

**DESIGN AND CONTROL OF A TELE-OPERATED MOBILE
PLATFORM**

AMAREN PRASANNA DAS



**DEPARTMENT OF MECHANICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY DELHI**

DECEMBER 2018

DESIGN AND CONTROL OF A TELE-OPERATED MOBILE PLATFORM

by

AMAREN PASANNA DAS

Department of Mechanical Engineering

Submitted

in fulfilment of the requirements of the degree of Doctor of Philosophy

to the



INDIAN INSTITUTE OF TECHNOLOGY DELHI

DECEMBER 2018

Certificate

This is to certify that the thesis entitled **DESIGN AND CONTROL OF A TELE-OPERATED MOBILE PLATFORM**, submitted by **Shri. Amaren Prasanna Das** to the Indian Institute of Technology Delhi, for the award of the degree of **Doctor of Philosophy** in Mechanical Engineering, is a record of the original, bona-fide research work carried out by him under our supervision and guidance. The thesis has reached the standards fulfilling the requirements of the regulations related to the award of the degree.

The results contained in this thesis have not been submitted in part or in full to any other university or institute for the award of any degree or diploma.

Prof. S. K. Saha

Professor

Department of Mechanical Engineering
Indian Institute of Technology Delhi
New Delhi - 110016, India

Prof. S. Bhasin

Associate Professor

Department of Electrical Engineering
Indian Institute of Technology Delhi
New Delhi - 110016, India

Dr. D. N. Badodkar

Ex-Group Director RDDG & Ex-Prof. HBNI
Bhabha Atomic Research Center
Mumbai, India

In memory of my father

Late Shri Shyam Narayan Das

Acknowledgements

The work presented in this thesis would not have been possible without the assistance of countless individuals. I extend my eternal gratitude to all those people. However, despite being a dangerous practice, I would like to single out a few, select, individuals for special recognition.

This thesis would not have seen the light of the day without the continuous and persistent motivation of my guide Prof. S. K. Saha. His advice and guidance on technical issues helped give direction to this research. Moreover, he played a pivotal role in providing the much needed psychological and emotional support. Each and every meeting with him was a booster to my self confidence.

I am deeply grateful to my co-supervisor, Professor S. Bhasin in the Department of Electric Engineering at IIT-Delhi, for his continued encouragement and invaluable suggestions in the field of control engineering and in particular adaptive control during this work. He has been instrumental in introducing me to the latest research results in the field of time delayed non-linear systems. I also express my gratitude to Dr. Deepak N Badodkar, my supervisor at my parent organization BARC, Mumbai for his encouragement and the logistic support he had provided throughout the tenure of my research.

It will be imprudent not to mention the help and support provided by my colleague Shri. Rahul V Sakrikar. One person, on whom I fell back in case of any hardware issues be it electrical or mechanical. While discussing and sorting out technical glitches I found his profound experience and field knowledge highly enriching. I still remember his *encouraging* words, "Are tera PhD ka kya hua? Saha kuch kaam nahi deya kya?" Another colleague, Shri Shishir Kr. Singh has provided his selfless service to help me develop all the software required for running the actual system. Without his help and support it would have been impossible to conduct experimental runs and data collection. I express my deepest regards

to both of them.

I cannot forget the support and help of my co-researchers and classmates Dr. Riby Abraham, Dr. Parmanand Nandihal, Dr. Abdullah Aamir Hayat and Dr. Arun Dayal Uday during my stay and course work at IIT-Delhi. Thereafter, providing all kinds of logistic support at IIT-Delhi. Without their help it would have been impossible to continue my work at IIT-Delhi. I thank them from the bottom of my heart and wish them grand success in their life.

Finally, I thank my two daughters, Akaisha and Sayuri, my mother for their patience and support. My regards to my father-in-law Shri S. N. Das, who had a grand hope on me for being the first in the family to achieve the Doctoral Degree. Special thanks to my wife Namrata, who had taken upon herself the responsibility of managing the family during my course work at IIT-Delhi and providing the required time and space for my research work thereafter. In absence of her support, patience and companionship this journey would not have been completed.

Amaren Pasanna Das

Abstract

A customized mobile manipulator was designed, developed and fabricated for radiation measurement and mapping. The detail design aspects based on the environment and mission requirement is presented in the thesis. The dynamic model of the system was developed and simulations were performed for specific paths to verify the actuator requirements and response time. The user interface and control architecture implemented on the mobile robot for teleoperation were developed for convenient and reliable functioning of the same. The time delay introduced due to video data transfer and its effects on system's stability and poor operator performance is demonstrated using simulations. A predictive display of the remote environment based on mathematical model of the mobile robot and RGB-Depth sensor data received from remote location was proposed, which was practically implemented. This strategy has largely improved robot's navigation by the operator even over significantly delayed communication network.

Contents

Certificate	i
Acknowledgements	v
Abstract	vii
Table of Contents	ix
List of Figures	xiii
List of Tables	xvii
1 Introduction	1
1.1 Literature Survey	4
1.1.1 Special Purpose Robots	4
1.1.2 Mathematical Modeling	5
1.1.3 Stability of Mobile Manipulator	7
1.1.4 Path Tracking	7
1.1.5 Obstacle avoidance and path planing	9
1.1.6 Tele-operation	10
1.1.6.1 Human interface	11
1.1.6.2 Control	12
1.1.7 Predictive Display	12
1.2 Research Contributions	13
1.3 Thesis Organization	14
2 Design of a Mobile Robot	17
2.1 Design Overview	17

2.2	Design of the Traction System	20
2.2.1	Stability Of Mobile Manipulator	21
2.2.2	Selection of motor and gearbox	23
2.3	Design of Steering System	24
2.3.1	Minimum turning radius	26
2.3.2	Calculation of steering torque	27
2.4	Scissor Mechanism for Manipulating Arm	28
2.5	Summary	29
3	Dynamics of Wheeled Mobile Robots	31
3.1	Modeling using the Natural Orthogonal Compliment (NOC)	32
3.2	Dynamic Equation of WMR	35
3.2.1	Kinematic analysis	36
3.2.2	Special cases	40
3.2.2.1	Standard caster ($d_1 = 0$)	41
3.2.2.2	Underactuated case ($d_2 = 0$)	41
3.3	Dynamic equations	41
3.3.0.1	Generalized inertia matrix, I	41
3.3.0.2	Matrix of convective inertia term C	43
3.3.1	Simulation	45
3.3.2	Inverse dynamics	46
3.4	Dynamics with Wheel Slip	48
3.4.1	Defining Slip	48
3.4.2	Kinematics	50
3.4.3	Simulation	55
3.5	Dynamical Model of Steering System	55
3.5.1	Design of PID controller	59
3.5.2	Simulation and results	61
3.6	Summary	62

4 Control of a Mobile Manipulator	63
4.1 Control Architecture and Hardware	63
4.1.1 Local onboard controller	64
4.1.2 Details of the motor controller	66
4.2 Control Algorithm	68
4.2.1 Safety Interlocks	73
4.2.2 Wheel odometry	73
4.3 Remote control station	76
4.4 Summary	77
5 Simulation of Tele-operation	79
5.1 Modeling of Mobile Robot	80
5.2 Model of Human Operator	80
5.3 Simulation and Results	83
5.3.1 Simulation algorithm	83
5.4 Summary	85
6 Predictive Display	87
6.1 Time Delay Compensation	87
6.1.1 Proposed controller	88
6.1.2 Simulation algorithm and results	89
6.2 Delay Compensation using Predictive Display	92
6.2.1 The Kinect sensor	93
6.2.2 Onboard data processing and transmission	94
6.2.3 3-D reconstruction	94
6.2.4 Extrapolation of remote scene	95
6.3 Summary	97
7 Conclusions	99
7.1 Thesis Summary	99
7.2 Current Status	101

7.3 Future Scope	101
Bibliography	105
A	115
A.1 Measurment of Time Delay in Video Feedback	115
B	117
B.1 Stability of Pure Pursuit under Time Delay	117
C	121
C.1 Optimal Design of Steering Linkages	121
Publications from the Thesis	123
Brief Bio-data of the Author	125

List of Figures

1.1	Robotics in Factory (Source: Kuka Robot)	2
1.2	Health Care (Source: da Vinci Surgical System)	2
1.3	Exploration and surveillance Source: Autonomous Space Robotics Lab - University of Toronto	2
1.4	Honda Asimo Source: http://asimo.honda.com	2
1.5	Sony Abio Source: https://us.aibo.com/	2
1.6	Robot for Fukushima Daiichi Source: Ohno, et al [1]	5
1.7	NERO III Source: Luk et al [2]	5
1.8	Follow the carrot	8
1.9	Vector pursuit	8
2.1	3-D Model of the mobile manipulator	18
2.2	Photograph of the actual system	19
2.3	Rear suspension	20
2.4	Mobile manipulator on slope	21
2.5	Centrifugal Forces	23
2.6	Mobile Manipulator on slope	23
2.7	Ackerman steering condition	25
2.8	Davis Steering Gear	27
2.9	Steer Torque	27
2.10	Scissor mechanism	28
3.1	WMR-Std. castor	35
3.2	WMR-general	35

3.3	Path traced by robot	46
3.4	Inverse dynamics of the mobile robot	48
3.5	49
3.6	Longitudinal slip	49
3.7	Longitudinal Slip or Skid	49
3.8	Tipical Curve of Magic Formula [3]	50
3.9	Steering Assembly	55
3.10	Link	55
3.11	Change in steer angle	61
3.12	Control effort	61
4.1	Control architecture	63
4.2	Data from PC to the robot	64
4.3	Data from the robot to PC	64
4.4	Wiring diagram of the WMR	65
4.5	Block digram of EPOS4 controller	66
4.6	Current control block	67
4.7	Velocity control bolck	67
4.8	Position control bolck	69
4.9	Block digram of WMR controller	70
4.10	Ackerman Steering Condition	71
4.11	Davis Steering Gear	71
4.12	Tracing a line	74
4.13	Motor torque	74
4.14	Linear velocity	74
4.15	Angular velocity	74
4.16	Tracing a circle	75
4.17	Motor torque	75
4.18	Linear velocity	75
4.19	Angular velocity	75

4.20	User interface for teleoperation	76
5.1	Teleoperation architecture	79
5.2	Assumed driving strategy	81
5.3	Geometry of Pure Pursuit	82
5.4	Simulation scheme	83
5.5	Block diagram for teleoperation	84
5.6	Simulation with no time delay in either direction	85
5.7	Simulation with time delay $h_1 = .5sec$ and $h_2 = 0$	86
5.8	Simulation with time delay $h_1 = .8sec$ and $h_2 = 0$	86
6.1	Smith predictor [4]	88
6.2	Block diagram of time delayed system	89
6.3	Delay measurement	89
6.4	Smith predictor applied to the developed mobile robot	90
6.5	Simulation block Diagram	90
6.6	Time delay $T_1 = .5 sec$ and $T_2 = 0$	92
6.7	Time delay $T_1 = .8 sec$ and $T_2 = 0$	92
6.8	Kinect Sensor from Microsoft	93
6.9	Predictive display architecture	95
6.10	PCD at time T	97
6.11	Predicted Scene	97
7.1	QR code	101
A.1	User interface for teleoperation	116
A.2	Experimental Setup: Schematic	116
A.3	Screen Shot of Operator PC	116
B.1	Results obtained by Ollero [5]	119

List of Tables

2.1	Key parameters and specifications of the mobile manipulator.	19
3.1	Dynamic & kinematic parameters	47
3.2	Key parameters steering assembly.	61
4.1	Proportionality constant table	66
4.2	Parameters for left and right rear wheel motor controllers	68
4.3	Parameters of steering motor controller	69

Chapter 1

Introduction

The use of robots such as robotic arm shown in Figure 1.1 has been used in factories for a long time, basically for repetitive kind of job. Though it started with the intention to reduce human labour, production cost, and increased productivity, with technological development their scope has expanded beyond manufacturing domain. Robots are now being used for health care, surveillance, exploration, etc. as illustrated in Figures 1.2 and 1.3. The reduction in development cost has resulted in introduction of robotic systems in entertainment industries and personal care as well. Robots have matured from heavy duty serial linked mechanical arms to a more presentable form such as ASIMO (Figure 1.4) by Honda and Aibo by Sony (Figure 1.5).

In areas where human access is not preferred or restricted due to risk of life or inhospitable environmental conditions as in chemical, space or nuclear industries, robotic systems have gained huge popularity in providing services as surveillance, rescue, exploration and remote maintenance. Research in teleoperated and autonomous mobile robotics has been fuelled largely by these requirement. Teleoperated mobile robots are suitable for these applications as the workspace required to be covered is very large, and it is essential to maintain physical separation between the robot and its control station. Moreover, the remote environment is in general unknown.



Figure 1.1: Robotics in Factory (Source: Kuka Robot)



Figure 1.2: Health Care
(Source: da Vinci Surgical System)



Figure 1.3: Exploration and surveillance
Source: Autonomous Space Robotics
Lab - University of Toronto



Figure 1.4: Honda Asimo
Source:<http://asimo.honda.com>



Figure 1.5: Sony Aibo
Source: <https://us.aiobo.com/>

The present research too was motivated by a similar requirement for in-situ measurement of the ionizing radiation and neutron field, inside the vault and cave areas of K-130, K-500 and Medical Cyclotron operational at VECC, Kolkata, West Bengal. Cyclotrons are used to accelerate charged particles to high energies. These are required for experiments in nuclear physics and nuclear medicine. The particles are accelerated to high energy using a high frequency alternating voltage, which is applied between two hollow "D"-shaped sheet metal electrodes called "Dees" inside a vacuum chamber. The area surrounding the Dee is called the vault. The cave is the area where beam line (beam of accelerated charged particles) is available for experimentation. Radiation mapping of these areas is a mandatory requirement for getting safety clearance from regulators during commissioning of new units and at regular intervals during operational life of the cyclotron facility. Though, there are radiation detectors placed at different locations in these areas they can only measure radiation levels at discrete locations, but can not provide the 3-D radiation map. The advantage of having a radiation map is that it provide detailed input to health physicist of the dose a person may receive and accordingly plan emergency operations. These maps also provide the plant operators with the location of radiation leakage and accordingly tune the system to improve its efficiency.

The challenge faced for in-situ inspection during operation of cyclotron is that the interaction between an accelerated beam of charged particles and the target produce Bremsstrahlung and characteristic x-rays, prompt γ -rays, neutrons and delayed radiation (β and γ) this makes human presence unacceptable. A teleoperated mobile robot with wireless communication link is the obvious solution. This thesis discusses the design, analysis and development of a prototype robot to carry out in-situ measurement and mapping of radiation level.

1.1 Literature Survey

In this section r, we discuss some of the important works published pertaining to the scope of this thesis. Few of the techniques and methods published in these literatures are directly used. The section on *Special Purpose Robots* lists literature which were reviewed to arrive at the overall design of the mobile robot developed in this research. The dynamic analysis of the mobile platform was based on the works cited in section *Mathamatical Modeling*. The literature discussed in *Path Tacking* helped to arrive at the "human model" proposed in this thesis for simulation of teleoperation. The section on *Tele-operation* discusses literature in a much broader sense; such as force feedback, haptic interface design, etc., than what was adapted in this thesis. This was done for completeness of the subject. Last section on *Predictive Display*, though a part of human interface for tele-operation is discussed separately because it forms one of the major components of the tele-operation system network designed for the mobile robot presented in this thesis.

1.1.1 Special Purpose Robots

This section reviews some of the special purpose robots built for various typical applications. Design and fabrication of a low cost, solar powered mobile robot for scientific missions on the Antarctic plateau was presented by Ray [6]. Honeycomb-glass-fibre composite was used to provide high strength and low weight. Ohno, et al. [1] developed a robotic vehicle shown in Figure 1.6 for measuring the radiation in the Fukushima Daiichi Nuclear Power Plant. A survey of different types of climbing robots for non-destructive testing of pressure vessel is given by [2], such as NERO III shown in Figure 1.7. Briones [7] presents a vacuum cup based wall climbing robot for inspection of nuclear power plants. Galt [8] has developed eight-legged teleoperated mobile robot for the use in nuclear industry. Development of a magnetic-wheel based mobile robot for painting of ship is discussed by Cho [9].



Figure 1.6: Robot for Fukushima Daiichi

Source: Ohno, et al [1]

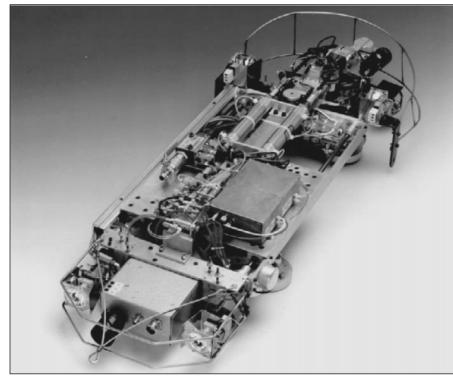


Figure 1.7: NERO III

Source: Luk et al [2]

Compliant link based mobile robot was designed and tested by Borenstein[10], in which the author claims that due to its unique design, better dead-reckoning accuracy was achieved compared to other contemporary designs. This vehicle has two independent drive units or "trucks" that are free to rotate about a vertical shaft connected to the vehicle body. Each truck comprises two drive motors on a common axes and forms a differential drive system. Mechanical compliance was implemented by means of a linear bearing that allows relative motion between the front and rear truck. Other literature giving details of mobile robots based on differential wheel, traction belt and omnidirectional wheel are given in the introductory part of Chapter 2.

1.1.2 Mathematical Modeling

A very comprehensive list of Wheeled Mobile Robots (WMR) using different wheel configuration is given by Muir and Neuman [11]. In the paper, kinematic equations of conventional, omnidirectional and ball wheels were presented. The kinematics of the WMR was derived by combining the kinematic information of individual wheel. Detection of wheel slip based on error in the least square solution was also discussed. Similar issues were addressed by Alexander in [12]. The major difference is that he uses physical friction model in the analysis of over actuated systems where rolling constrains are not satisfied. A seminal work by Champion [13] gives the structural classification of wheeled mobile robots based on the *degree of mobility*.

ity, δ_m , and *degree of steerability*, δ_s . It was based on the number of conventional fixed wheels and conventional centered orientable wheels. According to them any WMR fall in one of the 5 categories given by $(\delta_m, \delta_s) \rightarrow (3, 0), (2, 1), (1, 1), (1, 2)$. The configuration and posture kinematic models of each type was derived. Based on dynamic model, the minimal number of actuators required for full maneuverability of each type was presented. Kinematic analysis of omni-directional over-actuated mobile robot was presented in [14]. Two different methods for forward kinematics was also discussed along with singularity analysis. Actuator switching scheme based on load distribution to avoid singularity was also presented.

Dynamic modeling of mobile manipulator can be categorized as: force based ,i.e, the Newton -Euler (NE) formulation and energy based as in Eular-Lagrange (EL) equations. Hoostmans [15] used NE method to arrive at the dynamic model of a mobile manipulator that has two links mounted on a mobile platform. Chung [16] used EL method to arrive at the equations of motion for a mobile manipulator. Geometric mechanics was used to adapt Luh and walker [17] algorithm by Boyer and Ali [18] to apply recursive inverse dynamics formulation to wheeled systems.

Orthogonal compliment method utilizes the advantage of NE and EL approach to derive the equations of motion of a multibody system. It uses the fact that the motion can take place only in the null space of the constrains inducing matrix A defined as $Ax = 0$, where x is a vector of independent co-ordinates. The orthogonal compliment of the constraint inducing matrix A is used to eliminate the non-working constraint forces and moments from the equations of motion. Angeles and Lee [19] used the natural orthogonal compliment method to derive the equations of motion for holonomic mechanical systems. In this, orthogonal compliment was derived from the velocity constraints naturally, hence the name. This was used by Angeles [20] and Saha in [21],[22] to derive the equations of motion for a WMR.

1.1.3 Stability of Mobile Manipulator

Stability of mobile manipulators has been studied by both the vehicular community and the robotics community. The vehicular community has focused on characterizing the lateral rollover of the vehicle as in [23, 24]. The robotic community has discussed problem from motion planning of manipulator and has come up with different stability margins. Dubowsky [25] has studied the motion planning of mobile manipulator for stationary platform. The criterion for stability is that the support point should not loose ground contact. McGhee [26] proposed as shortest distance between the Center of Gravity and the edge of the support polygon projected on a horizontal plane. *Zero moment point* (ZMP) was developed to study the stability of biped mechanism by Vukobratovic et al. [27, 28]. It was later adapted by Ollero [5], Hung et al. [29, 30] and Sugano [31] to examine the stability of mobile manipulators. Furuno [32] propose the method for planning the trajectory of the nonholonomic mobile manipulator from its end-effector's path considering the dynamic stability. Then ZMP criterion proposed is used as an index for the system stability. Messuri et al. [33] proposed the stability measure called '*Energy Stability Margin*'. The uses minimum work required to tipover the legged vehicle, a measure which is sensitive to c.g. height. Ghasempoor and Sepehri [34] extend the method of Messuri by including include external and inertial loads. *Force-angle stability* measure which is a simple graphical method was proposed by Papadopoulos [35, 36]. The method is applicable to system subjected to inertial and external forces, operating over even and uneven terrains.

1.1.4 Path Tracking

Path tracing algorithm for the control of a mobile robot is used to arrive at the mathematical model of a human operator for simulation of tele-operation loop, as done in chapter 6. Geometry based path tracking algorithms are most intuitive and hence suitable for the present application. The major algorithms in this category reported in the literatures are *pure pursuit* [37], *follow the carrot* [38],

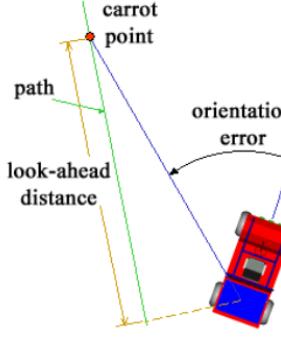


Figure 1.8: Follow the carrot

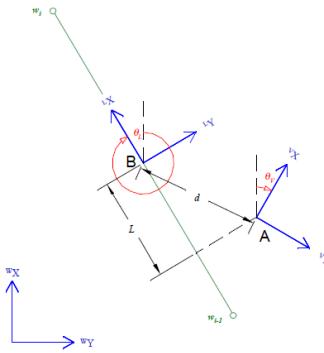


Figure 1.9: Vector pursuit

vector pursuit [39], and *follow the past* [40]. In pure-pursuit [37], the steer angle of the robot is set so that the robot moves in circle to reach a *goal point* on the desired path. The goal point is based on the "Look Ahead Distance", which is practically the maximum distance one can see from the current vehicle position. The detailed discussion is given in Chapter 5. Corrective action was based on position error of the vehicle, where orientation error was not taken into account explicitly.

In case of "Follow the Carrot" method [38], the steering angle is set proportional to the *orientation error* defined in Figure 1.8. The orientation error is defined as the difference between the current orientation of the vehicle and the orientation required from the present position of the vehicle to reach the goal point on the reference path. The proportionality constant is decided based on trial and error.

The two previous geometric path tracking techniques generate steering commands based upon the goal point on the reference trajectory to be traced. Wit in [39] suggested a strategy to uses the path orientation and curvature known at the goal point to improved path tracking, such that the vehicle arrived at the goal point with the correct orientation and curvature. Wit used Screw theory to find the error between the screw at the current location, point *A* and the required screw at the goal position, Point *b* as shown in Figure 1.9. Control is then generated proportional to this error.

Hellstrom [40] has proposed an algorithm which uses the knowledge of previ-

ously recorded steer angle, associated with the path traced earlier. In this algorithm, the steer angle of the vehicle is set based on the orientation error, position error and the past recorded steer angle. A recent survey by Paden [41] provides extensive review of other control strategies for path tracking of autonomous unmanned vehicles such as those based on Lyapunov method, Model Predictive Controller, adaptive control, etc.

1.1.5 Obstacle avoidance and path planing

The robotic system developed is not an autonomous mobile robot as it is controlled from the remote station by an operator. In future it might be required to provide autonomous navigation to the mobile robot. Autonomous navigation has two impotent ingredients; path planing and obstacle avoidance. In this section we will look at the available algorithms and techniques that has been proposed by different authors in the field of autonomous navigation.

The purpose of obstacle avoidance algorithms is to avoid collisions with obstacles. Obstacle avoidance algorithms deals with moving the robot based on the feedback information from its sensors. Virtual Force Field (VFF) method was proposed by Bornstein [42] for real-time obstacle avoidance for a mobile robot. In this method the robot work space is grids and each grid is given a value based on the chance of the obstacle being located in the cell. the amount of repulsive force acting on the robot is inversely proportional to the distance between the occupied cell and the robot. Borenstein and Koren [43] in this method polar histogram is around the robot. The sectors presented in the polar histogram show the *polar obstacle density*. The direction of the robot is computed by choosing the sector with least concentration of obstacles. The VFH+ suggested in [44] improves upon the VFH by introducing *threshold hysteresis* to improve the shape of the trajectory,

and the use of a cost function. The Dynamic Window Approach (DWA) [45] is another method for reactive obstacle avoidance dealing with the kinematical and dynamic constraints of the vehicle in contrast to VFF and VFH methods. Real-Time Obstacle Avoidance using artificial potential field method has been used in [46],[47],[48]. Obstacle avoidance in highly dense and troublesome environment using Nearness Diagram (ND) is presented in [49]. This method uses *divide and conquer* approach splitting the environment into sectors to represent the location of obstacles.

Some of the major path planning methods are the well know A* method, Visibility Graph, Artificial potential method and Voronoi diagram method [50]. In the A* [51] is a heuristic method in which the space is divide into grids, a collision free path is found by joining the adjacent free cells. A modified version of A* is presented in [52]. Visibility graph method presented in [53], in which the obstacles are represented as convex polygon, a line is joined from between the vertices of the polygon with out passing through it. An algorithm search the path by joining these lines from the robot current location to the goal position. The Voronoi diagram [51, 54, 55] consists of arcs (lines) which are equidistant from the two nearest obstacles. The obstacles in the Voronoi diagram are presented as polygons. The maximized clearance between the Voronoi arc segments and the polygons helps the robot maintain safe distance away from the obstacles.

1.1.6 Tele-operation

Tele-operation deals with connecton of a human operator with the robot in order to reproduce human action at distance. Tele-operation is in general bidirectional or bilateral as the human needs to have a feedback in order to understand the results of his action and to perceive the remote environment. It started with its

use in nuclear and space industries [56, 57], but now it is used in underwater exploration, surgery, live-power line maintenance, mining, etc. All characterized by reducing the risk to human operators. The two major related research areas are the "human interface" and "control" design.

1.1.6.1 Human interface

Human interface is a means through which the operator interacts with the remote robot by perceiving the remote environment and sending commands accordingly. Thus, the human interface has two important purpose: one to excite the human senses to show the action of the executed task and to process the human command properly to execute it at the remote end. Force and haptic feedback of remote environment drastically improves operator's performance. Hence a serial link haptic device PHANTOM [58] was developed at MIT during 1994 to provide 3-DOF force feedback for touch feedback purpose. DELTA Haptice Device described in [59] provides 6-DOF force feedback with moderate force. Clover [60] has reported the use of off-the-shelf serial industrial robots for haptic realization of tasks requiring a large workspace and high force capability. Customized 10-DOF haptic device was reported for similar purpose in [61]. Design of a 6-DOF parallel mechanism for force feedback is discussed in [62].

Another major form of human interface is the visual feedback. The main challenge is to provide depth perception of remote environment. Most stereoscopic systems used in telerobotics are based on shutter glasses [63, 64], head-mounted displays [64] or polarized images [65]. Systems based on shutter glasses hide user's eyes alternately in synchronization to screen refreshment, which projects images for left and right eye alternately. A second type of interfaces is based on polarized images. The user is also required to wear glasses that filter the left and right images. The third type of interface is the head mounted display such as "Google cardboard", especially designed to immerse users into virtual environments where the left and right images are projected on each eye using two separate screens or

split screens.

1.1.6.2 Control

Control of a tele-operetion system deals with two issues, *transparency* and *stability*. Transparency deals with what information is to be exchanged between the remote and local station so that the operator can have a natural feel of the remote environment. A position-position architecture is suggested by Goertz [66], where master position is passed as a command to the slave servo (position) controller, and slave position is returned to the master as a position command. A position-force architecture has been proposed by Flatau [67] in which the master sends the position to the slave and the slave sends back the force felt by it in the remote environment. A general 4-channel architecture been suggested by Lawrence [68], and transparency has been defined as a measure of performance in teleoperation and evaluated for different architectures.

An excellent survey article on control of bilateral teleoperation was given by Hokayem and Spong [69]. Few of these are briefly presented here. A teleoperation system, comprised of a master and slave with their corresponding controllers, residing between the human operator and the environment, can be modeled as a two port network. Passivity based design of stabilizing control using wave-variable concept and scattering theory has been proposed by Anderson and Spong [70], Rebelo [71] and Anderson and Slotin [72]. Port-Hamiltonian based approach was used in [73, 74]. Design of controller for time delayed systems based on back-stepping method in combination with partial differential heat equation was studied by Kristic [75].

1.1.7 Predictive Display

Delays are inherent in teleoperation over wireless network. Practically, much of the delay is due to relay stations and limited bandwidth of the network. As little as a half second delay in the visual feedback significantly reduces human performance

[76]. The operator tends to adopt an inefficient "move then wait and see" policy in order to complete the task.

To overcome performance deterioration of the operator due to time delay in visual feedback, two approaches have been reported in the literature, namely, *supervisory control or tele-assistance* and *predictive display*. In *supervisory control* [77, 78, 79] the robot is partly guided by operator by giving the robot intermittent commands to achieve the goal. The drawback of such system is that operator loses direct contact with the task. In predictive display systems, a natural and widely used techniques, synthesised view of the remote environment is displayed to the operator based on his movements. It has been used for space teleoperation as early as in 1993, which was reported by Sheridan [80], Bejczy [81] and Kim [82]. Whereas the above two used a-prior modeling and calibration of remote environment, Jagersand [83] used delayed visual feedback and operator control signal to build predicated image which was presented to the operator. The system was implemented with a fixed remote environment with a manipulator arm with two wall mounted cameras. An estimation function was proposed $I_i \approx \phi_k(x_i)$, $i \in 1,..k$, that approximated each image I_i seen so far on the trajectory, i.e., x_1, x_2, \dots, x_k . Uncalibrated monocular camera mounted on manipulator (eye-in-hand) based image predication method was discussed by Yeres [84] and Deng [85]. Multiple sensors based dense 3-D map of a remote scene was reported by Kelly [86] and [87]. While Kelly used fusion of lidar and camera, Burkert used stereo cameras. Hu [88] has used SLAM based Predictive Display (PD) system for telemanipulation of a mobile robot. In his approach, texture and geometry of the remote site was transmitted instead of video stream. This, the author, claims reduces bandwidth utilization.

1.2 Research Contributions

The original contributions of the present research are listed below:

- (i) Design of a redundantly actuated redundantly steered (RARS) teleoperated mobile robot for remote surveillance and mapping application.

*The challenge was to design a fault tolerant mobile robot with a relatively less number of actuators compared to what has been explained in section *.*.*. Introduction of the Ackerman steering with two differentially driven conventional wheels allowed us to control the robot with three actuators, while having the ability to rescue the robot if one of the motor fails during remote operation.*

- (ii) *Kinematic and dynamic modellings of the RARS robot.*

*This redundantly actuated RARS mobile robot has a combination of differentially driven rear wheels and front wheels steered by Ackerman mechanism. Such a design was not found in the literature as per the knowledge of the author. Hence none of the existing models was suitable for its analysis. Modeling of mobile robot with slip including both longitudinal and lateral using DNoC has also be presented for the first time in section *.*.*.*

- (iii) Synchronised control architecture required for intuitive teleoperation of the RARS mobile robot.

Since the actuator controlling the Ackerman mechanism needs to be synchronised with differential drive of the rear wheel actuators appropriate control stratagey was introduced as described in section 4.2, "Control Algorithm.

1.3 Thesis Organization

The thesis contains eight chapters and three appendix. They are organized as follows:

Chapter 1: Introduction

This chapter, discusses the scope of mobile robotics in general and the motivation which led to this research work. It also includes literature review in the following areas: kinematics and dynamics of mobile robots, control of mobile robot, control for time delay systems, performance of operator under tele-operation, and predictive display systems. Finally the research contribution of this thesis is listed.

Chapter 2: Design of a Mobile Robot

This chapter highlights the design considerations of a tele-operated mobile robot based on mission requirements and environmental conditions. It discusses the mechanical design for the traction system, the steering gear, and the scissor mechanism. Selection of steering system based on terrain condition and power requirement is also discussed.

Chapter 3: Dynamics of Wheeled Mobile Robots

In this chapter, the dynamic equations of a four wheeled differentially driven robotic platform derived using natural orthogonal compliment method is presented. The platform has two actuated wheels and two passive wheels. Different types of passive wheels were studied and corresponding dynamic equations were derived.

Chapter 4: Control of a mobile manipulator

The control architecture and the hardware used for tele-operation is presented along with the detailed description of implementation of the controller software at both the remote (mobile robot) and the local station. The experimental results of robots position based on wheel odometry and is torque requirement for few predefined paths are presented.

Chapter 5: Simulation of Tele-operation

In this chapter simulation of a teleoperated mobile robot is presented both without

time delay and under time delay due to communication link. It is shown via simulations that with increase in time delay, teleoperation loop become unstable.

Chapter 6: Predictive Display

In this chapter, we propose a Predictive Display strategy to counter the time delay in video feedback by extrapolating in time the camera view based on the predicted position of the robot at remote location.

Chapter 7: Conclusions

This chapter summarizes major results of this research work. Limitations of the system along with the future scope of work based on present experiences are also addressed.

Bibliography

Appendix A: Measurement of Time Delay in Video Feedback

Here the experimental set up and methodology used to determine the time delay is presented.

Appendix B: Stability of Pure Pursuit under Time Delay

Based on the paper by Ollero [5] stability analysis of pure pursuit tracking algorithm with input delay is discussed.

Appendix C: Optimal Design of Steering Linkages

This appendix describes the formulation of optimization problem for steering linkage.

Chapter 2

Design of a Mobile Robot

Most of the mobile robots presented in literature use differential wheel drive with passive castor as in [21], [89] and [90]. The other common methods for locomotion of mobile robots are the omnidirectional wheels [91] and [92], and tracked wheel system [93] and [94]. According to Nagatani [95], a vehicle with Mecanum wheels is susceptible to slippage and same is the case for tracked vehicle, which are inherently skid steered. The slippage of the wheels prevents the most popular dead-reckoning method using rotary shaft encoders from being performed well. This chapter discusses the design methodology of a novel mobile manipulator based on environmental requirements. Here, the advantage of Davis steering mechanism is highlighted over castor wheels or other steering methods from the perspective of the proposed mobile manipulator. The Davis steering system modifies the heading of front wheels in a way that, at low speeds, all the wheels are in pure rolling without lateral sliding [96].

2.1 Design Overview

The objective of a mobile robot under consideration is to navigate inside the cyclotron vault and collect radiation intensity data at all the required points decided by the operator. Data is to be collected not only at different planer locations of the floor but also at varying height from the floor. To cater to this operational

requirement, a mobile platform with a vertically extendable manipulating arm was developed. Together, they are referred henceforth as "mobile manipulator" or simply "mobile platform". The 3-D model of the mobile-manipulator with its major subsystems are shown in Figure 2.1, whereas and the actual system is shown in Figure 2.2.

The environmental condition required that the vehicle be either autonomous or teleoperated. To keep the complexity low, it was decided to have wireless teleoperated navigation and control. This gives an operator full flexibility to drive and control the system from a remote station using visual feedback provided from the on-board camera. The key parameters of the mobile manipulator are listed in Table 2.1.

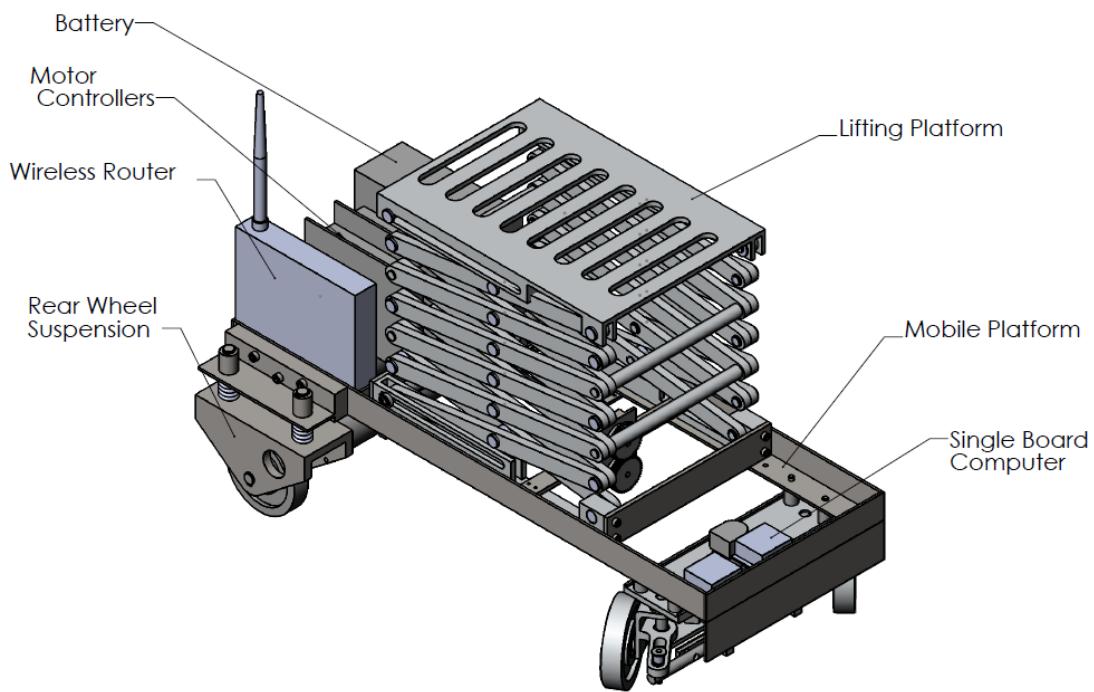


Figure 2.1: 3-D Model of the mobile manipulator

The mobile manipulator has a footprint of $700\text{ mm} \times 400\text{ mm}$ based on the narrow passage through which the system has to negotiate. These passages are formed inside the vault area by the pipelines and structural supports of the cyclotron and its associated equipment. Two DC motors, with speed servo controller, provide the traction to each rear wheels. The two front wheels are inter-connected

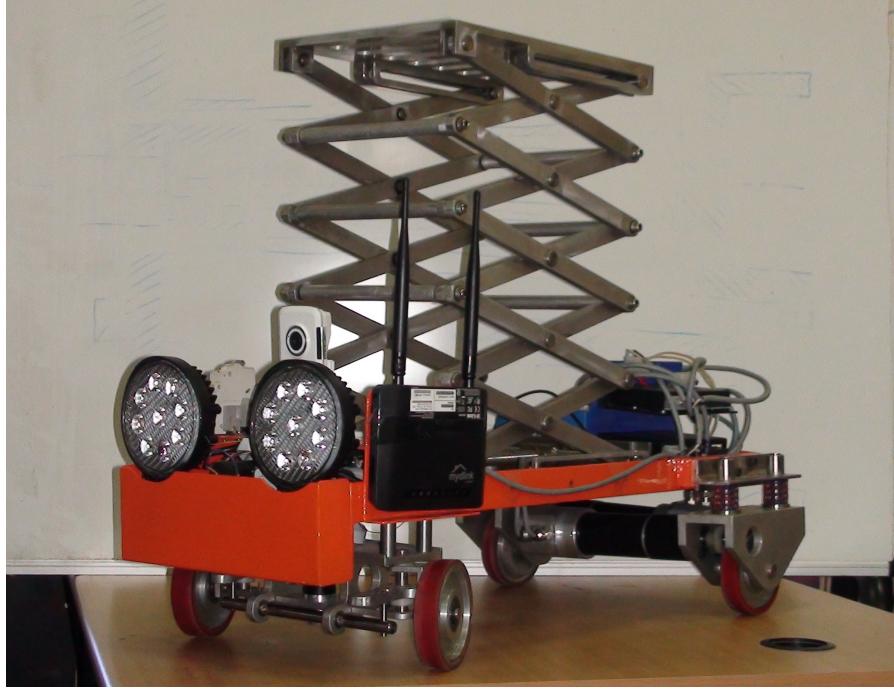


Figure 2.2: Photograph of the actual system

Table 2.1: Key parameters and specifications of the mobile manipulator.

Weight	70 Kg	Without payload
Payload	10 Kg	—
Footprint	700 mm × 400 mm	—
Height Collapsed	500 mm	Along Z-Axis
Height Extended	1500 mm	Along Z-Axis
Steering mechanism	Davis Steering	—
Turning radius	415 mm	—
Ground clearance	45 mm	—
Maximum traction speed	2 m/min	On flat terrain
Ramp climb angle	30°	Checkerboard surface

with a Davis steering mechanism [97]. A scissor mechanism provides the vertical motion to the detector that is mounted on the manipulating arm.

In order to keep the self weight of the system small, all the structural parts are made of aluminum alloy AL6061, apart from the base frame. Stainless Steel (SS304) angle sections was used for the base frame, which give it excellent strength to weight ratio.

2.2 Design of the Traction System

Traction is provided by the two rear wheels driven independently. This makes the system over actuated. A mechanical differential connecting the two rear wheels, as used in cars, would overcome this. It was not proposed to do so as the proposed vehicle is planned to be teleoperated in a environment inaccessible to humans. This calls for a single-failure-safe design. The proposed design gives two major advantages. Firstly, in case one of the wheel loosing contact with the ground due to overhang in small pits or while over an obstacle, the mechanical differential system would keep supplying power to the free hanging wheel. The system will hence get stuck, maybe in an unrecoverable location. This situation is avoided in the present design as the motor having traction can be independently powered, to move the vehicle. Secondly, using the proposed design, in case of one actuator failure, either the traction or the steering motor, still the vehicle can be manoeuvred to a safe location, albeit with dragging of the wheel with failed actuator.

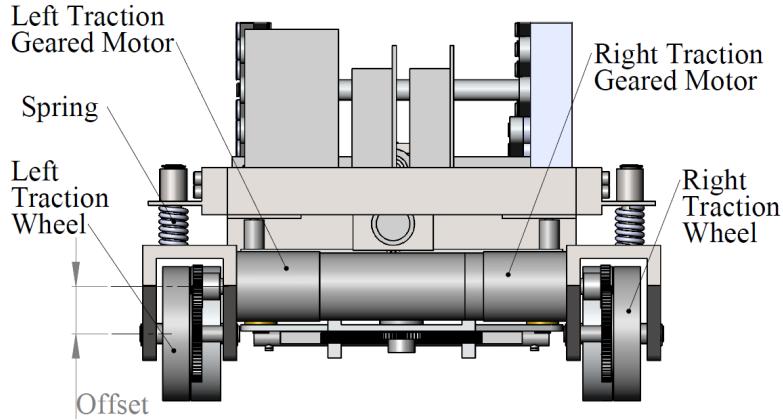


Figure 2.3: Rear suspension

Each wheel is driven by a Maxon DC RE50 200W Motor through a 26:1 reduction gearbox. The motors are mounted at an offset to the wheel axis for increased ground clearance and lateral compactness, as shown in Figure 2.3. Spring suspension is provided at each wheel to ensure sufficient contact force on uneven ground. The diameter of the wheel is 100 mm (D_w), which is sufficient to ride over obstacle of height 20 mm (Max). They are made of Aluminum alloy-6061 with 5mm thick

molded polyurethane (PU) liner. The PU liner provides large traction on cement flooring while being resistant to wear.

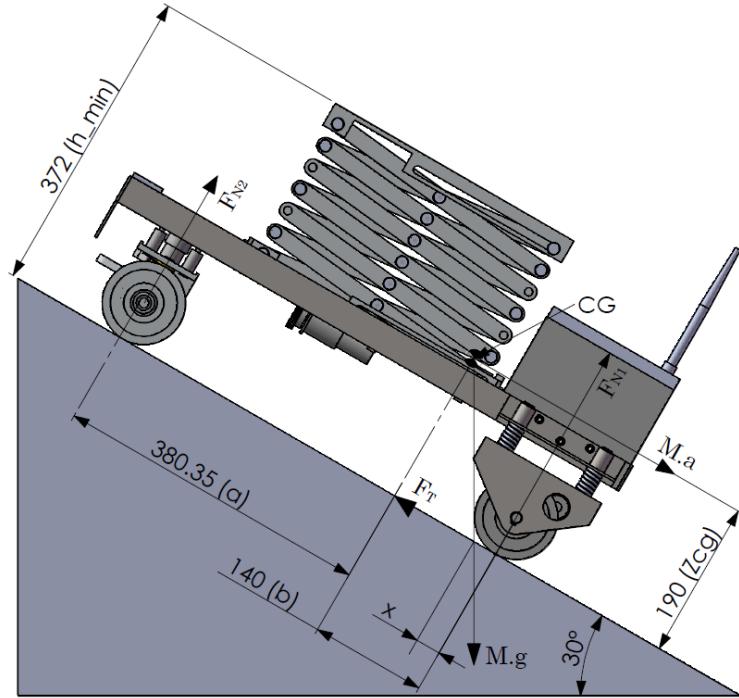


Figure 2.4: Mobile manipulator on slope

2.2.1 Stability Of Mobile Manipulator

The load distribution was optimized to generate maximum normal reaction, F_n , at the rear wheels without overturning while moving up the ramp of 30° . Maximizing rear wheel reaction by increasing "b" as per Equation 2.1 ensures increased traction, $F_T = \mu F_N$ (μ is the coefficient of friction), but at the same time decreases the stability margin indicated by "X" in Figure 2.4. The moment about the rear wheel contact point yield

$$(a + b)F_{N1} = b(m_r g \cos \theta) - z_{cg}(m_r g \sin \theta + m_r a) \quad (2.1)$$

The critical condition, which initiates over turning about the rear axle is given by $F_{N1} = 0$. This reduces the above equation 2.1 to

$$m_r g b \cos \theta = (m_r g \sin \theta + m_r a) z_{cg}, \Rightarrow g \left(\frac{b}{z_{cg}} \cos \theta - \sin \theta \right) = a \quad (2.2)$$

where

F_{N1} , F_{N2} are normal reaction on the front and rear wheels.

a and b are the distance of the vehicle centre of gravity (CG) from the rear and front wheels.

Z_{cg} is the height of CG from the plane containing the contact point of the wheels.

m_r is mass of the vehicle.

g is acceleration due to gravity.

θ is the inclination of the traction surface from horizontal.

The mass distribution of the mobile robot was adjusted such that the stability margin X as shown in Figure 2.2 was fixed to 30mm. Which limits the acceleration of the mobile robot while climbing up over a ramp of 30° to $0.144g$. Operation of system below this acceleration limit safeguards it against overturning along the longitudinal direction.

The stability analysis was carried out with the manipulator in the retracted condition. In case the manipulator is extended the CG of the system shift towards the rear wheel axle. A condition will be reached when the vector Mg crosses the rear wheel contact point, and the mobile robot will over turn. To insure against this unstable situation the vehicle is never moved the ramp with manipulator extended.

Another instability is encountered during turning or negotiating the curve. The centrifugal force tends to overturn the vehicle in lateral

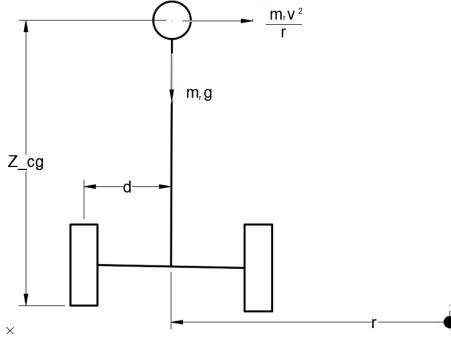


Figure 2.5: Centrifugal Forces

Figure 2.6: Mobile Manipulator on slope

direction as shown in Figure 2.5. This limits the linear velocity of the vehicle for a given steer angle. If v is the linear velocity of the vehicle and r the radius of the path, the moment balance is given as

$$\frac{m_r v^2}{r} Z_{cg} = m_r g d \quad (2.3)$$

Thus the limiting velocity as a function of r the turning radius

$$v = \sqrt{\frac{rgd}{Z_{cg}}} \quad (2.4)$$

This give the limiting velocity as $v = 1.9m/sec$ based on the $Z_{cg} = 190mm$ and $d = 168$ as given Figure 2.4 and 2.5. The minimum turning radius of the vehicle is $415mm$ as given in Table 2.1 is used for r in the above equation.

2.2.2 Selection of motor and gearbox

The torque requirement for the rear wheels were calculated based on the static moment balance with the assumption that each rear wheel shares equal load and the total suspended weight is 80 Kg. From the freebody diagram (Figure 2.4), using moment and force balance we get the following:

$$F_{N1} = \frac{aM \cos \theta}{a + b - \mu Z_{cg}}, \quad F_T = \mu F_{N1} \quad (2.5)$$

In order to estimate the traction motor size, we take worst case scenario of $\theta = 30^\circ$ and $\mu = 0.3$. This leads to

$$F_{N1} = 66Kg, \quad F_T = 0.3 \times 66 \approx 20Kg$$

Since the traction is provided by the two rear wheels, the torque required per wheel (T_w) is given by

$$T_w = (F_T/2)(D_w/2) = (20/2) \times 50 = 500Kg-mm \simeq 5Nm \quad (2.6)$$

The motor torque T_M , required based on the assumption of factor of safety, $FS = 1.5$ is

$$T_M = (FS) \times T_w = 1.5 \times 5 = 7.5 \simeq 8Nm \quad (2.7)$$

Assuming the maximum speed, $V_{ramp} = 1m/s$, of the mobile manipulator over a ramp, the required power, P_M , of the traction motor is calculated as,

$$\omega_w = V_{ramp}/(D_w/2) \simeq 200rpm \quad (2.8)$$

$$P_m = \omega_w T_m = 20 \times 8 = 160W$$

The nearest Maxon motor available as per the catalogue [98] is 200W, RE50-370354 motor. The nominal speed, N_s is 5680rpm. Therefore, the gearing ratio required is, $N_s/\omega_w = 5680/200 \simeq 28.4$. The nearest gear box available is of ratio 26 : 1, which was chosen.

2.3 Design of Steering System

The design objective of a steering system should be to ensure rolling motion of all the wheels during every possible manurers of the mobile robot. This is to reduce the friction drag due to sliding motion which degrades the energy efficiency of a mobile robot. In case of four wheeled vehicle, with the orientation of the rear

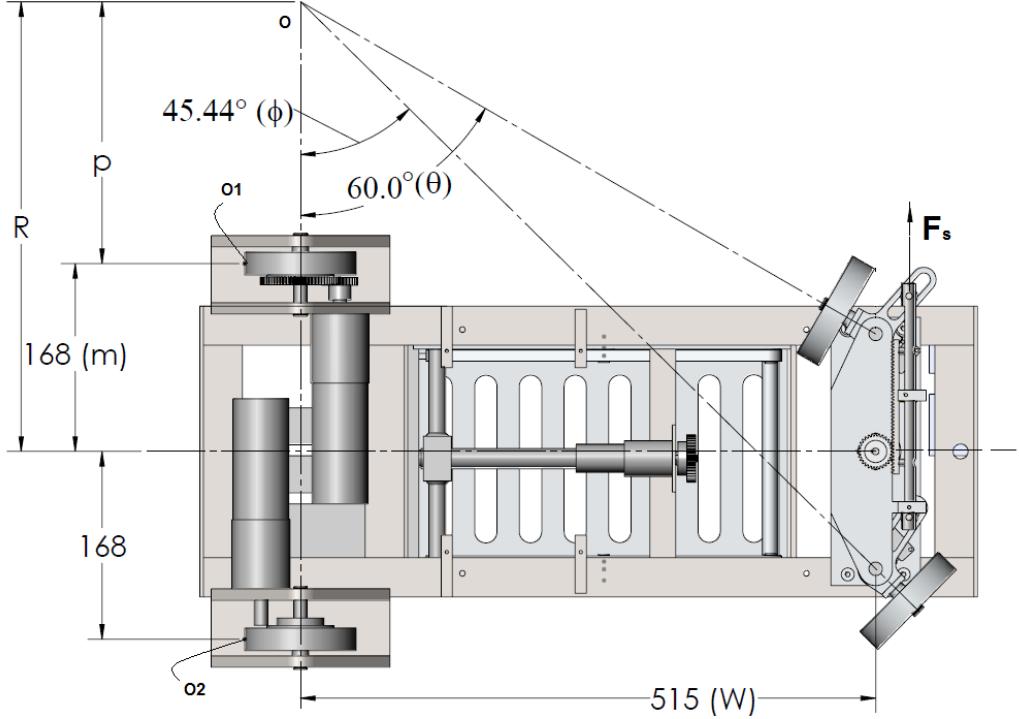


Figure 2.7: Ackerman steering condition

wheels fixed and front wheels steered, the condition shown in Figure 2.7, referred in some literature as Ackerman steering condition, must be satisfied to ensure pure rolling of all the wheels.

This mobile manipulator uses Davis steering mechanism, Figure 2.8, on the front wheels. Caster wheels were not used as they tend to align with obstacles and thus get stuck. On the other hand tracked wheels have excellent rough terrain capabilities, but is power intensive due to skid steering. Another option was to use Omnidirectional wheels, which need complex controller for coordination and an extra actuator. Moreover, the floor of the cyclotron are in general not clean of loose small objects, which may get stuck in between the free rollers of the omnidirectional wheels. This will reduce the efficiency of the vehicle.

Davis mechanism was chosen over Ackerman steering gear as it satisfies the steering condition given by Equation 2.9, which ensures pure rolling of all wheels, over the entire steering range. This makes the system, suitable for passive wheel odometry and also energy efficient. The steering gear being positively driven by position controlled servo motor does not align with the obstacles and thus are

able to crossover it. The dimensions of the links used in the steering mechanism is given in Figure 2.8, and are based on the Ackerman steering law given below:

$$\cot \phi - \cot \theta = a/w, \quad \frac{2b}{h} = \frac{a}{w} \quad (2.9)$$

where a and b is limited by the over all size of the vehicle, discussed earlier.

2.3.1 Minimum turning radius

The mechanical construction of this steering mechanism limits the steering angle. This in turn limits the minimum radius the vehicle can negotiate. The parameters of the steering linkages were found using optimization with constraints on over all dimensions and location of links. The details of the method used is given in Appendix C.1. The Figure 2.7 shows the extreme values of ϕ and θ , one side of the steering limits. The **turning radius** R for a given steer angle θ is calculated by the geometry of Figure 2.8 as

$$\tan \theta = \frac{w}{p + m - \frac{a}{2}} \quad \text{and} \quad R = m + p$$

Eliminating p from the above equation and rearranging , we get

$$R(\theta) = \frac{a}{2} + w \cot \theta \quad (2.10)$$

The extreme value of $\theta = 60^\circ$. as shown in Figure 2.7, this gives the *minimum turning radius* R_{min} as

$$R_{min} = \frac{210}{2} + 515 \times \cot(60^\circ) = 402 \text{ mm}$$

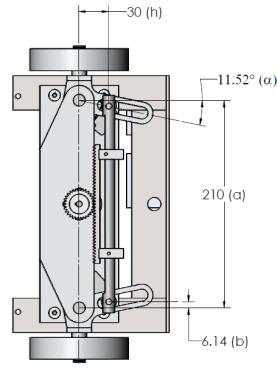


Figure 2.8: Davis Steering Gear

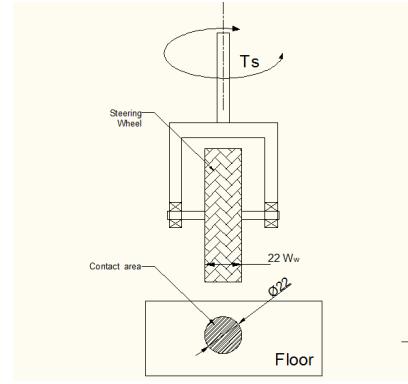


Figure 2.9: Steer Torque

2.3.2 Calculation of steering torque

The torque required to steer the front wheel is estimated based on a simplified assumption that the wheel deforms under normal load and the contact area thus generated is circular in shape with diameter that of the wheel width, W_w , as shown in Figure 2.9. In order to estimate the normal reaction on each wheel, we assumed that the total weight of 80kg was equally shared by the four wheels. Therefore, $N_s = 80/4 = 20Kg$. Next, the uniform pressure formula used for brakes/clutches design was applied to find the resistance torque T_s , between the ground and the wheel i.e.,

$$T_s = \frac{N_s \mu}{3} W_w = 0.4Nm \quad (2.11)$$

The resistance torque, T_s , of both the wheels are balanced by the force F_s acting on the rack as shown in Figure 2.7. The rack is coupled to the steering motor by a pinion of diameter, $D_p = 40mm$. The motor torque, T_{m_s} in Equation 2.12 is calculated with a high factor of safety, $FS = 3$. This is because T_S is estimated based on a simplified model of brake design. The power, P_{m_s} of the steering motor based on torque T_{m_s} and the steering speed ω_s of 100rpm is

$$T_{m_s} = (FS) \frac{2T_s D_p}{h} = 1.6Nm \quad \text{and} \quad P_{m_s} = T_{m_s} * \omega_s = 17W \quad (2.12)$$

Based on the above specifications, a 20W, RE25 DC motor of Maxon make and a gear box GP32 of ratio 159:1 was chosen for the steering mechanism.

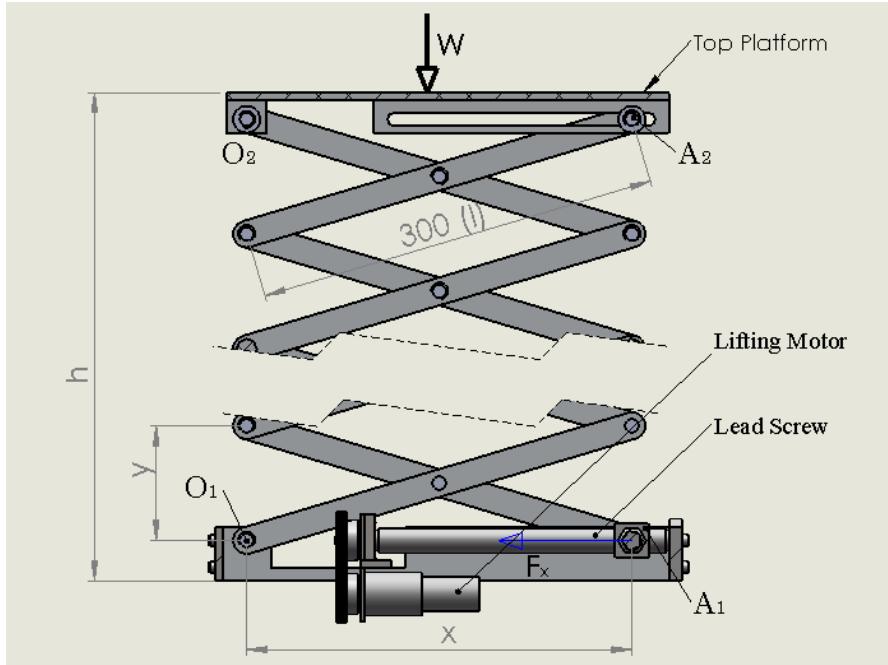


Figure 2.10: Scissor mechanism

2.4 Scissor Mechanism for Manipulating Arm

The manipulating arm was designed to move the radiation sensor mounted on the "top platform" up to a height of 1.5m from the floor level. This motion was generated using a scissor mechanism, as shown in Figure 2.10. The scissor mechanism has two major advantages over other lifting methods such as telescopic pillar, etc. First, the ratio of height in extended and collapsed condition is very large. In our case it is 3 : 1. Second, the self weight of the mechanism is less as it is made of rectangular links.

The scissor mechanism, Figure 2.10, has 6 stages, where each "X" denotes one stage. The Scissor is connected to the top platform by a pivot joint O_2 and a prismatic joint A_2 , and coupled to the base frame by pivot joint O_1 and a prismatic joint A_1 . The linear actuation of joint A_1 is provided by a lead screw of pitch (P) 1.5 mm and mean diameter (d_m) 10mm. This results in vertical motion of the top platform.

The relation between the vertical motion of the platform and the horizontal displacement of point A_1 is given by geometry of the mechanism shown in Figure

2.10

$$\begin{aligned} y &= l \sin \theta & \Rightarrow dy = l \cos \theta d\theta; \\ x &= l \cos \theta & \Rightarrow dx = -l \sin \theta d\theta \\ h &= Ny & \Rightarrow dh = Ndy \end{aligned} \quad (2.13)$$

where l is the link length, θ the angle of the link with horizontal plane, N the number of stages, and h the height of platform.

The number of stages used in the scissor mechanism is six ($N=6$). From the principle of virtual work, we get

$$-F_x dx = W dh, \Rightarrow F_x = \frac{WN}{\tan \theta} \quad (2.14)$$

where, F_x is the axial force on the prismatic joint, A_1 , and W is the payload. From Equation 2.14, it is clear that as $\theta \rightarrow 0$, the force $F_x \rightarrow \infty$. In the present design, $\theta_{min} = 5^\circ$ and $\theta_{max} = 45^\circ$. Therefore, the extended height $h_{max} = Nl \sin \theta_{max} = 1.3 \text{ m}$ and the collapsed height $h_{min} = 156 \text{ mm}$. Assuming $W = 8 \text{ kg}$ as payload the maximum force $F_x = 342 \text{ Kg}$ is required at $\theta_{min} = 5^\circ$.

The motor torque required for the scissor mechanism is calculated using screw jack formula given in [97] and presented here as Equation 2.15.

$$T_L = \frac{F_x d_m}{2} \left(\frac{p + \pi \mu d_m \sec \alpha}{\pi d_m - \mu p \sec \alpha} \right) = 7.5 \text{ Nm} \quad (2.15)$$

where coefficient of friction, $\mu = 0.1$, ACME thread angle, $2\alpha = 60^\circ$, pitch diameter, $d_m = 15 \text{ mm}$, and pitch, $p = 1.5 \text{ mm}$, is used. Based on the above specification 10 W, RE20 DC motor with a gear box of 25:1 ratio was chosen from Maxon motor catalogue [98].

2.5 Summary

Design calculations for the proposed mobile manipulator are presented in this chapter. Different aspects based on the requirements of radiation inspection

around cyclotron was taken into account. Advantage of positively steered wheels over caster wheel was highlighted for the proposed mobile robot.

Chapter 3

Dynamics of Wheeled Mobile Robots

In the field of mobile robotics, extensive research has been carried out. Mobile robots can broadly be divided into three categories, namely wheeled robots, legged robots [99], and aerial vehicles [100]. There are few mobile robots which use both wheels and legs for locomotion. For example, Creadapt [101] in order to take advantage of both modes of locomotion. Among these, the most extensively studied are the Wheeled Mobile Robots (WMR). They have been classified into five generic classes by Champion et. al. [13], [102] based on their mobility resulting from the kinematic constraints due to different wheel types. The most common among these are the 3 wheeled differentially driven WMR with one castor wheel. Because of its simplicity in modelling, they have been used in most of the control and motion planning algorithms [103], [104] and [105].

In order to develop a model-based control algorithm, it is imperative to have a good dynamic model of the WMR. These dynamic models are used in simulation software, Software in Loop (SIL) testing and Hardware in Loop (HIL) testing of the controllers. Different methods have been adopted to derive the dynamic model of WMRs. A general dynamical model was derived for three-wheel mobile robots with nonholonomic constraints by B. d'Andrea-Novel [106] using Lagrange

formulation. Alternatively, Thanjavur and Rajagopalan [107] have used Kane's method, and Saha et al. [22],[21] used Natural Orthogonal Compliment (NOC) method for the same.

In this chapter we first present the mathematical model of the mobile robot with two passive wheel at the front. It is then proved using kinematics why a passive wheel with center offset along the plane of the wheel self orients. This has not been explicitly presented in literature. Using the same set of equations it is proved that zero offset castors must be actuated.

The concept of the NOC which inherently takes into account the non-holonomic constraint of the wheels has been used. Moreover, being a matrix based approach, it is convenient for numerical simulations.

3.1 Modeling using the Natural Orthogonal Compliment (NOC)

Let us consider a system with n rigid bodies interconnected with different types of joints. Let, f_i be the net force acting at the center of mass (CM) of the i^{th} body and n_i is the net moment. If m_i is the mass, I_{ci} is the moment of inertia with respect to the CM, c_i is the position vector of the CM and ω_i is the angular velocity of the same body, then equations of motion of the i^{th} rigid body are given by Newton-Euler equations as

$$f_i = m_i \ddot{c}_i \quad \text{and} \quad n_i = I_{ci} \omega_i + \omega_i \times I_{ci} \omega_i \quad (3.1)$$

Let us define twist (t) and wrench(w) as

$$t_i \equiv \begin{pmatrix} \omega_i \\ \dot{c}_i \end{pmatrix} \quad w_i \equiv \begin{pmatrix} n_i \\ f_i \end{pmatrix}$$

Note that the wrench w_i acting on the i^{th} body can be decomposed into w_i^w , called the *working component* and w_i^c , the *non-working component*. The working component consists of all the external moments and forces, which imparts/extracts energy to/from the system, e.g., motor actuating torque. The non-working component of the wrench consists of the moments and forces that are used to constrain the motion of the body at the joints. Then, Newton-Euler equations (3.1) can be rewritten in a single matrix equation as

$$M_i \dot{t}_i + W_i M_i t_i = w_i^w + w_i^c \quad \because w_i \equiv w_i^w + w_i^c \quad (3.2)$$

where

$$M_i \equiv \begin{pmatrix} I_{ci} & 0 \\ 0 & m_i \tilde{I} \end{pmatrix}, \quad W_i \equiv \begin{pmatrix} \Omega_i & 0 \\ 0 & 0 \end{pmatrix}, \quad \Omega_i \equiv \omega_i \times \tilde{I} \quad (3.3)$$

in which Ω_i , is referred as the cross-product matrix of vector ω_i and \tilde{I} denotes the identity matrix. For details, refer to [20],[108]. If we define

$$M \equiv \text{diag}[M_1, M_2, \dots M_n], \quad W \equiv \text{diag}[W_1, W_2, \dots W_n], \quad t \equiv [t_1^T, t_2^T, \dots t_n^T]^T$$

and

$$w^j \equiv [w_1^{jT}, w_2^{jT}, \dots w_n^{jT}]^T, \quad j = c, w$$

then the equations of all the n rigid bodies in the system can be collected and written as a single matrix equation

$$M \dot{t} + W M t = w^c + w^w \quad (3.4)$$

The above equation is referred to as decoupled equations of motion of the system.

The kinematic constraints, both holonomic and non-holonomic (e.g., pure rolling), between two bodies i and j of a system can be expressed as a linear

homogeneous system of algebraic equations [20], namely

$$A_i t_i + A_j t_j = 0 \quad (3.5)$$

where A_i, A_j depend on the kinematic parameters.

The constraint equations corresponding to all the joints in the system can be written in terms of the *generalized twist vector* t . Furthermore, if $\dot{\theta} \equiv [\dot{\theta}_1, \dot{\theta}_2, \dots, \dot{\theta}_n]^T$ denote the *independent generalized joint rates*. One can then write t in terms of $\dot{\theta}$ as $t = T\dot{\theta}$. Using the fact, that $\dot{\theta}$ can take any arbitrary value, we get

$$At = 0, \Rightarrow AT\dot{\theta} = 0 \Rightarrow AT = 0 \quad (3.6)$$

The Equation 3.6 indicates that T is the orthogonal complement of A . Since this relation arises naturally, hence the name *Natural Orthogonal Complement*. It can be shown [20] that the non-working wrench w^c lies in the range space of A^T . In view of Equation 3.6, it can be proved that w^c lies in the null space of T^T . Therefore,

$$T^T w^c = 0 \quad (3.7)$$

To eliminate the non-working moments and forces, i.e., w^c from the uncoupled equation of motion (3.4), we multiply both sides of the equation by T^T ,

$$T^T M \dot{t} + T^T W M t = T^T w^W, \Rightarrow T^T M T \ddot{\theta} + T^T (M \dot{T} + W M T) \dot{\theta} = T^T w^T \quad (3.8)$$

Equation 3.8 represents the dynamic equation of interconnected n -body system. This equation is expressed in terms of the independent generalized joint rates $\dot{\theta}$ and cosponsoring acceleration $\ddot{\theta}$. Further, using the relations $t = T\dot{\theta}$ and $\dot{t} = \dot{T}\dot{\theta} + T\ddot{\theta}$ in Equation 3.8 the final equations of motion can be written as

$$I(\theta) \ddot{\theta} = C(\theta, \dot{\theta}) \dot{\theta} + \tau \quad (3.9)$$

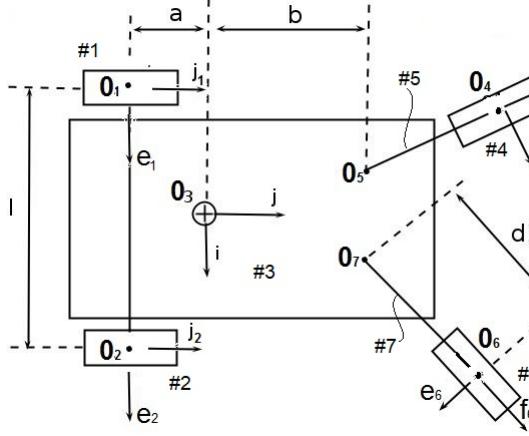


Figure 3.1: WMR-Std. castor

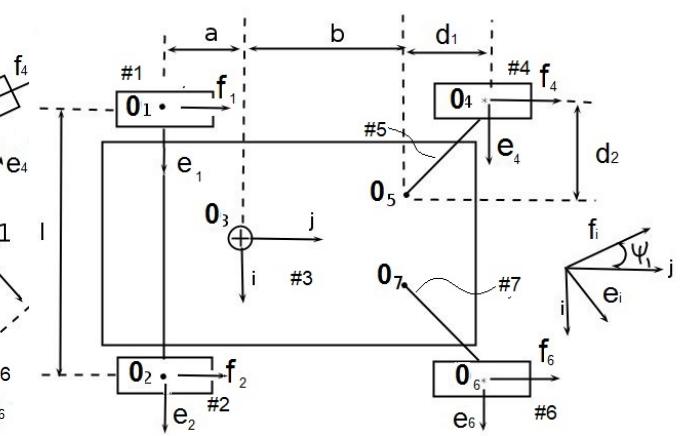


Figure 3.2: WMR-general

where

$$I(\theta) \equiv T^T M T \quad : \text{generalized inertia matrix}$$

$$C(\theta, \dot{\theta}) \equiv -T^T (M\dot{T} + WMT)\dot{\theta} \quad : \text{generalized convective inertia matrix}$$

$$\tau \equiv T^T w^T \quad : \text{generalized vector of driving forces}$$

3.2 Dynamic Equation of WMR

The dynamic equation of a differentially driven 3 wheeled mobile robot based on the Natural Orthogonal Compliment (NOC) has been presented by Saha [22]. The vehicle consisted of 2 driven wheels and one standard caster wheel. In general, for large vehicles, it is necessary to have at least four wheels from the point of stability of the vehicle. Such a vehicle is shown in Figure 3.1. It may be noted that the caster wheels used in this case are the standard caster, where the angle between line O_4O_5 and vector e_4 is 90° . Which is a special case of the more general configuration of the passive wheels shown in figure 3.2, where angle between line O_4O_5 and vector e_4 is not 90°

The vehicle considered for analysis in this chapter is shown in Figure 3.2. It consists of two independently driven wheels at the back and two passive wheels at the front. The actuated wheels are labelled as body #1 and #2 as indicated

in Figure 3.2. The platform is body #3. The first caster wheel and its bracket is labelled as #4 and #5 respectively, with the castor pivoted at O_5 . Similarly, the second castor is pivoted at O_7 , and its bracket and wheel are labelled as #6 and #7, respectively, all the wheels are assumed to be rolling without slipping.

3.2.1 Kinematic analysis

In order to proceed with the kinematic analysis of the vehicle in Figure 3.2, we define an orthogonal triad of vectors i, j, k at point O_3 , the control point of the platform, as shown in Figure 3.2. If $\dot{\theta}_1$ and $\dot{\theta}_2$ (positive when pointing along i) denote the rates of rotation of wheels #1 and #2 then the linear velocity of points O_1 and O_2 under pure rolling condition is given by

$$\dot{o}_i = -r\dot{\theta}_i j, \quad r=\text{radius of wheel} \quad (3.10)$$

The angular velocity of the platform ω_3 can be written as

$$\omega_3 = (r/l)(\dot{\theta}_1 - \dot{\theta}_2)k \quad (3.11)$$

Further, the velocity of point O_3 can be written as $\dot{o}_3 = \dot{o}_i + \omega_3 \times (c - o_i)$, $i = 1, 2$, where o_3 and o_i is the position vector of points O_3 and O_i respectively, with respect to some point fixed to the ground. Eliminating ω_3 one gets the following:

$$\dot{o}_3 = (ar/l)(-\dot{\theta}_1 + \dot{\theta}_2)i - (r/2)(\dot{\theta}_1 + \dot{\theta}_2)j \quad (3.12)$$

Now, the angular velocity of the drive wheel #1 can be expressed as $\omega_1 = -\dot{\theta}_1 i + \omega_3 k$. Using Equation 3.11, the same can be rewritten as

$$\omega_1 = \begin{pmatrix} -i + (r/l)k & -(r/l)k \end{pmatrix} \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix} \quad (3.13)$$

Similarly for the second wheel (#2) we get

$$\omega_1 = \begin{pmatrix} (r/l)k & i - (r/l)k \end{pmatrix} \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix} \quad (3.14)$$

Based on Equations 3.10 and 3.13, the twist for wheel #1 in terms of $\dot{\theta}_a \equiv (\dot{\theta}_1, \dot{\theta}_2)^T$, can be written as

$$t_1 = \begin{pmatrix} \omega_1 \\ \dot{\omega}_1 \end{pmatrix} = \begin{pmatrix} i + (r/l)k & -(r/l)k \\ -rj & 0 \end{pmatrix} \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix} \quad (3.15)$$

Similarly, for the other actuated wheel #2, one gets

$$t_2 = \begin{pmatrix} \omega_2 \\ \dot{\omega}_2 \end{pmatrix} = \begin{pmatrix} (r/l)k & i - (r/l)k \\ 0 & -rj \end{pmatrix} \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix} \quad (3.16)$$

To calculate the twist, t_3 of the platform body #3, Equations 3.11 and 3.12 are combined to get

$$t_3 = \begin{pmatrix} \omega_3 \\ \dot{\omega}_3 \end{pmatrix} = \begin{pmatrix} \rho\delta & -\rho\delta \\ r(\lambda i + (1/2)j) & r(-\lambda i + (1/2)j) \end{pmatrix} \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix} \quad (3.17)$$

where

$$\delta \equiv d/l, \quad \rho \equiv r/d, \quad \lambda \equiv a/l$$

From the above three equations, one gets

$$T_1 = \begin{pmatrix} i + (r/l)k & -(r/l)k \\ -rj & 0 \end{pmatrix}, \quad T_2 = \begin{pmatrix} (r/l)k & i - (r/l)k \\ 0 & -rj \end{pmatrix}$$

,

$$T_3 = \begin{pmatrix} \rho\delta & -\rho\delta \\ r(\lambda i + (1/2)j) & r(-\lambda i + (1/2)j) \end{pmatrix}$$

In order to calculate the twists of the caster bracket and the caster wheel, it

is necessary to express the un-actuated joint rates, $\dot{\psi}_1$ and $\dot{\phi}_1$, in terms of the actuated joint rate vector $\dot{\theta}_a$. Note here that $\dot{\psi}_1$ denotes the rate of rotation of the bracket body (#5) about O_5 with respect to the platform, and $\dot{\phi}_1$ is the rate of rotation of caster wheel body (#4) about its axis e_4 with respect to the bracket.

$$\omega_5 = \dot{\psi}_1 k + \omega_3, \quad \omega_4 = \dot{\phi}_1 e_4 + \omega_5$$

The velocity of O_5 can be expressed in two independent forms, namely, one in terms of the velocity of O_3 and the other one in terms of the velocity of O_4 , i.e.,

$$\dot{o}_5 = \dot{o}_4 + \omega_5 \times (d_2 e_4 - d_1 f_4), \quad \dot{o}_5 = \dot{o}_3 + \omega_3 \times (bj - mi) \quad (3.18)$$

On equating the above two equations together, and using the rotation matrix (R) between coordinate system $\{i, j, k\}$ and $\{e_4, f_4, k\}$ given Equation 3.19,

$$R = \begin{pmatrix} \cos(\psi_1) & -\sin(\psi_1) \\ \sin(\psi_1) & \cos(\psi_1) \end{pmatrix} \quad (3.19)$$

the equation in terms e_4 and f_4 , is obtained as

$$(-\dot{\phi}_1 r + \dot{\psi}_1 d_1) f_3 + d_3 \dot{\psi}_1 e_3 = \dot{o}_3 + \omega_3 (m \cos \psi_1 - b \sin \psi_1 - d_1) e_4 \quad (3.20)$$

Taking the dot product of the above equation first with e_4 and then with f_4 , and using Equation 3.12 for \dot{o}_3 , one gets

$$\begin{aligned} \begin{pmatrix} d_2 & 0 \\ -d_1 & r \end{pmatrix} \begin{pmatrix} \dot{\psi}_1 \\ \dot{\phi}_1 \end{pmatrix} &= \begin{pmatrix} (-ar/l)S_{\psi_1} + (r/2)C_{\psi_1} + \delta_1 & (ar/l)S_{\psi_1} + (r/2)C_{\psi_1} - \delta_1 \\ (ar/l)C_{\psi_1} + (r/2)S_{\psi_1} + \delta_2 & (-ar/l)C_{\psi_1} + (r/2)S_{\psi_1} - \delta_2 \end{pmatrix} \dot{\theta}_a \\ &= [F_{ij}] \dot{\theta}_a \end{aligned} \quad (3.21)$$

where,

$$\delta_1 = (r/l)(mC_{\psi_1} - bS_{\psi_1} - d2), \quad \delta_2 = (r/l)(mS_{\psi_1} + bC_{\psi_1} + d1)$$

Similarly, for the other caster wheel one gets,

$$\begin{aligned} \begin{pmatrix} d_2 & 0 \\ -d_1 & r \end{pmatrix} \begin{pmatrix} \dot{\psi}_2 \\ \dot{\phi}_2 \end{pmatrix} &= \begin{pmatrix} (-ar/l)S_{\psi_2} + (r/2)C_{\psi_2} - \delta_3 & (ar/l)S_{\psi_2} + (r/2)C_{\psi_2} + \delta_3 \\ (ar/l)C_{\psi_2} + (r/2)S_{\psi_2} + \delta_4 & (-ar/l)C_{\psi_2} + (r/2)S_{\psi_2} - \delta_4 \end{pmatrix} \dot{\theta}_a \\ &= [G_{ij}] \dot{\theta}_a \end{aligned} \quad (3.22)$$

where,

$$\delta_3 = (r/l)(mC_{\psi_2} + bS_{\psi_2} + d2), \quad \delta_4 = (r/l)(mS_{\psi_2} + bC_{\psi_2} + d1)$$

The angular and the liner velocity of the CM of the caster wheel #4, is written in terms of the co-ordinate frame fixed to the bracket #5, i.e. $\{e_4, f_4, k\}$, as

$$\omega_4 = \dot{\phi}_1 e_4 + (\omega_3 + \dot{\psi}_1)k, \quad o_4 = \dot{\phi}_1 e_4 \quad (3.23)$$

Using Equations 3.21 and 3.11, the twist t_4 can be written as

$$t_4 = \begin{pmatrix} \Theta_4 \\ C_4 \end{pmatrix} \dot{\theta}_a \quad (3.24)$$

Using the definition of $F(i, j)$ in Equation 3.21, Θ_4 and C_4 can be written as

$$\Theta_4 = [F_{11}e_4 + \bar{F}_{21}k \quad F_{12}e_4 + \bar{F}_{22}k], \quad C_4 = r[-F_{11}f_4 \quad -F_{12}f_4]$$

$$\bar{F}_{21} = F_{21} + \rho\delta, \quad \bar{F}_{22} = F_{22} - \rho\delta$$

The angular and the liner velocity of the CM of the caster bracket #5 expressed

in the co-ordinate frame fixed to the bracket is given as

$$\omega_4 = \dot{\phi}_1 e_3 + \dot{\psi}_1 k, \quad \dot{o}_4 = \dot{o}_4 + \omega_5 \times [-df_3] \quad (3.25)$$

Using Equations 3.11 and 3.22, the twist t_5 can next be expressed as

$$t_5 = \begin{pmatrix} \Theta_5 \\ C_5 \end{pmatrix} \dot{\theta}_a \quad (3.26)$$

where

$$\Theta_5 \equiv [\bar{F}_{21}k \quad \bar{F}_{22}k], \quad C_5 \equiv d[(1/2)\bar{F}_{21}e_4 - \rho F_{11}f_4 \quad (1/2)\bar{F}_{22}e_4 - \rho F_{12}f_4]$$

In a similar manner, the twists t_6 and t_7 of the other caster wheel and its bracket, respectively can be written as

$$t_6 = \begin{pmatrix} \Theta_6 \\ C_6 \end{pmatrix} \dot{\theta}_a, \quad t_7 = \begin{pmatrix} \Theta_7 \\ C_7 \end{pmatrix} \dot{\theta}_a \quad (3.27)$$

Using $G(i, j)$ defined in Equation 3.22, Θ_6 , C_6 , Θ_7 and C_7 are given by

$$\Theta_6 \equiv [G_{11}e_6 + \bar{G}_{21}k \quad G_{12}e_6 + \bar{G}_{22}k], \quad C_6 \equiv r[-G_{11}f_6 \quad -G_{12}f_6]$$

$$\Theta_7 \equiv [\bar{G}_{21}k \quad \bar{G}_{22}k], \quad C_7 \equiv d[(1/2)\bar{G}_{21}e_6 - \rho G_{11}f_6 \quad (1/2)\bar{G}_{22}e_6 - \rho G_{12}f_6]$$

$$\bar{G}_{21} \equiv G_{21} + \rho\delta, \quad \bar{G}_{22} \equiv G_{22} - \rho\delta$$

3.2.2 Special cases

Based on the above kinematic equations 3.21 and 3.22 for the caster wheels two special cases can be recognized one with $d_1 = 0$ and another ($d_2 = 0$). The RARS mobile robot designed fall in the second category and hence it requires a steering actuator as discussed below.

3.2.2.1 Standard caster ($d_1 = 0$)

The standard caster wheel configuration can be obtained by setting the value of $d_1 = 0$. In such condition, the left hand side matrix of Equations 3.21 and 3.22 become a diagonal matrix. Therefore, the first and second row of Equations 3.21 and 3.22 get divided by d_2 and r , respectively. The resulting equations relating the un-actuated joint rate to actuated joint rates are similar to those reported by [22],[20]. These passive wheel are self orienting. Hence there is no need for steering motors.

3.2.2.2 Underactuated case ($d_2 = 0$)

It can be seen from Equation 3.21 that when the *caster offset*, $d_2 = 0$, the LHS matrix becomes singular. So the unactuated joint rates cannot be determined from $\dot{\theta}_a$. It is therefore essential to have proper caster offset in case we need caster like behaviour from a passive wheel.

An alternative solution is to put an extra actuator to control the bracket motion i.e., $\dot{\psi}_i$. This is the case in Ackerman steering mechanism, where the steering wheel controls the orientation of the front passive wheels of a car

3.3 Dynamic equations

Based on the twists calculated in terms of the independent joint rate vector vector $\dot{\theta}_a$, the generalized inertia matrix and the matrix of convective inertia term for the coupled equation of motion 3.9 can be derived

3.3.0.1 Generalized inertia matrix, I

The equations gives in Section 4.2.1 of the twist of individual body, i.e $t_i = T_i \dot{\theta}_a$ are used to obtain $t = [t_1^T, t_2^T, \dots, t_7^T]^T$ and $t = T\dot{\theta}$ where $T = [T_1^T, T_2^T, \dots, T_7^T]^T$. Since the matrix M is block diagonal, the inertia matrix of the full system denoted by

I is given by

$$I = T^T M T = T_1^T M_1 T_1 + T_2^T M_2 T_2 + \dots T_7^T M_7 T_7 \quad (3.28)$$

Next, the contribution of the rear wheels to the inertia matrix, i.e., I_m is given by

$$I_m = \sum_{i=1,2} T_i^T M_i T_i$$

or

$$I_m = \begin{pmatrix} I_w + (\rho\delta)^2 H + m_w r^2 & -2(\rho\delta)^2 H \\ -2(\rho\delta)^2 H & I + (\rho\delta)^2 H + m_w r^2 \end{pmatrix} \quad (3.29)$$

where

$$M_i \equiv \begin{pmatrix} \tilde{I}_w & 0 \\ 0 & m_w \mathbf{1} \end{pmatrix}, \quad \tilde{I}_w \equiv \begin{pmatrix} I_w & 0 & 0 \\ 0 & H & 0 \\ 0 & 0 & H \end{pmatrix}$$

Matrix \tilde{I}_w is the 3×3 moment of inertia matrix of the wheel in co-ordinate frame $\{i, j, k\}$, m_w is mass of the motorized wheels and $\mathbf{1}$ is the 3×3 identity matrix. If the mass of the platform is m_p and its moment of inertia about vector k is I_p , then [20]

$$I_3 = T_3^T M_3 T_3 = I_p (\rho\delta)^2 \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} + m_p r^2 \begin{pmatrix} (1/4) + \gamma^2 & (1/4) - \gamma^2 \\ (1/4) - \gamma^2 & (1/4) + \gamma^2 \end{pmatrix} \quad (3.30)$$

Similarly, if m_c is the mass of the castor wheel and it is assumed to be a solid disk, then the generalized inertia matrix can be written as

$$\begin{aligned} I_c = \sum_{i=4,6} T_i^T M_i T_i &= (m_c r^2 / 4) \begin{pmatrix} 6F_{11}^2 + \bar{F}_{21}^2 & 6F_{11}F_{12} + \bar{F}_{21}\bar{F}_{22} \\ 6F_{11}F_{12} + \bar{F}_{21}\bar{F}_{22} & 6F_{12}^2 + \bar{F}_{22}^2 \end{pmatrix} \\ &+ \begin{pmatrix} 6G_{11}^2 + \bar{G}_{21}^2 & 6G_{11}G_{12} + \bar{G}_{21}\bar{G}_{22} \\ 6G_{11}G_{12} + \bar{G}_{21}\bar{G}_{22} & 6G_{12}^2 + \bar{G}_{22}^2 \end{pmatrix} \end{aligned} \quad (3.31)$$

If the mass of the brackets, i.e., body #5 and #7, are small compared to the mass of the caster wheels, then the contributions of $T_5^T M_5 T_5$ and $T_7^T M_7 T_7$ can be neglected.

3.3.0.2 Matrix of convective inertia term C

The matrix of convective inertia terms of Equation 3.9 can be broken down into two parts, $T^T M \dot{T}$ and $T^T W M T$. As can be seen from Equations 3.15, 3.16 and 3.17, T_1, T_2, T_3 associated with rear wheels and the platform is constant. Therefore

$$T^T M \dot{T} = 0$$

. The generalized inertia matrix is constant for the rear wheels and the platform, therefore the vector $I_i \omega_i$ is parallel to $\omega_i, i = 1..3$

$$\Rightarrow \omega \times I\omega = 0$$

$$\Rightarrow T^T W M T = 0$$

. This shows that contribution of the rear wheels and the platform to the convective inertia term is zero. Moreover, the mass of the brackets are assumed to be zero, so they also do not contribute to the convective inertia term. Hence

$$C = T^T M \dot{T} + T^T W M T = \sum_{i=4,6} T_i^T M_i \dot{T}_i + \sum_{i=4,6} T_i^T W_i M_i T_i \quad (3.32)$$

The expression for the first term is found by using Equations 3.21, 3.22, 3.24 and 3.27. The terms \dot{F}_{ij} and \dot{G}_{ij} denote the derivatives of the elements of the matrix F and G defined in Equations 3.21 and 3.22 . To find \dot{T}_4 and \dot{T}_6 we have used the

fact $\dot{e}_4 = \omega_4 \times e_4$ and $\dot{e}_4 = \omega_6 \times e_6$.

$$T^T M \dot{T} = (m_c r^2 / 4) \left[\begin{pmatrix} 6F_{11}\dot{F}_{11} + \bar{F}_{21}\dot{F}_{21} & 6F_{11}\dot{F}_{12} + \bar{F}_{21}\dot{F}_{22} \\ 6\dot{F}_{11}F_{12} + \dot{F}_{21}\bar{F}_{22} & 6F_{12}^2\bar{F}_{22} \end{pmatrix} \right. \\ \left. + \begin{pmatrix} 6G_{11}\dot{G}_{11} + \bar{G}_{21}\dot{G}_{21} & 6G_{11}\dot{G}_{12} + \bar{G}_{21}\dot{G}_{22} \\ 6\dot{G}_{11}G_{12} + \dot{G}_{21}\bar{G}_{22} & 6G_{12}^2\bar{G}_{22} \end{pmatrix} \right] \quad (3.33)$$

The second term of Equation 3.32 i.e, $\sum_{i=4,6} T_i^T W_i M_i T_i$ evaluates to zero, as shown below.

Next, consider passive wheel (#4), using Equation 3.24 and W defined in the Equation 3.3 one gets,

$$T_4^T W_4 M_4 T_4 = [\Theta_4, C_4] \begin{pmatrix} \Omega_4 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} I_4 & 0 \\ 0 & m_4 \mathbf{1} \end{pmatrix} \begin{pmatrix} \Theta_4 \\ C_4 \end{pmatrix} = \Theta_4 \Omega_4 I_4 \Theta_4 \quad (3.34)$$

To evaluate the above equation, we express all the terms in the coordinate system $\{e_4, f_4, k\}$. Moreover, $\omega_4 = \Theta_4 \dot{\theta}_a$, using definition of Θ_4 from Equation 3.24 we get

$$\omega_4 = (F_{11}\dot{\theta}_1 + F_{12}\dot{\theta}_2)e_4 + (F_{21}\dot{\theta}_1 + F_{22}\dot{\theta}_2)k$$

and the cross product matrix of ω_4 as

$$\Omega_4 \equiv \begin{pmatrix} 0 & -(F_{21}\dot{\theta}_1 + F_{22}\dot{\theta}_2) & 0 \\ (F_{21}\dot{\theta}_1 + F_{22}\dot{\theta}_2) & 0 & -(F_{11}\dot{\theta}_1 + F_{12}\dot{\theta}_2) \\ 0 & (F_{11}\dot{\theta}_1 + F_{12}\dot{\theta}_2) & 0 \end{pmatrix}$$

When the above expressions are substituted in Equation 3.34, we get

$$T_4^T W_4 M_4 T_4 = 0, \Rightarrow T^T W M T = 0 \quad (3.35)$$

Therefore, the matrix of convective inertia term C of Equation 3.9 evaluates to

$$C = (m_c r^2 / 4) \left[\begin{pmatrix} 6F_{11}\dot{F}_{11} + \bar{F}_{21}\dot{F}_{21} & 6F_{11}\dot{F}_{12} + \bar{F}_{21}\dot{F}_{22} \\ 6\dot{F}_{11}F_{12} + \dot{F}_{21}\bar{F}_{22} & 6F_{12}^2\bar{F}_{22} \end{pmatrix} + \begin{pmatrix} 6G_{11}\dot{G}_{11} + \bar{G}_{21}\dot{G}_{21} & 6G_{11}\dot{G}_{12} + \bar{G}_{21}\dot{G}_{22} \\ 6\dot{G}_{11}G_{12} + \dot{G}_{21}\bar{G}_{22} & 6G_{12}^2\bar{G}_{22} \end{pmatrix} \right] \quad (3.36)$$

All the components of Equation 3.9 are now been evaluated, except the τ . These are simply the torques exerted by the actuated wheels. This completes the dynamic model of the WMR with generalized passive wheel configuration.

3.3.1 Simulation

The mobile manipulator proposed in Chapter 3 was modelled as a differentially driven robot. The front wheels and steering mechanism were not included in the dynamic model as their masses are too small compared to the platform. The weight of each wheel is 0.3Kg, the steering mechanism 0.24Kg, whereas the weight of the platform is 70Kg.

In simulation the vehicle reference point O_3 is required to trace a circle of radius 5m. As shown in the Figure 3.3. Let, β be the angle between the line joining point O_3 and O with respect to $X - axis$. The function $\beta(t)$ was defined such that the full circle is completed in 60Sec and the velocity and acceleration of the robot is zero at the beginning ($t=0$) and end of travel ($t=60$).

$$\beta(t) = \frac{20\pi}{60^3}t^3 - \frac{30\pi}{60^4}t^4 + \frac{12\pi}{60^5}t^5 \quad (3.37)$$

The initial pose of the vehicle is parallel to the $Y - axis$ i.e. $\beta = 0$ as shown by dotted line in Figure 3.3.

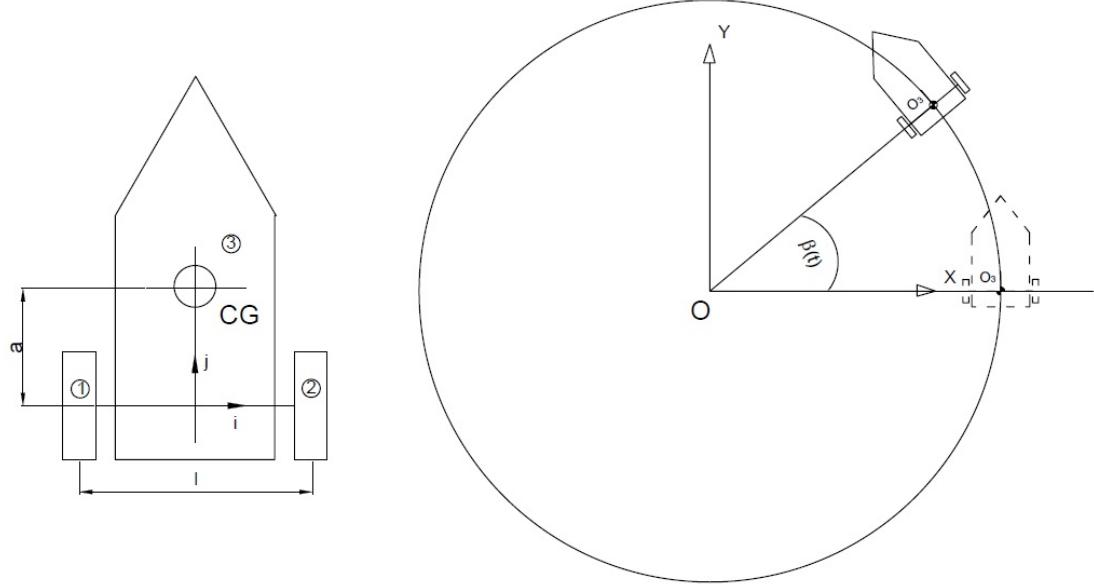


Figure 3.3: Path traced by robot

3.3.2 Inverse dynamics

In order to find the torque required to trace the mobile platform on the circular curve given in Figure 3.3, inverse dynamics was carried out. Using the model's inverse kinematic, the wheel velocity and acceleration were determined using pseudo inverse. The wheel angle, velocity and acceleration were used in the dynamics equation 3.38 to calculate the torque required by each motor. The results are plotted in figure 3.4.

The equation of dynamic model is given in equation 3.38, where the actuated joints are $\theta_a(t) = (\theta_l, \theta_r)$, the rear left and right wheel rotation angles. The dynamic and kinematic parameters used in the simulation is listed in table 3.1. The rear wheels ($i = 1, 2$) and the platform $i = 3$, twist $t_i^T = T_i\theta_a$, is given by equations 3.15, 3.16 and 3.17. The dynamic equation is of the vehicle on slope is

given as

$$\begin{aligned}
T^T M T \ddot{\theta}_a &= -T^T (M \dot{T} + W M T) \dot{\theta}_a + T^T (w^J + w^G) \\
\text{where } T &= (T_1^T T_2^T T_3^T)^T, \quad M = \text{diag}(M_1, M_2, M_3) \\
T^T M T &= I_1 + I_2 + I_m + I_3 = I_m + I_3 \\
W &= \text{diag}(W_1, W_2, W_3), \quad w^G = g \sin \alpha, \quad \alpha = 10^\circ
\end{aligned} \tag{3.38}$$

where w^G , w^J and α represent the gravitational force acting along the inclined plane, the external torque applied by a motor and the inclination of the plane to horizontal respectively. The generalized inertia matrix I_m and I_3 are given in 3.29 and 3.30. As stated earlier the convective inertia term is zeros as the inertia matrices are constant. The expression for motor torque is then given as

$$T_1^T w_1^j = \tau_1, \quad T_2^T w_2^j = \tau_2 \tag{3.39}$$

$$T_3^T w^g = \begin{pmatrix} \rho \delta k & r(\lambda i + (1/2)j) \\ -\rho \delta k & r(-\lambda i + (1/2)j) \end{pmatrix} \begin{pmatrix} 0 \\ -j g \sin \alpha \end{pmatrix} = \begin{pmatrix} -\frac{1}{2} r g \sin \alpha \\ -\frac{1}{2} r g \sin \alpha \end{pmatrix} \tag{3.40}$$

Figure 3.4 presents the torques required at the wheels of the vehicle while move up a spiral ramp of slope 10° and radius 5m.

Table 3.1: Dynamic & kinematic parameters

<i>Part Name</i>	<i>Property</i>	<i>Value</i>
Rear Wheels		
m_1, m_2	mass	300g
I_1, I_2	Moment of Inertia	$\text{diag}(242, 242, 465)\text{kg mm}^2$
Base Frame		
m_3	mass	70Kg
I_3	Moment Of Inertia	$\begin{pmatrix} 1.18 & 0.01 & -0.05 \\ 0.01 & 1.28 & 0.08 \\ -0.05 & 0.08 & 0.53 \end{pmatrix} \text{Kg} - \text{m}^2$
l	length	400mm
r	wheel radius (r)	50mm
a	see Figure 3.3	220 mm

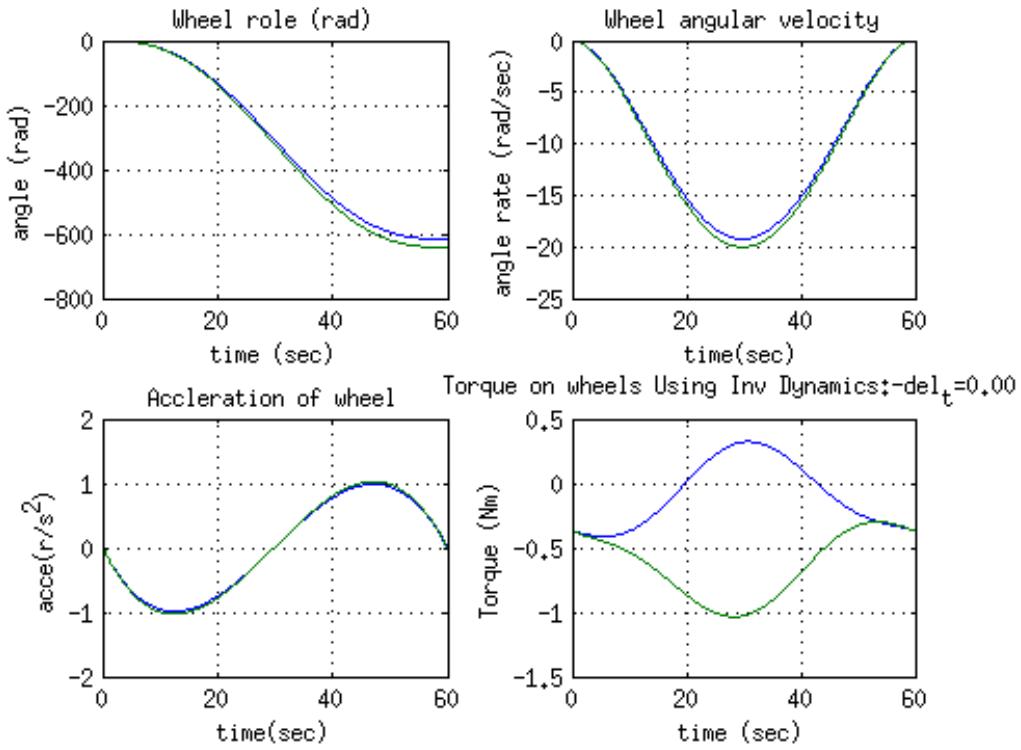


Figure 3.4: Inverse dynamics of the mobile robot

3.4 Dynamics with Wheel Slip

The above derivation of the dynamics of the mobile robot was based on the no slip condition between the ground and the wheel, i.e. pure rolling condition. In case of RARA robot the steering wheels are passive wheel but power steered, hence it is not assured always that passive wheel will correctly orient itself to assure pure rolling condition of these wheels. Therefore ideal rolling condition removed from the dynamic equation, both the longitudinal slip and lateral slip (also called skid) is introduced.

3.4.1 Defining Slip

Longitudinal slip occurs when the peripheral velocity of the wheel at the point of contact with the ground, $\dot{\phi} \times r$ is different from the linear velocity v_l of the wheel cg as shown in figure 3.6. Where, $\dot{\phi}$ is the angular velocity of wheel and r is the effective radius of the wheel. Under pure rolling assumption $\dot{\phi} \times r = v_l$.

Figure 3.5

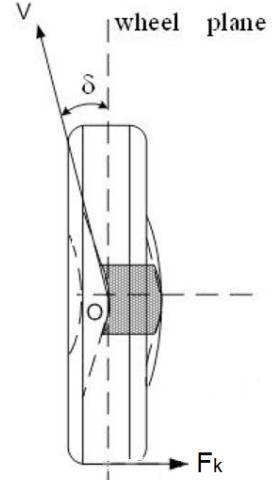


Figure 3.7: Longitudinal Slip or Skid

However, this is not the case for a deformed wheel. The wheel slipping can be characterized by slippage (or Slip ratio)

$$\rho = \frac{\dot{\phi} \times r - v_l}{\max(\dot{\phi} \times r, v_l)}$$

that has a value range of $\rho \in [-1, 1]$. $i = 0$ indicating no wheel slippage whereas $i = 1$ implies a complete slippage, i.e., the wheel is not moving linearly despite its angular rotation. In normal road condition, the wheel's slippage is usually $-1 < \rho < 1$. The lateral longitudinal force due to slip is related to the slip ratio ρ . One of the well known empirical relation to find F_s is the *magic formula* [3] used extensively in automotive industry to modeling Tire forces.

Lateral slip also called **skidding** is experienced when the wheel moves perpendicular to it's plain, this is general encountered during turning. This lateral movement is called skidding. The skidding produces frictional force perpendicular to the wheel plane as shown in figure 3.7 by F_k . The force F_k is related to the slip angle δ , which is the angle formed between the wheel plane and the velocity vector of the wheel's velocity, when projected on the ground plane. The force F_k can again be found using magic formula [3]. where insted of the slip ratio we use the slip angle as the parameter.

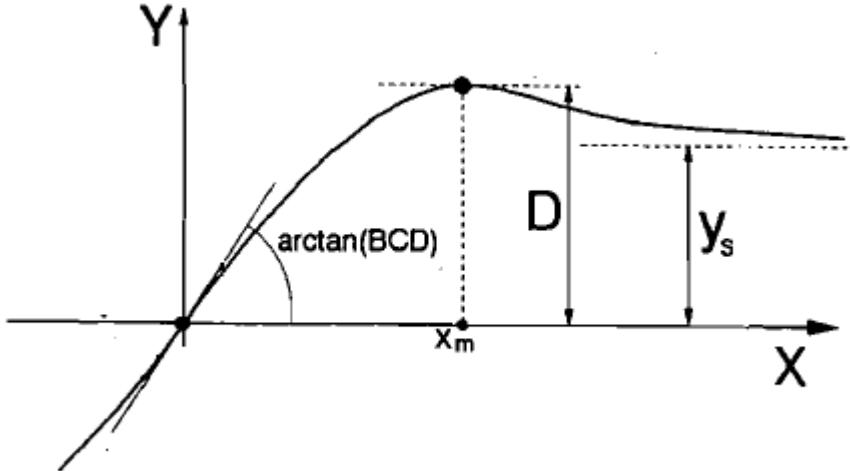


Figure 3.8: Tipical Curve of Magic Formula [3]

The magic formula 3.41 is reproduced below for completeness. In case of longitudinal slip $Y(x)$ represents longitudinal force F_s and x is the slip ratio ρ . For computation the lateral frictional force due to skidding x is replaced by the slip angle δ . The constants D represents the peak frictional force. The angle of tangent at the origin is equal to BCD . C controls the shape of the curve, $C = 2.4$ for F_k and $C = 1.65$ for F_s . B controls the slope at origin. The remaining constant E influences both the curvature of the curve near peak and also the position of x_m , where the peak is located.

$$Y(x) = D \sin[C \arctan Bx - E(Bx - \arctan(bx))] \quad (3.41)$$

3.4.2 Kinematics

The Figure shows the line diagram of the RARS robot. Even though the actual RARS robot uses Ackerman linkage for steering the front wheel, in this we have take the steer angle ψ_1, ψ_2 of each wheel independent, this makes the mathematical model more general. In order to simply things the link connecting the front wheels to the platform, tie rod etc associated with steering mechanism are not included in the analysis as there weight is small. The steering axis is assumed to pass through

the centre of the front wheel, i.e the offset is zero. This means $d_1 = d_2 = 0$ for figure 3.2, thus $O_4 = O_5$ and $O_6 = O_7$. Therefore in Figure ?? we use only $O4$ for wheel #4 and $O5$ for wheel #5, to denote both the centre of the wheel and also the steering pivot. DNoc will be used to derive the equations of motion of the RARS with wheel slip taken into account for all the four wheels.

The set of independent variables for this model is given by

$$\dot{\theta}_{9x1} = (\dot{\theta}_1, \dot{\theta}_2, \dot{y}_1, \dot{y}_2, \dot{x}, \dot{\phi}_1, \dot{\phi}_2, \dot{\theta}_4, \dot{\theta}_5)^T \quad (3.42)$$

It may be noted that θ_i, ϕ_i is same as defined earlier in Section 3.2. \dot{y}_1, \dot{y}_2 are the linear velocity of centre of the rear wheels i.e. of points point O_1 and O_2 , in the direction j of the body coordinate system. The lateral slip of the rear wheels are denoted by x_1 and x_2 as shown in the Figure ??. Due to rigidity constrains of the rear wheels

$$\dot{x}_1 = \dot{x}_2 = \dot{x}$$

The velocity of point o_1 and o_2 in the world coordinate X, Y, z is given by

$$\dot{o}_1 = \dot{x}i + \dot{y}_1j, \quad \dot{o}_2 = \dot{x}i + \dot{y}_2j \quad (3.43)$$

Equating the velocity of point O_3 expressed in terms of o_1 and o_2 , We get

$$\dot{o}_3 = \dot{o}_1 + \omega_3 \times o_1 o_3 = \dot{o}_2 + \omega_3 \times o_2 o_3 \quad (3.44)$$

Where, ω_3 is the angular velocity of platform, body #3.

$$\omega_3 = \frac{\dot{y}_2 - \dot{y}_1}{l}k = \bar{\omega}_3 k \quad (3.45)$$

using Equations 3.46 and 3.45 , we get

$$\dot{o}_3 = (\dot{x} + \frac{a\dot{y}_1}{L} - \frac{a\dot{y}_2}{L})i + (\frac{\dot{y}_1}{2} + \frac{\dot{y}_2}{2})j \quad (3.46)$$

We can write twist t_3 for body #3 as

$$t_3 = \begin{pmatrix} \omega_3 \\ \dot{o}_3 \end{pmatrix} = T_3 \dot{\theta}, \quad \text{Where, } T_3 = \begin{pmatrix} 0 & 0 & -k/l & k/l & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & ai/l + j/2 & -ai/l + j/2 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (3.47)$$

or in expanded form

$$T_3 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{L} & \frac{1}{L} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{a}{L} & -\frac{a}{L} & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (3.48)$$

Using the fact

$$\frac{di}{dt} = \bar{\omega}_3 k \times i = \bar{\omega}_3 j, \quad \frac{dj}{dt} = \bar{\omega}_3 k \times j = -\bar{\omega}_3 i, \quad \frac{dk}{dt} = \bar{\omega}_3 k \times k = 0$$

$$\dot{T}_3 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{\bar{\omega}_3}{2} i + \frac{a\bar{\omega}_3}{l} j & -\frac{\bar{\omega}_3}{2} i - \frac{a\bar{\omega}_3}{l} j & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (3.49)$$

The angular velocities ω_1 and ω_2 of wheels #1 and #2 is given as

$$\omega_1 = \dot{\theta}_1 i + \omega_3, \quad \omega_2 = \dot{\theta}_2 i + \omega_3 \quad (3.50)$$

using Equations 3.43 and 3.50 for wheel #1 we get the twist matrix as

$$t_1 = \begin{pmatrix} \omega_1 \\ \dot{o}_1 \end{pmatrix} = T_1 \dot{\theta} \quad (3.51)$$

$$T_1 = \begin{pmatrix} i & 0 & -k/l & k/l & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & j & 0 & i & 0 & 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{L} & \frac{1}{L} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (3.52)$$

and

$$\dot{T}_1 = \begin{pmatrix} \bar{\omega}_3 j & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\bar{\omega}_3 i & 0 & \bar{\omega}_3 j & 0 & 0 & 0 & 0 \end{pmatrix} \quad (3.53)$$

Using Equations 3.43 and 3.50 for wheel #2 we get the twist matrix t_2 as

$$t_2 = \begin{pmatrix} \omega_2 \\ \dot{o}_2 \end{pmatrix} = T_2 \dot{\theta} \quad (3.54)$$

$$T_2 = \begin{pmatrix} 0 & i & -k/l & k/l & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & j & i & 0 & 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{L} & \frac{1}{L} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (3.55)$$

and

$$\dot{T}_2 = \begin{pmatrix} 0 & \bar{\omega}_3 j & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\bar{\omega}_3 i & \bar{\omega}_3 j & 0 & 0 & 0 & 0 \end{pmatrix} \quad (3.56)$$

Next the angular velocity of the steered wheel i.e body #4 relative to the absolute frame is given as

$$\omega_4 = \dot{\theta}_4 e_4 + \dot{\phi}_1 k + \omega_3$$

The transformation matrix between the Frame $\{e_4, f_4\}$ is given by the rotation

matrix R_4 . From Figure R_4 can be written as

$$Rr = \begin{pmatrix} \cos \psi_1 & -\sin \psi_1 \\ \sin \psi_1 & \cos \psi_1 \end{pmatrix} \quad (3.57)$$

Then, $e_4 = \cos \psi_1 i + \sin \psi_1 j$ and ω_4 can be written as

$$\omega_4 = \dot{\theta}_4 (\cos \psi_1 i + \sin \psi_1 j) + \dot{\phi}_1 k + \omega_3$$

Linear velocity of point o_4 is given as

$$\dot{o}_4 = \dot{o}_3 + \omega_3 \times \left(-\frac{c}{2} i + b j \right)$$

or

$$t_4 = \begin{pmatrix} \omega_4 \\ \dot{o}_4 \end{pmatrix} = T_4 \dot{\theta} \quad (3.58)$$

where

$$T_4 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & \text{Cos}[\psi_1] & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \text{Sin}[\psi_1] & 0 \\ 0 & 0 & -\frac{1}{L} & \frac{1}{L} & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & \frac{a}{L} + \frac{b}{L} & -\frac{a}{L} - \frac{b}{L} & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} + \frac{c}{2L} & \frac{1}{2} - \frac{c}{2L} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (3.59)$$

Next, for wheel #5

$$\omega_5 = \dot{\theta}_5 (R_5 \cdot e) + \dot{\psi}_2 k + \omega_3 = \begin{pmatrix} \dot{\theta}_5 \cos \psi_2 \\ \dot{\theta}_5 \sin \psi_2 \\ \dot{\psi}_2 + \frac{-y_1 + y_2}{L} \end{pmatrix}$$

the linear velocity of point o_5 is given as

$$\dot{o}_5 = \dot{o}_3 + \omega_3 \times o_3 o_5$$

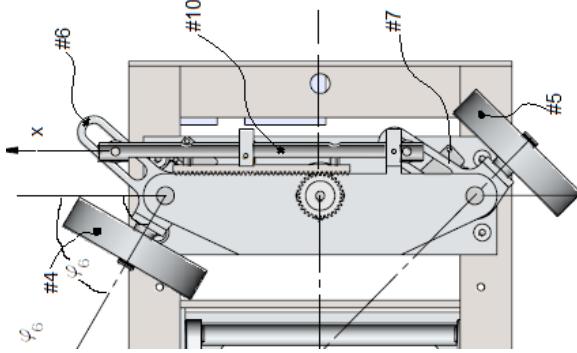


Figure 3.9: Steering Assembly

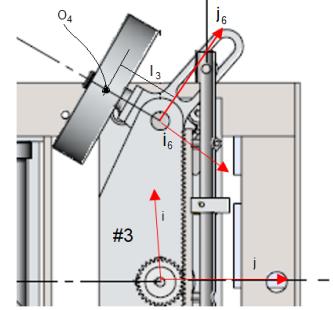


Figure 3.10: Link

The twist of body 5 , ie wheel 5 is given as

$$t_5 = \begin{pmatrix} \omega_5 \\ \dot{o}_5 \end{pmatrix} = T_5 \dot{\theta} \quad (3.60)$$

Where

$$T_5 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cos[\psi_2] \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \sin[\psi_2] \\ 0 & 0 & -\frac{1}{L} & \frac{1}{L} & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{a}{L} + \frac{b}{L} & -\frac{a}{L} - \frac{b}{L} & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} - \frac{c}{2L} & \frac{1}{2} + \frac{c}{2L} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (3.61)$$

3.4.3 Simulation

3.5 Dynamical Model of Steering System

The dynamical model of the steering system shown in Figure 3.9 was derived using Lagrangian method. The equations of motion were derived under the assumptions, that the vehicle body #3 was at rest and second that the mass of links #8 and #9 were negligible compared to other links. The kinematic energy of all the body except #8 and #9 is presented next.

The kinetic energy of link #6 is given by

$$K_6 = \frac{1}{2}(m_6((x_6^b)^2 + (y_6^b)^2) + I_{6zz})\dot{\phi}_6 \quad (3.62)$$

where, m_6 is the mass, (x_6^b, y_6^b) the coordinate of c.m. expressed in the body coordinate frame $B : \{i_6, j_6, k_6\}$ as shown in Figure 3.10 and I_{6zz} is the moment of inertia in the body coordinate of link #6.

Next we derive the kinetic energy of body # 4, the wheel. The coordinate of c.g of wheel, O_4 in the world $F : \{i, j, k\}$ is given by

$$\begin{pmatrix} x_4 \\ y_4 \end{pmatrix}_F = \begin{pmatrix} -\frac{l}{2} \\ 0 \end{pmatrix} + \begin{pmatrix} \cos \phi_6 & -\sin \phi_6 \\ -\sin \phi_6 & \cos \phi_6 \end{pmatrix} + \begin{pmatrix} -l_3 \\ 0 \end{pmatrix}$$

The linear velocity \dot{O}_4 and its angular velocity ω_6 expressed in the world frame $F : \{i, j, k\}$ is given by

$$\dot{O}_4 = l_6 \dot{\phi}_6 [\sin \phi_6, -\cos \phi_6]^T \quad (3.63)$$

$$\omega_6 = \dot{\phi}_6 k_6 + \dot{\theta}_4 i_6 = [\dot{\theta}_4 \cos \phi_6, \dot{\theta}_4 \sin \phi_6, \dot{\phi}_6]^T \quad (3.64)$$

where, l_6 is the length as shown in Figure 3.10 and $\dot{\theta}_4$ is the spinning rate of the wheel about axis i_6 . Then the kinetic energy of body #4 is given by

$$K_4 = \frac{1}{2}(m\dot{O}_4^T \dot{O}_4 + \omega_6^T [I_6]_F \omega_6) \quad (3.65)$$

$$K_4 = \frac{1}{2} l_6^2 \dot{\phi}_6 m \begin{pmatrix} \sin \phi_6, & -\cos \phi_6 \end{pmatrix} \begin{pmatrix} \sin \phi_6 \\ -\cos \phi_6 \end{pmatrix} + \begin{pmatrix} \dot{\theta}_4 \cos \phi_6, & \dot{\theta}_4 \sin \phi_6, & \dot{\phi}_6 \end{pmatrix} [I_6]_F \begin{pmatrix} \dot{\theta}_4 \cos \phi_6 \\ \dot{\theta}_4 \sin \phi_6 \\ \dot{\phi}_6 \end{pmatrix} \quad (3.66)$$

Where $[I_6]_F$ is the inertia matrix of wheel about its c.g. expressed in world frame F . In general the moment of inertia matrix of the body is known in the body frame B , ie $[I_6]_B$. This can be transformed to the world coordinate frame F by using the formula

$$[I_4]_F = R_B^F [I_4]_B [R_B^F]^T$$

where, R_B^F represents rotation transformation matrix between the fixed Frame, F , and the body frame, B . The above Equation, after above transformation can be written as

$$K_4 = \frac{1}{2} (l_3^2 \dot{\phi}_6^2 m_4 + [I_{4xx}]_B \theta_4^2) \quad (3.67)$$

where, I_{4zz} is the moment of inertia of the wheel about its z axis in the body coordinate frame.

The kinetic energy of the tie rod, i.e., body #10, which under goes linear reciprocating motion in a plane is given by

$$K_{10} = \frac{1}{2} m_{10} \dot{x} \quad (3.68)$$

where, x is the displacement of the tie rod and m_{10} is its mass.

The kinetic energy of body #5, i.e., second wheel and second link #7 can be derived in a similar fashion as that of body #4 and #6 respectively. The kinetic energy of body #7 is given as

$$K_7 = \frac{1}{2} (m_7 ((x_7^b)^2 + (y_7^b)^2) + I_{7zz}) \dot{\phi}_7 \quad (3.69)$$

where, m_7 is the mass and I_{7xx} moment of inertia of body #7.

The kinetic energy of body #5 is given as

$$K_4 = \frac{1}{2}(l_3^2\dot{\phi}_7^2m_5 + [I_{5xx}]_B\theta_5^2) \quad (3.70)$$

where m_5 is the mass and I_{5xx} moment of inertia of body #5.

From the geometry of Figure 2.8 following relations can be derived

$$\tan(\alpha - \phi_6) = \frac{b+x}{h}, \quad \tan(\alpha - \phi_7) = \frac{b-x}{h} \quad (3.71)$$

$$x_6^b + y_6^b = x_7^b + y_7^b = r_0 \quad (3.72)$$

$$\dot{\phi}_6 = \frac{h\dot{x}}{h^2 + b^2 + 2bx + x^2} = f_1(x)\dot{x}, \quad \dot{\phi}_7 = \frac{h\dot{x}}{h^2 + b^2 - 2bx + x^2} = f_2(x)\dot{x} \quad (3.73)$$

Therefore the total kinetic energy of steering unit is

$$K_s = \sum_{i=4..10} K_i$$

using the expression for K_i and using Equations 4.48, 4.49 and 4.50, the above equation for K_s is written as

$$K_s = \frac{\dot{x}^2}{2} \left(m_{10} + (m_l r_0^2 + I_{zz}) + l_3^2 m_w \right) \left(f_1^2(x) + f_2^2(x) \right) + \frac{1}{2} I_{wxx} (\dot{\theta}_4^2 + \dot{\theta}_5^2) \quad (3.74)$$

under the following assumptions

$$l_3 = l_4, \quad m_4 = m_5 = m_w, \quad m_6 = m_7 = m_l, \quad I_{7zz} = I_{6zz} = I_{zz}$$

The Lagrangian in this case is simply the kinetic energy, i.e, $L = K_s$. Since the external forces acts only along the x ordinate, via steering motor. We get

$$F_x = \frac{\partial}{\partial t} \frac{\partial L}{\partial \dot{x}} - \frac{\partial L}{\partial x}$$

or

$$F_x = \ddot{x} \left[m_{10} + (m_l r_o^2 + Izz + l_3^2 m_w) \right] (f_1^2 + f_2^2) + 2\dot{x}^2 (m_l r^2 + Izz + l_3^2 m_w) (f_1 \dot{f}_1 + f_2 \dot{f}_2) + F_1 + F_2 \quad (3.75)$$

The external force F_1 and F_2 in the above equation is given by

$$F_i = \frac{T_s}{h}, i = 1, 2$$

where T_s is evaluated using Equation 2.11 assuming symmetric loading of both the wheels. The derivative of $f_1(x)$ and $f_2(x)$ is given by

$$\frac{df_1}{dt} = -\frac{2(x+b)h}{(h^2 + b^2 + 2bx + x^2)^2}, \quad \frac{df_2}{dt} = -\frac{2(x-b)h}{(h^2 + b^2 - 2bx + x^2)^2}$$

3.5.1 Design of PID controller

To control the steer angle of the front wheels a computed torque controller [109] was designed. If U denotes the auxiliary control input then F_x is given by

$$F_x = U \left[m_{10} + (mr_o^2 + Izz + l_3^2 m_w) \right] (f_1^2 + f_2^2) + 2\dot{x}^2 (mr^2 + Izz + l_3^2 m_w) (f_1 \dot{f}_1 + f_2 \dot{f}_2) + F_1 + F_2 \quad (3.76)$$

eliminating F_x using Equation 3.75 and Equation 3.76, we get

$$\ddot{x} = U \quad (3.77)$$

If x_d , is the set point for the displacement of the rack of the steering system and $e(t) = (x(t) - x_d)$ is the position error, we define auxiliary input U as

$$U = -K_d \dot{x} - K_p (x(t) - x_d) - K_i \int e(t) dt \quad (3.78)$$

then Equation 3.77 which represents the overall dynamics of the steering mechanism along with the controller, can be written as

$$\ddot{e}(t) + K_d \dot{e}(t) + K_p e(t) + K_i \int e(t) dt = 0$$

It is be noted that $\dot{e}(t) = \dot{x}$ and $\ddot{e}(t) = \ddot{x}$, since $x_d = const.$. Given that $y_1(t) = \int e(t)$ and $Y = [y_1, y_2, y_3]^T$, the above equation can be rewritten in state space

$$\dot{Y} = AY$$

or

$$\begin{pmatrix} \dot{y}_1 \\ \dot{y}_2 \\ \dot{y}_3 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -K_i & -K_d & -K_p \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} \quad (3.79)$$

With $K_d = 20$, $K_p = 60$, $K_i = 100$ the system is stable as the state transition matrix A has all its eigen values, λ , with negative real part, as given below

$$\lambda = \{-16.7794 + 0.0000i, -1.6103 + 1.8348i, -1.6103 - 1.8348i\}$$

These controller parameters were arrived by trial and error using multiple simulation. The guiding principle behind the selection of these parameters was to choose one root far away from the imaginary axis so that its dynamics dies very fast with the respect to the other two roots. The system then behaves as second order system. The remaining two roots thus governs the dynamics of the system. they were chosen as complex conjugate with negative real part and near to imaginary axis so as to make the steering unit dynamics under damped.

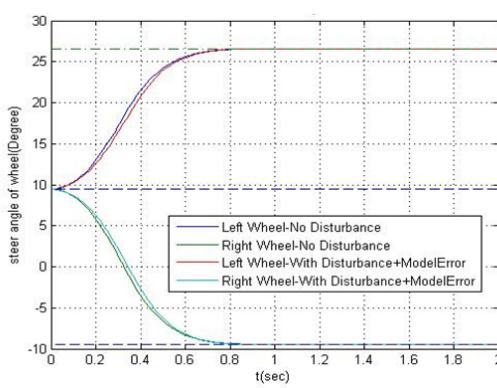


Figure 3.11: Change in steer angle

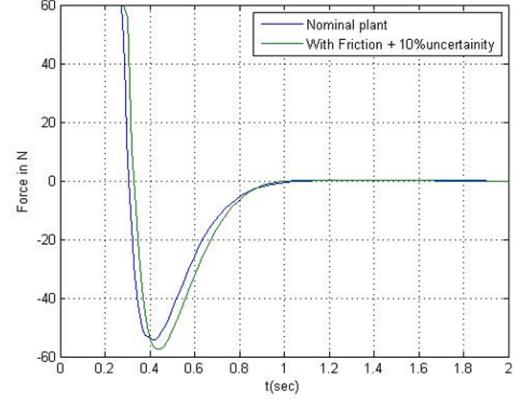


Figure 3.12: Control effort

3.5.2 Simulation and results

The simulation of the dynamic Equation 3.75 and controller given by Equations 3.76 and 3.78 was carried out with the step change in steering angle responding to 20 mm rack displacement (x). The parameter used for the nominal plant were obtained from solid model of the parts and are listed in table 3.2. The results of the wheel orientations and the actuator effort required are given in Figure 3.11 and 3.12 respectively.

Table 3.2: Key parameters steering assembly.

Wheel Mass	m_w	350 g
Wheel Inertia	I_{wxx}	463 Kg mm ²
Connecting Rod Mass	m_{10}	200 g
Link mass	m_l	80g
Link Inertia	I_{zz}	60834 Kg mm ²
Link Cg distance	r_o	18mm

The simulation results establishes that with 10% error in plant parameters there is practically no deviation in the performance of the controller. The results also predicts a settling time of 0.8 sec for the system. It may be noted that the roll velocity of the wheel does not affect the dynamics of the steering system as is indicated by the absence of $\dot{\theta}_4$ and $\dot{\theta}_5$ from Equation 3.75 .

3.6 Summary

In this chapter, the dynamic equations were derived for the most general form of passive wheel configuration. Even though only two passive wheel configuration was considered the same formulation can be extended to any number of wheels. It is shown that the dynamics of standard caster wheels is a special case of the general case with $d_1 = 0$. It is also proven why a caster needs non-zero caster offset. There is a need of extra actuator in case $d_2 = 0$. The dynamic equation for the Davis steering mechanism was formulated and using simulation the settling time for the same was obtained.

Chapter 4

Control of a Mobile Manipulator

In this chapter, the control architecture of the tele-operated mobile manipulator or platform is presented. The user interface for teleoperation is discussed. The control algorithm running on the mobile manipulator and the hardware used for the control of traction and steering is discussed. The protocol used for communication between the robot and the user interface is also described in detail.

4.1 Control Architecture and Hardware

The mobile manipulator explained in chapter 2 was planned to be teleoperated over a wireless network. The control block diagram and architecture are shown in Figure 4.1. It has a remote control station which is the interface for the operator

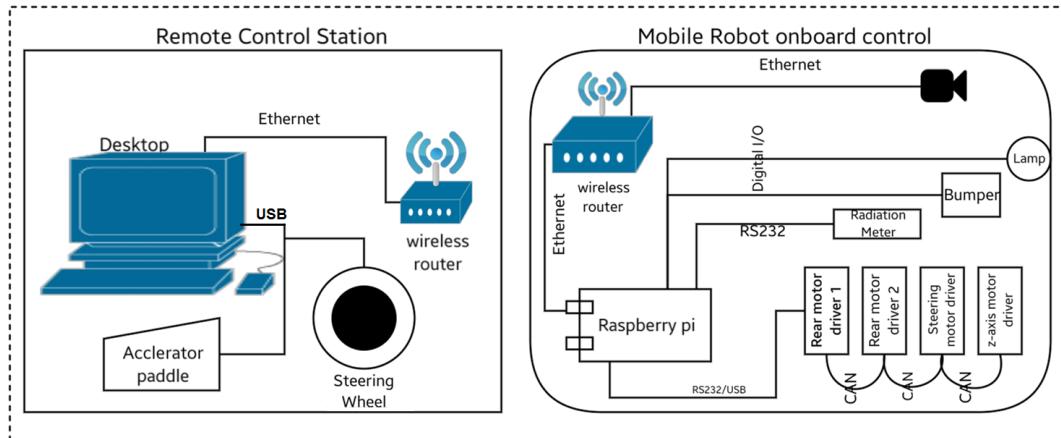


Figure 4.1: Control architecture

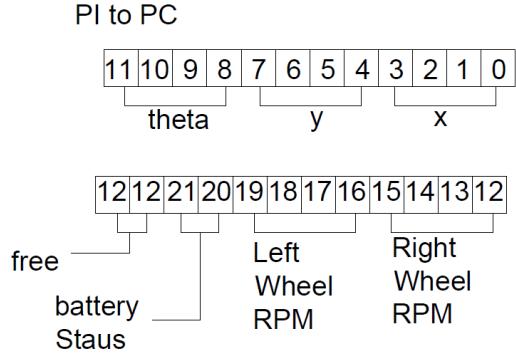
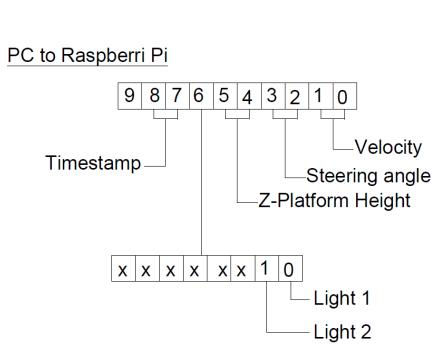


Figure 4.2: Data from PC to the robot Figure 4.3: Data from the robot to PC

and a local onboard controller of the mobile robot. They communicate over a dedicated wireless network. The remote station sends data packet in every 50 milliseconds (20Hz) to the mobile robot. The commanded velocity, steer angle, z position of the platform, and state of the detector and headlamps constitute the data packet sent by the remote station, as shown in Figure 4.2. The onboard controller of the mobile robot replies with a data packet consisting of the X, Y position and orientation θ of the robot, the current steer angle, angular velocities of each wheel, the z position of the top platform, battery voltage and current of each motor. They are indicated in Figure 4.3.

4.1.1 Local onboard controller

The onboard computer which is Raspberry Pi running Raspian (linux) OS receives command from the remote station and controls the robot hardware through customized C++ application. The Raspberry Pi is daisy chained to the four Maxon make, EPOS2 motor controllers/drivers. The communication between the onboard computer and the first Maxon controller is over usb/RS232 interface using Maxon's proprietorial protocol [110]. The first controller serves as CAN master for the rest of the controllers. The rear wheel motor drivers were configured in velocity servo loop. The drivers for steering and the z-axis motors were configured in position control loop. The camera mounted on the mobile robot and Raspberry Pi were connected over Ethernet via a wireless hub. The wiring diagram of the

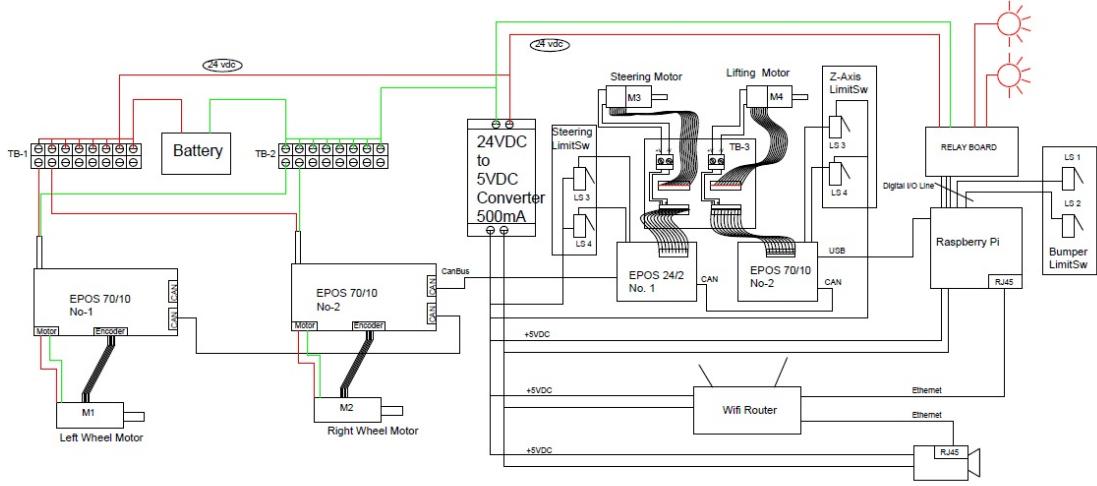


Figure 4.4: Wiring diagram of the WMR

robot is given in Figure 4.4. Since the onboard camera is connected to the wireless network directly, it does not interfere with the command loop between the Raspberry Pi and the PC.

The mobile robot is teleoperated using position-speed command as in [111]. The workspace for the mobile robot was assumed infinite compared to the input device. In case of manipulators position-position, control approach is generally used with scaling. In this case, the mixed approach was used. The steering angle was controlled in position-position mode whereas the mobile robot's speed was controlled by foot pedal's position, i.e. in position-velocity mode. This can be given by the following equation:

$$\begin{pmatrix} V \\ \theta_s \end{pmatrix} = \begin{pmatrix} K_v & 0 \\ 0 & K_s \end{pmatrix} \begin{pmatrix} \tilde{X}_p \\ \tilde{\Theta}_s \end{pmatrix} \quad (4.1)$$

where \tilde{X}_p is the displacement of pedal, $\tilde{\Theta}_s$ is the twist of steering wheel, K_v and K_s are the proportionality constants, V is the velocity of point O_3 and θ_s is the displacement of the steer motor. These proportionality constants are derived based on extreme limits. They are listed in Table 4.1.

Table 4.1: Proportionality constant table

Robot parameters	range	Joy-Stick Parameter	range	parameter Value
θ_s	-60 to +60	$\dot{\Theta}_s$	-90° to 90°	$K_s = 2/3$
V	0 to +60 mm/sec	\tilde{X}_p	0 to 30mm	$K_v = 2$

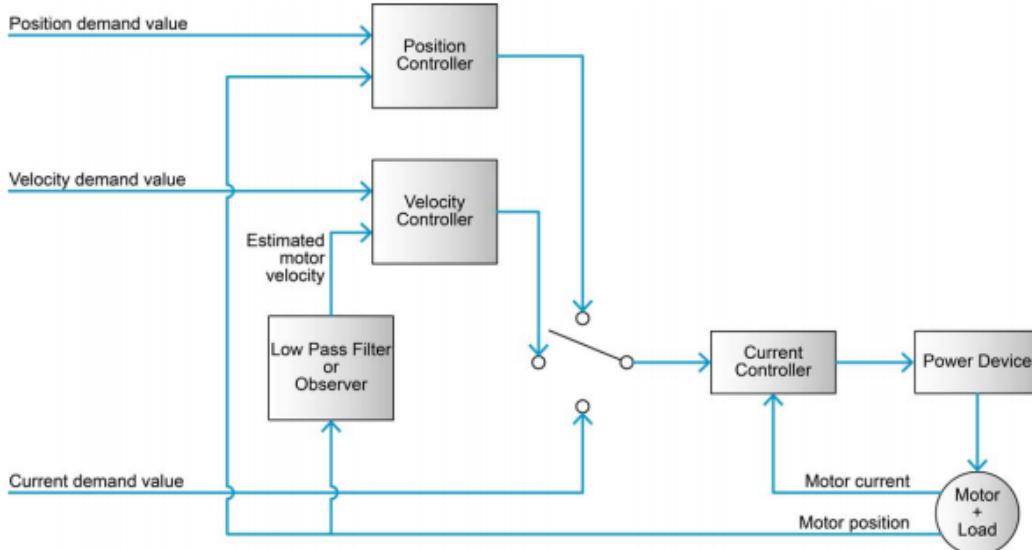


Figure 4.5: Block diagram of EPOS4 controller

4.1.2 Details of the motor controller

Three EPOS2 controllers from Maxon Motors were used to control the mobile robot's rear wheel velocity and the steering gears position. Each controller can control one motor. The overall architecture of the controller as given in [112], [113] is shown in Figure 4.5. The controller can be configured in either current, position or velocity control mode. The inner most current loop controls the torque of the motor. The current feedback loop shown in Figure 4.6 is a Proportional Integrator (PI) controller, running at 25KHz and the transfer function of the PI block is given as 4.2.

$$C(s) = K_p + \frac{K_I}{s} \quad (4.2)$$

where K_p and K_I are the proportional and integral gains.

The rear motor controllers are configured in the velocity control mode. The block diagram of the velocity loop is given in Figure 4.7. The velocity controller is a PI controller with velocity and acceleration feed-forward. The transfer function

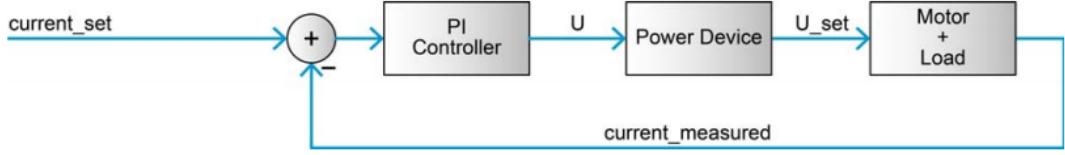


Figure 4.6: Current control block

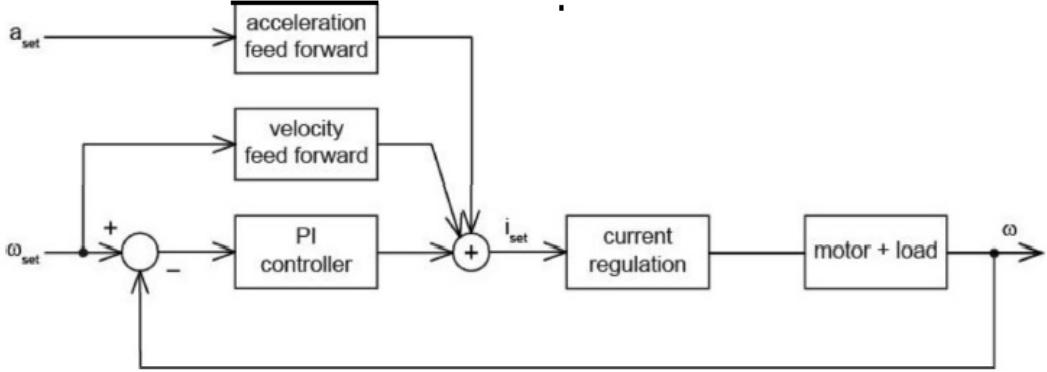


Figure 4.7: Velocity control bolck

$V(s)$ of the velocity loop PI block is given by

$$V(s) = K_{p\omega} + \frac{K_{I\omega}}{s} \quad (4.3)$$

where $K_{p\omega}$ and $K_{I\omega}$ are proportional and integral gains for velocity, respectively. The sampling rate of the velocity loop is 2.5 KHz. The feedforward acceleration and velocity was used to compensate for the known inertial load and viscous frictional load [113] respectively. Velocity is estimated from differentiation of the position data, the low-pass filter in Figure 4.5 eliminates noise due to differentiation. The transfer function $H(s)$ for the low-pass filter is given by

$$H(s) = \frac{1}{1 + \frac{K_{p\omega}}{48K_{I\omega}}} \quad (4.4)$$

The gain values used for each rear motor in the velocity control mode is listed in Table 4.2. No acceleration or velocity feedforward was used. The gain parameters were determined by auto tuning software provided by Maxon Motors.

The steering motor is in position control mode. The block diagram is shown in

Table 4.2: Parameters for left and right rear wheel motor controllers

<i>Gain Parameter</i>	<i>Right motor Value</i>	<i>Left motor Value</i>	<i>Unit</i>
K_p	300	230	$\frac{mV}{A}$
K_I	100	53	$\frac{mA}{A.mS}$
$K_{P\omega}$	1000	5182	$\frac{rad}{mA.sec}$
$K_{I\omega}$	100	425	$\frac{rad}{rad}$

Figure 4.8. It is a PID controller with transfer function given as

$$P(s) = K_{PP} + K_{IP}s + \frac{K_{DP}s}{1 + \frac{K_{DP}}{10K_{PP}}s} \quad (4.5)$$

where K_{PP} , K_{IP} and K_{DP} are position proportional, integral and derivative gains respectively. The velocity feed-forward $F_{\omega P}$ and acceleration feed-forward $F_{\alpha P}$ were used in position control loop to take care of viscous friction and known inertial load. The gains for controller were decided using auto tuning software provided by Maxon Motors. The values are reported in Table 4.3.

It may be noted that the controller parameters listed in table 4.3 is very different from that used in simulation. This difference is due to the fact that the control structures are different. In the actual system motor dynamics is also included, which is not the case of simulation. In simulation, position error directly affects the torque to the motor through the control parameters K_p , K_d and K_I , whereas as the position error in actual system sets the motor current. Hence there units too are different. Therefore direct comparison between simulation PID parameters and actual controllers PID values cannot be done.

4.2 Control Algorithm

This section discusses in detail the algorithm running on the onboard controller. Command received by the controller was parsed to extract the velocity and the steer angle information. They were suitably scaled to get command velocity V in

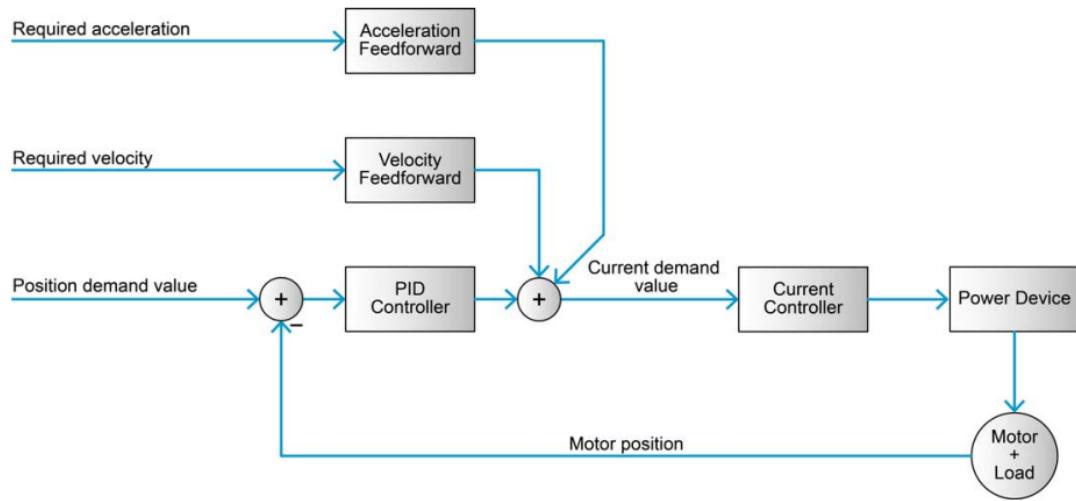


Figure 4.8: Position control bolck

Table 4.3: Parameters of steering motor controller

Gain Parameter	Value	Unit
K_p	537	$\frac{mV}{A}$
K_I	307	$\frac{mV}{A \cdot mS}$
K_{PP}	128	$\frac{mA \cdot sec}{rad}$
K_{IP}	663	$\frac{mA}{rad}$
K_{ID}	200	$\frac{mA}{rad}$
$F_{\omega P}$	0	$\frac{rad}{mA \cdot sec}$
$F_{\alpha P}$	54	$\frac{mA \cdot sec^2}{rad}$

mm/sec, and steering angle θ_s in radians. It may be noted that the velocity V corresponds to the velocity of point O_r the reference point of the mobile robot. Next the set point for each motor was calculated and sent to individual drive. The algorithm is listed below and the block diagram for the same in Figure 4.9

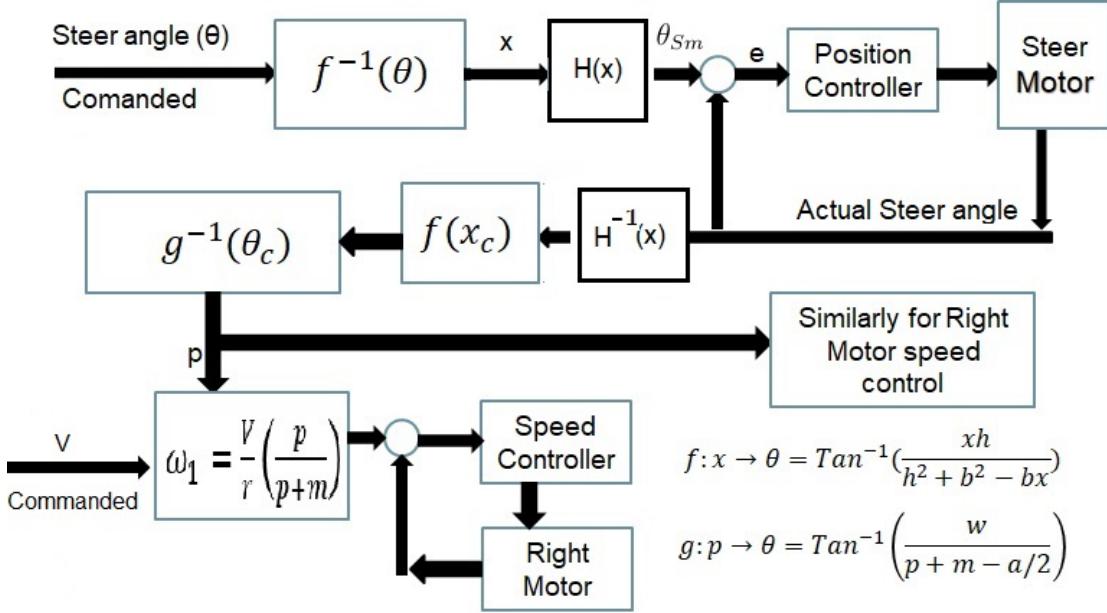


Figure 4.9: Block diagram of WMR controller

1. Calculate the setpoint for steering motor θ_{SM} based on θ_s .
2. Read the current steering angle ϕ_{ic} .
3. Calculate the velocity setpoints ω_i and ω_o of rear wheels based on the V and ϕ_{ic} .
4. Command setpoints ω_i , ω_o and θ_{SM} to each motor.

The above loop is repeated every 50 mSec.

It may be noted that the steer angle command received from the control station is not directly sent to the steer motor as set point after suitable scaling. The steer set point is based on the current rear wheel velocities. This is important as the response time of the motors are different. The above methodology helps minimize the deviation from the Ackerman steering condition even during transit condition,

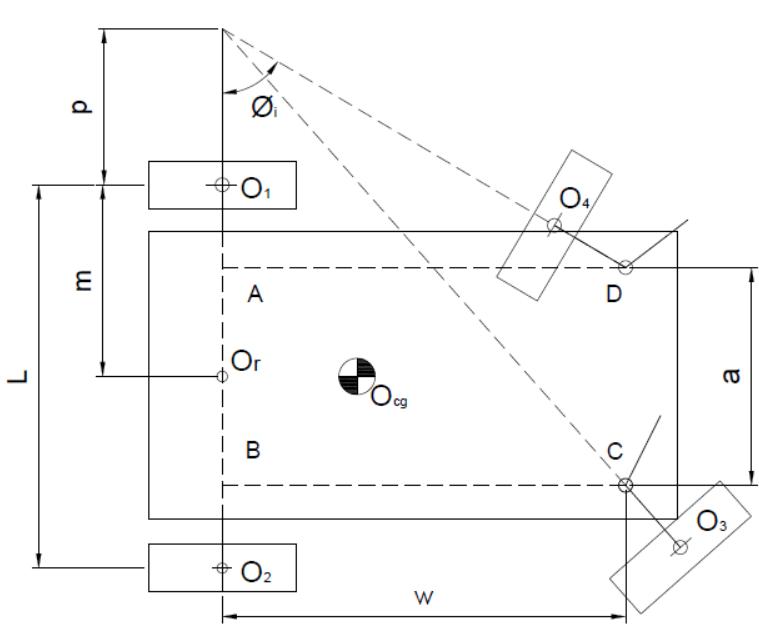


Figure 4.10: Ackerman Steering Condition

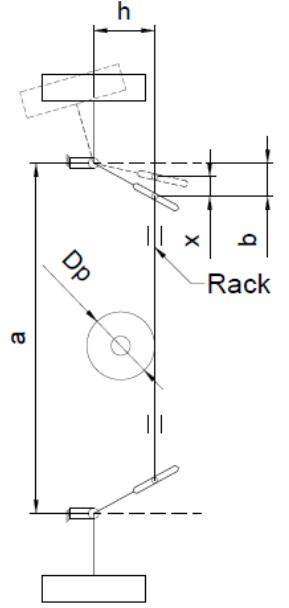


Figure 4.11: Davis Steering Gear

particularly, in case of large change in commanded v and θ_s . Each block in Figure 4.9 is discussed next.

It may be noted that the θ_s always refers to the steer angle of the inner front wheel ϕ_i and V refers to O_r as shown in Figure 4.10, i.e.

$$\theta_s = \phi_i \quad (4.6)$$

The set point of the steering motor at the output of gear box θ_{SM} is given by equations 4.7 and 4.8 based on the geometry of Davis steering gear [97].

$$\tan \phi_i = \frac{xh}{h^2 + b^2 - bx}$$

or

$$f(\phi_i) : x = \frac{\tan \phi_i (h^2 + b^2)}{h + b \tan \phi_i} \quad (4.7)$$

where x is the displacement of the rack and h and b are link lengths. The rack is connected to the steering motor by pinion of PCD D_p (30mm) as shown in Figure

4.11. Therefore, the steering motor angle, θ_{Sm} , is given below as

$$H(x) : \theta_{Sm} = x \frac{360}{\pi D_p} \quad (4.8)$$

Next, the equation relating current steer angle ϕ_i and rear wheel set point velocities ω_{RS} and ω_{LS} is presented. From the geometry of Figure 4.10, one gets

$$\begin{aligned} \tan \phi_i &= \frac{\bar{BD}}{\bar{OB}} = \frac{w}{p + m - a/2} \\ g : \phi_i \rightarrow p, \quad p &= \frac{w}{\tan \phi_i} - m + \frac{a}{2} \end{aligned} \quad (4.9)$$

Now using Equations 3.10 and 3.11 in equations relating the right and left wheel velocities to the WMR platform angular velocity ω_3 and velocity, V , of the reference point O_r , presented below

$$\begin{aligned} \dot{O}_r &= \dot{O}_i + \omega_3 \times (O_r - O_i) \\ \dot{O}_r &= \dot{O}_o + \omega_3 \times (O_r - O_o) \end{aligned}$$

We get the velocity of each rear wheel as

$$\begin{aligned} \omega_i &= \frac{Vp}{r(p+m)} \\ \omega_o &= \frac{V(p+2m)}{r(p+m)} \end{aligned}$$

Using the above equations and Equation 4.9, the setpoints for the rear wheels are as follows

$$\omega_i = \frac{V}{r} \frac{\left(\frac{w}{\tan \theta_o} - m + \frac{a}{2}\right)}{\frac{w}{\tan \theta_o} - m + \frac{a}{2} + m} \quad (4.10)$$

$$\omega_o = \frac{V}{r} \frac{\left(\frac{w}{\tan \theta_o} - m + \frac{a}{2}\right) + 2m}{\frac{w}{\tan \theta_o} - m + \frac{a}{2} + m} \quad (4.11)$$

4.2.1 Safety Interlocks

The control algorithm has safety interlocks built into it. The vehicle speed has been limited to 0.5m/sec, this was done based on operator feedback for convenience in driving the vehicle remotely. The vehicle does not move if the manipulator is in the extended condition. This is to avoid over turning of the vehicle in case it has to climb a ramp. Acceleration of the vehicle is never exceeded above the limiting value of 0.144g which was arrived at based on dynamic stability of the mobile robot as calculated in Equation 2.2. To avoid overturning of vehicle while following a circular path the linear velocity of the vehicle is limited by Equation 2.4 which is function of the turning radius.

4.2.2 Wheel odometry

The dead reckoning odometry can be performed based on either the differential drive or Bicycle model. In the present case, we use the differential drive model. Where the rear wheel velocities were used to determine the position and orientation of the mobile robot. The position here means the position of the reference point O_r . The steps are follows.

1. calculate V and ω_3 from current wheel velocities ω_1 and ω_2 using equations 3.11 and 3.12 with $a = 0$.
2. integrate V and ω_3 over time step.

If $x(t)$, $y(t)$ are the coordinate of O_r and $\beta(t)$ be the orientation of the robot with some global coordinate system, the kinematic model of the differential wheel robots is given by [13]

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\beta} \end{pmatrix} = \begin{pmatrix} \cos \beta & 0 \\ \sin \beta & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} V \\ \omega_3 \end{pmatrix} \quad (4.12)$$

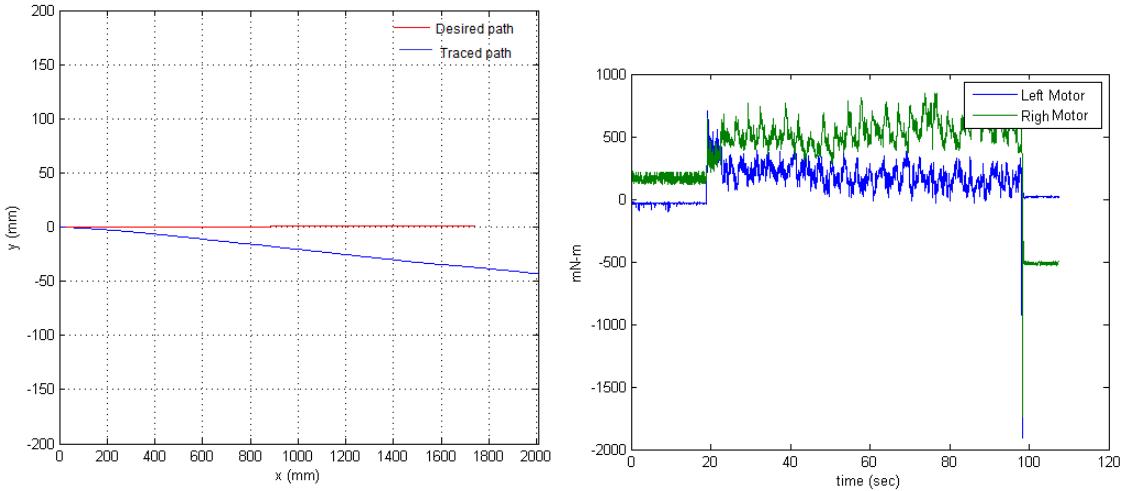


Figure 4.12: Tracing a line

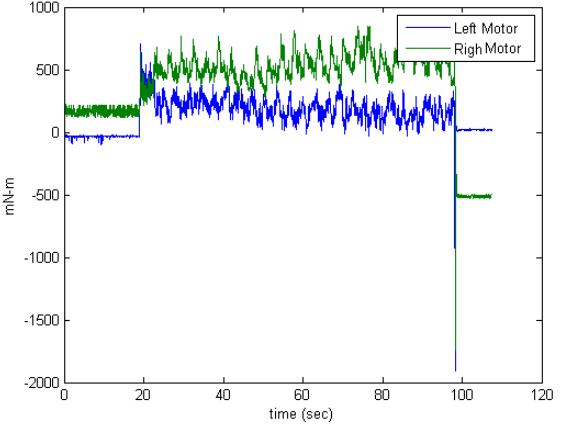


Figure 4.13: Motor torque

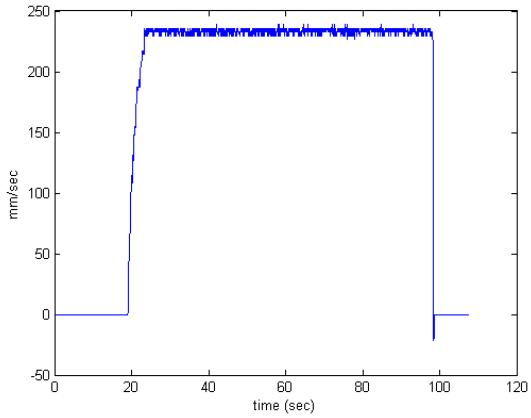


Figure 4.14: Linear velocity

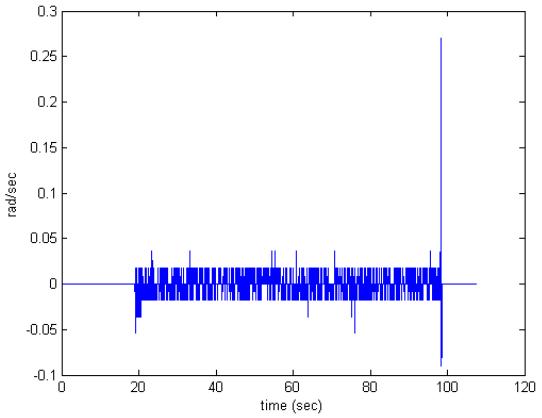


Figure 4.15: Angular velocity

Equation 4.12 is numerically integrated for time δt using the following expressions:

$$x(i + 1) = x(i) + \delta t V(i) \cos \beta(i); \quad (4.13)$$

$$y(i + 1) = y(i) + \delta t V(i) \sin \beta(i); \quad (4.14)$$

$$\beta(i + 1) = \beta(i) + \delta t \omega_3(i); \quad (4.15)$$

where i is at time step t_i . Next, the actual odometric results for the vehicle moving in a circle and a straight line are presented.

As seen in the graph of Figure 4.12 there is a lateral shift in the robots path calculated using odometry. There is a linear shift too, which can be observed due to longer path calculated by odometry. The lateral shift of 200 mm and a linear shift

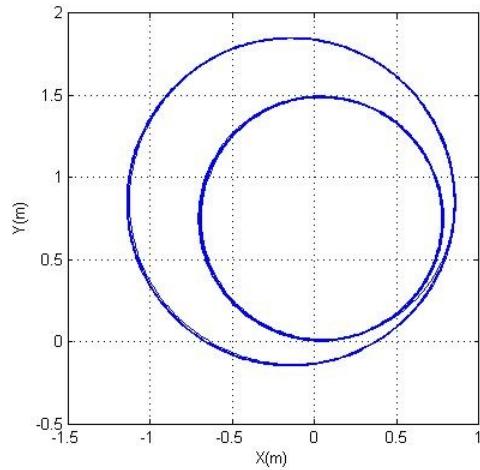


Figure 4.16: Tracing a circle

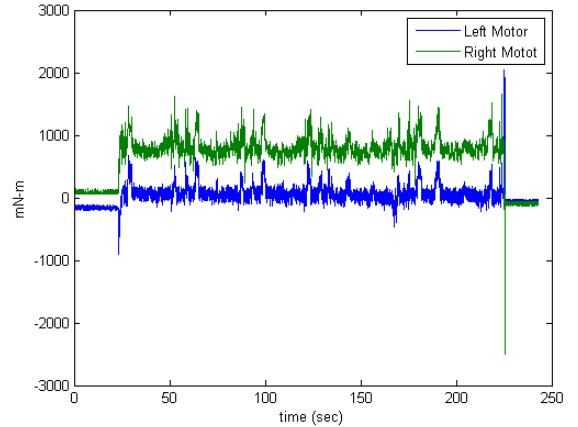


Figure 4.17: Motor torque

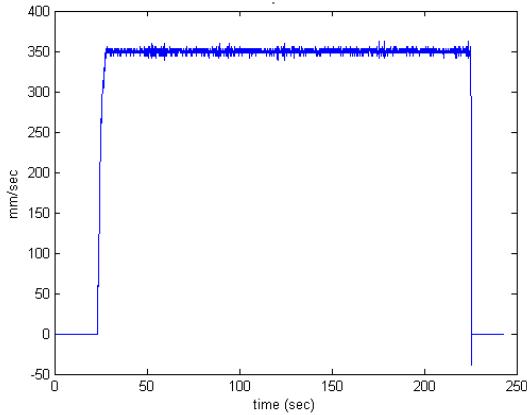


Figure 4.18: Linear velocity

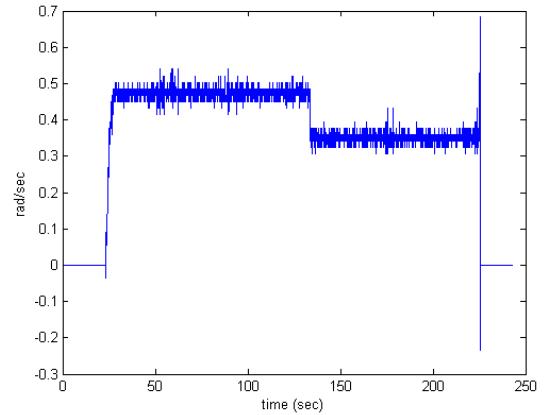


Figure 4.19: Angular velocity

of 300 mm for 15000 mm long path was calculated. This clearly indicates slip in the wheels. The torque curves also shows that one wheel is more loaded than the other this is expected as the battery weight was on one side of the robot.

In Figure 4.18 the robot traces circular path of two different radius. It can be seen that there is no lateral (side) slip during the motion. The step change in the angular velocity at time $t \approx 125 \text{ sec}$, shown in Figure 4.19, indicate transition to a larger radius path. The linear velocity of the robot is maintained constant at 300 mm/sec .

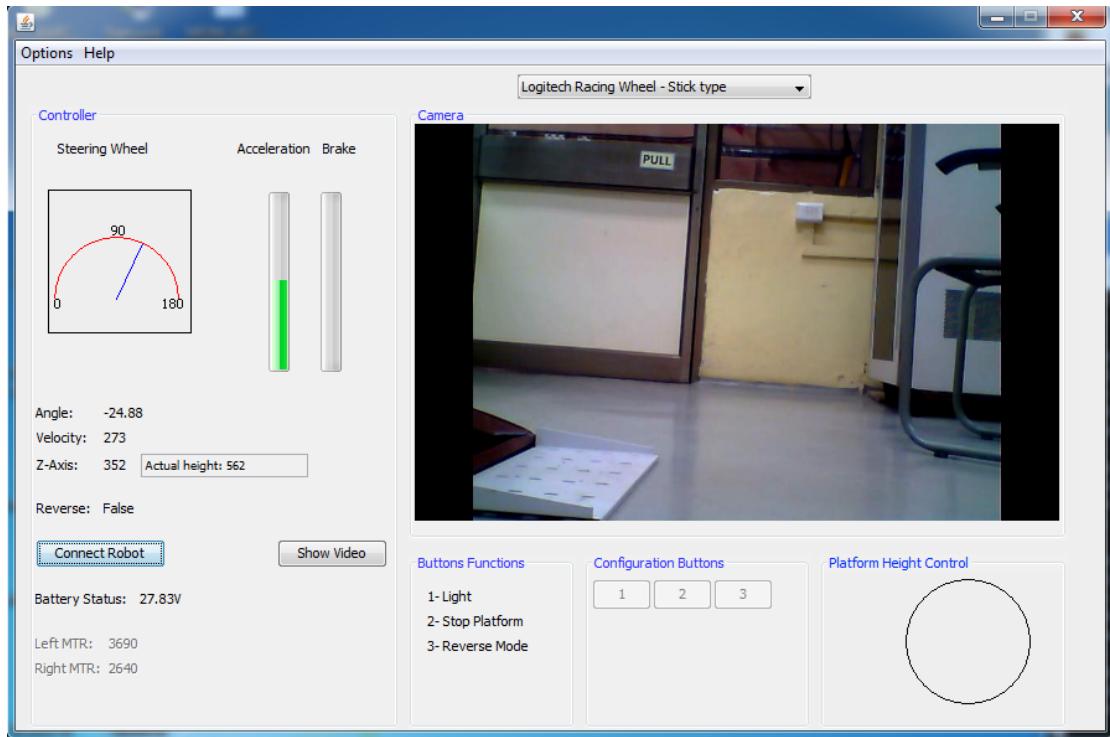


Figure 4.20: User interface for teleoperation

4.3 Remote control station

The operator controls the vehicle from a local station away from the robot over a wireless network. The control station consists of a desktop computer running Windows XP. A steering wheel and two foot switches are connected to the desktop. The steering wheel sets the steering of the remote mobile robot and the footpadel is used to set the velocity V of the robot. A push button in the steering wheel is used to reverse the direction of motion of the robot.

The screen of the desktop displays video streaming from the mobile robot's on board camera. A graphical user interface (GUI) shown in Figure 4.20 displays the robot's parameters such as current steer angle, velocity of each rear wheels and the position of the z-axis. Buttons on the GUI operates the z-axis, head lamps, etc.

4.4 Summary

In this chapter, the control architecture of the mobile manipulator has been presented. The algorithm used to move the robot was discussed in detail. The odometry used for pose estimation of the robot was also presented with the experimental results.

Chapter 5

Simulation of Tele-operation

In this chapter, simulation of a tele-operated mobile robot is presented. In tele-operation, the human operator observes a remote scene through camera(s), and manipulates the local steering wheel and accelerator pedal, as illustrated in Figure 5.1 . The command is transmitted to the mobile robot over wireless network. The operator's response is based on the latest feedback images from the cameras. In general, there is a time lag when communication takes place over wireless network. The time lag deteriorates the human performance as discussed in [76] and references therein. This chapter simulates the tele-operation both without and with time delay in transmission network.

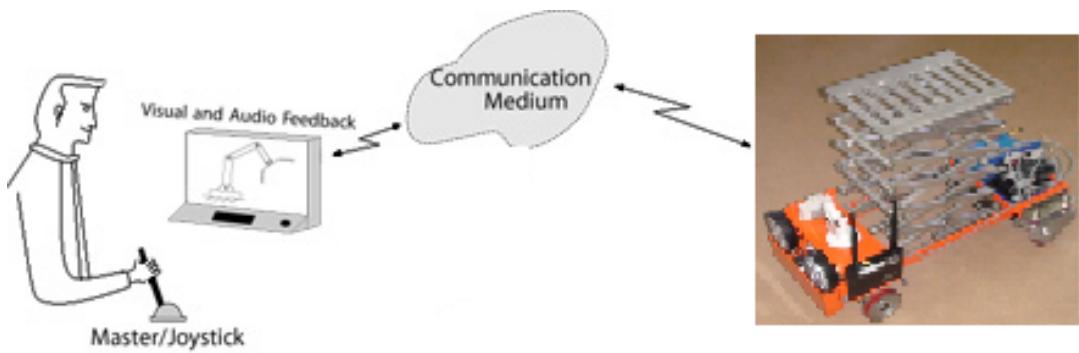


Figure 5.1: Teleoperation architecture

5.1 Modeling of Mobile Robot

The standard kinematic model, as described in [13], of the mobile robot was used for the simulation. The use of kinematic model is justified as the vehicle is expected to move at relatively slow speed and model is simple. Inputs to the model are left and right rear wheel velocities. The front wheels are steered to satisfy the Ackerman condition, as presented in Chapter 2 and are assumed to attain the desired angle instantaneously. Therefore, the robot can be treated as differential drive robot. The kinematic model of the platform is presented below:

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} r_w/2 & r_w/2 \\ 1/b & -1/b \end{pmatrix} \begin{pmatrix} \dot{\phi}_L \\ \dot{\phi}_R \end{pmatrix} \quad (5.1)$$

where , b is the distance between the rear wheels, r_w wheel radius. $\dot{\phi}_R$ and $\dot{\phi}_L$ are the rotational velocities of left and right wheels. The operator station sends the command u_1 and u_2 over the wireless network. In general, it will be delayed by time δ . These commands are interpreted by the robot controller as the left and right wheel velocities. Therefore, by taking the time delay into consideration one can write

$$\begin{pmatrix} \dot{\phi}_R(t) \\ \dot{\phi}_L(t) \end{pmatrix} = \begin{pmatrix} u_r(t - \delta) \\ u_l(t - \delta) \end{pmatrix} \quad (5.2)$$

The control inputs to the mobile robot u_r and u_l are generated by the operator based on the visual data available to the person. Hence, a model of the human operator needs to be used for the simulation of the complete loop.

5.2 Model of Human Operator

In order to simulate the tele-operation loop, one needs a mathematical model of a human operator. The mathematical model of the operator's action is approached assuming a car driving metaphor. The video feedback, which the operator receives

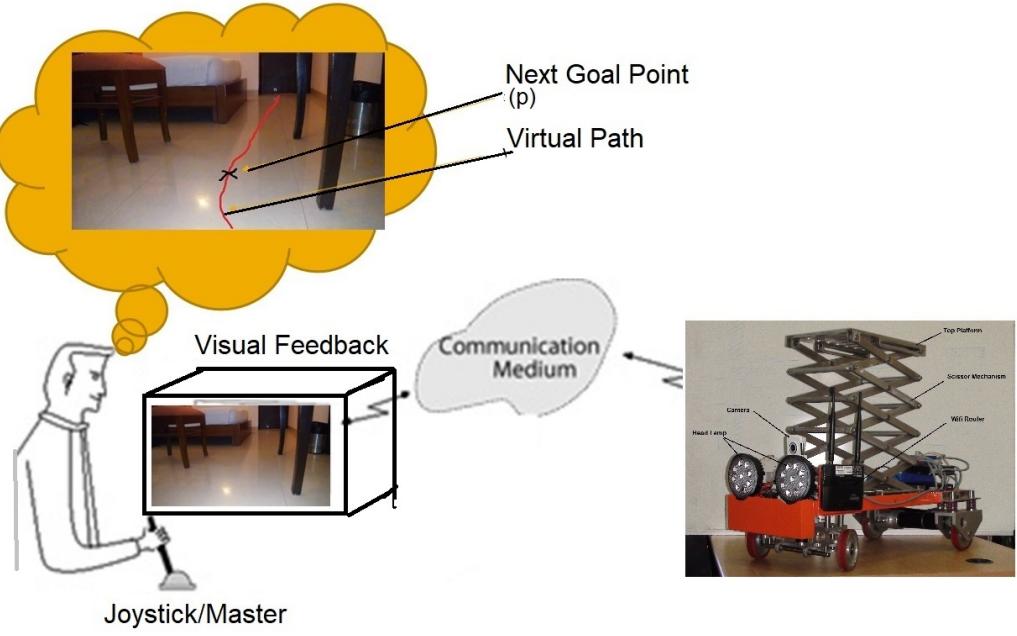


Figure 5.2: Assumed driving strategy

of the remote environment, give him the idea of the vehicle's position and the tentative next goal point (p) based on a lookahead distance (l). He then constructs a virtual path mentally and tries to manoeuvre or steers the robot to follow that path as shown in Figure 5.2. As he moves forward the goal point keeps changing until he reaches the desired location. This methodology of path tracing is known as pure pursuit [37].

The mathematical model for the pure pursuit method of path following can be derived as given in Figure 5.3. As shown in Figure 5.3, the origin of the coordinate system is at point o , the middle of rear axis of the robot. As the differential drive robot can move only about a circle with center lying on the line along its rear axis. An arc OP of radius r , is drawn with center O_1 and passing through o and p . Where p is a point on the path to be traced by the robot. The linear distance between the points o and p is called the *look ahead distance* l . This distance in the case of a tele-operated robot will depend on the field of view of camera at the remote location and the obstacles present in the remote environment.

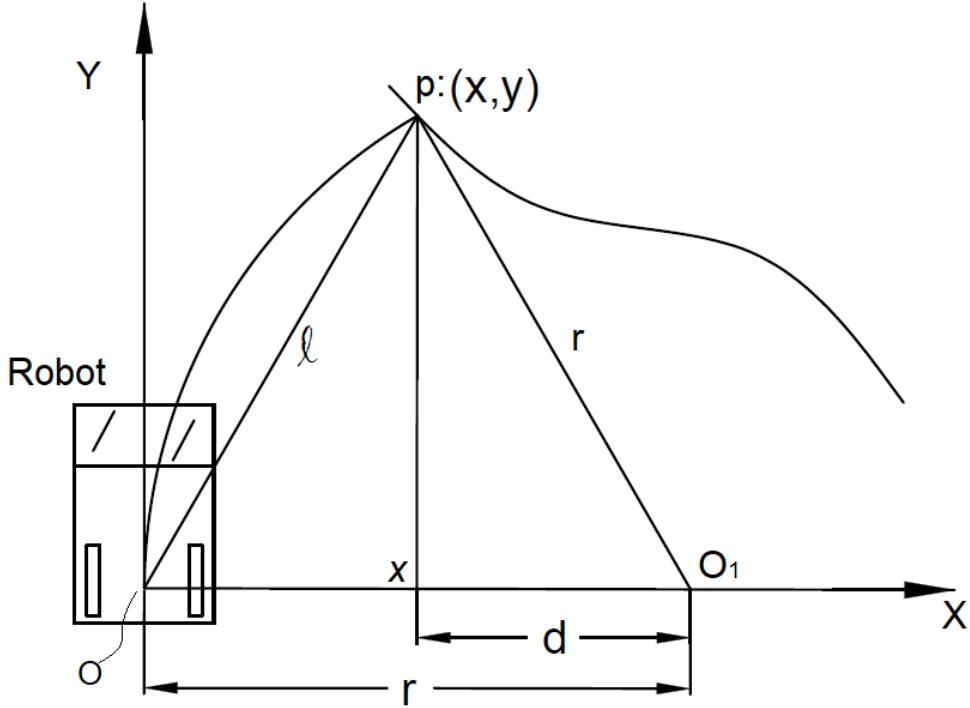


Figure 5.3: Geometry of Pure Pursuit

If (x, y) is the coordinate of point p in $X - Y$ coordinate system, then

$$x^2 + y^2 = l^2, \quad d = r - x \quad (5.3)$$

Similarly, from triangle p, x, o_1 we get

$$d^2 + y^2 = r^2 \Rightarrow (r - x)^2 + y^2 = r^2 \Rightarrow x^2 + y^2 - 2rx = 0 \quad (5.4)$$

Replacing x^2 and y^2 in Equation 6.4 with Equation 6.3, we get

$$2rx = l^2 \Rightarrow r = \frac{l^2}{2x} \quad (5.5)$$

Once the radius r , the desired linear velocity of the robot v are known, the angular velocity of the vehicle is $\dot{\theta} = -v/r$. The rear wheel $\dot{\phi}_L(t)$ and $\dot{\phi}_R(t)$ can be calculated from Equation 6.1. Where $\dot{y} = v$ and $\dot{x} = 0$. To match the orientation of the vehicle with that in Figure 5.3, $\theta = 90^\circ$. We then get

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ r_w/2 & r_w/2 \\ b/2 & -b/2 \end{pmatrix} \begin{pmatrix} \dot{\phi}_L \\ \dot{\phi}_R \end{pmatrix} \Rightarrow \begin{pmatrix} \dot{\phi}_L \\ \dot{\phi}_R \end{pmatrix} = \begin{pmatrix} 1/r_w & 1/r_w \\ 1/b & -1/b \end{pmatrix} \begin{pmatrix} v \\ \dot{\theta} \end{pmatrix} \quad (5.6)$$

The operator station sends v and $\dot{\theta}$ as the command over the communication network to the robot. The Figure 5.4 shows the simulation strategy

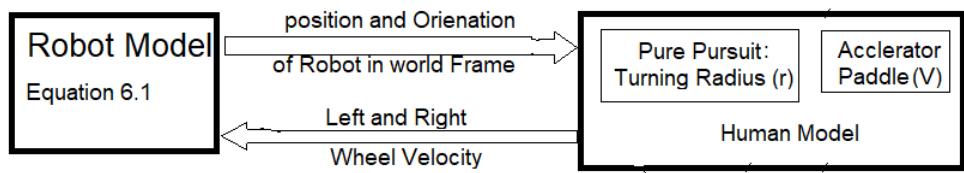


Figure 5.4: Simulation scheme

5.3 Simulation and Results

The teleoperation loop consists of the operator model described in Section 6.2 at one end of the communication link and the mobile robot model described in Section 6.1 on the other end . As shown in Figure 5.5, there will a delay in both directions of communication. In real system, the video image is streamed by the robot. Due to large quantity of data and limited bandwidth the delay $T_1 \gg T_2$. The amount of delay h_1 was experimentally measured and it was found around 0.5sec, as described in Appendix A. The command sent by the operator is v and $\dot{\theta}$ which is few bites only. Therefore in simulation $h_2 = 0$ is assumed.

5.3.1 Simulation algorithm

The algorithm for simulation is explained in the following steps.

1. Convert the path from global coordinate system (CS) to Robots Local Coordinate System

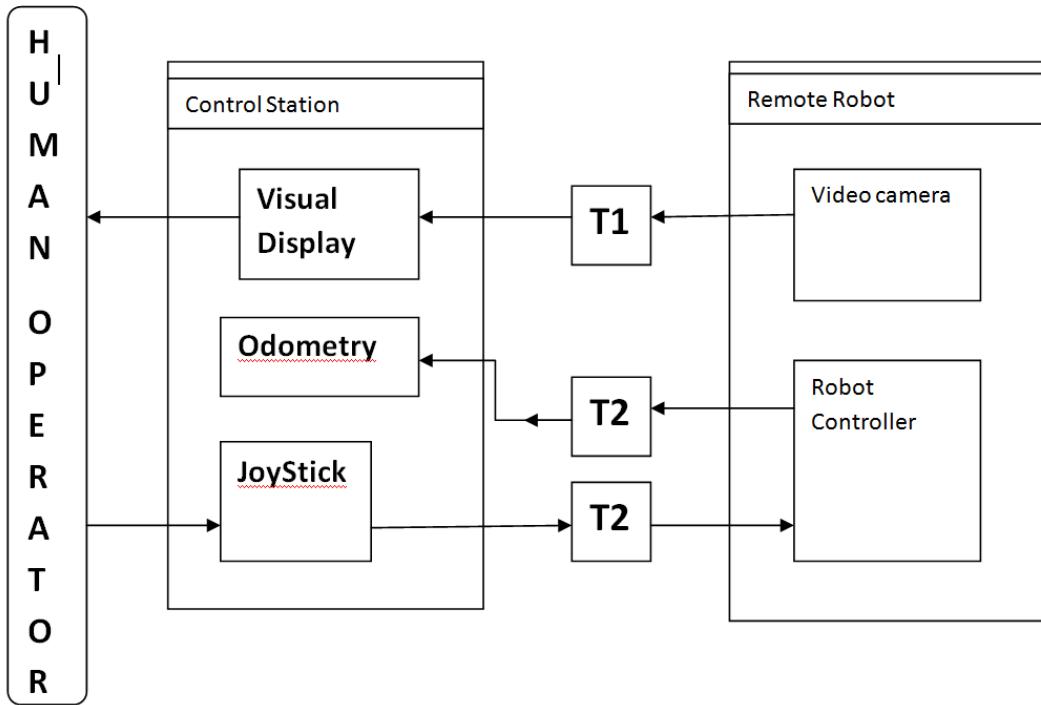


Figure 5.5: Block diagram for teleoperation

2. With a given look ahead distance (l) search for a point on the path
 - If point is found goto Step 3
 - If not found increase the look ahead distance
3. Determine the turning radius (r) using Equation 6.5
4. Calculate the command to the robot based on Equation 6.6. Note that these commands are based on the old visual data the operator saw.
5. Solve delayed differential equation.

Simulation was carried using Matlab. Delay differential equation solver "*dde24*" was used to solve Equations 6.1 and 6.2 with delayed inputs u_r and u_l . The desired path was a circle of radius 5m centred at origin of the global coordinate system. The human action was modelled with look ahead distance of 0.5m and linear velocity of 0.5m/s. The initial position of the robot was (4.5,0,0).

The performance of the system with zero delay, i.e. $T_1 = T_2 = 0$ is shown in Figure 5.6. Figures 5.7 and 5.8 show the robot's motion under delay of 0.5 and

0.8 sec. It is seen that oscillation becomes visible at 0.5sec delay, and with the delay of 0.8 sec the system was on the verge of instability.

It was also observed that the with large vehicle velocity, v and large look ahead distance, l the instability commences with smaller time delay, δ in Equation 6.2.

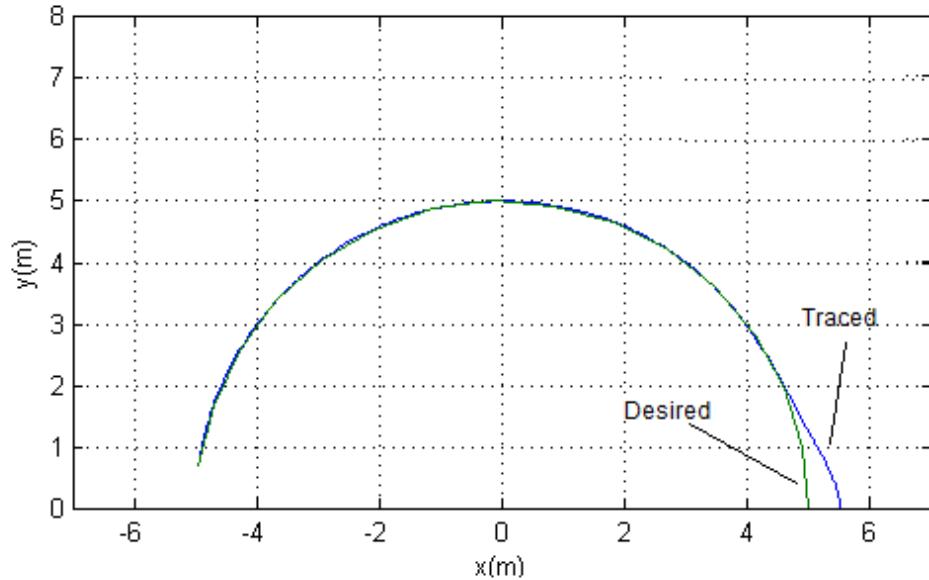


Figure 5.6: Simulation with no time delay in either direction

In the next section we propose a predictive model based feedback control which is used to stabilize robot motion under time delay teleoperation.

5.4 Summary

In this chapter, simulation study of the developed teleoperated mobile robot is presented. A mathematical model of the action of human operator while driving the robot based on visual feedback was presented. Simulation results show that the behaviour of the system deteriorates with increase in delay in communication between the local and remote station. With large time delay the system becomes unstable.

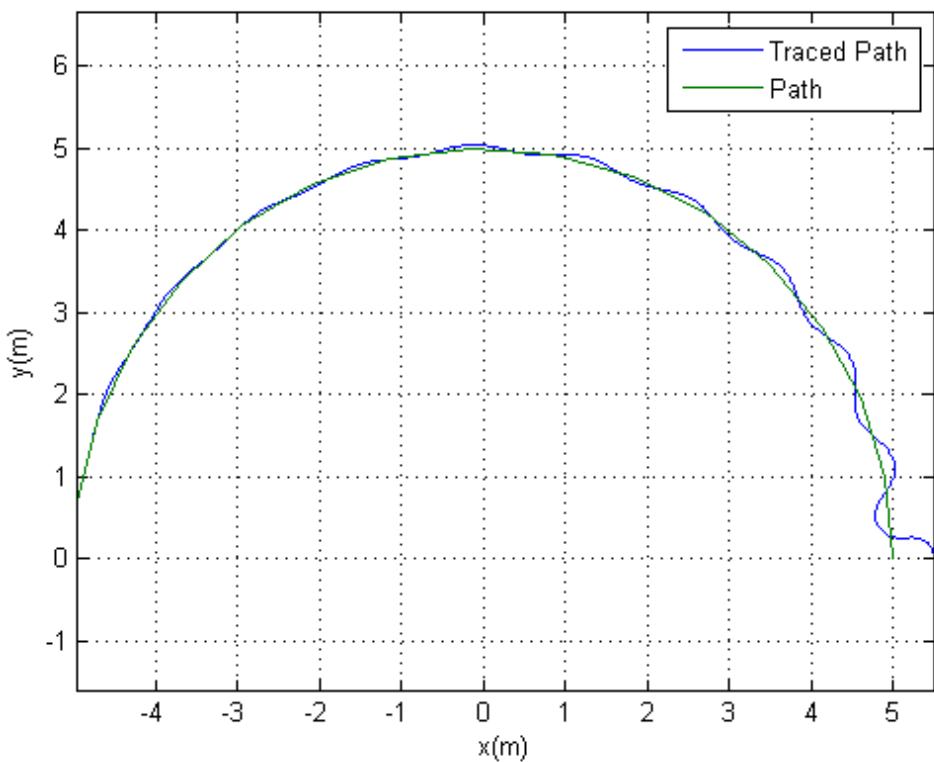


Figure 5.7: Simulation with time delay $h_1 = .5\text{sec}$ and $h_2 = 0$

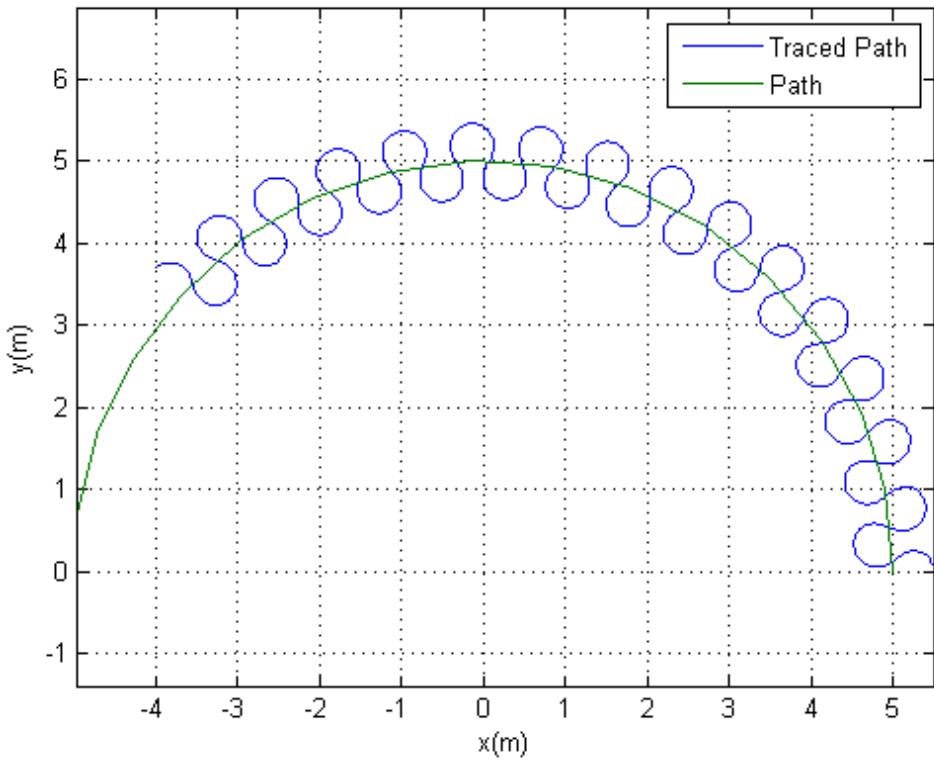


Figure 5.8: Simulation with time delay $h_1 = .8\text{sec}$ and $h_2 = 0$

Chapter 6

Predictive Display

In the last chapter, it was shown using simulation that the time delay between the remote and local stations leads the system towards instability. In this chapter, first a predictive model controller for time delayed teleoperation is presented. Associated issues of time delay on human performance are then highlighted. To alleviate the problem of time delay in visual feedback, predictive display using a RGB-Depth sensor is implemented.

6.1 Time Delay Compensation

Simulation of a time delayed system with pure pursuit model for human controller was presented in Chapter 5. Stability aspects with input delay to the human model was pointed out. In paper by Ollero [5], stability of pure pursuit with input delay is presented and for completeness it is briefly discussed in Appendix B. In view of the above theoretical analysis and the simulation results presented in Chapter 5, it is required to design a stabilizing controller to take care of large delays, e.g., 0.8 sec. One such design based on model predictive control is presented next.

One of the earliest predictor based controller for linear system with time delay was proposed by Smith [4] called the Smith Predictor or Smith Controller. The schematic diagram of the smith controller is shown in Figure 6.1. There are two loops, the inner and the outer loop, where $G(z)$ is the plant, $C(z)$ is the stabilizing

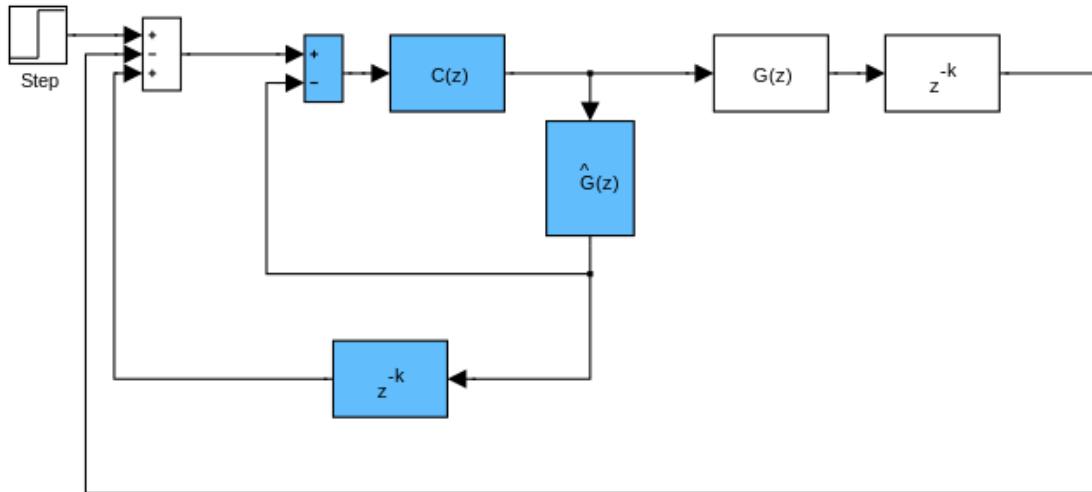


Figure 6.1: Smith predictor [4]

controller for the plant without delay, $\hat{G}(z)$ is the model of the plant. As can be seen from Figure 6.1, during the period (k time unit) when the feedback (output) is not available the model of the plant is used to predict the actual plant behaviour and generate the control signal accordingly.

In the presented case, the plant is the mobile robot developed in this PhD research. The block diagram shown in Figure 6.2 depicts the architecture of time delayed system which is applicable to the current mobile robot and the local station. The tele-operation over wireless network for our system resulted in a delay of $T_1 = 500 \text{ ms}$ with update frequency of 2Hz. The delay was caused due to large amount of data being transmitted as video feedback from the robot's onboard camera. Figure 6.3 shows the measured delay in the video link. The frequency of odometric and control data exchange between the two stations was at the rate of 20Hz, i.e., a delay $T_2 = 50 \text{ ms}$ only. This loop runs independent of the video feedback link. It may be noted that $T_2 \ll T_1$.

6.1.1 Proposed controller

The proposed control strategy to mitigate the effect of time delay is to predict the current position from the last known position, based on the delayed video image; using the dynamic model of the mobile robot. As shown in Figure 6.4, let us

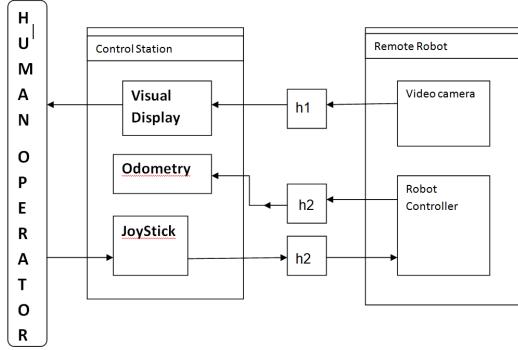


Figure 6.2: Block diagram of time delayed system

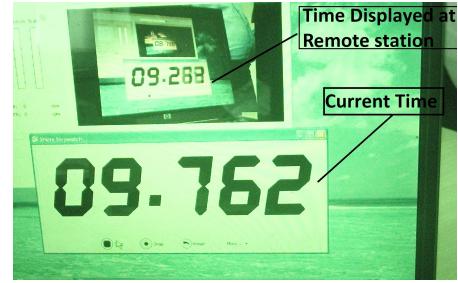


Figure 6.3: Delay measurement

assume that we are at time $t + \delta$, but the latest pose data of the robot available is that of at time t . The present pose at time $t + \delta$ was predicted using the dynamic model of the robot derived in Chapter 3, Equation 3.9, and presented here again for convenience.

$$I(\theta)\ddot{\theta} = C(\theta, \dot{\theta})\dot{\theta} + \tau \quad (6.1)$$

It may be noted that the dynamic model uses torque τ as its input, whereas the local station which uses pure pursuit for simulation of human action generates linear and angular velocities of the robot, v and ω as shown in Figure 6.5. This is taken care by first converting $v \equiv \dot{o}_3$ and $\omega \equiv \omega_3$ to rear wheel velocities using Equations 3.11 and 3.12. The rear wheel velocities are then used in PI controller of the rear wheel motors to generate the τ for Equation 6.1.

This predicted position is given to the pure pursuit algorithm to generate required control outputs for the remote robot. In the simulation, it was assumed that this control inputs reaches the remote robot instantaneously, as $T_1 \gg T_2$. As discussed in the Kinematic model of Chapter 5. Equation 5.1 was used to simulate the remote robot.

6.1.2 Simulation algorithm and results

The simulation scheme is shown in block diagram of Figure 6.5. The algorithm is explained below.

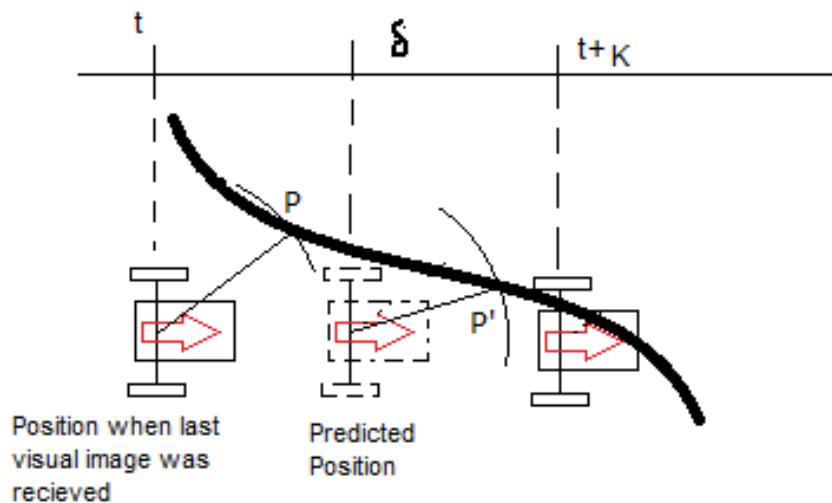


Figure 6.4: Smith predictor applied to the developed mobile robot

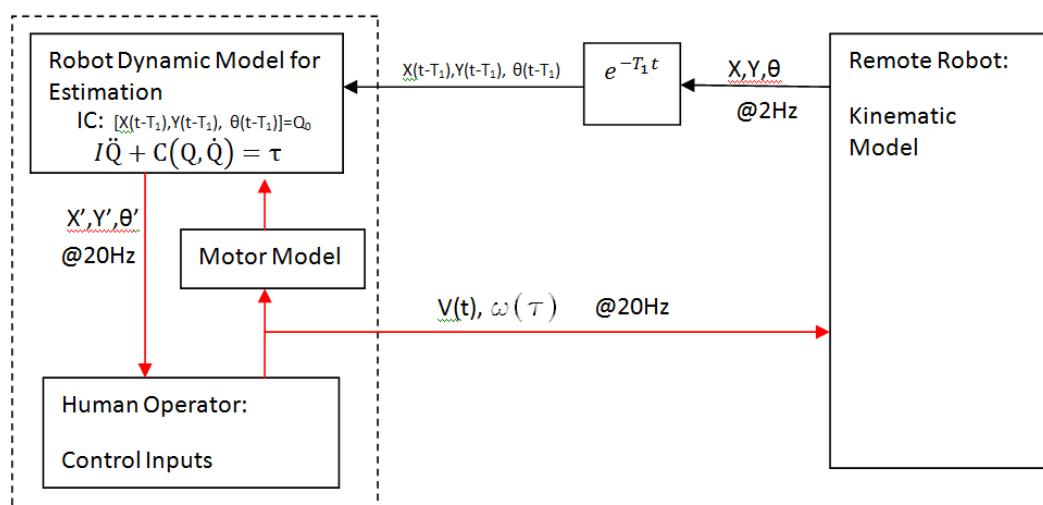


Figure 6.5: Simulation block Diagram

- 1: Convert the path from global coordinate system (CS) to Robot's Local Coordinate System based on the current pose (x, y, θ) of the robot.
- 2: With a given look ahead distance (l) search for a goal point on the path.
- 3: Determine the turning radius (r) using Equation 6.5
- 4: Calculate ω using turning radius (r) and given Linear Velocity v
- 5: Command Robot ω and v
- 6: **if** new pose of the robot is available from remote station **then**
- 7: Update robot pose (x, y, θ)
- 8: **else**
- 9: Calculate the predicted pose of the robot based on command given in Step 4, and using dynamic model of the robot given by Equation 6.1.
- 10: Update robot's pose (x, y, θ)
- 11: **end if**
- 12: **Goto** Step 1

Simulation was carried using Matlab. Differential equation solver "*Ode24*" was used to solve Equations 5.1 and 6.1. The desired path was a circle of radius 5m centred at origin of the global coordinate system. The human action was modelled with look ahead distance l of 0.5m and linear velocity v of 0.5 m/s. The initial position of the robot was (4.5,0.0). The robot's motion under feedback delay of 0.5 sec and 0.8 sec, i.e., $T_1 = 0.5$ sec and $T_1 = 0.8$ sec, are shown in Figures 6.6 and 6.7 respectively. It is seen that oscillation are no more visible at delay of 0.5 sec and 0.8 sec. The system shows no instability.

The above results show that model-based prediction of the robot's pose helps in removing the instability of the system. In actual teleoperation, the operator's control actions are based on the visual display available to him. The delay in visual feedback results in inefficient operator's performance as the person tends to adopt a wait and watch policy to see the effect of his control action. In the next section, an implementation and adaptation of above discussed *model based prediction* for visual display feedback is presented.

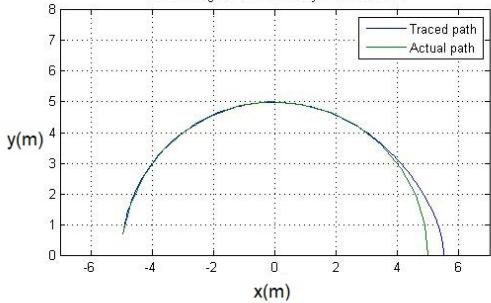


Figure 6.6: Time delay $T_1 = .5 \text{ sec}$
and $T_2 = 0$

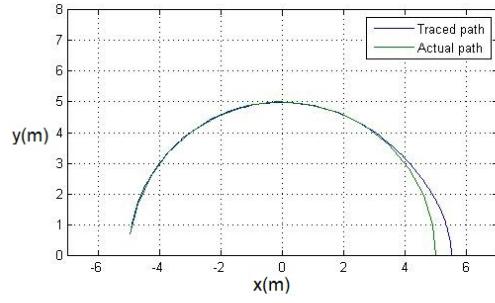


Figure 6.7: Time delay $T_1 = .8 \text{ sec}$
and $T_2 = 0$

6.2 Delay Compensation using Predictive Display

As indicated in Chapter ?? on Literature survey, the delay of visual feedback leads to inefficient and unstable performance of a teleoperated system. There are two major methods to overcome time delay induced problems in teleoperation. One is to use *supervisory control* and the other is *predictive display*. The second methodology, i.e., the Predictive Display, was adapted here. This is because it is more intuitive to human operators, and the onboard controller required on the mobile robot is very much simplified.

Predictive display has been defined as using the computer for extrapolating the display forward in time [114]. In this, a local model of the remote scene is used to predict and render the remote scene in response to operator's command. It replaces the delayed video feedback with extrapolated synthesised image of the remote environment. This enables the operator to perform the task normally. Predictive display has been implemented in the past using different sensors such as monocular camera fixed on the wall, camera mounted on the robot arm, fusion of Lidar scanner and RGB camera, etc. The proposed approach here is to use a low-cost Kinect Sensor from Microsoft Inc. to generate the 3-D model of the remote environment and use kinematic model of the mobile robot to predict the

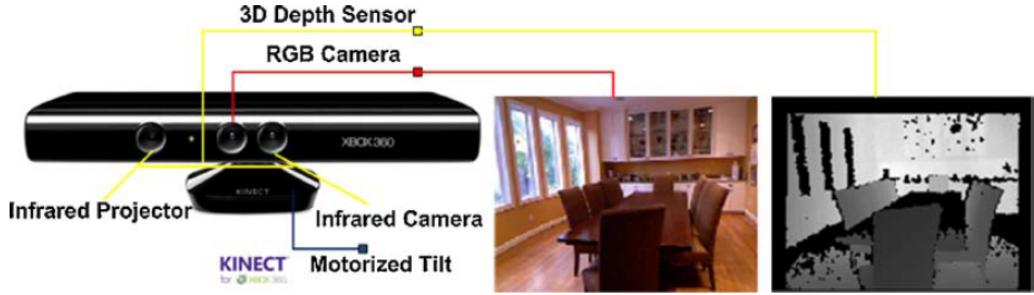


Figure 6.8: Kinect Sensor from Microsoft

motion of the actual robot on the 3-D model of the environment to generate a delay-free estimated image of the remote scene operator.

6.2.1 The Kinect sensor

Kinect sensor from microsoft is shown in Figure 6.8. The hardware contains a normal RGB camera, a depth sensor and a four-microphone array, which are able to provide RGB images, depth signals, and audio signals simultaneously. The depth sensor comprises of an Infra Red (IR) projector and the IR camera. The IR projector casts an IR speckle dot of known pattern into the 3-D scene while the IR camera captures the reflected IR speckles. To determine the depth, triangulation method is used. Kinect is therefore an instance of a structured light depth sensor. More details concerning the structured light 3-D imaging technology can be found in [115].

The *RGB camera* delivers three basic colour components of the video. The camera operates at 30 Hz, and can offer images at 640×480 pixels with 8 bits per channel. The *3-D Depth Sensor* creates a depth map, which provides the distance information between the camera and an object. The sensor has a practical range limit of 0.8m-3.5m distance, and outputs video at a frame rate of 30 frames/sec with the resolution of 640×480 pixels. The angular field of view is 57° horizontally and 43° vertically. Number of different open source software libraries are available which includes OpenNI [116], Microsoft Kinect SDK [117] and OpenKinect [118] to access data from the Kinect sensor. OpenNi was used in this research, as it is

compatible with RTabMap, an Open Source Software used for reconstruction of the 3-D point cloud model of the remote scene.

6.2.2 Onboard data processing and transmission

In teleoperation, the Kinect sensor mounted on the mobile robot was connected with the onboard Raspberry Pi single board computer with limited processing power. The onboard controller uses OpenNi library to interface with the kinect hardware. The Kinect sensor has two separate cameras. The transformation between the camera centres are known and provided by the manufacturer. It therefore generates two streams, the colour and depth data. It is possible to send the two raw data streams over the network without any processing at the robot's side. This puts less stress on the onboard computer but requires high communication bandwidth. The other option is to stream *registered depth* data over the network. This requires less bandwidth as each pixel has both the colour and depth value associated with it when it is transmitted. Depth Registration requires minimal computing power. Known transformation between the two cameras was used to align the depth pixel and the colour pixel so that they correspond to the same point of the 3-D scene. This is referred to as *depth registration* or *image registration* in the literature. The word "depth" in the *depth registration* indicates that the final image data is with respect to the depth camera's frame. The second method was adapted here. The robot transmits registered depth data, i.e., the RGB and the Depth (RGB-D) values associated with each pixel to the control station over network.

6.2.3 3-D reconstruction

The registered depth data received at the operator-station was processed using the RTabMap library [119] and 3-D Point Cloud data was generated. The details of the working of RTabMap can be found in [120] and [121]. Point cloud map is a set of points in 3-D space derived using the camera model and the depth

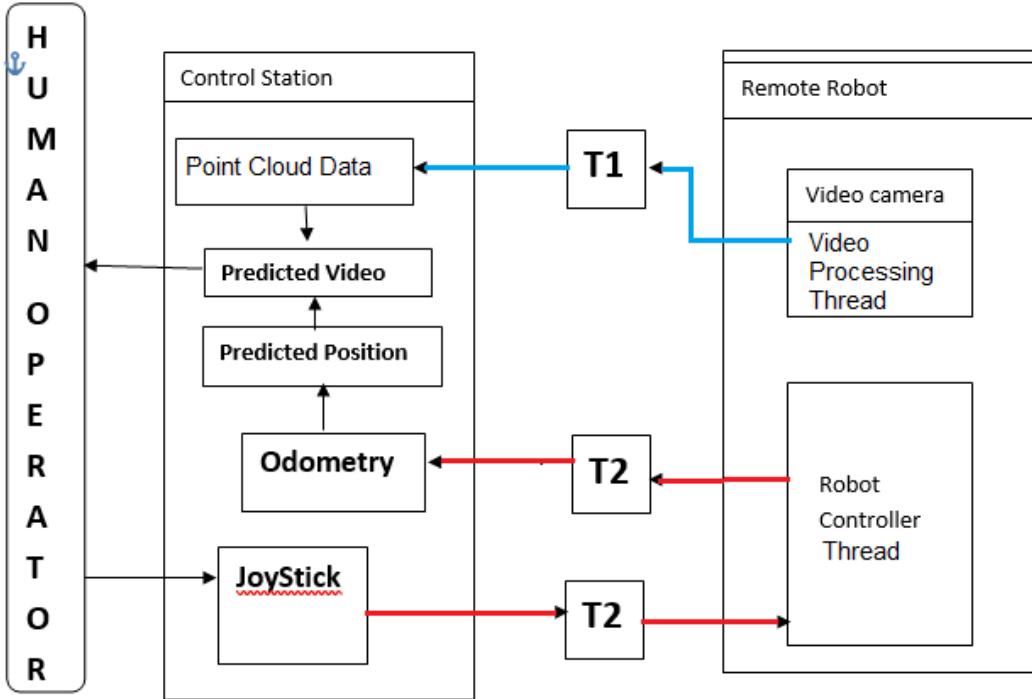


Figure 6.9: Predictive display architecture

data associated with each pixel of the depth registered data sent from the mobile robot. Each new frame that arrives is added to the current data set after proper transformation. The transformation between the two frames of data was created by matching the common feature in the two frames. The RTabMap library thus outputs a set of 3-D point alongwith their RGB color. The coordinates of the points are always with respect to camera coordinate frame. It may be noted that the registered depth image that has arrived was delayed by T_1 unit of time. So the local station always has a 3-D map w.r.t the robot at $(t - T_1)$ sec.

6.2.4 Extrapolation of remote scene

The visual data present in the current frame is the view that the robot has seen T_1 seconds earlier. Let this data be denoted as $P(t - T_1) = Set\{P_k\}$. The point P_i is associated with the coordinates $P_{x_k}, P_{y_k}, P_{z_k}$ and color data c_k . In order to predict the current scene that the robot might be seeing one needs to estimate the current position of the robot. The dynamic model developed in chapter 3 is used

to estimate the current position of the robot. To accomplish this the architecture of the teleoperation shown in Figure 6.9 was developed and used to compensate for the time delay. It has been stated earlier that the delay T_2 in exchange of command data and the wheel velocity data from the robot to the operator station was 20ms. It may thus be assumed that using the odometric data in Equation 4.13, 4.14 and 4.15, the current position of the robot was always known to the operator-station. This was performed by Odometric Block of Figure 6.9 by using Equations 4.13, 4.14 and 4.15 from time $t - T_2$ to t . Where t denotes the current time and with initial condition

$$x(i=0) = 0, \quad y(i=0) = 0, \quad \beta(i=0) = 0$$

and

$$i = 0 \text{ to } n; \quad n = T_1/T_2$$

Let the predicted position of the robot at time t be given by

$$[x(i), \quad y(i), \quad \beta(i)] \rightarrow [v(t), y_v(t), \beta_v(t)]^T$$

This was basically the amount by which the robot has moved after the last video frame had arrived. Next, the transformation matrix, T_r^c between the mobile robot's coordinate frame at point O_r in Figure 4.10 and the camera frame was used to calculate the change in pose of the camera. Since the points P_i are in the camera frame their current coordinates $P'x_i, P'y_i, P'z_i$ was arrived at by using

$$\begin{pmatrix} P'x_i \\ P'y_i \\ P'z_i \end{pmatrix} = T_r^c \begin{pmatrix} Px_i \\ Py_i \\ Pz_i \end{pmatrix} \quad (6.2)$$

This new location of the points were then projected onto the screen of the local operator, thus, giving him the estimated current view of the remote scene. The



Figure 6.10: PCD at time T



Figure 6.11: Predicted Scene

view available at the operator station at time t is shown in Figure 6.10 which is T_1 sec old. Based on this old data, the predicted scene is shown in Figure 6.11. The operator sees Figure 6.11 instead of Figure 6.10.

With the predictive display model the operator was able to move the robot without "*wait and see*" stratagey adopted earlier. The motions were smooth and speed of operation improved.

6.3 Summary

In this chapter, the model-based predictive control for control of mobile robot over time delayed network is presented. The stability of the control was verified using simulation. This model was then adopted for teleoperation using visual feedback. Kinect sensor was used to generate the 3-D model of the remote environment in real time. The estimated position of the mobile robot using the odometry was used to project a synthesised image of the 3-D model of the remote environment.

Chapter 7

Conclusions

7.1 Thesis Summary

This thesis describes the design, development and experimentation of a customized mobile manipulator developed at Bhabha Atomic Research Centre (BARC), Mumbai, India for mapping radiation in different areas of Cyclotron building where human access is restricted during Cyclotron operation. The customized mobile manipulator is a four wheeled mobile robot with a vertical lifting platform on which the radiation detector is mounted. The mobile manipulator or platform has two rear wheels individually coupled to two motors and the front two wheels which are mutually connected using a motorized Davis steering mechanism.

The mechanical design was obtained in Chapter 3, and through optimization of steering mechanism, minimum turning radius of 415 mm was achieved while maintaining the over all size constraint of the mobile platform. The optimal location of the CG for the mobile manipulator was decided so as to provide maximum traction as well as stability over 30° ramp. Among different lifting platforms for the vertical motion to the sensor for scanning, a simple scissor based lifting platform was chosen due to several advantages it offers.

The kinematic and dynamic modeling of the customized mobile manipulator was presented in Chapter 4, where the method based on the Natural Orthogonal

Complement was used to derive the dynamic equations of motion of the wheeled mobile robot under study. This analysis highlighted the effect of caster offsets on the maneuverability of a mobile robot. The need of having a positive drive for certain castor offsets were also brought in the analysis.

Control structure for intuitive teleoperation of the mobile robot was presented in Chapter 5. This chapter proposed a user interface at the operator's side, along with the hardware and software required for the teleoperation. The control algorithm running on the robot's on-board computer was also presented. The contribution there was that it synchronizes all three motors; two in velocity control and one in position control mode (steering). In spite of difference in response time, Ackerman relation for wheel rolling was always satisfied.

The developed mobile robot was controlled by an operator who was physically separated from the mobile manipulator. A dedicated wireless network and the video signal of the remote environment from the robot's on-board camera was available. The limited bandwidth resulted in the delay of the video. Chapter 6 dealt with the simulation of a tele-operation for the mobile robot under time delay. A mathematical model of the human operator based on pure pursuit algorithm was presented. The simulation predicted instability with increase in time delay and linear velocity of the mobile robot. These studies qualitatively matched with the operator behaviour under similar conditions.

Note that the time delay in the video feedback affects the efficiency of the operator. In order to overcome this, predictive display was proposed. In Chapter 7, a new methodology was presented for the predictive display based on the 3-D model generated using the RGB-Depth data provided by a Kinect Sensor mounted on the mobile robot and the predicted position of the robot based on dynamic model of the robot presented in Chapter 4. This novel method did not require the 3-D model of the remote environment known before hand as reported in the literature earlier. This flexibility allows the tele-operation of the mobile robot in an unknown environment.

In summary, a mobile manipulator was successfully conceived, designed, assembled and controlled remotely for the purpose for which it was intended, i.e, to map the radiation of a given area.

7.2 Current Status

A prototype of mobile manipulator and its tele-operation system were manufactured and assembled at BARC, Mumbai, India based on the analysis and algorithms presented in this thesis. The robot was lab tested and the video of the same can be accessed on at the following link <https://youtu.be/B5OngfLQjo4> or using the QR code shown in Figure 7.1. The system will shortly be deployed at Variable Energy Cyclotron Centre (VECC) at Kolkata, India for the same purpose.



Figure 7.1: QR code

7.3 Future Scope

The testing and evaluation of teleoperation system during actual trial opened up further scope of improvement. Few of these are listed below:

- (i) Detection of slip and estimation of slip parameters using inertial sensors such as gyroscope and accelerometer mounted on the mobile robot in combination with wheel odometer. *Number of algorithm are available in the literature which uses sensor fusion to arrive at better estimation of robots pose. This estimated pose can then be used to calculate*

the estimated real time slip based on the actual wheels encoder readings. Based on the estimated slip and using the slip model presented above, actual slip parameters can then be estimated.

- (ii) Pose estimation using encoder mounted on the front passive wheels for slip detection. *The two front wheel are passive wheel, which in general will see less slip than the actuated wheels. On flat terrain and with wheels with low inertia they can give a better estimate of the robots position. Since the contact point of the two wheel keep changing with respect to the robot body an algorithm based on kinematics of of the steering linkage can be worked out to give the position of the robot body based on the odometry of theses wheels. It needs to be seen if this in combination with the inertial sensors can produce better estimate.*
- (iii) Semi-automated navigation using intermediate goal point. *In this mode the operator will give goal point/s on the screen. This 2D pint on the screen would then be mapped to the 3-D point in the actual environment. A planing algorithm will then generate a path between the current position of the robot and goal point. This path must to satisfy the minimum turning radius constrain of the mobile robot. This path will then be feedback to the robots on board controller. A local tracking controller will then be activated to track the path.*
- (iv) Texture mapping of predicted visual image instead of 3-D point cloud model currently used. *Since the color of the environment is also available along with its distance. An algorithm needs be developed or existing algorithm implemented which will use the 3-D point cloud which contains the distance information and the colour to produce surface with texture. This will give a more natural feel of*

the environment to the operator.

- (v) Some kind of force feedback to the operator for estimation of obstacle location in the remote environment. *This can be achieved using ultrasonic sensors mounted on the periphery of the robot. Once obstacles comes too close to the robot, the distance of the obstacle from the robot will be transmitted to the control station. The software at the operator side then will give some vibratory input to the steering wheel. The amplitude of vibration should be function of the distance of the robot from the obstacle, the heading direction and the speed of approach. A proper function to give intuitive feel to the operator has to be worked out.*

Bibliography

- [1] K. Ohno, S. Kawatsuma, T. Okada, E. Takeuchi, K. Higashi, and S. Tadokoro, “Robotic control vehicle for measuring radiation in fukushima daiichi nuclear power plant,” in *Safety, Security, and Rescue Robotics (SSRR), 2011 IEEE International Symposium on*, pp. 38–43, IEEE, 2011.
- [2] B. L. Luk, K. Liu, A. A. Collie, D. S. Cooke, and S. Chen, “Tele-operated climbing and mobile service robots for remote inspection and maintenance in nuclear industry,” *Industrial Robot: An International Journal*, vol. 33, no. 3, pp. 194–204, 2006.
- [3] H. B. Pacejka and E. Bakker, “The magic formula tyre model,” *Vehicle system dynamics*, vol. 21, no. S1, pp. 1–18, 1992.
- [4] O. J. Smith, “A controller to overcome dead time,” *ISA J.*, vol. 6, pp. 28–33, 1959.
- [5] A. Ollero and G. Heredia, “Stability analysis of mobile robot path tracking,” in *Intelligent Robots and Systems 95. Human Robot Interaction and Cooperative Robots’, Proceedings. 1995 IEEE/RSJ International Conference on*, vol. 3, pp. 461–466, IEEE, 1995.
- [6] L. Ray, A. Price, A. Streeter, D. Denton, and J. H. Lever, “The design of a mobile robot for instrument network deployment in antarctica,” in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pp. 2111–2116, IEEE, 2005.
- [7] L. Briones, P. Bustamante, and M. A. Serna, “Wall-climbing robot for inspection in nuclear power plants,” in *Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on*, pp. 1409–1414, IEEE, 1994.
- [8] S. Galt, B. Luk, D. Cooke, and A. Collie, “A tele-operated semi-intelligent climbing robot for nuclear applications,” in *Mechatronics and Machine Vision in Practice, 1997. Proceedings., Fourth Annual Conference on*, pp. 118–123, IEEE, 1997.
- [9] C.-S. Cho, J.-D. Kim, S.-G. Lee, S. K. Lee, S.-C. Han, and B.-S. Kim, “A study on automated mobile painting robot with permanent magnet wheels for outer plate of ship,” in *Robotics (ISR), 2013 44th International Symposium on*, pp. 1–4, IEEE, 2013.

- [10] J. Borenstein, “Control and kinematic design of multi-degree-of freedom mobile robots with compliant linkage,” *IEEE transactions on robotics and automation*, vol. 11, no. 1, pp. 21–35, 1995.
- [11] P. F. Muir and C. P. Neuman, “Kinematic modeling of wheeled mobile robots,” *Journal of robotic systems*, vol. 4, no. 2, pp. 281–340, 1987.
- [12] J. C. Alexander and J. H. Maddocks, “On the kinematics of wheeled mobile robots,” *The International Journal of Robotics Research*, vol. 8, no. 5, pp. 15–27, 1989.
- [13] G. Campion, G. Bastin, and B. Dandrea-Novel, “Structural properties and classification of kinematic and dynamic models of wheeled mobile robots,” *IEEE transactions on robotics and automation*, vol. 12, no. 1, pp. 47–62, 1996.
- [14] B.-J. Yi and W. K. Kim, “The kinematics for redundantly actuated omnidirectional mobile robots,” *Journal of Robotic Systems*, vol. 19, no. 6, pp. 255–267, 2002.
- [15] N. A. M. Hootsmans *et al.*, *The motion control manipulators on mobile vehicles*. PhD thesis, Massachusetts Institute of Technology, 1992.
- [16] J. H. Chung, S. A. Velinsky, and R. A. Hess, “Interaction control of a redundant mobile manipulator,” *The International Journal of Robotics Research*, vol. 17, no. 12, pp. 1302–1309, 1998.
- [17] J. Y. Luh, M. W. Walker, and R. P. Paul, “On-line computational scheme for mechanical manipulators,” *Journal of Dynamic Systems, Measurement, and Control*, vol. 102, no. 2, pp. 69–76, 1980.
- [18] F. Boyer and S. Ali, “Recursive inverse dynamics of mobile multibody systems with joints and wheels,” *IEEE Transactions on robotics*, vol. 27, no. 2, pp. 215–228, 2011.
- [19] J. Angeles and S. K. Lee, “The formulation of dynamical equations of holonomic mechanical systems using a natural orthogonal complement,” *Journal of applied mechanics*, vol. 55, p. 243, 1988.
- [20] J. Angeles, *Fundamentals of robotic mechanical systems: theory, methods, and algorithms*, vol. 124. Springer Science & Business Media, 2013.
- [21] S. K. Saha and J. Angeles, “Kinematics and dynamics of a three-wheeled 2-dof agv,” in *Robotics and Automation, 1989. Proceedings., 1989 IEEE International Conference on*, pp. 1572–1577, IEEE, 1989.
- [22] S. K. Saha and J. Angeles, “Dynamics of nonholonomic mechanical systems using a natural orthogonal complement,” *Journal of Applied Mechanics*, vol. 58, no. 1, pp. 238–243, 1991.
- [23] W. Buchele, L. Xie, *et al.*, “Computer analysis of the lateral stability of agricultural tractors.,” *Paper-American Society of Agricultural Engineers*, no. 90-1589, 1990.

- [24] I. S. Jones and M. B. Penny, “Engineering parameters related to rollover frequency,” tech. rep., SAE Technical Paper, 1990.
- [25] S. Dubowsky and E. Vance, “Planning mobile manipulator motions considering vehicle dynamic stability constraints,” in *Proceedings, 1989 International Conference on Robotics and Automation*, pp. 1271–1276, IEEE, 1989.
- [26] R. B. McGhee and A. A. Frank, “On the stability properties of quadruped creeping gaits,” *Mathematical Biosciences*, vol. 3, pp. 331–351, 1968.
- [27] M. Vukobratovic and D. Juricic, “Contribution to the synthesis of biped gait,” *IEEE Transactions on Biomedical Engineering*, no. 1, pp. 1–6, 1969.
- [28] M. Vukobratovic, B. Borovac, D. Surla, and D. Stokic, *Biped locomotion: dynamics, stability, control and application*, vol. 7. Springer Science & Business Media, 2012.
- [29] Q. Huang and S. Sugano, “Manipulator motion planning for stabilizing a mobile-manipulator,” in *Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots*, vol. 3, pp. 467–472, IEEE, 1995.
- [30] Q. Huang, S. Sugano, and I. Kato, “Stability control for a mobile manipulator using a potential method,” in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS’94)*, vol. 2, pp. 839–846, IEEE, 1994.
- [31] S. Sugano, Q. Huang, and I. Kato, “Stability criteria in controlling mobile robotic systems,” in *Proceedings of 1993 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS’93)*, vol. 2, pp. 832–838, IEEE, 1993.
- [32] S. Furuno, M. Yamamoto, and A. Mohri, “Trajectory planning of mobile manipulator with stability considerations,” in *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)*, vol. 3, pp. 3403–3408, IEEE, 2003.
- [33] D. Messuri and C. Klein, “Automatic body regulation for maintaining stability of a legged vehicle during rough-terrain locomotion,” *IEEE Journal on Robotics and Automation*, vol. 1, no. 3, pp. 132–141, 1985.
- [34] A. Ghasempoor and N. Sepehri, “A measure of machine stability for moving base manipulators,” in *Proceedings of 1995 IEEE International Conference on Robotics and Automation*, vol. 3, pp. 2249–2254, IEEE, 1995.
- [35] E. Papadopoulos and D. A. Rey, “A new measure of tipover stability margin for mobile manipulators,” in *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 4, pp. 3111–3116, IEEE, 1996.
- [36] E. Papadopoulos and D. A. Rey, “The force-angle measure of tipover stability margin for mobile manipulators,” *Vehicle System Dynamics*, vol. 33, no. 1, pp. 29–48, 2000.

- [37] R. C. Coulter, “Implementation of the pure pursuit path tracking algorithm,” tech. rep., Carnegie-Mellon UNIV Pittsburgh PA Robotics INST, 1992.
- [38] M. J. Barton, “Controller development and implementation for path planning and following in an autonomous urban vehicle,” *Undergraduate thesis, University of Sydney*, 2001.
- [39] J. Wit, C. D. Crane III, and D. Armstrong, “Autonomous ground vehicle path tracking,” *Journal of Robotic Systems*, vol. 21, no. 8, pp. 439–449, 2004.
- [40] T. Hellstrom and O. Ringdahl, “Follow the past: a path-tracking algorithm for autonomous vehicles,” *International journal of vehicle autonomous systems*, vol. 4, no. 2, pp. 216–224, 2006.
- [41] B. Paden, M. Čáp, S. Z. Yong, D. Yershov, and E. Frazzoli, “A survey of motion planning and control techniques for self-driving urban vehicles,” *IEEE Transactions on intelligent vehicles*, vol. 1, no. 1, pp. 33–55, 2016.
- [42] J. Borenstein and Y. Koren, “Real-time obstacle avoidance for fast mobile robots,” *IEEE Transactions on systems, Man, and Cybernetics*, vol. 19, no. 5, pp. 1179–1187, 1989.
- [43] J. Borenstein and Y. Koren, “The vector field histogram-fast obstacle avoidance for mobile robots,” *IEEE transactions on robotics and automation*, vol. 7, no. 3, pp. 278–288, 1991.
- [44] I. Ulrich and J. Borenstein, “Vfh+: Reliable obstacle avoidance for fast mobile robots,” in *Proceedings. 1998 IEEE international conference on robotics and automation (Cat. No. 98CH36146)*, vol. 2, pp. 1572–1577, IEEE, 1998.
- [45] O. Brock and O. Khatib, “High-speed navigation using the global dynamic window approach,” in *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)*, vol. 1, pp. 341–346, IEEE, 1999.
- [46] O. Khatib, “Real-time obstacle avoidance for manipulators and mobile robots,” in *Autonomous robot vehicles*, pp. 396–404, Springer, 1986.
- [47] L. Tang, S. Dian, G. Gu, K. Zhou, S. Wang, and X. Feng, “A novel potential field method for obstacle avoidance and path planning of mobile robot,” in *2010 3rd International Conference on Computer Science and Information Technology*, vol. 9, pp. 633–637, IEEE, 2010.
- [48] M. G. Park, J. H. Jeon, and M. C. Lee, “Obstacle avoidance for mobile robots using artificial potential field approach with simulated annealing,” in *ISIE 2001. 2001 IEEE International Symposium on Industrial Electronics Proceedings (Cat. No. 01TH8570)*, vol. 3, pp. 1530–1535, IEEE, 2001.
- [49] J. Minguez and L. Montano, “Nearness diagram (nd) navigation: collision avoidance in troublesome scenarios,” *IEEE Transactions on Robotics and Automation*, vol. 20, no. 1, pp. 45–59, 2004.

- [50] S. Garrido, L. Moreno, and D. Blanco, “Voronoi diagram and fast marching applied to path planning,” in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pp. 3049–3054, IEEE, 2006.
- [51] J.-C. Latombe, *Robot motion planning*, vol. 124. Springer Science & Business Media, 2012.
- [52] F. Duchoň, A. Babinec, M. Kajan, P. Beňo, M. Florek, T. Fico, and L. Juríšica, “Path planning with modified a star algorithm for a mobile robot,” *Procedia Engineering*, vol. 96, pp. 59–69, 2014.
- [53] N. J. Nilsson, “A mobile automaton: An application of artificial intelligence techniques,” tech. rep., SRI INTERNATIONAL MENLO PARK CA ARTIFICIAL INTELLIGENCE CENTER, 1969.
- [54] P. Bhattacharya and M. L. Gavrilova, “Roadmap-based path planning-using the voronoi diagram for a clearance-based shortest path,” *IEEE Robotics & Automation Magazine*, vol. 15, no. 2, pp. 58–66, 2008.
- [55] O. Takahashi and R. J. Schilling, “Motion planning in a plane using generalized voronoi diagrams,” *IEEE Transactions on robotics and automation*, vol. 5, no. 2, pp. 143–150, 1989.
- [56] H. L. Martin and D. P. Kuban, *Teleoperated robotics in hostile environments*. Society of Manufacturing Engineers, 1985.
- [57] J. Vertut and P. Coiffet, “Teleoperations and robotics: Evolution and development,âĂĲrobot technology,âĂĪ vol. 3a,” 1986.
- [58] T. H. Massie, J. K. Salisbury, *et al.*, “The phantom haptic interface: A device for probing virtual objects,” in *Proceedings of the ASME winter annual meeting, symposium on haptic interfaces for virtual environment and teleoperator systems*, vol. 55,1, pp. 295–300, Citeseer, 1994.
- [59] S. Grange, F. Conti, P. Helmer, P. Rouiller, and C. Baur, “Overview of the delta haptic device,” tech. rep., 2001.
- [60] C. Clover, G. R. Luecke, J. J. Troy, and W. A. McNeely, “Dynamic simulation of virtual mechanisms with haptic feedback using industrial robotics equipment,” in *Robotics and Automation, 1997. Proceedings., 1997 IEEE International Conference on*, vol. 1, pp. 724–730, IEEE, 1997.
- [61] M. Ueberle, N. Mock, and M. Buss, “Vishard10, a novel hyper-redundant haptic interface,” in *Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2004. HAPTICS'04. Proceedings. 12th International Symposium on*, pp. 58–65, IEEE, 2004.
- [62] J. Yoon and J. Ryu, “Design, fabrication, and evaluation of a new haptic device using a parallel mechanism,” *IEEE/ASME Transactions on mechatronics*, vol. 6, no. 3, pp. 221–233, 2001.

- [63] R. Aracil, M. Ferre, M. Hernando, and J. Sebastian, “Telerobotic system for live-power line maintenance: Robtet,” *IEEE Transactions on Industrial Electronics*, vol. 44, no. 5-44, pp. 630–637, 1997.
- [64] L. Matthies, “Stereo vision for planetary rovers: Stochastic modeling to near real-time implementation,” *International Journal of Computer Vision*, vol. 8, no. 1, pp. 71–91, 1992.
- [65] G. Hirzinger, “Robots in space-a survey,” *Advanced Robotics*, vol. 9, no. 6, pp. 625–651, 1994.
- [66] R. Goertz, R. Blomgren, J. Grimson, G. Forster, W. Thompson, and W. Kline, “The anl model 3 master-slave electric manipulator—its design and use in a cave,” *Trans. Am. Nuclear Soc.*, vol. 4, no. 2, 1961.
- [67] C. Flatau, “Sm-229: a new compact servo master-slave manipulator,” in *Proceedings of the 25th conference on remote systems technology*, 1977.
- [68] D. A. Lawrence, “Stability and transparency in bilateral teleoperation,” *IEEE transactions on robotics and automation*, vol. 9, no. 5, pp. 624–637, 1993.
- [69] P. F. Hokayem and M. W. Spong, “Bilateral teleoperation: An historical survey,” *Automatica*, vol. 42, no. 12, pp. 2035–2057, 2006.
- [70] R. J. Anderson and M. W. Spong, “Bilateral control of teleoperators with time delay,” *IEEE Transactions on Automatic control*, vol. 34, no. 5, pp. 494–501, 1989.
- [71] J. Rebelo and A. Schiele, “Time domain passivity controller for 4-channel time-delay bilateral teleoperation,” *IEEE transactions on haptics*, vol. 8, no. 1, pp. 79–89, 2015.
- [72] G. Niemeyer and J.-J. Slotine, “Stable adaptive teleoperation,” *IEEE Journal of oceanic engineering*, vol. 16, no. 1, pp. 152–162, 1991.
- [73] S. Stramigioli, R. Mahony, and P. Corke, “A novel approach to haptic teleoperation of aerial robot vehicles,” in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pp. 5302–5308, IEEE, 2010.
- [74] S. Stramigioli, C. Secchi, A. J. van der Schaft, and C. Fantuzzi, “Sampled data systems passivity and discrete port-hamiltonian systems,” *IEEE Transactions on Robotics*, vol. 21, no. 4, pp. 574–587, 2005.
- [75] M. Krstic, *Delay compensation for nonlinear, adaptive, and PDE systems*. Springer, 2009.
- [76] J. Y. Chen, E. C. Haas, and M. J. Barnes, “Human performance issues and user interface design for teleoperated robots,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 37, no. 6, pp. 1231–1245, 2007.

- [77] T. Sheridan, “Human supervisory control of robot systems,” in *Robotics and Automation. Proceedings. 1986 IEEE International Conference on*, vol. 3, pp. 808–812, IEEE, 1986.
- [78] P. K. Pook and D. H. Ballard, “Teleassistance: Contextual guidance for autonomous manipulation,” in *AAAI*, pp. 1291–1296, 1994.
- [79] M. Jagersand and R. Nelson, “Visual space task specification, planning and control,” in *Computer Vision, 1995. Proceedings., International Symposium on*, pp. 521–526, IEEE, 1995.
- [80] T. B. Sheridan, “Space teleoperation through time delay: Review and prognosis,” *IEEE Transactions on robotics and Automation*, vol. 9, no. 5, pp. 592–606, 1993.
- [81] A. K. Bejczy and W. S. Kim, “Predictive displays and shared compliance control for time-delayed telemanipulation,” in *Intelligent Robots and Systems' 90.'Towards a New Frontier of Applications', Proceedings. IROS'90. IEEE International Workshop on*, pp. 407–412, IEEE, 1990.
- [82] W. S. Kim and A. K. Bejczy, “Demonstration of a high-fidelity predictive/preview display technique for telerobotic servicing in space,” *IEEE Transactions on Robotics and Automation*, vol. 9, no. 5, pp. 698–702, 1993.
- [83] M. Jagersand, “Image based predictive display for tele-manipulation,” in *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*, vol. 1, pp. 550–556, IEEE, 1999.
- [84] K. Yerex, D. Cobzas, and M. Jagersand, “Predictive display models for tele-manipulation from uncalibrated camera-capture of scene geometry and appearance,” in *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, vol. 2, pp. 2812–2817, IEEE, 2003.
- [85] Z. Deng and M. Jagersand, “Predictive display system for tele-manipulation using image-based modeling and rendering,” in *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, vol. 3, pp. 2797–2802, IEEE, 2003.
- [86] A. Kelly, N. Chan, H. Herman, D. Huber, R. Meyers, P. Rander, R. Warner, J. Ziglar, and E. Capstick, “Real-time photorealistic virtualized reality interface for remote mobile robot control,” *The International Journal of Robotics Research*, vol. 30, no. 3, pp. 384–404, 2011.
- [87] T. Burkert, J. Leupold, and G. Passig, “A photorealistic predictive display,” *Presence: Teleoperators & Virtual Environments*, vol. 13, no. 1, pp. 22–43, 2004.
- [88] H. Hu, C. P. Quintero, H. Sun, and M. Jagersand, “On-line reconstruction based predictive display in unknown environment,” in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pp. 4446–4451, IEEE, 2015.

- [89] Y. Yamamoto and X. Yun, “Coordinating locomotion and manipulation of a mobile manipulator,” in *Decision and Control, 1992., Proceedings of the 31st IEEE Conference on*, pp. 2643–2648, IEEE, 1992.
- [90] S. Rajendran and et.al., “Mobile robot for reactor vessel inspection,” in *National Conference on Advanced Manufacturing and Robotics*, pp. 2527–2532, CMERI, Durgapur, WB,, 2004.
- [91] F. G. Pin and S. M. Killough, “A new family of omnidirectional and holonomic wheeled platforms for mobile robots,” *IEEE transactions on robotics and automation*, vol. 10, no. 4, pp. 480–489, 1994.
- [92] J. E. M. Salih, M. Rizon, S. Yaacob, A. H. Adom, and M. R. Mamat, “Designing omni-directional mobile robot with mecanum wheel,” *American Journal of Applied Sciences*, vol. 3, no. 5, pp. 1831–1835, 2006.
- [93] J. Suthakorn, S. S. H. Shah, S. Jantarajit, W. Onprasert, W. Saensupo, S. Saeung, S. Nakdhamabhorn, V. Sa-Ing, and S. Reaungamornrat, “On the design and development of a rough terrain robot for rescue missions,” in *Robotics and Biomimetics, 2008. ROBIO 2008. IEEE International Conference on*, pp. 1830–1835, IEEE, 2009.
- [94] M. Guarnieri, R. Debenest, T. Inoh, E. Fukushima, and S. Hirose, “Development of helios vii: an arm-equipped tracked vehicle for search and rescue operations,” in *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, vol. 1, pp. 39–45, IEEE, 2004.
- [95] K. Nagatani, S. Tachibana, M. Sofne, and Y. Tanaka, “Improvement of odometry for omnidirectional vehicle using optical flow information,” in *Intelligent Robots and Systems, 2000.(IROS 2000). Proceedings. 2000 IEEE/RSJ International Conference on*, vol. 1, pp. 468–473, IEEE, 2000.
- [96] J. Y. Wong, *Theory of ground vehicles*. John Wiley & Sons, 2008.
- [97] T. Bevan, *Theory of Machines, Third Edition*. Delhi, India: CBS Publishers & Distributors, 1984.
- [98] *Motor catalogue 2014*. Switzerland: Maxon Motors, 2017.
- [99] J. T. Machado and M. F. Silva, “An overview of legged robots,” in *International symposium on mathematical methods in engineering*, MME Press Ankara, Turkey, 2006.
- [100] K. P. Valavanis and G. J. Vachtsevanos, *Handbook of unmanned aerial vehicles*. Springer Publishing Company, Incorporated, 2014.
- [101] J.-B. Mouret, *Evolutionary Adaptation in Natural and Artificial Systems*. PhD thesis, Université Pierre et Marie Curie, 2015.
- [102] G. Campion and W. Chung, “Wheeled robots,” in *Springer Handbook of Robotics*, pp. 391–410, Springer, 2008.

- [103] R. M. DeSantis, “Modeling and path-tracking control of a mobile wheeled robot with a differential drive,” *Robotica*, vol. 13, no. 04, pp. 401–410, 1995.
- [104] K. C. Koh and H. S. Cho, “A smooth path tracking algorithm for wheeled mobile robots with dynamic constraints,” *Journal of intelligent & robotic systems*, vol. 24, no. 4, pp. 367–385, 1999.
- [105] B. d’Andréa Novel, G. Campion, and G. Bastin, “Control of nonholonomic wheeled mobile robots by state feedback linearization,” *The International journal of robotics research*, vol. 14, no. 6, pp. 543–559, 1995.
- [106] B. d’Andrea Novel, G. Bastin, and G. Campion, “Modelling and control of non-holonomic wheeled mobile robots,” in *Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference on*, pp. 1130–1135, IEEE, 1991.
- [107] K. Thanjavur and R. Rajagopalan, “Ease of dynamic modelling of wheeled mobile robots (wmrs) using kane’s approach,” in *Robotics and Automation, 1997. Proceedings., 1997 IEEE International Conference on*, vol. 4, pp. 2926–2931, IEEE, 1997.
- [108] S. Saha, *Introduction to robotics*. Tata McGraw-Hill, New Delhi, 2010.
- [109] J. J. Craig, *Introduction to robotics: mechanics and control*, vol. 3. Pearson/Prentice Hall Upper Saddle River, NJ, USA:, 2005.
- [110] *EPOS Application Note: RS232 to CANopen Getway*. Maxon Motors, 2012.
- [111] I. Farkhatdinov and J.-H. Ryu, “Hybrid position-position and position-speed command strategy for the bilateral teleoperation of a mobile robot,” in *Control, Automation and Systems, 2007. ICCAS’07. International Conference on*, pp. 2442–2447, IEEE, 2007.
- [112] M. Z. Zeroual, “maxon auto tuning,” *iNaCoMM2013,IIT Roorkee, India*, 2013.
- [113] *EPOS4 Application Note: Positioning Controllers*. Maxon Motors, 2012.
- [114] S. T. B, “Space teleoperation through time delay: review and prognosis,” *IEEE Transactions on Robotics and Automation*, vol. 9, no. 5, October 1993.
- [115] J. Geng, “Structured-light 3d surface imaging: a tutorial,” *Advances in Optics and Photonics*, vol. 3, no. 2, pp. 128–160, 2011.
- [116] “Openni [online].” <http://www.openni.org/>. Accessed: 2018-09-30.
- [117] “Microsoft kinect sdk [online].” <https://www.microsoft.com/en-hk/download/details.aspx?id=44561>. Accessed: 2018-09-30.
- [118] “Openkinect [online].” <https://github.com/OpenKinect/libfreenect/>. Accessed: 2018-09-30.
- [119] “Rtabmap [online].” <https://github.com/introlab/rtabmap>. Accessed: 2018-09-30.

- [120] M. Labb   and F. Michaud, “Rtab-map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation,” *Journal of Field Robotics*, vol. 36, no. 2, pp. 416–446, 2019.
- [121] M. Labbe and F. Michaud, “Appearance-based loop closure detection for online large-scale and long-term operation,” *IEEE Transactions on Robotics*, vol. 29, no. 3, pp. 734–745, 2013.

Appendix A

A.1 Measurment of Time Delay in Video Feed-back

The experiment used to find the time delay in video feedback is described here. The operator station's graphical user interface shown in Figure A.1. This displays the video received form the camera mounted on the mobile robot in the window marked with a green boundary.

To measure the time delay a digital clock and the user interface runs side by side on the operator screen. The camera on the robot is now oriented to face the user interface, so that the camera captures the video of the digital clock running on the user PC. The schematic of the experimental set-up is shown in Figure A.2. The screen shot of the operator screen is shown in Figure A.3. The red box marks the clock running of on the PC. The blue box marks the time captured by the robot camera in the most recent video frame received at the operator station. We thus see the delay of $9.2 - 8.7 = 0.5 \text{ sec}$, for the video frame to travel from robot to the operator station. This delay includes both the processing time and the transmission time.

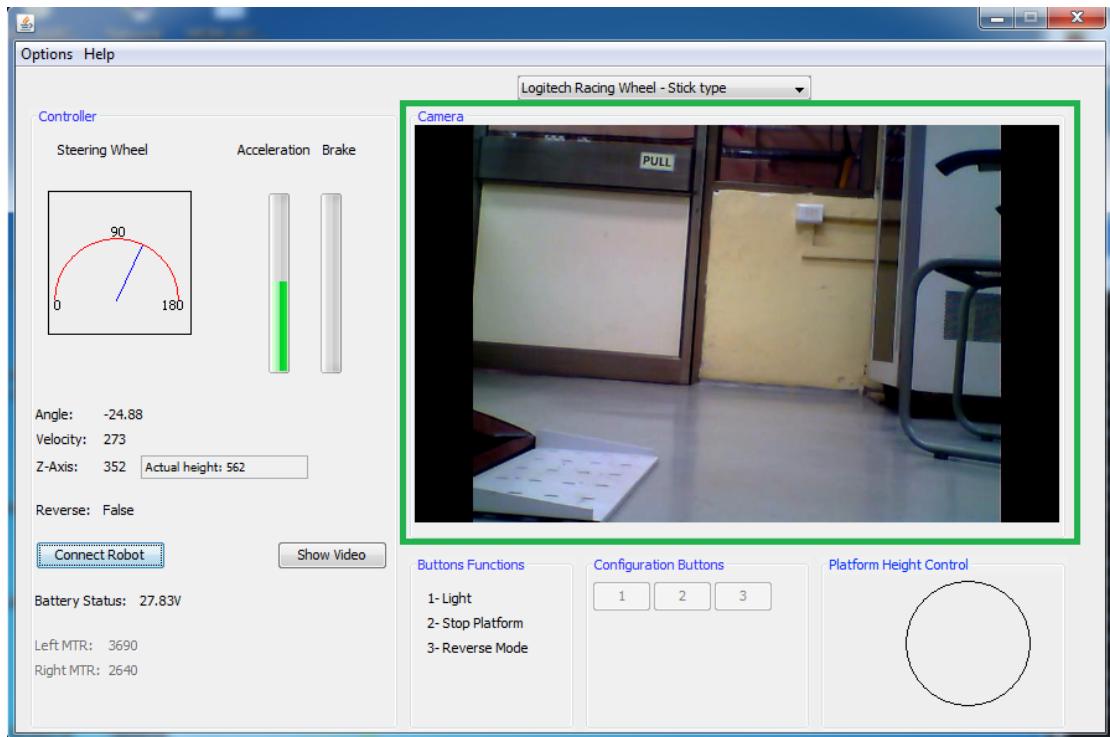


Figure A.1: User interface for teleoperation

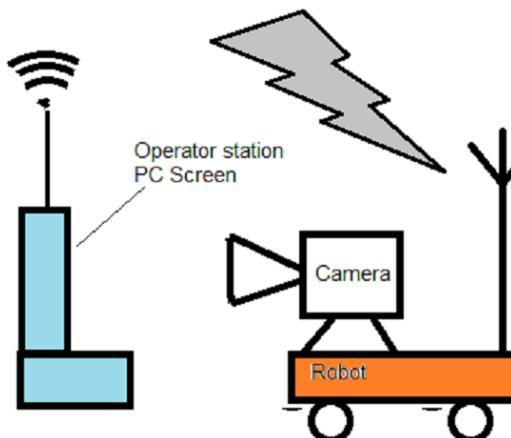


Figure A.2: Experimental Setup: Schematic



Figure A.3: Screen Shot of Operator PC

Appendix B

B.1 Stability of Pure Pursuit under Time Delay

The kinematic model of a mobile robot is given by Equation B.1 and first order model for steering dynamics is given by Equation B.2

$$\begin{pmatrix} \dot{x}'_w \\ \dot{y}'_w \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} -V \sin \theta \\ V \cos \theta \\ V\gamma \end{pmatrix} \quad (\text{B.1})$$

$$\frac{d\gamma'}{dt} = -\frac{1}{T}(\gamma' - \gamma_r) \quad (\text{B.2})$$

where, (x', y', θ) is the posture of robot in coordinate system attached to the robot, V is the longitudinal velocity, γ is the angular velocity, T time constant of steering dynamics and γ_r the control input. Since the control input is generated by Pure Pursuit it is given by

$$\gamma_r = \frac{1}{r} = \frac{2x}{L^2} \quad (\text{B.3})$$

where r is defined in Equation 5.5 and L is the *look ahead distance*. The above Equations B.1 and B.2 using the fact that $\dot{y}'_w = 0$, because of the non-holonomic

constraint of the robot, can be written in the state space form as

$$\begin{pmatrix} \dot{x} \\ \dot{\theta} \\ \dot{\gamma} \end{pmatrix} = \begin{pmatrix} -\sin \theta \\ \gamma \\ -\gamma - \frac{2x}{l^2} \end{pmatrix} \quad (\text{B.4})$$

where the following scaling of variables were used to render it non-dimensional.

$$t = \frac{t'}{T}, \quad x = \frac{x'_w}{VT}, \quad \gamma = VT\gamma', \quad l = \frac{L}{VT}$$

The above system was linearised about the origin and delay τ in input was introduced to get Equation B.5

$$\begin{pmatrix} \dot{x} \\ \dot{\theta} \\ \dot{\gamma} \end{pmatrix} = J \begin{pmatrix} x(t) \\ \theta(t) \\ \gamma(t) \end{pmatrix} + J_\tau \begin{pmatrix} x(t-\tau) \\ \theta(t-\tau) \\ \gamma(t-\tau) \end{pmatrix} \quad (\text{B.5})$$

where J is the Jacobian with respect to state and J_τ is Jacobian with respect to τ . Based on location of roots of the Characteristic Quasi-Polynomial $q(s)$ given by

$$q(s) = \det[sI - J - J_\tau e^{s\tau}]$$

limiting look ahead distance L , for stable path tracking of a circular path for different input delays was presented by Ollero [5]. The results are represented here for in Figure B.1.

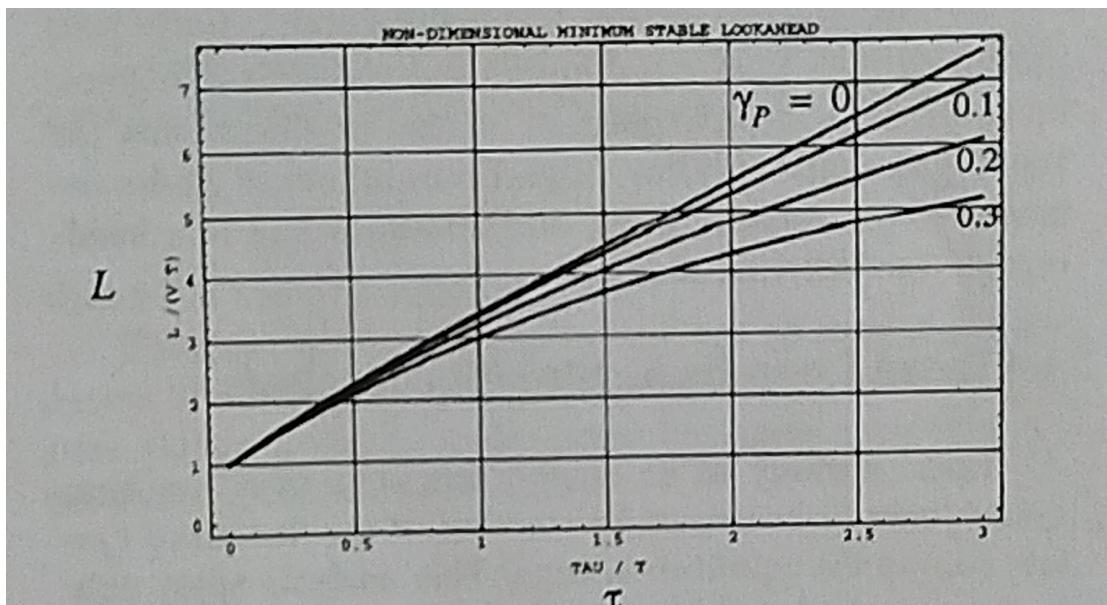


Figure B.1: Results obtained by Ollero [5]

Appendix C

C.1 Optimal Design of Steering Linkages

Objective is to minimize turning radius within the space available after installation of the rear wheel assembly, battery compartment and scissor mechanism on a given over all frame size of the vehicle. The design variables are x , a , h , w . These variables are described in Chapter 3 and shown in Figures 2.7 and 2.8

Objective function to be minimized is the turning radius is given by

$$R = \frac{a}{2} + \frac{w(h^2 + b^2 - bx)}{hx} \quad (\text{C.1})$$

where

$$\frac{2b}{h} = \frac{a}{w}$$

The constrain equations are

$$x < \frac{a}{2} - b - 20$$

$$h^2 + \left(\frac{a}{2}\right)^2 < 100^2$$

The bonds on the design variables are given by

$$30 < h < 200$$

$$500 < w < 600$$

$$50 < a < 270$$

$$0 < x$$

The minimization was carried out in Matlab using *fmincon()* function. The resulting minimum radius thus obtained was

$$R_{min} = 228 \text{ mm}$$

The corresponding design variables obtained were

$$h = 30 \text{ mm}, w = 515 \text{ mm}, a = 190 \text{ mm}, x = 69.8 \text{ mm}$$

The achieved minimum turning radius was $R_{min} = 415 \text{ mm}$. This was due to introduction of support for the rack and links which were not taken into consideration in the optimization process.

Publications from the Thesis

Research papers published/presented/under preparation are listed below

Conference

1. Das Amaren P., et al. "Design of a Teleoperated Mobile Manipulator for Inspection of Cyclotron Vault." *Machines, Mechanism and Robotics-Proceedings of iNaCoMM 2017*. Springer, Singapore, 2019. 529-540.
2. Das Amaren P., Saha S. K., Bhasin S., and Badodkar D. N., "RGB-D based Predictive Display for Tele-operation of Mobile Robot in Unknown Environment", To be Submitted to *Advances in Robotics, International Conference of Robotics Society of India, (AiR 2019)*, IIT Madras, Chennai, July 2-6, 2019
3. Das Amaren P., Saha S. K., Bhasin S., and Badodkar D. N., "Control Architecture for Tele-operated Mobile Platform for Radiation Mapping in Radioactive Areas", *under preparation*

Brief Bio-data of the Author

Shri Amaren Prasanna Das, graduated in Mechanical Engineering from Kalyani Government Engineering College, Kalyani, West Bengal, India there after joined BARC Training School in the year 1999. After successful completion of the one year orientation training in Nuclear Science and Technology, he joined Division of Remote Handling and Robotics, BARC, Mumbai in 2000 as Scientific Officer-C. Subsequently, he obtained his M.Tech. degree in Systems and Control Engineering from IIT-Bombay, Mumbai in 2007. He joined Ph.D. in 2012 in the Department of Mechanical Engineering at Indian Institute of Technology Delhi as an external part-time Ph. D. scholar.