4*; now, the same is implemented in the BG model of the quadruped model. The actual values of *x* and *y*-displacement of the

robot body is compared with the reference values and the error is fed to the PI controller. The values of proportional and integral gains considered for simulation are determined by hit and trial method.

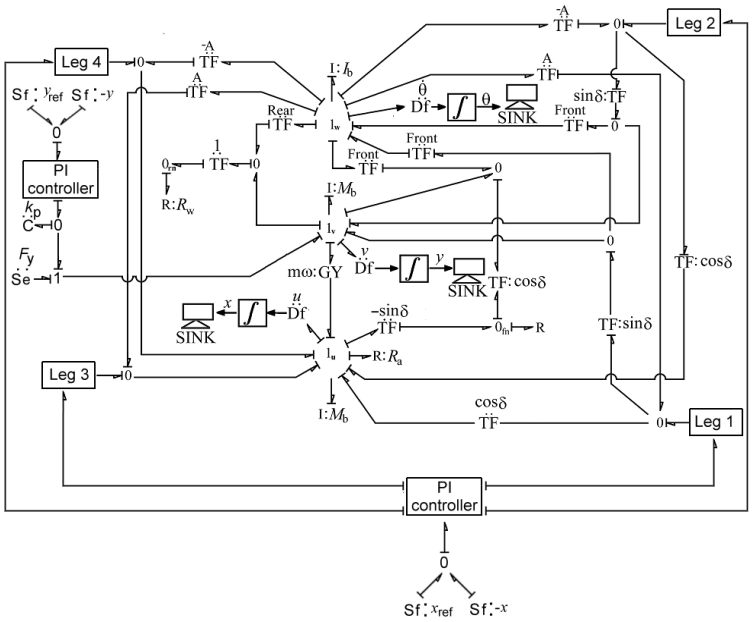


Fig. 5.38 Bond graph model of the quadruped robot with PI controller

5.4.4.2 Simulink Model

The schematic diagram of *SimMechanics* model of the quadruped model with PI controller is depicted in Fig. 5.39.

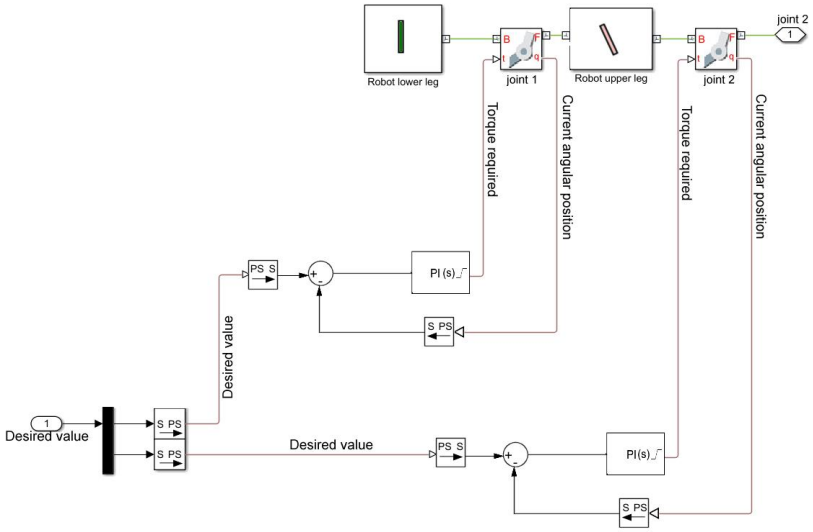


Fig. 5.39 *SimMechanics* based quadruped robot model with PI controller

5.4.4.3 Simulation Results

In this section, the comparison of simulation results of BG model and *SimMechanics* model of quadruped robot is discussed when the robot moves in a straight path on a horizontal plane. For both simulations, the parameters are kept same. The comparisons of the different models such as BG model and *Simulink* model with or without PI controller are shown in Fig. 5.39. For the model with PI controller, *x*-position of the robot is more nearer to the desired profile than for model without PI controller. Similarly, Fig. 5.40 shows results of robot velocity along *X*-direction. Figure 5.40 shows the results for trot gait while undergoing walking locomotion. The robot starts to move from its home position as input is given and travels through a distance of approximately 4.8 m in simulation time of 10s. The velocity in *X*-direction is also shown in Fig. 5.40. As depicted from Fig. 5.40, the velocity increases up to 1 m/s. The robot follows the straight path.

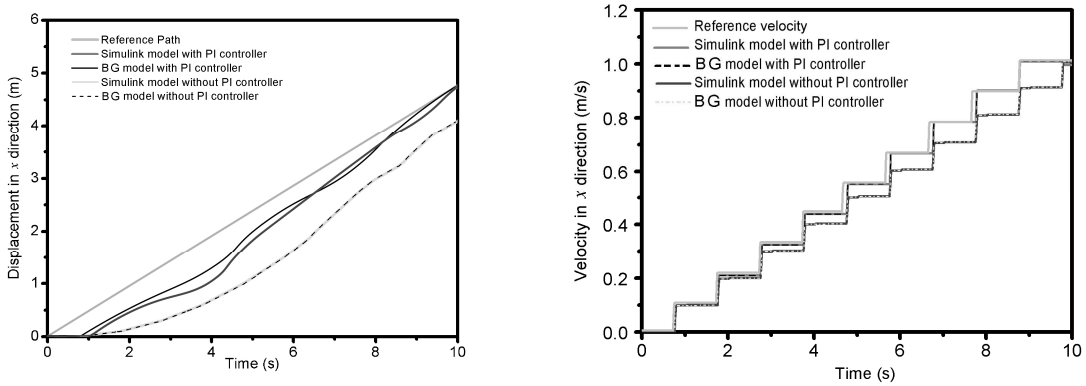


Fig. 5.40 Simulation (BG and *Simulink*) results for displacement and velocity in *X*-direction

5.4.5 Case Study: Side Ramp Walking

The application of the proposed PI-controller for quadruped robot is considered under case study of side ramp walking. In this scenario, the walking plane (*XY*-horizontal plane) is tilted about *X*-axis with some angle and the robot tends to walk on it along *X*-direction on the *xy*-plane without falling into lateral *y*-direction due to the weight (29.4 N) of the robot. The slope of side ramp is 10° with the horizontal plane. The component of the

weight along *y*-direction is about 5.1 N and this force compels the robot to move along *y*-direction. The simulation environment is shown in Fig. 5.41(a–d). The results were shown in Fig. 5.41(e, f). The performance of bond graph and *Simulink* model with PI controller is quite acceptable as compared with model without controller. PI controller helps the robot to move along the desired path. As shown in Fig. 5.41(f), the lateral (*y*-direction) movement of robot having PI control is less (nearly 1 m) as compared to model without PI control (approximately 2 m). Hence, the proposed control model of quadruped robot for walking is quite satisfactory. The simulation is carried out for a time period of 10s.

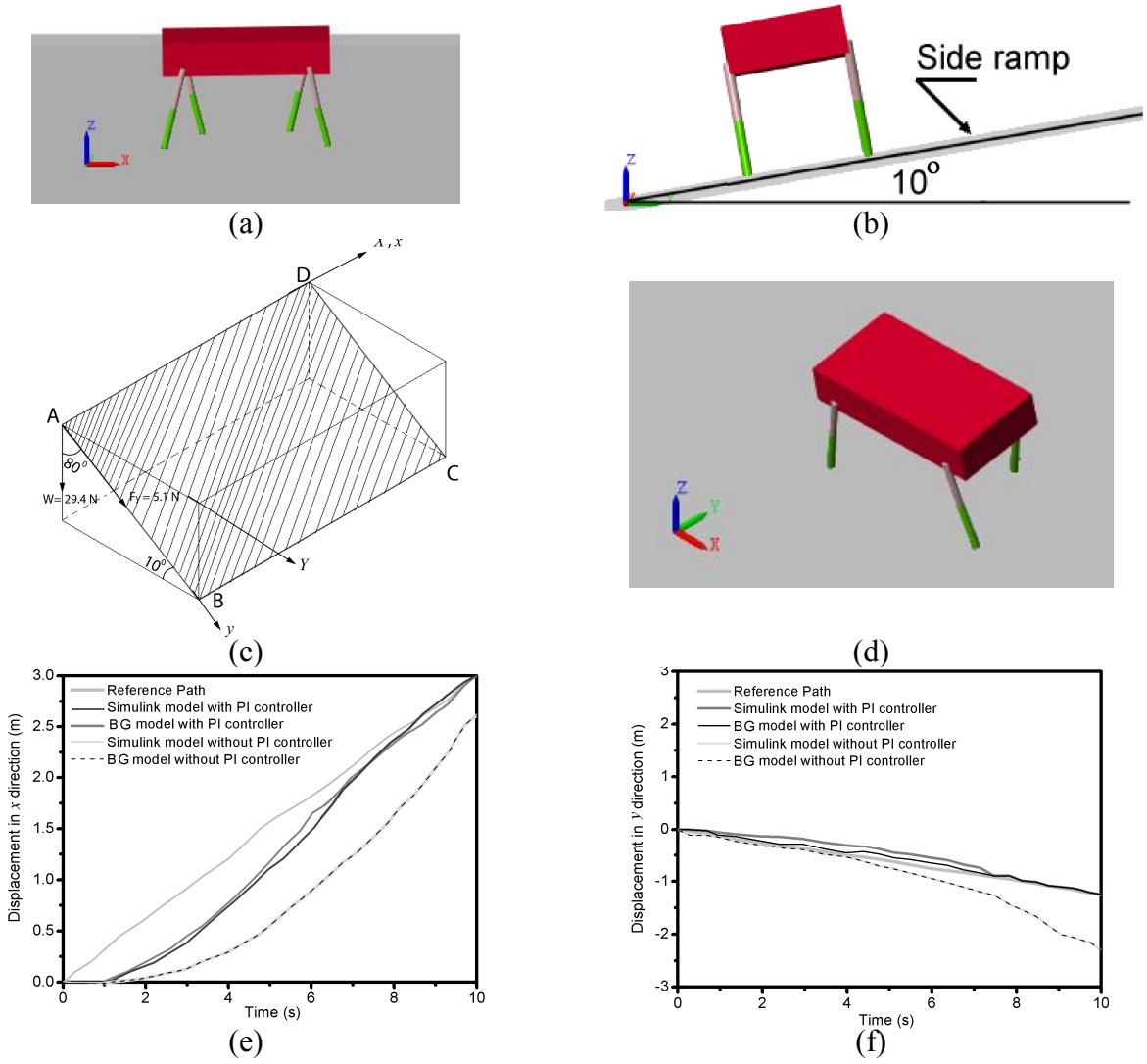


Fig. 5.41 Side ramp simulation scenario (a) Front view, (b) Side view, (c) Inclined plane ABCD, (d) Isometric view, (e) Displacement along *x* and (f) *y*-direction

5.5 Conclusions

In this chapter, the walking mechanism for quadruped robot and bipedal robot with fault detection, isolation and reconfiguration were proposed. The BG based model of quadruped robot was developed and simulated for different gaits such as trot, bound, pace and amble. The results for the models are very much similar to the gaits followed by animals in real life. The data generated by *Matlab* for mathematical formulation provides accurate results. Therefore, this shows the reliability of bond graph as a tool in field of modelling techniques. A detailed observation of the graphs obtained gives very minute details about the movement of legs. For example, as leg takes step forward, the lower link goes backward and upper link moves forward, whereas both the links reach their peak amplitude together and come down at the same time. Another observation depicts a phase lag in the movement of links of legs taking the steps simultaneously. This phase lag occurs as when the leg taking first step comes down, the leg taking next step starts lifting up to maintain the continuous movement. From the results, it is seen that present model is stable irrespective of the gaits being followed and moves forward. As noticed from the comparison of the displacement achieved by the gaits implemented, it is observed that the dynamically balanced gaits are faster and more energy efficient than statically stable gaits as they traverse more distance than statically balanced gaits. The trot gait is the most energy efficient as it traverses the maximum distance among all the gaits. The gait efficiency is usually established in terms of energy efficiency. In the current research, model order reduction is performed with eigenvalue sensitivity approach. Another approach called MORA (Model Order Reduction Approach) relies on the relative activity index of components and constraints (junction structure elements) and removes the least active elements from the BG model. The activity indices are computed from the absolute power exchanged with an element during specific period of system dynamics evolution. This BG models based approach all most always leads to identical results as the eigenvalue sensitivity approach, but provides more physical insights in to the system dynamics and identifies the physical sub-systems which are not participating sufficiently in the system dynamics.

Unfortunately, we have not used MORA in this research. Had we used MORA, the most energy efficient path identification would have been a by-product and could have added greater importance to bond graph modelling. Never-the-less, MORA and energy efficiency can be developed later based on the developed models. The tools and methods for this are well established. Thus, the BG model development does not exclude the possibilities of energy efficiency analysis in the future.

Later, the bond graph based fault detection and isolation for walking mechanism of bipedal robot was developed. The fault accommodation by reconfiguration is considered for biped-legged robot that has faced an actuator fault. The fault-accommodation by reconfiguration is only possible when standby devices are present in the system. ARRs were developed from the bond graph model of the legged robot. These relations are used to derive the theoretical FSM in order to detect the fault. This FDI approach for the mobile robot is suitable for controlling the system using the redundant components/devices. Then, an on-board real-time fault diagnosis for locomotion of legged robot is also considered. The results for fault detection, isolation and reconfiguration in experimentation are compared with the simulation results. At last, the modelling and simulation of quadruped robot was done using bond graph and *SimMechanics* tools. The comparison was done for bond graph model and *Simulink* model of the quadruped robot. Also, the PI controller was proposed to the quadruped robot model for better locomotion of the robot for the desired path. The proposed PI controller was implemented on case study of side ramp walking. As noticed from the comparison results of different models, PI based model is more effective than that of model without controller.

Chapter 6

Conclusions

This dissertation shared the static and dynamic obstacle avoidance, bond graph based modelling of bicycle vehicle and four-wheel vehicle model for mobile robot, obstacle avoidance using overwhelming controller, fuzzy logic based obstacle avoidance controller of Jansen mechanism based quadruped robot, model order reduction, modelling of two-wheel mobile robot, walking mechanism of quadruped robot, fault detection, isolation and reconfiguration of bipedal robot and walking mechanism of quadruped robot with PI controller.

The hybrid algorithm by combining the qualities of line, wall and tangent bug algorithm for static obstacle detection and avoidance was proposed in Chapter 3. The simulation results were compared with those of experimental results to check the feasibility of the algorithm. This algorithm was found to have adopted the shortest possible path based on the geometry of the obstacle by following its circumferential boundary. Also, the fuzzy logic based obstacle detection controller for four-wheel vehicle was proposed to avoid the static and dynamic obstacles. The four-wheel vehicle model of mobile robots using bond graph was developed considering all kinematic constraints. The validation of the controller was checked under different obstacle scenarios. This algorithm is useful in identifying not only the position of the mobile robot but also it is capable to determine the orientation of the robot with respect to the target. An overwhelming control strategy was illustrated with a pedagogical example to prove its efficiency in building a computationally efficient yet simple inverse dynamic system. The efficiency of the controller was tested with the application to avoid the obstacles of different shapes. The bond graph model of the bicycle vehicle model for mobile robot was constructed and linked with the overwhelming controller. This algorithm allows the mobile robot to follow any obstacle free path with very high accuracy in order to reach the target.

Further, the modelling and dynamic analysis of the quadruped robot based on the Jansen mechanism was done using *SimMechanics* software in Chapter 4. The obstacle

avoidance in a closed environment-scenario was considered as an application for the quadruped robot. The fuzzy logic controller with a set of 23 rules was developed to avoid obstacles and to control the movement of the robot. The comparison of the paths travelled by the robot in a closed boundary in simulation and experimental work was made. The velocities in different directions, pitch angle, yaw angle and roll angle show the effectiveness of the controller. This fuzzy logic controller is suitable for the distances of the mobile robot either from the target or from the obstacle ranging from small, medium to large. Then, the model reduction for four-wheel vehicle model was done using eigenvalue sensitivity method. The reduced system is modelled using bond graph approach. The results are compared for the complete and the reduced model. The simulation time and the number of iterations are reduced using the reduced model even though the system dynamics remains unchanged. Finally, the bond graph based model for two-wheel mobile robot using inverted pendulum technique was constructed. The heuristic and PI based controllers were developed to control the motion of the inverted pendulum. The simulation results of the inverted pendulum were validated with the experimentation results. This heuristic controller for two wheeled mobile robots is capable of moving the swinging bar to the upright unstable equilibrium position using voltage as a control input.

The walking mechanism for quadruped robot and bipedal robot with fault detection and isolation was proposed in Chapter 5. The bond graph based model of quadruped robot was developed and simulated for different gaits such as trot, bound, pace and amble. From the results, it is seen that present model is stable irrespective of the gaits being followed. As noticed from the comparison of the displacement achieved by the gaits, it is observed that the dynamically balanced gaits are faster and more energy efficient than statically stable gaits as they traverse more distance than statically balanced gaits. The trot gait is the most energy efficient as it traverses the maximum distance among all the gaits. Later, the bond graph based fault detection and isolation for walking mechanism of bipedal robot was developed. The fault-accommodation by reconfiguration is considered for biped-legged robot that faces an actuator fault. The fault-accommodation by reconfiguration is only possible when standby devices are present in

the system. The ARRs were developed from the bond graph model of legged robot. These relations are used to derive the theoretical FSM in order to detect the fault. This FDI approach for the mobile robot is suitable for controlling the system using the redundant components/devices. Then, an on-board real-time fault diagnosis for locomotion of legged robot is also considered. The results for fault detection, isolation and reconfiguration in experimentation are compared with the simulation results.

A few areas have been suggested to explore further based on the dissertation work. These areas are as follows:

1. An overwhelming controller may be used along with fuzzy logic controller for obstacle avoidance in the presence of external disturbances.
2. Two wheel and four wheel mobile robots with model order reduction for reducing controller's computational time may be developed.
3. Fault detection, isolation and reconfiguration can be done for mobile robot during static or dynamic obstacles avoidance.
4. The hybrid obstacle avoidance algorithm in combination with vision (camera) may be used to calculate the boundary of the upcoming obstacles for accurate results.
5. The on-board real-time fault diagnosis for obstacle avoidance of wheeled mobile robot may be considered.
6. The proposed obstacle avoidance algorithm may be validated with SLAM navigation model under ROS environment.
7. The FDI with LFT model for biped robot during obstacle avoidance may be developed.
8. For more accurate navigation of mobile robot, the sensors such as IMU, GPS and accelerometer may be used on real prototype mobile robot.
9. Different control schemes may be used on real time two-wheel mobile robot (segway) to check the effectiveness of the controller.
10. The fuzzy based obstacle avoidance and detection algorithm may be compared with other existing algorithms.

References

Abdi H and Nahavandi S. Optimal Actuator Fault Tolerance for Static Nonlinear Systems Based On Minimum Output Velocity Jump. In: *Proceedings of the IEEE International Conference on Information and Automation*. Harbin, China. 2010; 1165–1170.

Banerjee N and Karmakar R. Bond graph modeling of rail wheelset on curved track, *Simulation*. 2007; 83(10): 695–706.

Beres J and Wettergreen D. Dante II: Technical Description, Results And Lessons Learned. *International Journal Of Robotics Research*. 1999; 18(7): 621–649.

Bartsch S, Timo B, Malte R, Jens H. Development of The Six- Legged Walking and Climbing Robot Spaceclimber, *Journal of Field Robot.* 2012; 29(3): 506–532.

Bell G, Blythe M, Sengers P and Wright P. Designing Culturally Situated Technology for the Home. *CHI 2003 Extended Conferenc*e. New York. 2003; 1062–1063.

Bera TK, Merzouki R, Bouamama B and Samantaray AK. Design and Validation of A Reconfiguration Strategy For A Redundantly Actuated Intelligent Autonomous Vehicle. *Journal of Systems and Control Engineering.* 2012; 226(8): 1060–1076.

Bera TK, Bhattacharya K and Samantaray AK. Bond Graph Model-Based Evaluation of A Sliding Mode Controller For A Combined Regenerative And Antilock Braking System. *Proceeding IMechE, Part I: Journal Of System And Control Engineering.* 2011; 225(7): 918–934.

Bera TK, Bhattacharya K and Samantaray AK. Evaluation Of Antilock Braking System With An Integrated Model of Full Vehicle Dynamics. *Simulation Modelling Practice and Theory.* 2011; 19(10): 2131–2150.

Bera TK, Samantaray AK and Karmakar R. Bond Graph Modeling of Planar Prismatic Joint. In: *Proceeding of the 14th National Conference Machines and Mechanism.* Durgapur, India. 2009.

Bera TK, Samantaray AK and Karmakar R. Bond Graph Modeling of Planar Prismatic Joints. *Mechanism and Machine Theory.* 2011; 49: 2–20.

Bera TK, Samantaray AK and Karmakar R, Robust Overwhelming Control of a Hydraulically Driven Three Degrees of Freedom Parallel Manipulator Through a Simplified Fast Inverse Model. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering.* 2010; 224 (2): 169–184.

Borutzky W. Bond Graph Modeling From An Object Oriented Modeling Point of View. *Simulation Practice And Theory.* 1999; 7(5): 439-461.

Borutzky W. Bond Graphs-A Methodology for Modelling Multidisciplinary Dynamic Systems. *SCS Publishing House*. 2004.

Borutzky W. Bond Graph Methodology: Development and Analysis of Multidisciplinary Dynamic System Models. *SCS Publishing House*. 2004.

Bos A. Modeling Multibody Systems in terms of Multibond Graphs, with Application to a Motorcycle. *PhD thesis.* Twente University. 1986.

Breedveld P. Physical Systems Theory in Terms of Bond Graphs. *Ph.D. Thesis*. Twente University, Enschede. 1984.

Broggi A, Cerri P and Antonello P. Multi-Resolution Vehicle Detection Using Artificial Vision. *Intelligent Vehicle Symposium.* 2004; 310–314.

Brown F. Engineering System Dynamics: A Unified Graph-Centered Approach.. *Florida: Taylor & Francis Group*. 2007.

Bruns F and Erbe H. Mixed reality with hyper-bonds—A means for remote labs. *Control Engineering Practice.* 2007; 15(11): 1435–1444.

Ceccarelli M and Kececi F. Designs And Prototypes of Mobile Robots. *Momentum Press*. 2015.

Charlie O, Konolige K, Vincent R. Centibots: Very Large Scale Distributed Robtotic Teams. *Intelligent System Demonstrations*. 2004; 1022–1023.

Chatti N, Gehin AL, Ould Bouamama B and Merzouki R. Functional Behaviour Models for The Supervision of an Intelligent and Autonomous System. *IEEE Transactions on Automation Science and Engineering*. 2013; 10 (2): 431–445.

Chaudhari SP, Patil SL, Pandey SK and Sinha S. Performance Analysis of BLDC Motor on Sinusoidal and Square Wave Supply. In: *IEEE International Conference on Power Electronics, Drives and Energy Systems (PEDES)*. Trivandrum, India. 2016; 1–6.

Chen J and Patton R.J. Robust Model-Based Fault Diagnosis for Dynamic Systems. *Springer*. 1999.

Chen X, Chen YQ and Chase JG. Mobile Robots- State of The Art in Land, Sea, Air and Collaborative Missions. *In-Tech publications*. 2009.

Chi C, Zhang W, Zhu J and Liu X. A Method For Comprehensive Diagnosis of Sensors of Flight Control System Using Analytical Redundancy. In: *Proceedings of the International Conference on Electrical and control engineering*. 2010; 4892–4895.

Ching H and Tsai W. Obstacle Avoidance for Autonomous Land Vehicle Navigation in Indoor Environments by Quadratic Classifier. *Transactions On Systems, Man, And Cybernetics.* 1999; *29*(3): 416–426.

Chu K. Local Path Planning for Off-Road Autonomous Driving with Avoidance of Static Obstacles. *Transaction On Intelligent Transportation Systems.* 2012; 13(4): 1599–1616.

Dasgupta K and Watton J. Dynamic analysis of proportional solenoid controlled piloted relief valve by bond graph, *Simulation Modelling Practice and Theory*. 2005; 13: 21–38.

Dauphin T, Rahmani A and Sueur C. Bond Graph Aided Design of Controlled Systems. *Simulation Practice and Theory.* 1999; 7(5-6): 493–513.

Dauphin T. Les Bond Graphs. *Paris: Hermes Science Europe Ltd*. 2000.

Davidson AM and Walters IR. Linear System Reduction Using Approximate Moment Matching. In: *Proceedings on Control Theory and Applications*. 1988; 135 (2): 73–78.

Dewang H S, Mohanty P K and Kundu S. A Robust Path Planning for Mobile Robot Using Smart Particle Swarm Optimization. *Procedia Computer Science*. 2018; 133: 290–297.

Dickmanns E. Dynamic Vision For Perception And Control of Motion. *Springer*. 2007.

Du Y and Papanikolopoulos N. Real-Time Vehicle Following Through a Novel Symmetry-Based Approach. In: *Proceeding of the IEEE International Conference Robotics & Automation*. 1997; 3160–3165.

Ersal T, Fathy H and Stein J. Structural Simplification of Modular Bond-Graph Models Based on Junction Inactivity. *Simulation Modelling Practice and Theory.* 2009; 17: 175–196.

Fakoor M, Kosari A and Jafarzadeh M. Humanoid Robot Path Planning with Fuzzy Markov Decision Processs. *Journal of applied research and technology*. 2016; 14(5): 300–310.

Ferguson D, Darm M, Urmson C and Kolski S. Detection, Prediction, and Avoidance of Dynamic Obstacles in Urban Environments. *Intelligent Vehicles Symposium.* 2008; 149–1154.

Gawthrop P. Bond Graphs: A Representation for Mechatronic Systems. *Mechatronics.* 1991; 1(2): 127–156.

Gawthrop P. Physical Model-Based Control: A Bond Graph Approach. *Journal of the Franklin Institute.* 1995; 332(3): 285–305.

Gawthrop P and Smith L. Metamodelling: Bond Graphs and Dynamic Systems. UK: *Prentice Hall International*. 1996.

Gawthrop P, Jones R and Mackenzie S. Identification of Partially-Known Systems. *Automatica.* 1992; 28(4): 831–836.

Gawthrop P, Neild S, Gonzalez-Buelga A and Wagg D. Causality in Real-Time Dynamic Substructure Testing. *Mechatronics.* 2009a; 19(7): 1105–1115.

Ghangrekar S and Conrad J. Modeling and Simulating a Path Planning and Obstacle Avoidance Algorithm for an Autonomous Robotic Vehicle. *International Symposium on Modeling, Analysis & Simulation of Computer and Telecommunication Systems.* USA. 2009; 1–3.

Ghoshal SK and Samantaray AK. Model Based Fault Diagnosis, Fault Tolerant Control and Reconfiguration of Hydraulic and Thermofluid Processes using Analytical Redundancy. *International Journal of Automation Control.* 2009; 3(4): 363–384.

Ghoshray S and Yen K. A Comprehensive Robot Collision Avoidance Scheme by Two-Dimensional Geometric Modeling. In: *Proceeding of the International Conference on Robotics and Automation.* Minnesota. 1996; 1087–1092.

Granda J. The Role of Bond Graph Modeling and Simulation in Mechatronics Systems: An Integrated Software Tool: CAMP-G, MATLAB–SIMULINK. *Mechatronics.* 2002; 12(9-10): 1271–1295.

Hamid M, Adom A and Rahim N. Navigation of Mobile Robot Using Global Positioning System (GPS) and Obstacle Avoidance System with Commanded Loop Daisy Chaining Application Method. *International Colloquium on Signal Processing & Its Applications*. Malaysia. 2009; 176–181.

Heieh A and Lacroix S. Multi Autonomous Ground Robotics International Challenge. *Journal of Field Robotics.* 2012; 29(5): 687–695.

Hutter M, Diethelm R and Bachmann S. Towards A Generic Solution for Inspection of Industrial Sites. *Field and service robotics*, Springer. 2017; 575–589.

Hyo J, Kim S and Chung M. Object and Ground Classification for a Mobile Robot in Urban Environment. In: *Proceeding of International Conference on Control, Automation and Systems*. Korea. 2012; 2068–2070.

Iida F, Gomez G, Pfeifer R. Exploiting body dynamics for controlling a running quadruped robot. In: *Proceeding of the International Conference Advanced Robot ICAR*. 2005; 229–233.

Isermann R. Mechatronic Systems. *Springer*. 2005.

Jakimovski B, Meyer B and Maehle E. Swarm Intelligence for Self-Reconfiguring Walking Robot. In: *Proceeding of the IEEE Symposium on Swarm intelligence*. St. Louis. 2008; 1–8.

Jun J, Zhan W and Juang J. Application of Image Process and Distance Computation to WMR Obstacle Avoidance and Parking Control. In: *Proceeding of the Conference on Industrial Electronics and Applications.* Singapore. 2010; 1264–1269.

Junco S, Diéguez G and Ramírez F. On Commutation Modeling in Bond Graphs. In: *Proceeding of the 8th International Conference on Bond Graph Modeling and Simulation*. San Diego, California, USA. 2007.

Junco S and Donaire A. Bond Graph Modeling and Simulation of Electrical Machines, Chapter in: *Bond Graph Modeling of Engineering Systems*. 2011.

Kakoty NK, Mazumdar M and Sonowal D. Mobile Robot Navigation in unknown Dynamic Environment Inspired by Human Pedestrian Behaviour. In: *Progress in Advanced Computing and Intelligent Engineering*. Springer. 2018; 2: 441–451.

Kaneko K, Kanehirao F, Morisawa M, Miura K, Nakaoka S I, Kajita S. Cybernetic Human HRP-4C, Humanoid Robots. In: *Proceedings of 9th IEEE-RAS International Conference*. Paris, France. 2009.

Karnoop D, Margolis D and Rosenberg R. Modeling and Simulation of Mechatronic Systems. *Wiley*. 2000.

Karnopp D, Margoli D and Rosenberg R. System Dynamics, Modeling and Simulation of Mechatronic Systems. *John Wiley & Sons*. 2000.

Khadabadi UB and Pilli SC. Analysis of Mechanisms Using Bond Graphs. In: *Proceeding of the National Conference on Machines and Mechanisms*.IIT Kharapur, India. 2005; 31–37.

Komoda K, Wagatsuma H. A Study of Availability and Extensibility of Theo Jansen Mechanism Toward Climbing Over Bumps, In: *Proceeding of the 21st Annual Conference of the Japanese Neural Networks Society*. Japan. 2011.

Krishnan V, Pathak P, Jain S and Samantaray AK. Reconfiguration of Four-Legged Walking Robot for Actuator Faults. *Journal of Systems and Control Engineering IMechE.* 2011; 226(1): 11–26.

Kurabayashit D and Koga S. Local Path Re-planning for Unforeseen Obstacle Avoidance by an Autonomous Sweeping Robot. In: *Proceeding of the International Conference On Robotics and Automation.* Belgium.1998; 3153–3158.

Lalonde RJ, Hartley TT and De Abreu-Garcia JA. Least Squares Model Reduction. *Journal of the Franklin Institute*. 1992; 329 (2): 215–240.

Lategahn H, Graf T and Hasberg C. Mapping in Dynamic Environments Using Stereo Vision. *Intelligent Vehicles Symposium*. Germany. 2011; 150–156.

Leibe B, Cornelis N, Cornelis K and Van L. Dynamic 3D Scene Analysis from a Moving Vehicle. In: *Proceeding of the Conference on Computer Vision and Pattern.* Minneapolis, USA. 2007.

LeSage JR and Longoria RG. Mission Feasibility Assessment for Mobile Robotic Systems Operating In Stochastic Environments, *Journal of Dynamic Systems, Measurement and Control*. 2014; 137(3): 1–11.

Liaquat A.K, Juwairiyah N, Umar K and Zahid H. PID control of a biped robot. In: *Proceeding of the 8th WSEAS International Conference on Robotics, Control and Manufacturing Technology* (ROCOM '08). Hangzhou, China. 2008.

Lin C, Jiang S, Pu Y and Song K. Robust Ground Plane Detection for Obstacle voidance of Mobile Robots Using a Monocular Camera. In: *Proceeding of the International Conference on Intelligent Robots and Systems*. Taiwan. 2010; 3706–3711.

Lin C, Tseng C and Tseng T. A Hardware-In-The-Loop Dynamics Simulator for Motorcycle Rapid Controller Prototyping. *Control Engineering Practice.* 2006; 14(12): 1467–1476.

Linhui L, Mingheng Z, Lie G and Yibing Z. Stereo Vision Based Obstacle Avoidance Path-planning for Cross-country Intelligent Vehicle. In: *Proceeding of the IEEE International Conference on Fuzzy system and knowledge Discovery*. Tianjin, China. 2009; 463–467.

Liu Y, Chen D and Zhang S. Obstacle Avoidance Method Based on the Movement Trend of Dynamic Obstacles. In.*: Proceeding of 3rd International conference on control and robotics engineering*. Nagoya, Japan. 2018; 45–50.

Louca LS, Stein J and Rideout D. Generating Proper Integrated Dynamic Models for Vehicle Mobility Using a Bond Graph Formulation. In: *Proceedings of the 2001 International conference on Bond Graph Modeling*. Phoenix. 2001; 339–345.

Louca LS, Stein JL, Hulbert GM and Sprague J. Proper Model Generation: An Energy Based Methodology. *Simulation Series*. 1997; 29: 44–49.

Margolis D and Shim T. A Bond Graph Model Incorporating Sensors, Actuators, and Vehicle Dynamics for Developing Controllers for Vehicle Safety. *Journal of the Franklin Institute.* 2001; 338: 21–34.

Medjaher K, Samantaray AK, Ould-Bouamama B and Staroswiecki M. Supervision of an industrial steam generator. Part II: Online implementation, *Control Engineering Practice*. 2006; 14(1): 85–96.

Merzouki R, Ould-Bouamama B, Djeziri M and Bouteldja M. Modelling and Estimation of Tire–Road Longitudinal Impact Efforts Using Bond Graph Approach. *Mechatronics.* 2007a; 17(2-3): 93–108.

Merzouki R, Medjaher K, Djeziri M. and Ould-Bouamama B. Backlash Fault Detection in Mechatronic System. *Mechatronics.* 2007b; 17(6): 299–310.

Mishra AK, Raina R, Yadav SB, Verma A, Sarangi S and Saha A. Modeling and Simulation of Levitating Ball by Electromagnet using Bond Graph. In: *Proceedings of the 1st International and 16th National Conference on Machines and Mechanisms (iNaCoMM2013).* Roorkee, India. 2013; 42–48.

Mohomed YA and Ali HA. Wheeled Mobile Robot Obstacle Avoidance using Compass and Ultrasonic. *Universal journal of control and automation*. 2018; 6(1): 13–18.

Moore BC. Principal Component Analysis in Linear Systems: Controllability, Observability and Model Reduction. *IEEE Transactions on Automation and Control*. 1981; 26 (1): 17–32.

Mondada F, Franzi E and Guignard A. The Development of Khepera. In: *Proceeding of first International Khepera workshop*. Paderborn, 1999.

Moravec H. Obstacle Avoidance and Navigation in the Real World by A Seeing Robot Rover. *PhD thesis in computer science*. 1980.

Mordechai BA and Mondada F. Local Navigation: Obstacle Avoidance. *Chapter in Elements of Robotics*. Springer. 2017; 111–126.

Mukherjee A. Users manual of SYMBOLS Shakti. *http://www.htcinfo.com*. 2006.

Mukherjee A, Karmakar R and Samantaray AK. Bond Graph in Modelling, Simulation and Fault Identification. *CRC press*. 2006.

Nagar SK and Singh SK. An Algorithmic Approach for System Decomposition and Balanced Realized Model Reduction. *Journal of the Franklin Institute*. 2004; 341: 615–630.

Nagel J, Trepagnier P and Koutsougeras C. The Culebra Algorithm for Path Planning and Obstacle Avoidance in Kat-5. In: *Proceeding of the 18th IEEE International Conference on Tools with Artificial Intelligence*. Arlington. 2006; 247–253.

Nehmzow U. Mobile Robotics: A Practical Introduction. *Springer-Verlag*. London. 2003.

Ngwompo R and Gawthrop P. Bond Graph-Based Simulation of Non-Linear Inverse Systems using Physical Performance Specifications. *Journal of the Franklin Institute.* 1999; 336(8): 1225–1247.

Orbak AY, Eskinat E and Turkay OS. Physical Parameter Sensitivity of System Eigen Values and Physical Model Reduction. *Journal of the Franklin Institute*. 2004; 341 (7): 631–655.

Orbak AY, Türkay OS, Eskinat E and Youcef-Toumi K. Model Reduction in the Physical Domain. *Journal of Systems and Control Engineering*. 2003; 217 (6): 481–496.

Orbak AY. Physical Model Reduction: A Bond Graph Approach for Engineering System. *Lap Lambert Academics*. 2010.

Otter M, Elmqvist H and Cellier F. Modeling of Multibody Systems with the Object-Oriented Modeling Language Dymola. *Nonlinear Dynamics.* 1996; 9(1-2): 91–112.

Ould-Bouamama B, Harabi ER, Abdelkrim MN and Gayed BMK. Bond Graphs for the Diagnosis of Chemical Processes. *Computers and Chemical Engineering*. 2012; 36: 301–324.

Ould Bouamama B, Medjaher K, Bayart M, Samantaray AK and Conrard B. Fault Detection and Isolation of Smart Actuators using Bond Graphs and External Models. *Control Engineering Practice.* 2005; 13(2): 159–175.

Ould Bouamama B, Medjaher K, Samantaray AK and Staroswiecki M. Supervision of an Industrial Steam Generator. Part I: Bond Graph Modelling. *Control Engineering Practice.* 2006; 14(1): 71–83.

Pacejka H. Modeling Complex Vehicle Systems Using Bond Graphs. *Journal of the Franklin Institute.* 1985; 319(1): 67–81.

Pacejka H. Tyre and Vehicle Dynamics. *Oxford*. 2001.

Parkash O, Samantaray AK and Bhattacharyya R. Model-based diagnosis of multiple faults in hybrid dynamical systems with dynamically updated parameters, *IEEE Transactions On Systems, Man And Cybernetics: Systems*. 2017; 99: 1–20.

Pathak K, Franch J and Agrawal S. Velocity and Position Control of a Wheeled Inverted Pendulum by Partial Feedback Linearization. *IEEE Transactions on Robotics.* 2005; 21: 16–28.

Pathak PM, Kumar R, Mukherjee A and Dasgupta A. A Scheme for Robust Trajectory Control of Space Robots. *Simulation Modelling Practice and Theory.* 2008; 16(9): 1337–1379.

Pathak P, Mukherjee A and Dasgupta A. Impedance Control of Space Robots using Passive Degrees of Freedom in Controller Domain. *Transactions of ASME, Journal Dynamic System Measurement and Control.* 2005; 127(4): 564–578.

Pathak P, Samantaray AK and Merzouki R. Reconfiguration of Directional Handling of an Autonomous Vehicle. In: *Proceeding of the IEEE Region 10 colloquium and the third ICIIS*. Kharagpur. 2008; 1–6.

Paynter H. Analysis and Design of Engineering Systems. *M.I.T. Press. Cambridge*.1961.

Phuc TD, Hoang DV, Hak K.K and Sang BK. A New Approach for Development of Quadruped Robot Based on Biological Concepts. *International journal of precision engineering and manufacturing*. 2010; 11(4): 559–568.

Pingzeng L, Shusheng B and Zang G. Obstacle Avoidance System for Agricultural Robots Based on Multi-sensor Information Fusion. In: *Proceeding of the International Conference on Computer Science and Network Technology*. China. 2011; 1181–1185.

Playter R, Buehler M and Raibert M. Big Dog. In: *Proceeding of Unmanned system technology.* Florida. 2006; 6220–6230.

Ponticelli R, & Armada M. Dynamic Simulation System for the Biped Robot Silo2, In: *Proceedings of the 9th International Conference on Climbing and Walking Robots*. Brussels, Belgium. 2006; 490–494.

Raibert M. BigDog, The Rough-Terrain Quadruped Robot. In: *Proceeding of the IFAC*. Seoul, Korea. 2008; 17: 6–9.

Rashmi A and Bera TK. Tarjectory Tracking of 3D Hybrid Manipulator Through Human Hand Motion. *Arabian Journal for Science and Engineering*. 2018; 1–13.

Rath AK, Parhi DR, Das HC, Muni MK and Kumar PB. Analysis and use of Fuzzy Intelligent Technique For Navigation of Humanoid Robot in Obstacle Prone Zone. *Defence Technology*. 2018; In press.

Remy C, Siegwart R and Gehring C. A Compliant Quadrupedal Robot for Fast, Efficient, and Versatile Locomotion. In: *Proceeding of the 15th International Conference on Climbing and Walking Robot - CLAWAR*. USA. 2012; 1–8.

Rocchi A, Hoffman EM, Fanioli E, Tsagarakis NG. A Whole-Body Stack-of-Tasks Compliant Control for the Humanoid Robot. In: *Proceeding of the COMAN*. Germany. 2015.

Safonov MG and Chiang RY. A Schur Method for Balanced-Truncation Model Reduction. *IEEE Transactions on Automation and Control*. 1989; 34: 729–733.

Saha SK and Schiehlen WO. Recursive kinematics and dynamics for parallel structured closed loop multibody systems, *Mechanics of Structures and Machines*. 2001; 29(2): 143–175.

Sahin E. Swarm-Bot: Pattern Formation in A Swarm of Self-Assembling Mobile Robots. In: *Proceeding of the IEEE International conference on systems, man and cybernetics*. Tunisia. 2002; 63–67.

Samantaray AK and Ould-Bouamama B. Model-based Process Supervision. *London: Springer*. 2008.

Sangok S, Wang A, Meng Y, Otten D, Lang J, Sangbae K. Design Principles for Highly Efficient Quadrupeds and Implementation on The MIT Cheetah Robot In.: *Proceeding of IEEE ICRA*. Karlsruhe, Germany. 2013; 3307–3312.

Schilders W. Introduction to Model Order Reduction. In: *Model order reduction: Theory and applications*. Springer. 2008.

Singh M. Model Reduction in Vehicle Dynamic Systems. *M.E. Thesis*. Thapar Institute of Engineering and Technology (Deemed to be University), Patiala, India. 2015.

Skelton RE. and Yousuff A. Component Cost Analysis of Large Scale Systems. *International Journal of Control*. 1983; 37 (2): 285–304.

Slone RD, Lee FAJ and Lee R. A Comparison of Some Model Order Reduction Techniques. *Electromagnetics*. 2002; 22: 275–289.

Sudhakara P, Ganapathy V, Priyadharshini B and Sundaran K. Obstacle Avoidance and Navigation Planning of A Wheeled Mobile Robot Using Amended Artificial Potential Field Method. *Procedia Computer Science*. 2018; 133: 998–1004.

Sueur C and Dauphin-Tanguy G. Bond Graph Approach for Multi-Time Scale System Analysis. *Journal of the Franklin Institute*. 1991; 328 (5): 1005–1026.

Thoma J and Ould Bouamama B. Modelling and Simulation in Thermal and Chemical Engineering. *Springer*. 2000.

Tomar P and Das S. Bond graph modeling and simulation of a closed-loop feed drive system for a CNC machine. *SAE world Congress*. Detroit, Michigan. 2007.

Wang L, Liu Y and Deng H. Obstacle-avoidance Path Planning for Soccer Robots Using Particle Swarm Optimization. In: *Proceeding of the International Conference on Robotics and Biomimetics.* China. 2006; 1233–1238.

Walter G. A Machine That Learns. *Scientific American.* 1948; 185(2): 60–63.

William JC and Zemach K. The MARCbot: The Army Program That Revolutionized Robotics for Patrol War Fighters. *IQT Quarterly*. 2016; 3(1): 15–17.

Xihang H. FF-Pade Method of Model Reduction in Frequency Domain. *IEEE Transactions of Automation and Control*. 1987; 32 (3): 243–246.

Xu F, Wang S C and Yang W. Local Logic Optimization Algorithm for Autonomous Mobile Robot Based on Fuzzy Logic. In*.: Proceeding of Chinese control and decision conference*. Shenyang, China. 2018; 4160–4165.

Yang J. Tripod Gaits for Fault Tolerance Of Hexapod Walking Machines with a Locked Joint Failure. *Robotics Autonomous System.* 2005; 52(2-3): 180–189.

Yang J. Omni-Directional Walking of Legged Robots with a Failed Leg. *Mathematical Computer Model.* 2008a; 47(11-12): 1372–1388.

Yang J. Two-Phase Discontinuous Gaits For Quadruped Walking Machines With A Failed Leg. *Robotics Autonomous System.* 2008b; 56(9): 728–737.

Yang JM, Ko DI, Shim KH and Hwang SK. Fault Tolerant Gaits of Legged Robots for Locked Joint Failures. In: *Proceeding of the IEEE International Conference on Systems, man, and cybernetics*. Nashville, Tennessee. 2000; 5: 3300–3305.

Ye Y and Youcef-Toumi K. Subsystem's Influence on a System Eigen Value. In: *Proceedings of the IEEE Southeastcon*. Cambridge, MA. 2000; 261–267.

Yen CT and Cheng MF. A Study of Fuzzy Control with Ant Colony Algorithm used in Mobile Robot for Shortest Path Planning and Obstacle Avoidance. *Microsystem Technology*. Springer. 2018; 24(1): 125–135.

Yu Z, Huang Q, Ma G, Chen X, Zhang W, Li J, Gao J. Design and Development of the Humanoid Robot BHR-5. *Advance Mechanical Engeering.* 2014; (6): 56–61.

Zarate L, Becker M and Garrido B. An Artificial Neural Network Structure able to Obstacle Avoidance Behavior Used in Mobile Robots. In: *Proceeding of the Annual Conference of IEEE Electronic Society.* Austria. 2002; 2457–2461.

Zhao X, & Liu Y. Modeling of Biped Robot, In: *Proceedings of the Chinese Control and Decision Conference (CCDC)*. Xuzhou, China. 2010; 3233–3238.

Zheng N, Tang S, Cheng H and Li Q. Toward Intelligent Driver Assistance and Safety Warning System. *IEEE Intelligent System*. 2004; 19(2): 8–11.

Zhou K, Doyle JC and Glover K. Robust and Optimal Control. *Prentice Hall*, Inc. Engelwood Cliffs, New Jersey. 1996.

APPENDIX A

I. Newton-Euler Equations: The Newton-Euler equations of the vehicle body with attached body fixed axes aligned with the principal axes of inertia are as follows:

$$\sum F_{\mathrm{x}} = m_{\mathrm{c}}\ddot{x}_{\mathrm{c}} + m_{\mathrm{c}}(\dot{z}_{\mathrm{c}}\dot{\theta}_{\mathrm{cy}} - \dot{y}_{\mathrm{c}}\dot{\theta}_{\mathrm{cz}}) \tag{A1}$$

$$\sum F_{\mathrm{y}} = m_{\mathrm{c}}\ddot{y}_{\mathrm{c}} + m_{\mathrm{c}}(\dot{x}_{\mathrm{c}}\dot{\theta}_{\mathrm{cz}} - \dot{z}_{\mathrm{c}}\dot{\theta}_{\mathrm{cx}}) \tag{A2}$$

$$\sum F_{\mathrm{z}} = m_{\mathrm{c}}\ddot{z}_{\mathrm{c}} + m_{\mathrm{c}}(\dot{y}_{\mathrm{c}}\dot{\theta}_{\mathrm{cx}} - \dot{x}_{\mathrm{c}}\dot{\theta}_{\mathrm{cy}}) \tag{A3}$$

$$\sum M_{\mathrm{x}} = J_{\mathrm{cx}}\ddot{\theta}_{\mathrm{cx}} - \dot{\theta}_{\mathrm{cy}}\dot{\theta}_{\mathrm{cz}}(J_{\mathrm{cy}} - J_{\mathrm{cz}}) \tag{A4}$$

$$\sum M_{\mathrm{y}} = J_{\mathrm{cy}}\ddot{\theta}_{\mathrm{cy}} - \dot{\theta}_{\mathrm{cz}}\dot{\theta}_{\mathrm{cx}}(J_{\mathrm{cz}} - J_{\mathrm{cx}}) \tag{A5}$$

$$\sum M_{\mathrm{z}} = J_{\mathrm{cz}}\ddot{\theta}_{\mathrm{cz}} - \dot{\theta}_{\mathrm{cx}}\dot{\theta}_{\mathrm{cy}}(J_{\mathrm{cx}} - J_{\mathrm{cy}}) \tag{A6}$$

II. Coordinate Transformation (CTF): If ψ, θ and ϕ are the Z-Y-X Cardan angles, $\dot{X}, \dot{Y}$ and $\dot{Z}$ are the inertial velocities and $\dot{x}, \dot{y}$ and $\dot{z}$ are the body-fixed velocities, the equations for body-fixed to inertial coordinate transformation ($\mathrm{CTF_{BI}}$) are given by

$$\dot{X} = \mu_1\dot{x} + \mu_2\dot{y} + \mu_3\dot{z} \tag{A7}$$

$$\dot{Y} = \mu_4\dot{x} + \mu_5\dot{y} + \mu_6\dot{z} \tag{A8}$$

$$\dot{Z} = \mu_7\dot{x} + \mu_8\dot{y} + \mu_9\dot{z} \tag{A9}$$

where

$\mu_1 = \cos\psi\cos\theta$ $\quad$ $\mu_2 = \cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi$ $\quad$ $\mu_3 = \cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi$

$\mu_4 = \sin\psi\cos\theta$ $\quad$ $\mu_5 = \sin\psi\sin\theta\sin\phi + \cos\psi\cos\phi$ $\quad$ $\mu_6 = \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi$

$\mu_7 = -\sin\theta$ $\quad$ $\mu_8 = \cos\theta\sin\phi$ $\quad$ $\mu_9 = \cos\theta\cos\phi$

Similarly, the equations for inertial to body-fixed coordinate transformation ($\mathrm{CTF_{IB}}$) are given by

$$\dot{x} = \lambda_1\dot{X} + \lambda_2\dot{Y} + \lambda_3\dot{Z} \tag{A10}$$

$$\dot{y} = \lambda_4\dot{X} + \lambda_5\dot{Y} + \lambda_6\dot{Z} \tag{A11}$$

$$\dot{z} = \lambda_7\dot{X} + \lambda_8\dot{Y} + \lambda_9\dot{Z} \tag{A12}$$

Where

$\lambda_1 = \cos\psi\cos\theta$ $\quad$ $\lambda_2 = \sin\psi\cos\theta$ $\quad$ $\lambda_3 = -\sin\theta$

$\lambda_4 = \cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi$ $\quad$ $\lambda_5 = \sin\psi\sin\theta\sin\phi + \cos\psi\cos\phi$ $\quad$ $\lambda_6 = \cos\theta\sin\phi$

$\lambda_7 = \cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi$ $\quad$ $\lambda_8 = \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi$ $\quad$ $\lambda_9 = \cos\theta\cos\phi$

APPENDIX B

$$ARR1 = R_b \times i - V + \frac{d\dot{\theta}}{dt} \times m_u \tag{B1}$$

$$ARR2 = L_m \times \frac{d\ddot{\theta}}{dt} - i \times m_u + K_{pad} \times \left(\frac{d\dot{\theta}}{dt} \times mu_{17} - \frac{d\,\dot{x}_{cg}}{dt} - mu_1 \times \frac{d\,\dot{\theta}_{cg}}{dt} \right) + R_{pad} \times \left(\frac{d\ddot{\theta}}{dt} \times mu_{17} - \frac{d\,\ddot{x}_{cg}}{dt} - mu_1 \times \frac{d\,\ddot{\theta}_{cg}}{dt} \right) +$$
$$mu_{15} \times \left(\left(\frac{d\ddot{\theta}}{dt} \times mu_{15} - \frac{\ddot{y}_{pg}}{dt} - mu_{12} \times \frac{d\,\ddot{\theta}_{pg}}{dt} \right) \times K_{216} \right) + mu_{16} \times \left(\left(\frac{d\ddot{\theta}}{dt} \times mu_{16} - \frac{\ddot{x}_{pg}}{dt} - mu_{11} \times \frac{d\,\ddot{\theta}_{pg}}{dt} \right) \times K_{215} \right) \tag{B2}$$

$$ARR3 = M_p \times \frac{d\,\ddot{x}_{pg}}{dt} - \left(\left(\frac{d\,\ddot{x}_{cg}}{dt} + mu_1 \times \frac{d\,\ddot{\theta}_{cg}}{dt} - \frac{d\,\dot{x}_{pg}}{dt} - mu_{11} \times \frac{d\,\ddot{\theta}_{pg}}{dt} \right) \times mu_6 + \left(\frac{d\,\ddot{y}_{cg}}{dt} + mu_2 \times \frac{d\,\ddot{\theta}_{cg}}{dt} - \frac{d\,\dot{y}_{pg}}{dt} - \frac{d\,\ddot{\theta}_{pg}}{dt} \times mu_{12} \right) \times mu_8 \right)$$
$$\times mu_6 \times R_f - \left(\frac{d\dot{\theta}}{dt} \times mu_{16} - \frac{d\,\dot{x}_{pg}}{dt} - mu_{11} \times \frac{d\,\dot{\theta}_{pg}}{dt} \right) \times K_{215} + K_{bend} \times \left(\left(mu_9 \times \frac{d\,\dot{\theta}_{pg}}{dt} + mu_{13} \times \frac{d\,\dot{x}_{pg}}{dt} + mu_{14} \times \frac{d\,\dot{y}_{pg}}{dt} \right) + 1 \Big/ mu_4 \right) \times$$
$$\frac{d\,\dot{\theta}_{cg}}{dt} - \left(\frac{d\,\dot{x}_{cg}}{dt} \Big/ mu_5 \right) - \left(\frac{d\,\dot{y}_{cg}}{dt} \Big/ mu_7 \right) + r_{pad} \times \left(\left(mu_9 \times \frac{d\,\ddot{\theta}_{pg}}{dt} + mu_{13} \times \frac{d\,\ddot{x}_{pg}}{dt} + mu_{14} \times \frac{d\,\ddot{y}_{pg}}{dt} \right) + 1 \Big/ mu_4 \right) \times \frac{d\,\ddot{\theta}_{cg}}{dt} - \left(\frac{d\,\ddot{x}_{cg}}{dt} \Big/ mu_5 \right) -$$
$$\left(\frac{d\,\ddot{y}_{cg}}{dt} \Big/ mu_7 \right) + K_{bend} \times \left(\left(mu_{10} \times \frac{d\,\dot{\theta}_{pg}}{dt} + mu_{13} \times \frac{d\,\dot{x}_{pg}}{dt} + mu_{14} \times \frac{d\,\dot{y}_{pg}}{dt} \right) + 1 \Big/ mu_3 \right) \times \frac{d\,\dot{\theta}_{cg}}{dt} - \left(\frac{d\,\dot{x}_{cg}}{dt} \Big/ mu_5 \right) - \left(\frac{d\,\dot{y}_{cg}}{dt} \Big/ mu_7 \right) +$$
$$r_{pad} \times \left(\left(mu_{10} \times \frac{d\,\ddot{\theta}_{pg}}{dt} + mu_{13} \times \frac{d\,\ddot{x}_{pg}}{dt} + mu_{14} \times \frac{d\,\ddot{y}_{pg}}{dt} \right) + 1 \Big/ mu_3 \right) \times \frac{d\,\ddot{\theta}_{cg}}{dt} - \left(\frac{d\,\ddot{x}_{cg}}{dt} \Big/ mu_5 \right) - \left(\frac{d\,\ddot{y}_{cg}}{dt} \Big/ mu_7 \right) \tag{B3}$$

$$ARR4 = M_p \times \frac{d\ddot{y}_{pg}}{dt} - \left(\left(\frac{d\ddot{x}_{cg}}{dt} + mu_1 \times \frac{d\ddot{\theta}_{cg}}{dt} - \frac{d\ddot{x}_{pg}}{dt} - mu_{11} \times \frac{d\ddot{\theta}_{pg}}{dt}\right) \times mu_6 + \left(\frac{d\ddot{y}_{cg}}{dt} + mu_2 \times \frac{d\ddot{\theta}_{cg}}{dt} - \frac{d\ddot{y}_{pg}}{dt} - \frac{d\ddot{\theta}_{pg}}{dt} \times mu_{12}\right) \times mu_8\right)$$
$$\times mu_6 \times R_f - \left(\frac{d\dot{\theta}}{dt} \times mu_{15} - \frac{d\dot{y}_{pg}}{dt} - mu_{12} \times \frac{d\dot{\theta}_{pg}}{dt}\right) \times K_{216} + K_{bend} \times \left(\left(mu_9 \times \frac{d\dot{\theta}_{pg}}{dt} + mu_{13} \times \frac{d\dot{x}_{pg}}{dt} + mu_{14} \times \frac{d\dot{y}_{pg}}{dt}\right) + \frac{1}{mu_4}\right) \times$$
$$\frac{d\dot{\theta}_{cg}}{dt} - \left(\frac{\frac{d\dot{x}_{cg}}{dt}}{mu_5}\right) - \left(\frac{\frac{d\dot{y}_{cg}}{dt}}{mu_7}\right) + r_{pad} \times \left(\left(mu_9 \times \frac{d\ddot{\theta}_{pg}}{dt} + mu_{13} \times \frac{d\ddot{x}_{pg}}{dt} + mu_{14} \times \frac{d\ddot{y}_{pg}}{dt}\right) - \frac{1}{mu_4}\right) \times \frac{d\ddot{\theta}_{cg}}{dt} - \left(\frac{\frac{d\ddot{x}_{cg}}{dt}}{mu_5}\right) -$$
$$\left(\frac{\frac{d\ddot{y}_{cg}}{dt}}{mu_7}\right) + K_{bend} \times \left(\left(mu_{10} \times \frac{d\dot{\theta}_{pg}}{dt} + mu_{13} \times \frac{d\dot{x}_{pg}}{dt} + mu_{14} \times \frac{d\dot{y}_{pg}}{dt}\right) - \frac{1}{mu_3}\right) \times \frac{d\dot{\theta}_{cg}}{dt} - \left(\frac{\frac{d\dot{x}_{cg}}{dt}}{mu_5}\right) - \left(\frac{\frac{d\dot{y}_{cg}}{dt}}{mu_7}\right) +$$
$$r_{pad} \times \left(\left(mu_{10} \times \frac{d\ddot{\theta}_{pg}}{dt} + mu_{13} \times \frac{d\ddot{x}_{pg}}{dt} + mu_{14} \times \frac{d\ddot{y}_{pg}}{dt}\right) - \frac{1}{mu_3}\right) \times \frac{d\ddot{\theta}_{cg}}{dt} - \left(\frac{\frac{d\ddot{x}_{cg}}{dt}}{mu_5}\right) - \left(\frac{\frac{d\ddot{y}_{cg}}{dt}}{mu_7}\right)$$

(B4)

$$ARR5 = M_c \times \frac{d\dddot{x}_{cg}}{dt} - \left(\left(\frac{d\ddot{x}_{cg}}{dt} + mu1 \times \frac{d\ddot{\theta}_{cg}}{dt} - \frac{d\ddot{x}_{pg}}{dt} - mu11 \times \frac{d\ddot{\theta}_{pg}}{dt}\right) \times mu6 + \left(\frac{d\ddot{y}_{cg}}{dt} + mu2 \times \frac{d\ddot{\theta}_{cg}}{dt} - \frac{d\ddot{y}_{pg}}{dt} - \frac{d\ddot{\theta}_{pg}}{dt} \times mu12\right) \times mu8\right)$$
$$\times mu6 \times R_f + K_{pad} \times \left(\frac{d\dot{\theta}}{dt} \times mu17 - \frac{d\dot{x}_{cg}}{dt} - mu1 \times \frac{d\dot{\theta}_{cg}}{dt}\right) + r_{pad} \times \left(\frac{d\ddot{\theta}}{dt} \times mu17 - \frac{d\ddot{x}_{cg}}{dt} - mu1 \times \frac{d\ddot{\theta}_{cg}}{dt}\right) + K_{pad\,body} \times$$
$$\left(\frac{d\dot{x}_{cg}}{dt} + \frac{d\dot{x}_{cg}}{dt} \times mu1 - \frac{d\dot{x}_b}{dt} - \frac{d\dot{\theta}_b}{dt}\right) + mu9 \times \frac{d\dot{\theta}_{pg}}{dt} + mu13 \times \frac{d\dot{x}_{pg}}{dt} + mu14 \times \frac{d\dot{y}_{pg}}{dt} - \frac{1}{mu_4} \times \frac{d\dot{\theta}_{pg}}{dt} - \left(\frac{\frac{d\dot{x}_{cg}}{dt}}{mu5}\right) - \left(\frac{\frac{d\dot{y}_{cg}}{dt}}{mu7}\right)$$
$$+ mu10 \times \frac{d\dot{\theta}_{pg}}{dt} + mu13 \times \frac{d\dot{x}_{pg}}{dt} + mu14 \times \frac{d\dot{y}_{pg}}{dt} - \frac{1}{mu3} \times \frac{d\dot{\theta}_{pg}}{dt} - \left(\frac{\frac{d\dot{x}_{cg}}{dt}}{mu5}\right) - \left(\frac{\frac{d\dot{y}_{cg}}{dt}}{mu7}\right) + mu9 \times \frac{d\ddot{\theta}_{pg}}{dt} + mu13 \times \frac{d\ddot{x}_{pg}}{dt}$$
$$+ mu14 \times \frac{d\ddot{y}_{pg}}{dt} - \frac{1}{mu4} \times \frac{d\ddot{\theta}_{cg}}{dt} - \left(\frac{\frac{d\ddot{x}_{cg}}{dt}}{mu5}\right) - \left(\frac{\frac{d\ddot{y}_{cg}}{dt}}{mu7}\right) + mu10 \times \frac{d\ddot{\theta}_{pg}}{dt} + mu13 \times \frac{d\ddot{x}_{pg}}{dt}$$
$$+ mu14 \times \frac{d\ddot{y}_{pg}}{dt} - \frac{1}{mu3} \times \frac{d\ddot{\theta}_{pg}}{dt} - \left(\frac{\frac{d\ddot{x}_{cg}}{dt}}{mu5}\right) - \left(\frac{\frac{d\ddot{y}_{cg}}{dt}}{mu7}\right)$$

(B5)

$$ARR6 = Mc \times \frac{d\ddot{y}_{cg}}{dt} - \left(\left(\frac{d\ddot{x}_{cg}}{dt} + mu1 \times \frac{d\ddot{\theta}_{cg}}{dt} - \frac{d\ddot{x}_{pg}}{dt} - mu11 \times \frac{d\ddot{\theta}_{pg}}{dt} \right) \times mu6 + \left(\frac{d\ddot{y}_{cg}}{dt} + mu2 \times \frac{d\ddot{\theta}_{cg}}{dt} - \frac{d\ddot{y}_{pg}}{dt} - \frac{d\ddot{\theta}_{pg}}{dt} \times mu12 \right) \times mu8 \right)$$
$$\times mu6 \times Rf + K_{pad} \times \left(-\frac{d\dot{\theta}}{dt} \times mu2 \times \frac{d\dot{\theta}_{cg}}{dt} \right) + r_{pad} \times \left(-\frac{d\ddot{\theta}}{dt} \times mu2 \times \frac{d\ddot{\theta}_{cg}}{dt} \right) + K_{pad}{}_{body} \times$$
$$\left(\frac{d\dot{y}_{cg}}{dt} + \frac{d\dot{\theta}_{cg}}{dt} \times mu2 - \frac{d\dot{y}_b}{dt} - \frac{d\dot{\theta}_b}{dt} \right) + mu9 \times \frac{d\dot{\theta}_{pg}}{dt} + mu13 \times \frac{d\dot{x}_{pg}}{dt} + mu14 \times \frac{d\dot{y}_{pg}}{dt} - 1/_{mu4} \times \frac{d\dot{\theta}_{pg}}{dt} - \left(\frac{d\dot{x}_{cg}}{dt} \Big/ mu5 \right) - \left(\frac{d\dot{y}_{cg}}{dt} \Big/ mu7 \right)$$
$$+ mu10 \times \frac{d\dot{\theta}_{pg}}{dt} + mu13 \times \frac{d\dot{x}_{pg}}{dt} + mu14 \times \frac{d\dot{y}_{pg}}{dt} - 1/_{mu3} \times \frac{d\dot{\theta}_{pg}}{dt} - \left(\frac{d\dot{x}_{cg}}{dt} \Big/ mu5 \right) - \left(\frac{d\dot{y}_{cg}}{dt} \Big/ mu7 \right) + mu9 \times \frac{d\ddot{\theta}_{pg}}{dt} + mu13 \times \frac{d\ddot{x}_{pg}}{dt}$$
$$+ mu14 \times \frac{d\ddot{y}_{pg}}{dt} - \frac{1}{mu4} \times \frac{d\ddot{\theta}_{cg}}{dt} - \left(\frac{d\ddot{x}_{cg}}{dt} \Big/ mu5 \right) - \left(\frac{d\ddot{y}_{cg}}{dt} \Big/ mu7 \right) + mu10 \times \frac{d\ddot{\theta}_{pg}}{dt} + mu13 \times \frac{d\ddot{x}_{pg}}{dt}$$
$$+ mu14 \times \frac{d\ddot{y}_{pg}}{dt} - 1/_{mu3} \times \frac{d\ddot{\theta}_{pg}}{dt} - \left(\frac{d\ddot{x}_{cg}}{dt} \Big/ mu5 \right) - \left(\frac{d\ddot{y}_{cg}}{dt} \Big/ mu7 \right)$$

(B6)

$$ARR7 = J_c \times \frac{d\ddot{\theta}_{cg}}{dt} - \left(\left(\frac{d\ddot{x}_{cg}}{dt} + mu1 \times \frac{d\ddot{\theta}_{cg}}{dt} - \frac{d\ddot{x}_{pg}}{dt} - mu11 \times \frac{d\ddot{\theta}_{pg}}{dt} \right) \times mu6 + \left(\frac{d\ddot{y}_{cg}}{dt} + mu2 \times \frac{d\ddot{\theta}_{cg}}{dt} - \frac{d\ddot{y}_{pg}}{dt} - \frac{d\ddot{\theta}_{pg}}{dt} \times mu12 \right) \times mu8 \right)$$
$$\times mu6 \times Rf + K_{pad} \times \left(-\frac{d\dot{\theta}}{dt} \times mu1 \times \frac{d\dot{\theta}_{cg}}{dt} \right) + r_{pad} \times \left(\frac{d\ddot{\theta}}{dt} \times mu1 \times \frac{d\ddot{\theta}_{cg}}{dt} \right) + K_{pad\,body} \times$$
$$\left(\frac{d\dot{x}_{cg}}{dt} + \frac{d\dot{\theta}_{cg}}{dt} \times mu1 - \frac{d\dot{x}_b}{dt} - \frac{d\dot{\theta}_b}{dt} \right) + \left(\begin{array}{l} \left(\frac{d\ddot{x}_{cg}}{dt} + mu1 \times \frac{d\ddot{\theta}_{cg}}{dt} - \frac{d\ddot{x}_{pg}}{dt} - mu11 \times \frac{d\ddot{\theta}_{pg}}{dt} \right) \times \\ mu6 + \left(\frac{d\ddot{y}_{cg}}{dt} + mu2 \times \frac{d\ddot{\theta}_{cg}}{dt} - \frac{d\ddot{y}_{pg}}{dt} - \frac{d\ddot{\theta}_{pg}}{dt} \times mu12 \right) \times mu8 \end{array} \right)$$
$$\times mu6 \times Rf + K_{pad} \times \left(-\frac{d\dot{y}_{cg}}{dt} - \frac{d\dot{\theta}}{dt} \times mu1 \times \frac{d\dot{\theta}_{cg}}{dt} \right) + r_{pad} \times \left(-\frac{d\dot{y}_{cg}}{dt} \frac{d\ddot{\theta}}{dt} \times mu1 \times \frac{d\ddot{\theta}_{cg}}{dt} \right) + K_{pad\,body} \times$$
$$\left(\frac{d\dot{y}_{cg}}{dt} + \frac{d\dot{\theta}_{cg}}{dt} \times mu2 - \frac{d\dot{y}_b}{dt} - \frac{d\dot{\theta}_b}{dt} \right) + K_{bend} \times \left(\left(mu9 \times \frac{d\dot{\theta}_{pg}}{dt} + mu13 \times \frac{d\dot{x}_{pg}}{dt} + mu14 \times \frac{d\dot{y}_{pg}}{dt} \right) - 1/mu4 \right)$$
$$\times \frac{d\dot{\theta}_{cg}}{dt} - \left(\frac{d\dot{x}_{cg}}{dt} \Big/ mu5 \right) - \left(\frac{d\dot{y}_{cg}}{dt} \Big/ mu7 \right) + r_{pad} \times \left(\left(mu9 \times \frac{d\ddot{\theta}_{pg}}{dt} + mu13 \times \frac{d\ddot{x}_{pg}}{dt} + mu14 \times \frac{d\ddot{y}_{pg}}{dt} \right) - 1/mu4 \right)$$
$$\times \frac{d\ddot{\theta}_{cg}}{dt} - \left(\frac{d\ddot{x}_{cg}}{dt} \Big/ mu5 \right) - \left(\frac{d\ddot{y}_{cg}}{dt} \Big/ mu7 \right) + K_{bend} \times \left(\left(mu10 \times \frac{d\dot{\theta}_{pg}}{dt} + mu13 \times \frac{d\dot{x}_{pg}}{dt} + mu14 \times \frac{d\dot{y}_{pg}}{dt} \right) - 1/mu4 \right)$$
$$\times \frac{d\dot{\theta}_{cg}}{dt} - \left(\frac{d\dot{x}_{cg}}{dt} \Big/ mu5 \right) - \left(\frac{d\dot{y}_{cg}}{dt} \Big/ mu7 \right) + r_{pad} \times \left(\left(mu10 \times \frac{d\ddot{\theta}_{pg}}{dt} + mu13 \times \frac{d\ddot{x}_{pg}}{dt} + mu14 \times \frac{d\ddot{y}_{pg}}{dt} \right) - 1/mu4 \right)$$
$$\times \frac{d\ddot{\theta}_{cg}}{dt} - \left(\frac{d\ddot{x}_{cg}}{dt} \Big/ mu5 \right) - \left(\frac{d\ddot{y}_{cg}}{dt} \Big/ mu7 \right)$$

(B7)

$$ARR8 = J_p \times \frac{d\dddot{x}_{pg}}{dt} - \left(\left(\frac{d\ddot{x}_{cg}}{dt} + mu1 \times \frac{d\ddot{\theta}_{cg}}{dt} - \frac{d\ddot{x}_{pg}}{dt} - mu11 \times \frac{d\ddot{\theta}_{pg}}{dt} \right) \times mu6 + \left(\frac{d\ddot{y}_{cg}}{dt} + mu2 \times \frac{d\ddot{\theta}_{cg}}{dt} - \frac{d\ddot{y}_{pg}}{dt} - \frac{d\ddot{\theta}_{pg}}{dt} \times mu12 \right) \times mu8 \right)$$
$$\times mu6 \times Rf - \left(\frac{d\dot{\theta}}{dt} \times mu15 - \frac{d\dot{x}_{pg}}{dt} - mu11 \times \frac{d\dot{\theta}_{pg}}{dt} \right) \times K215 + Kbend \times \left(\left(mu9 \times \frac{d\dot{\theta}_{pg}}{dt} + mu13 \times \frac{d\dot{x}_{pg}}{dt} + mu14 \times \frac{d\dot{y}_{pg}}{dt} \right) + 1/mu4 \right) \times$$
$$\frac{d\dot{\theta}_{cg}}{dt} - \left(\frac{d\dot{x}_{cg}}{dt} / mu5 \right) - \left(\frac{d\dot{y}_{cg}}{dt} / mu7 \right) + r\,pad \times \left(\left(mu9 \times \frac{d\ddot{\theta}_{pg}}{dt} + mu13 \times \frac{d\ddot{x}_{pg}}{dt} + mu14 \times \frac{d\ddot{y}_{pg}}{dt} \right) + 1/mu4 \right) \times \frac{d\ddot{\theta}_{cg}}{dt} - \left(\frac{d\ddot{x}_{cg}}{dt} / mu5 \right) -$$
$$\left(\frac{d\ddot{y}_{cg}}{dt} / mu7 \right) + Kbend \times \left(\left(mu10 \times \frac{d\dot{\theta}_{pg}}{dt} + mu13 \times \frac{d\dot{x}_{pg}}{dt} + mu14 \times \frac{d\dot{y}_{pg}}{dt} \right) + 1/mu3 \right) \times \frac{d\dot{\theta}_{cg}}{dt} - \left(\frac{d\dot{x}_{cg}}{dt} / mu5 \right) - \left(\frac{d\dot{y}_{cg}}{dt} / mu7 \right) +$$
$$r\,pad \times \left(\left(mu10 \times \frac{d\ddot{\theta}_{pg}}{dt} + mu13 \times \frac{d\ddot{x}_{pg}}{dt} + mu14 \times \frac{d\ddot{y}_{pg}}{dt} \right) + 1/mu3 \right) \times \frac{d\ddot{\theta}_{cg}}{dt} - \left(\frac{d\ddot{x}_{cg}}{dt} / mu5 \right) - \left(\frac{d\ddot{y}_{cg}}{dt} / mu7 \right) +$$
$$\frac{d\dddot{x}_{pg}}{dt} - \left(\left(\frac{d\ddot{x}_{cg}}{dt} + mu1 \times \frac{d\ddot{\theta}_{cg}}{dt} - \frac{d\ddot{x}_{pg}}{dt} - mu11 \times \frac{d\ddot{\theta}_{pg}}{dt} \right) \times mu6 + \left(\frac{d\ddot{y}_{cg}}{dt} + mu2 \times \frac{d\ddot{\theta}_{cg}}{dt} - \frac{d\ddot{y}_{pg}}{dt} - \frac{d\ddot{\theta}_{pg}}{dt} \times mu12 \right) \times mu8 \right)$$
$$\times mu6 \times Rf - \left(\frac{d\dot{\theta}}{dt} \times mu15 - \frac{d\dot{x}_{pg}}{dt} - mu11 \times \frac{d\dot{\theta}_{pg}}{dt} \right) \times K216$$

(B8)

$$ARR9 = Mb \times \frac{d\dddot{x}_b}{dt} + K\,pad\,body \times \left(\frac{d\dot{x}_{cg}}{dt} + \frac{d\dot{\theta}_{cg}}{dt} \times mu1 - \frac{d\dot{x}_b}{dt} - \frac{d\dot{\theta}_b}{dt} \right)$$

(B9)

$$ARR10 = Mb \times \frac{d\dddot{y}_b}{dt} + K\,pad\,body \times \left(\frac{d\dot{y}_{cg}}{dt} + \frac{d\dot{\theta}_{cg}}{dt} \times mu2 - \frac{d\dot{y}_b}{dt} - \frac{d\dot{\theta}_b}{dt} \right)$$

(B10)

$$ARR11 = Jb \times \frac{d\dddot{\theta}_b}{dt} + K\,pad\,body \times \left(\left(\frac{d\dot{x}_{cg}}{dt} + \frac{d\dot{\theta}_{cg}}{dt} \times mu1 - \frac{d\dot{x}_b}{dt} - \frac{d\dot{\theta}_b}{dt} \right) + \left(\frac{d\dot{y}_{cg}}{dt} + \frac{d\dot{\theta}_{cg}}{dt} \times mu2 - \frac{d\dot{y}_b}{dt} - \frac{d\dot{\theta}_b}{dt} \right) \right)$$

(B11)

Curriculum Vitae

Rajmeet Singh did his graduation (Bachelor of Technology) in Industrial Engineering from Saheed Bhagat Singh College of Engineering and Technology, Ferozepur, Punjab, India in the year 2005. He obtained his Master of Engineering (M.E.) degree in CAD/CAM & Robotics from Thapar Institute of Engineering & Technology (Deemed to be University), Patiala, Punjab in the year 2007. Then he worked as Assistant Manager (Maintenance) at Abhishek Industries (Trident Group) Barnala, Punjab. In the year 2010, he joined as Assistant Professor at Baba Banda Singh Bahadur Engineering College, Fatehgarh Sahib. In year 2013, he joined in the Ph.D. Program at Thapar Institute of Engineering & Technology (Deemed to be University), Patiala, Punjab as a Research Scholar to pursue his research work in the area of '**Static and Dynamic Obstacle Avoidance with Dynamic Analysis for Mobile Robots'**. He is presently working in Baba Banda Singh Bahadur Engineering College, Fatehgarh Sahib as Assistant Professor in Mechanical Engineering Department. His research areas are system dynamics, robotics, obstacle avoidances, vehicle dynamics and control.