

# **DESIGN AND CONTROL OF TELEOPERATED MOBILE PLATFORM**

**AMAREN PRASANNA DAS**



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

**DECEMBER 2018**

©Indian Institute of Technology Delhi (IITD), New Delhi, 2017

# **DESIGN AND CONTROL OF TELEOPERATED MOBILE PLATFORM**

by

**AMAREN P 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 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 my 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

**Dr. 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

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. To all these people, I extend my eternal gratitude. 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. I am deeply grateful to my co-supervisor, Professor S. Bhasin of Department of Electric Engineering, IIT-D, 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 through out the tenure of my research.

It will be imprudent not to mention the help and support provided by my colleagues Shri Rahul V Sakrikar. One person on whom I fell back in case of any hardware issues, it was him. His experience and field knowledge has been highly enriching, while discussing technical issues such as interpretation of graphs and results. I still remember his *encouraging* words, "Are tera PhD ka kya hua. Saha kuch kaam nahi deya kya". Another colleagues , Shri Shishir Kr. Singh has provided his selfless service to help me develop all the software required for running the actual system. With out his support it would have been impossible to take experimental runs and data collection. I express my deepest regards for them.

I cannot forget the support and help provide by Dr. Abraham Ribby, Dr. Parmannad, Dr. Aamir Abdulla and Dr. Arun Uday during my stay at IIT-Delhi for 6 months during my course work. Thereafter they provided all kind of

support needed from fee submission to physical submission of progress report in each semester, to name a few.

Finally, I thank my two daughters, Akaisha and Sayuri, my mother for their patience and support. Special thanks to my wife Namrata, who had taken upon herself the responsibility of managing the family's without my slightest help. In absence of her support, patience and companionship this journey would not have been completed.

**Amaren P 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 discussed. The dynamic model of the system was also developed and simulation was carried out for specific path to verify the actuator requirements and response time. The user interface and control architecture implemented on the mobile robot for teleoperation is described. The time delay introduced in video data transfer due to bandwidth limitation and image processing at camera end and its effect on system stability and poor operator performance is demonstrated using simulation. A predictive display of remote environment based on mathematical model of the mobile robot and RGB-Depth sensor data received from remote location is developed and implemented. This strategy has largely improved robot navigation by the operator even over delayed communication network.



# Contents

Certificate . . . . .	ii
Acknowledgements . . . . .	v
Abstract . . . . .	vii
List of Figures . . . . .	xi
List of Tables . . . . .	xvii
<b>1 Introduction</b>	<b>1</b>
1.1 Research Contributions . . . . .	3
1.2 Thesis Organization . . . . .	4
<b>2 Literature Review</b>	<b>7</b>
2.1 Special Purpose Robots . . . . .	7
2.2 Mathematical Modeling . . . . .	8
2.3 Path Tracking . . . . .	10
2.4 Tele-operation . . . . .	11
2.4.1 Human Interface . . . . .	12
2.4.2 Control . . . . .	12
2.5 Predictive Display . . . . .	13
2.6 Summary . . . . .	14
<b>3 Design of Mobile a Robot</b>	<b>15</b>
3.1 Design Overview . . . . .	15
3.2 Design of the Traction System . . . . .	17
3.2.1 Selection of Motor and Gearbox . . . . .	19

3.3	Design of Steering System . . . . .	20
3.3.1	Minimum Turning Radius . . . . .	21
3.3.2	Calculation of Steering Torque . . . . .	22
3.4	Design of Scissor Mechanism for Manipulating Arm . . . . .	23
3.5	Summary . . . . .	25
<b>4</b>	<b>Dynamics of Wheeled Mobile Robots</b>	<b>27</b>
4.1	Modeling using the Natural Orthogonal Compliment (NOC) . . . . .	28
4.2	Dynamic Equation of WMR . . . . .	31
4.2.1	Kinematic analysis . . . . .	32
4.2.2	Dynamic equations . . . . .	36
4.2.2.1	Generalized Inertia Matrix, $I$ . . . . .	36
4.2.2.2	Matrix of Convective Inertia term $C$ . . . . .	37
4.3	Special cases . . . . .	39
4.3.1	Standard caster ( $d_1 = 0$ ) . . . . .	39
4.3.2	Under Actuated Case ( $d_2 = 0$ ) . . . . .	39
4.4	Simulation . . . . .	40
4.4.1	Inverse Dynamics . . . . .	41
4.5	Summary . . . . .	43
<b>5</b>	<b>Control of Mobile Manipulator</b>	<b>45</b>
5.1	Control Architecture and Hardware . . . . .	45
5.1.1	Local Onboard Controller . . . . .	46
5.1.2	Details of the Motor Controller . . . . .	48
5.2	The control algorithm for the WMR . . . . .	50
5.2.1	Wheel Odometry . . . . .	54
5.2.2	The Remote Control Station . . . . .	56
5.3	Summary . . . . .	57
<b>6</b>	<b>Simulation of Tele-operation</b>	<b>59</b>
6.1	Modeling of Mobile Platform . . . . .	60

6.2	Model of Human Operator . . . . .	60
6.3	Simulation and Results . . . . .	63
6.3.1	The Simulation Algorithm . . . . .	63
6.4	Summary . . . . .	65
<b>7</b>	<b>Time Delay Compensation and Predictive Display</b>	<b>69</b>
7.1	Model Based Time Delay Compensation . . . . .	69
7.1.1	Proposed Controller . . . . .	71
7.1.2	Simulation algorithm and results . . . . .	72
7.2	Predictive Display . . . . .	74
7.2.1	Depth Sensor Kinect . . . . .	75
7.2.2	Onboard Data processing and transmission . . . . .	76
7.2.3	3-D Reconstruction at operator station . . . . .	77
7.2.4	Remote scene extrapolation . . . . .	78
7.3	Result and conclusion . . . . .	79
7.4	Summary . . . . .	79
<b>8</b>	<b>Conclusions</b>	<b>81</b>
8.1	Thesis Summary . . . . .	81
8.2	Current status . . . . .	83
8.3	Future Scope . . . . .	83
<b>Bibliography</b>		<b>84</b>
<b>A Simulation Time Delay Tele-operation</b>		<b>93</b>
A.1	Measurment of Time Delay in Video Feedback . . . . .	93
<b>B</b>		<b>95</b>
<b>Publications from the Thesis</b>		<b>99</b>
<b>Brief Bio-data of the Author</b>		<b>101</b>



# List of Figures

1.1	Robotics in Factory . . . . .	2
1.2	Health Care . . . . .	2
1.3	Exploration and surveillance . . . . .	2
1.4	Honda Asimo . . . . .	2
1.5	Sony Abio . . . . .	2
2.1	Robot for Fukushima Daiichi . . . . .	8
2.2	NERO III . . . . .	8
2.3	Follow the carrot . . . . .	11
2.4	Vector Pursuit . . . . .	11
3.1	3-D Model of the mobile manipulator . . . . .	16
3.2	Photograph of the actual System . . . . .	16
3.3	Rear Suspension . . . . .	19
3.4	Mobile Manipulator on slope . . . . .	19
3.5	Ackerman Steering Condition . . . . .	21
3.6	Davis Steering Gear . . . . .	22
3.7	Steer Torque . . . . .	22
3.8	Scissor Mechanism . . . . .	24
4.1	WMR-Std. Castor . . . . .	31
4.2	WMR-general . . . . .	31
4.3	Castor wheel configuration of a WMR . . . . .	31
4.4	Path Traced by Robot . . . . .	40

4.5	Inverse dynamics of the mobile robot . . . . .	42
5.1	Control architecture . . . . .	45
5.2	Data from PC to the robot . . . . .	46
5.3	Data from the robot to PC . . . . .	46
5.4	Wiring diagram of the WMR . . . . .	47
5.5	Block diagram of EPOS4 controller . . . . .	48
5.6	Current control block . . . . .	49
5.7	Velocity control bolck . . . . .	49
5.8	Position control bolck . . . . .	50
5.9	Block digram of WMR controller . . . . .	51
5.10	Ackerman Steering Condition . . . . .	53
5.11	Davis Steering Gear . . . . .	53
5.12	Tracing a line . . . . .	55
5.13	Motor torque . . . . .	55
5.14	Linear velocity . . . . .	55
5.15	Angular velocity . . . . .	55
5.16	Tracing a circle . . . . .	56
5.17	Motor torque . . . . .	56
5.18	Linear velocity . . . . .	56
5.19	Angular velocity . . . . .	56
5.20	User interface for teleoperation . . . . .	57
6.1	Teleoperation architecture . . . . .	59
6.2	Assumed driving strategy . . . . .	61
6.3	Geometry of Pure Pursuit . . . . .	62
6.4	Simulation Scheme . . . . .	63
6.5	Block diagram for teleoperation . . . . .	64
6.6	Simulation with no time delay in either direction . . . . .	65
6.7	Simulation with time delay $h_1 = .5sec$ and $h_2 = 0$ . . . . .	66

6.8	Simulation with time delay $h_1 = .8sec$ and $h_2 = 0$	67
7.1	Smith Predictor	70
7.2	Delay measurement	71
7.3	Block diagram of time delay system	71
7.4	Smith predictor applied to mobile robot	72
7.5	Simulation block Diagram	73
7.6	Time delay $T_1 = .5sec$ and $T_2 = 0$	74
7.7	Time delay $T_1 = .8sec$ and $T_2 = 0$	74
7.8	Kinect Sensor from Microsoft	75
7.9	Predictive display architecture	77
7.10	PCD at time T	79
7.11	Predicted Scene	79
B.1	Results obtained by Ollero [1]	97



# List of Tables

3.1	Key parameters and specifications of the mobile manipulator.	16
4.1	Dynamic & kinematic parameters	42
5.1	Proportionality constant table	48
5.2	Parameters for left and right rear wheel motor controllers	50
5.3	Parameters of steering motor controller	51



# Chapter 1

## Introduction

The use of robots such as robotic arm 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 figure 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 1.4 by Honda and Aibo by Sony 1.5.

In areas where human access is not preferred or restricted due to risk of life or inhospitable environmental conditions such as which exists 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 work space 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.

This research too was motivated by a similar requirement for in-situ mea-



Figure 1.1: Robotics in Factory



Figure 1.2: Health Care



Figure 1.3: Exploration and surveillance



Figure 1.4: Honda Asimo



Figure 1.5: Sony Aibo

surement of the radioactive radiation, mostly 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 particle beam to high energy. 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 and cave is the areas where beam line (beam of accelerated particles) is available for experimentation. Radiation mapping of these areas are mandatory requirements 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 area they can only measure radiation levels at discrete location 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 there 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  $\gamma$ -rays, neutrons and delayed radiation ( $\beta$  and  $\gamma$ ) thus making human presence unacceptable. A tele-operated mobile robot with wireless communication link is the obvious solution. This thesis discusses design analysis and develop of a prototype robot to carry out in-situ measurement and mapping of radiation level.

## 1.1 Research Contributions

The original contributions of the present research are listed below:

- (i) Design of a customized teleoperated mobile robot for remote surveillance and mapping application.
- (ii) Development of kinematic and dynamic model of the robot at hand.
- (iii) Control architecture required for intuitive teleoperation of this mobile robot.
- (iv) Predictive visual feedback control to alleviate problem arising due to time delay.

## 1.2 Thesis Organization

The thesis contains eight chapters and one 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.

### **Chapter 2: Literature Review**

This chapter presents the 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 teleopreration*, and *predictive display systems*.

### **Chapter 3: Design of a Mobile Robot**

This chapter highlights the design considerations of tele-operated mobile robot based on mission requirements and environmental conditions. It will 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 4: Mathematica Modelling of Wheeled Mobile Robots**

In this chapter, the orthogonal compliment method will be presented to derive the

dynamic equations of a four wheeled differentially driven platform. The platform has two actuated wheels and two passive wheels. Different types of passive wheels were studied and corresponding dynamic equations were derived.

### **Chapter 5:Control of 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. A comparison study with the simulated results based on dynamic analysis is also presented presented.

### **Chapter 6: 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 simulation that with increase in time delay teleoperation loop become unstable.

### **Chapter 7: 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 8: Conclusions**

This chapter summarizes major results of this research work. Limitations along with the future scope from the 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. In this appendix,



# Chapter 2

## Literature Review

In this chapter, 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 work 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 *Teleoperation* 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 teleoperation is discussed separately because it forms one of the major components of the teleoperation system network designed for the mobile robot presented in this thesis.

### 2.1 Special Purpose Robots

This section discusses 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 [2]. Honeycomb-glassfiber composite was used to provide high strength and low weight. Ohno, et



Figure 2.1: Robot for Fukushima Daiichi

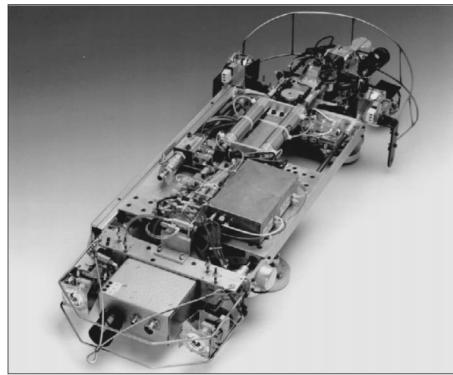


Figure 2.2: NERO III

al. [3] developed a robotic control vehicle shown in Figure 2.1 for measuring the radiation in the Fukushima Daiichi Nuclear Power Plant. A survey of different types of climbing robots for no-destructive testing of pressure vessel is given by [4], such as NERO III shown in Figure 2.2. Briones [5] presents a vacuum cup based wall climbing robot for inspection of nuclear power plants. Galt [6], 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 [7]. Compliant, link based mobile robot was designed and tested by Borenstein[8], 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 3.

## 2.2 Mathematical Modeling

A very comprehensive list of Wheeled Mobile Robots (WMR) using different wheel configuration is given by Muir and Neuman [9]. 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 [10]. The major difference is that he uses physical friction model in the analysis of over actuated systems where rolling constraints are not satisfied. A seminal work by Champion [11] gives the structural classification of wheeled mobile robots based on the *degree of mobility*,  $\delta_m$ , and *degree of steerability*,  $\delta_s$ . 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 [12]. 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 Euler-Lagrange (EL) equations. Hoostmans [13] used NE method to arrive at the dynamic model of a mobile manipulator that has two links mounted on a mobile platform. Chung [14] used EL method to arrive at the equations of motion for a mobile manipulator. Geometric mechanics was used to adapt Luh and Walker [15] algorithm by Boyer and Ali [16] 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 constraints 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 [17] 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 [18] and Saha in [19],[20] to derive the equations of motion for a WMR.

## 2.3 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* [21], *follow the carrot* [22], *vector pursuit* [23], and *follow the past* [24]. In pure-pursuit [21], 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 6. 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 [22], the steering angle is set proportional to the *orientation error* defined in Figure 2.3. 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. Witin [23] 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

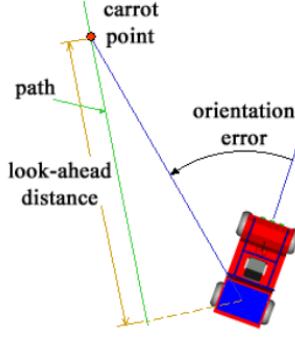


Figure 2.3: Follow the carrot

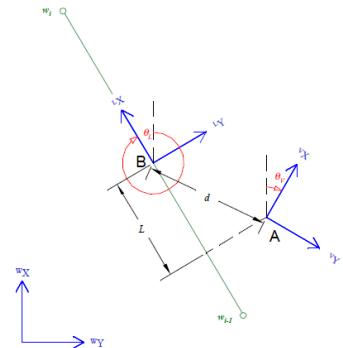


Figure 2.4: Vector Pursuit

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 2.4. Control is then generated proportional to this error.

Hellstrom [24] has proposed an algorithm which uses the knowledge of previously 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 [25] 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.

## 2.4 Tele-operation

Tele-operation deals with connection 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 [26, 27], 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.

#### **2.4.1 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 [28] was developed at MIT during 1994 to provide 3-DOF force feedback for touch feedback purpose. DELTA Haptice Device described in [29] provides 6-DOF force feedback with moderate force. Clover [30] 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 [31]. Design of a 6-DOF parallel mechanism for force feedback is discussed in [32].

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 [33, 34], head-mounted displays [34] or polarized images [35]. 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.

#### **2.4.2 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 [36], 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 [37] 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 [38], 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 [39]. 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 [40], Rebelo [41] and Anderson and Slotin [42]. Port-Hamiltonian based approach was used in [43, 44]. Design of controller for time delayed systems based on back-stepping method in combination with partial differential heat equation was studied by Krstic [45].

## 2.5 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 [46]. 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, *su-*

*pervisory control or tele-assistance* and *predictive display*. In *supervisory control* [47, 48, 49] 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 looses 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 [50], Bejczy [51] and Kim [52]. Whereas the above two used a-prior modeling and calibration of remote environment, Jagersand [53] 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,  $x_1, x_2.....x_k$ . Uncalibrated monocular camera mounted on manipulator (eye-in-hand) based image predication method was discussed by Yeres [54] and Deng [55]. Multiple sensors based dense 3-D map of a remote scene was reported by Kelly [56] and [57]. While Kelly used fusion of lidar and camera, Burkert used stereo cameras. Hu [58] has used SLAM based Predictive Dispaly (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.

## 2.6 Summary

Different literatures relevant to the topic addressed in this research were presented. Each topic is a separate area of research in its self. There are many more literatures in each topic which could be of great interest but was been skipped due to space constraints.

# Chapter 3

## Design of Mobile a Robot

Most of the mobile robots presented in literature uses differential wheel drive with passive castor, as in [59], [60] and [19]. The other common methods for locomotion of mobile robots are the omnidirectional wheels [61] and [62], and tracked wheel system [63] and [64]. According to Nagatani [65], 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. We also highlight the advantage of the Davis steering mechanism over castor wheels or other steering methods from the perspective of this 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 [66].

### 3.1 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

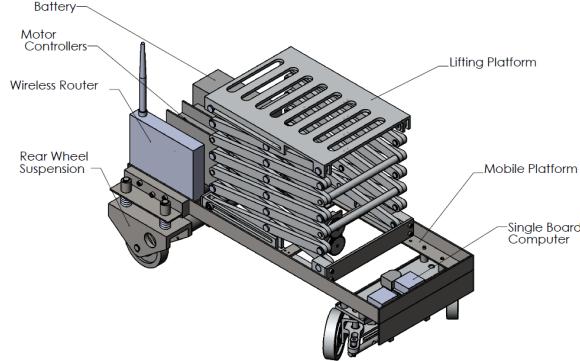


Figure 3.1: 3-D Model of the mobile ma-  
nipulator

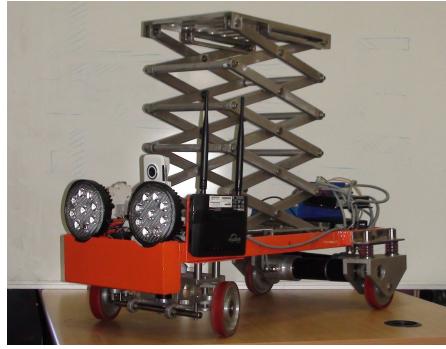


Figure 3.2: Photograph of the actual Sys-  
tem

requirement, a mobile platform with a vertically extendable manipulating arm was developed as shown in Figures 3.1 and 3.2. Together, they are referred henceforth as mobile-manipulator. The 3-D model of the mobile-manipulator with its major subsystems are shown in Figure 3.1, whereas and the actual system is shown in Figure 3.2.

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 by the on board camera. The key parameters of the mobile manipulator are listed in Table 3.1.

Table 3.1: 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

The mobile-manipulator has a footprint of 700 mm x 400 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 with a Davis steering mechanism [67]. A scissor mechanism provides the vertical motion of 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.

## 3.2 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.

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, as shown in Figure 3.3. 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 resistance to wear.

The load distribution was optimized to generate maximum normal reaction,  $F_n$ , at the rear wheels without overturning while moving up the ramp of  $30^\circ$ . Maximizing rear wheel reaction by increasing "b" as per Equation 3.1 ensures increased traction,  $F_T = \mu F_N$  ( $\mu$  is the coefficient of friction), but at the same time decreases the stability margin indicated by "X" in Figure 3.4. The static moment and force balance yield

$$F_{N2} = \frac{b}{a} F_{N1}, \quad F_{N1} + F_{N2} = mg \cos \theta \quad (3.1)$$

The stability margin  $X$  as shown in Figure 3.2 was fixed as 30mm so as to achieve maximum acceleration of  $0.144g$  over the ramp of  $30^\circ$  without overturning. This was done based on the condition of dynamic stability given by

$$mgb \cos \theta = (mg \sin \theta + ma)z_{cg}, \quad \Rightarrow g\left(\frac{b}{z_{cg}} \cos \theta - \sin \theta\right) = a \quad (3.2)$$

Where

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

a and b are the distance of the vehicle cg from the rear and front wheels.

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

$m$  is mass of the vehicle.

$g$  is acceleration due to gravity.

$\theta$  is the inclination of the traction surface from horizontal.

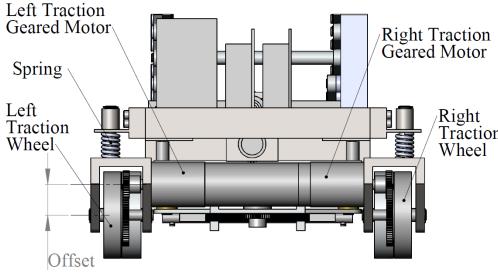


Figure 3.3: Rear Suspension

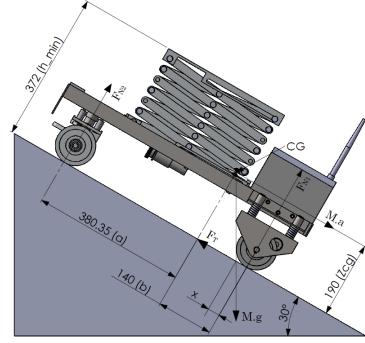


Figure 3.4: Mobile Manipulator on slope

### 3.2.1 Selection of Motor and Gearbox

The torque requirement for each rear wheel was 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 3.4), using moment and force balance we get the following:

$$F_{N1} = \frac{aM \cos \theta}{a + b - \mu Z_{cg}}, \quad F_T = \mu F_{N1} \quad (3.3)$$

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

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

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

$$T_w = (F_T/2)(D_w/2) = (20/2) * 50 = 500Kg-mm \simeq 5Nm \quad (3.4)$$

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

$$T_M = (FS) * T_w = 1.5 * 5 = 7.5 \simeq 8Nm \quad (3.5)$$

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,

$$\begin{aligned}\omega_w &= V_{ramp}/(D_w/2) \simeq 200\text{rpm} \\ P_m &= \omega_w T_m = 20 * 8 = 160W\end{aligned}\tag{3.6}$$

The nearest Maxon motor available as per the catalogue [68] 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.

### 3.3 Design of Steering System

The design objective of a steering system should ensure rolling motion of all the wheels during every possible manurers of the mobile robot. This is important to reduce the friction drag due to sliding motion which reduces the energy efficiency of the mobile robot. In case of four wheeled vehicle with rear wheels fixed and front wheels steered the condition is shown in Figure 3.5, referred in some literature as Ackerman steering conduction.

This mobile manipulator uses Davis steering mechanism, Figure 3.6, 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 operation are 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 3.7, which ensures pure rolling of all wheels over the entire steering range. This makes the system suitable for passive wheel

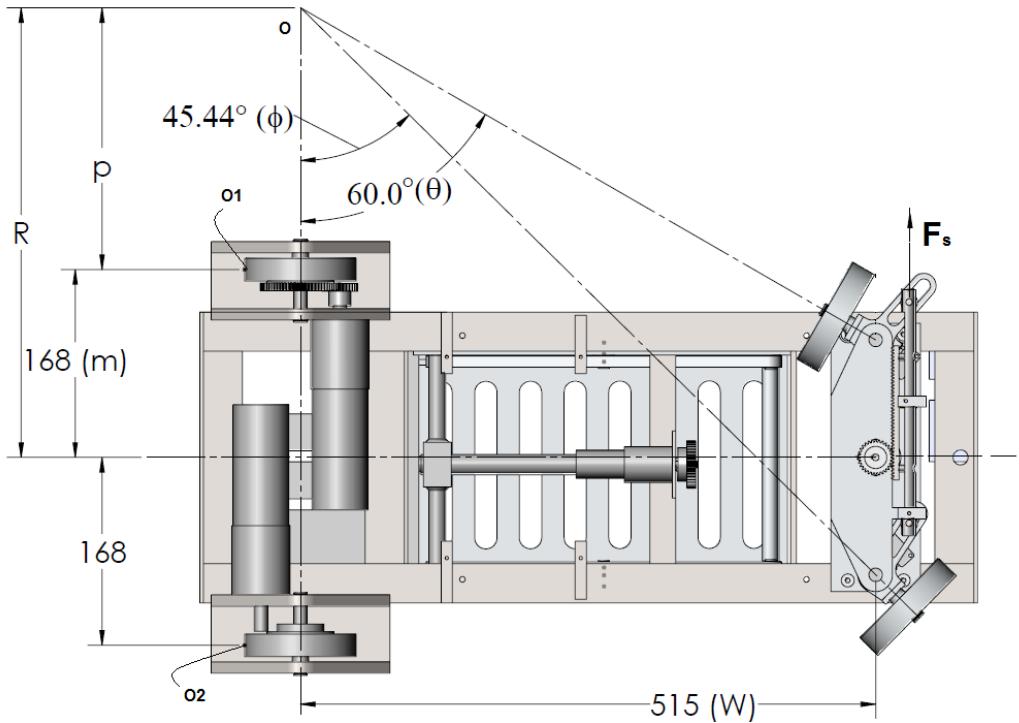


Figure 3.5: Ackerman Steering Condition

odometry and energy efficient. The mechanism 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 3.6, and are based on the Ackerman steering law given below:

$$\cot \phi - \cot \theta = a/w, \quad \frac{2b}{h} = \frac{a}{w} \quad (3.7)$$

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

### 3.3.1 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. Figure 3.5 shows the extreme values of  $\phi$  and  $\theta$ , one side of the steering limits. The **turning radius**  $R$  for a given steer angle  $\theta$  is calculated by the geometry of Figure 3.6 as

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

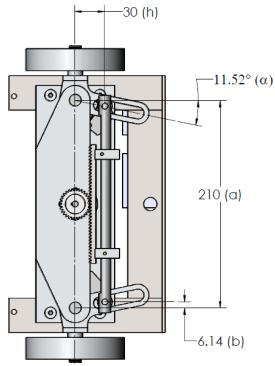


Figure 3.6: Davis Steering Gear

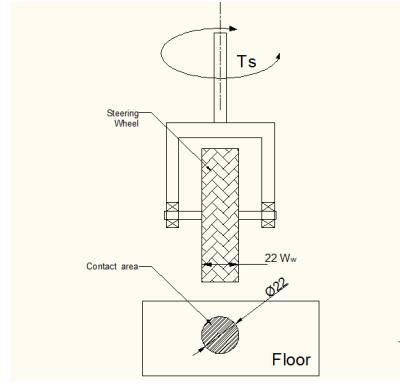


Figure 3.7: Steer Torque

Substituting  $R$  and rearranging the above equation, we get

$$R(\theta) = \frac{a}{2} + w \cot \theta \quad (3.8)$$

The extreme value of  $\theta = 60^\circ$ , as shown in Figure 3.5, this gives the *minimum turning radius*  $R_{min}$  as

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

### 3.3.2 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 3.7. In order to estimate the normal reaction on each wheel, we assume that the total weight of 80kg is 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.,

$$T_s = \frac{N_s \mu}{3} W_w = 0.4Nm \quad (3.9)$$

The resistance torque,  $T_s$ , of both the wheels are balanced by the force  $F_s$  acting on the rack as shown in Figure 3.6. The rack is coupled to the steering motor by a pinion of diameter,  $D_p = 40mm$ . The motor torque,  $T_{m_s}$  in Equation 3.10 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  $\omega_s$  of 100rpm is

$$T_{m_s} = (FS) \frac{2T_s D_p}{h} = 1.6Nm \quad \text{and} \quad P_{m_s} = T_{m_s} * \omega_s = 17W \quad (3.10)$$

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.

### 3.4 Design of Scissor Mechanism for Manipulating Arm

The manipulating arm was designed to move up to a height of 1.5m from the floor level. This motion was generated using a scissor mechanism, as shown in Figure 3.8. 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 3.8, has 6 stages, where one "X" denotes one stage. The Scissor is connected to the top platform by a pivot joint  $O_2$  and a prismatic joint  $A_2$ . This is 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

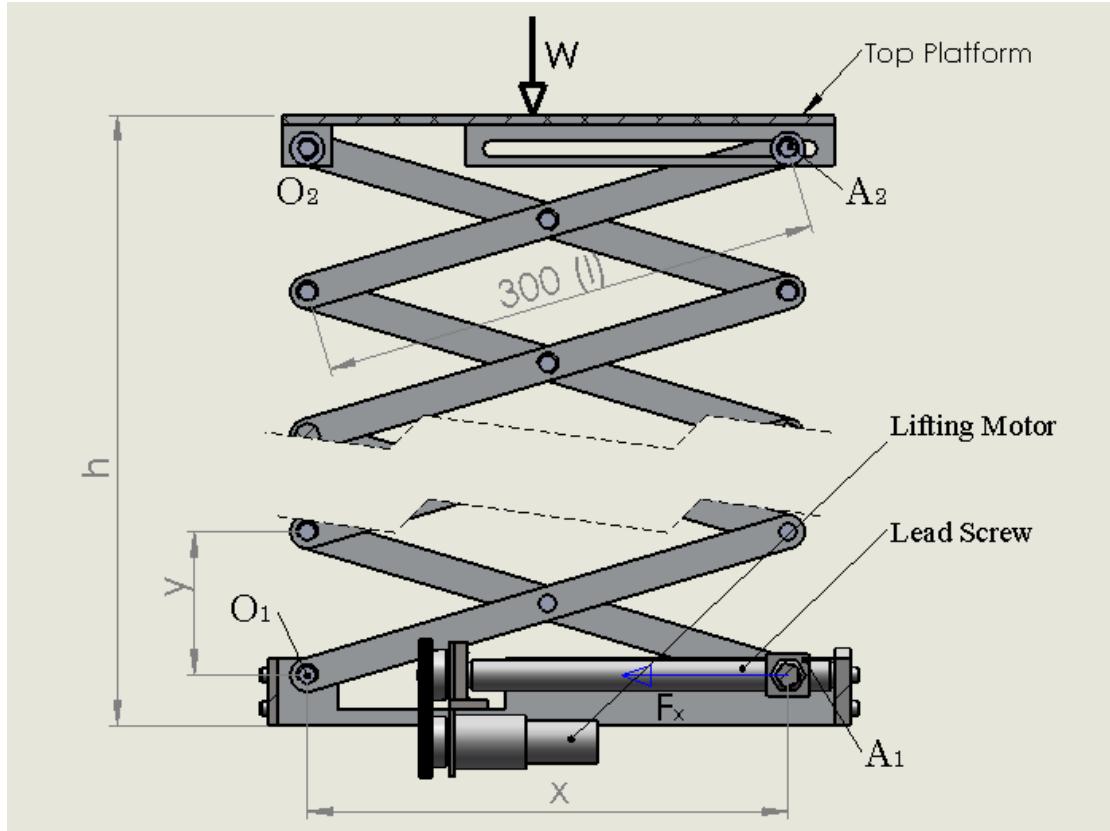


Figure 3.8: Scissor Mechanism

3.8

$$y = l \sin \theta, \quad x = l \cos \theta, \Rightarrow dy = l \cos \theta d\theta, \quad dx = -l \sin \theta d\theta \quad (3.11)$$

$$h = Ny \rightarrow dh = Ndy$$

where  $l$  is the link length,  $\theta$  the angle of the link with horizontal plane and  $h$  the height of platform.

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

$$-F_x dx = W dh, \Rightarrow F_x = \frac{WN}{\tan \theta} \quad (3.12)$$

where,  $F_x$  is the axial force on the prismatic joint,  $A_1$  and  $W$  is the payload. From Equation 3.12, 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  $h_{max} = Nl \sin \theta_{max} = 1.3m$  and the collapsed height  $h_{min} = 156mm$ . Assuming  $W = 8kg$  as payload the maximum force  $F_x = 342Kg$  is required at  $\theta_{min} = 5^\circ$ .

The motor torque required for the scissor mechanism is calculated using screw jack formula given in [67] and here presented as Equation 3.13.

$$T_L = \frac{F_x d_m}{2} \left( \frac{p + \pi \mu d_m \sec \alpha}{\pi d_m - \mu p \sec \alpha} \right) = 7.5 Nm \quad (3.13)$$

where coefficient of friction assumed,  $\mu = 0.1$ , ACME thread angle  $2\alpha = 60^\circ$ , pitch diameter  $d_m = 15mm$  and pitch  $p = 1.5mm$  is used. Based on the above specification 10W RE20 DC motor with a gear box of 25:1 ratio was chosen from Maxon motor catalogue [68].

### 3.5 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 4

## 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-wheeled robots, legged robots [69], and aerial vehicles [70]. There are few mobile robots which use both wheels and legs for locomotion for example, in Creadapt [71] 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. [11], [72] based on their mobility resulting from the kinematic constraints due to different wheel types. The most common among these are the 3 wheeled differential drive WMR with one castor wheel. Because of its simplicity in modelling, they have been used in most of the control and motion planning algorithms [73], [74] and [75].

In order to develop a model-based control algorithm it is imperative to have 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 [76] using Lagrange formu-

lation. Alternatively Thanjavur and Rajagopalan [77] has used Kane's method and Saha et al. [20],[19] used Natural Orthogonal Compliment (NOC) method for the same.

All the papers known to us use the standard castor wheel in deriving the dynamic model of the WMR. In this chapter the modelling is carried out considering the Castor wheel in the most general configuration. In the stranded castor wheel configuration the axis of rotation is perpendicular to the line joining the pivot and the axis of rotation, as shown in Figure 4.1 whereas in this study they are oriented in the most general way, as shown in figure Figure 4.2. The concept of the NOC which here 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.

## 4.1 Modeling using the Natural Orthogonal Compliment (NOC)

Let us consider a system  $n$  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  $\omega_i$  is the angular velocity of the same body, then equation of motion of the  $i-th$  rigid body is given by Newton-Euler equations as

$$f_i = m_i \ddot{c}_i \quad \text{and} \quad n_i = I_{ci} \omega_i + \omega_i \times I_{ci} \omega_i \quad (4.1)$$

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 forces and torques, which imparts/extracts energy to/from the system, e.g. motor actuating torque. The non-working component of the wrench consists of the forces and torques that are used to constrain the motion of the body at the joints. Then, Newton-Euler equation (4.1) can be rewritten in a single matrix equation as

$$M_i \dot{t}_i + W_i M_i t_i = w_i^w + w_i^c \quad \because w_i \equiv w_i^w + w_i^c \quad (4.2)$$

where

$$M_i \equiv \begin{pmatrix} I_{ci} & 0 \\ 0 & m_i \tilde{1} \end{pmatrix}, \quad W_i \equiv \begin{pmatrix} \Omega_i & 0 \\ 0 & 0 \end{pmatrix}, \quad \Omega_i \equiv \omega_i \times \tilde{1} \quad (4.3)$$

in which  $\omega_i \times 1$ , is referred as the crossproduct matrix of vector  $\omega_i$  and  $\tilde{1}$  denotes the identity matrix. For details, refer to [18],[78].

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

$$M \dot{t} + W M t = w^c + w^w \quad (4.4)$$

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 linear homogeneous

system of algebraic equations [18], namely

$$A_i t_i + A_j t_j = 0 \quad (4.5)$$

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,  $\dot{\theta} \equiv (\dot{\theta}_1, \dot{\theta}_2, \dots)^T$  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

$$At = 0, \Rightarrow AT\dot{\theta} = 0 \Rightarrow AT = 0 \quad (4.6)$$

The above equation (4.6) indicates that  $T$  is the orthogonal complement of  $A$ . Since this relation arises naturally, hence the name *Natural Orthogonal Complement*. It can be shown [18] that the non-working wrench  $w^c$  lies in the range space of  $A^T$ . In view of equation 4.6, it can also be proved that  $w^c$  lies in the null space of  $T^T$ , therefore

$$T^T w^c = 0 \quad (4.7)$$

To eliminate the non-working forces and moments, i.e.  $w^C$  from the uncoupled equation of motion (4.4), we multiply both sides of the equation by  $T^T$ ,

$$T^T M \dot{t} + T^T W M t = T^T w^W, \Rightarrow T^T M T \ddot{\theta} + T^T (M \dot{T} + W M T) \dot{\theta} = T^T w^T \quad (4.8)$$

Equation 4.8 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 4.8 the final equations of motion can be written as

$$I(\theta)\ddot{\theta} = C(\theta, \dot{\theta})\dot{\theta} + \tau \quad (4.9)$$

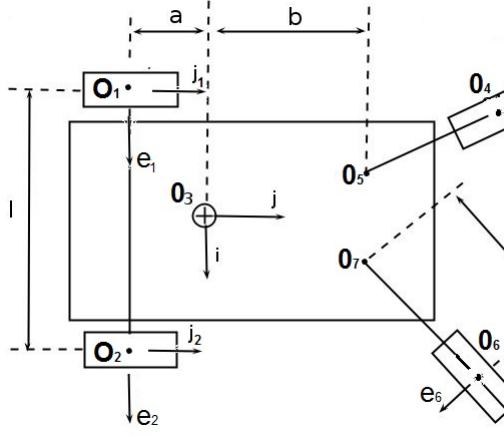


Figure 4.1: WMR-Std. Castor

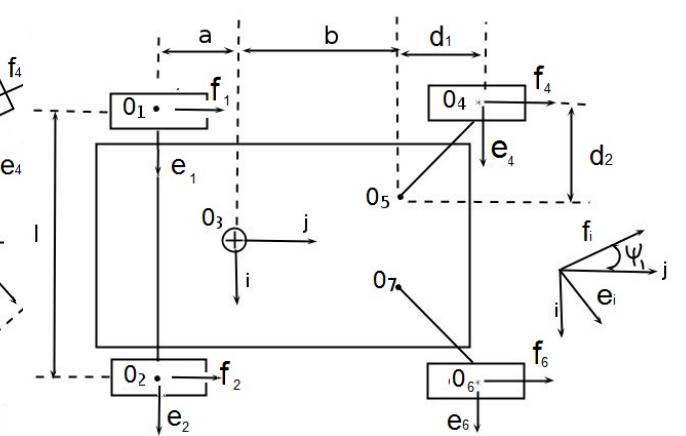


Figure 4.2: WMR-general

Figure 4.3: Castor wheel configuration of a WMR

where

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

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

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

## 4.2 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 [20]. The vehicle consisted of 2 driven wheel and one standard caster wheels. In general, for large vehicles, it necessary to have at least four wheels from the point of stability of the vehicle. Such a vehicle is shown in figure 4.1. It may be noted that the caster wheels used in this case are the standard caster wheel configuration, where the angle between line  $O_4O_5$  and vector  $e_4$  is 90 deg. Which is a special case of the more general configuration of the caster wheels shown in figure 4.2, where angle between line  $O_4O_5$  and vector  $e_4$  is not 90 deg

The vehicle considered for analysis in this chapter is shown in figure 4.2. It consists of two independently driven wheels at the back and two generalized caster

wheels at the front. The actuated wheels are labelled as body #1 and #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 is labelled as #6 an #7 respectively, all the wheels are assumed to be rolling without slipping.

#### 4.2.1 Kinematic analysis

In order to proceed with the kinematic analysis of the vehicle in figure 4.2, we define a orthogonal triad of vectors  $i, j, k$  at point  $O_3$ , the control point of the platform, as shown in the figure. If  $\dot{\theta}_1$  and  $\dot{\theta}_2$ <sup>1</sup> denote the rates of rotation of wheel #1 and #2 then the linear velocity of points  $O_1$  &  $O_2$  under pure rolling condition is given by

$$\dot{\mathbf{o}}_i = r\dot{\theta}_i j, \quad r=\text{radius of wheel} \quad (4.10)$$

The angular velocity of the platform  $\omega_3$  can be written as

$$\dot{\boldsymbol{\omega}}_3 = (r/l)(\dot{\theta}_1 - \dot{\theta}_2)k \quad (4.11)$$

Further, the velocity of point  $O_3$  can be written as  $\dot{\mathbf{o}}_3 = \dot{\mathbf{o}}_i + \boldsymbol{\omega}_3 \times (\mathbf{c} - \mathbf{o}_i)$ ,  $i = 1, 2$ . 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. By eliminate  $\omega_3$  we get the following:

$$\dot{\mathbf{o}}_3 = [(ar/l)(\dot{\theta}_1 - \dot{\theta}_2) + (r/2)(\dot{\theta}_1 + \dot{\theta}_2)]i \quad (4.12)$$

Now, the angular velocity of the drive wheel #1 can be expressed as  $\boldsymbol{\omega}_1 = -\dot{\theta}_1 i + \omega_3 k$ . Using equation 4.11, the same can be rewritten as

$$\boldsymbol{\omega}_1 = \begin{pmatrix} -i + (r/l)k & -(r/l)k \end{pmatrix} \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix} \quad (4.13)$$

---

<sup>1</sup> positive when pointing along  $i$

Based on equations 4.10 and 4.13, we can write the twist for wheel #1 in terms of  $\dot{\theta}_a \equiv (\dot{\theta}_1, \dot{\theta}_2)^T$ , as

$$t_1 = \begin{pmatrix} \omega_1 \\ \dot{o}_1 \end{pmatrix} = \begin{pmatrix} -i + (r/l)k & -(r/l)k \\ rj & 0 \end{pmatrix} \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix} \quad (4.14)$$

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

$$t_2 = \begin{pmatrix} \omega_1 \\ \dot{o}_1 \end{pmatrix} = \begin{pmatrix} -i + (r/l)k & -(r/l)k \\ 0 & rj \end{pmatrix} \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix} \quad (4.15)$$

To calculate the twist,  $t_3$  of the platform body #3, equation 4.11 and 4.12 are combined to get

$$t_3 = \begin{pmatrix} \omega_3 \\ \dot{o}_3 \end{pmatrix} = \begin{pmatrix} \rho\delta & -\rho\delta \\ r(\lambda i + (1/2)j) & r(-\lambda i + (1/2)j) \end{pmatrix} \begin{pmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix} \quad (4.16)$$

where

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

From the above three equations we get  $T_1 = \begin{pmatrix} -i + (r/l)k & -(r/l)k \\ rj & 0 \end{pmatrix}$ ,  $T_2 = \begin{pmatrix} -i + (r/l)k & -(r/l)k \\ 0 & rj \end{pmatrix}$ ,  $T_3 = \begin{pmatrix} \rho\delta & -\rho\delta \\ r(\lambda i + (1/2)j) & r(-\lambda i + (1/2)j) \end{pmatrix}$

In order to calculate the twist of the caster bracket and the caster wheel, we need to express the unactuated 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 bracket body #5 about  $O_5$  with respect to the platform, and  $\dot{\phi}_1$  the rate of rotation of caster wheel body #4 about its axis  $e_4$  with respect to bracket. 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  $O_5$ , i.e.,

$$\dot{o}_5 = \dot{o}_4 + \omega_5 \times (d_1 e_4 - d_2 f_4), \quad \dot{o}_5 = \dot{o}_4 + \omega_5 \times (d_1 e_4 - d_2 f_4) \quad (4.17)$$

On equating the above two equations together, and using the rotation matrix between coordinate system  $\{i, j, k\}$  and  $\{e_4, f_4, k\}$ , to express the equation in  $e_4$  and  $f_4$ , we get

$$(-\dot{\phi}_1 r + \dot{\psi}_1 d_1) f_3 + d_3 \dot{\psi}_1 e_3 = \dot{o}_3 + \omega_3(m \cos \psi_1 - b \sin \psi_1 - d_1) e_4 \quad (4.18)$$

Taking the dot product of the above equation first with  $e_4$  and then with  $f_4$ , and using equation 4.12 for  $\dot{o}_3$ , we get

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

$$= [F_{ij}] \dot{\theta}_a \quad (4.19)$$

where,

$$\delta_1 = (r/l)(mC_{\psi_1} - bS_{\psi_1} - d_2), \quad \delta_2 = (r/l)(mS_{\psi_1} + bC_{\psi_1} + d_1)$$

Similarly, for the other caster wheel we get,

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

$$= [G_{ij}] \dot{\theta}_a \quad (4.20)$$

where,

$$\delta_3 = (r/l)(mC_{\psi_2} + bS_{\psi_2} + d_2), \quad \delta_4 = (r/l)(mS_{\psi_2} + bC_{\psi_2} + d_1)$$

The angular and the linear 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

$$\omega_4 = \dot{\phi}_1 e_4 + (\omega_3 + \dot{\psi}_1) k, \quad \dot{o}_4 = \dot{\phi}_1 e_4 \quad (4.21)$$

Using equations 4.19 and 4.11, the twist  $t_4$  can be written as

$$t_4 = \begin{pmatrix} \Theta_4 \\ C_4 \end{pmatrix} \dot{\theta}_a \quad (4.22)$$

Using the definition of  $F(i, j)$  in equation 4.19,  $\Theta_4$  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 by

$$\omega_4 = \dot{\phi}_1 e_3 + \dot{\psi}_1 k, \quad \dot{o}_4 = \dot{o}_4 + \omega_5 \times [-df_3] \quad (4.23)$$

Using equations 4.11 & 4.20, the twist  $t_5$  can be written as

$$t_5 = \begin{pmatrix} \Theta_5 \\ C_5 \end{pmatrix} \dot{\theta}_a \quad (4.24)$$

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 twist  $t_6$  and  $t_7$  of the other caster wheel and its bracket can be written as

$$t_6 = \begin{pmatrix} \Theta_6 \\ C_6 \end{pmatrix} \dot{\theta}_a, \quad t_7 = \begin{pmatrix} \Theta_7 \\ C_7 \end{pmatrix} \dot{\theta}_a \quad (4.25)$$

Using  $G(i, j)$  defined in equation 4.20,  $\Theta_6$ ,  $C_6$ ,  $\Theta_7$  and  $C_7$  can be written as

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

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

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

### 4.2.2 Dynamic equations

Based on the twist calculated in terms of the independent joint rate vector vector  $\dot{\theta}_a$ , we now derive the generalized inertia matrix and the matrix of convective inertia term for the coupled equation of motion 4.9.

#### 4.2.2.1 Generalized Inertia Matrix, $I$

The above equations gives the twist of individual body, i.e  $t_i = T_i\dot{\theta}_a$ . As defined earlier  $t = [t_1^T, t_2^T \dots t_7^T]^T$  and  $t = T\theta$  we get

$$T = [T_1^T, T_2^T, \dots T_7^T]^T$$

Since the matrix  $M$  is block diagonal, the inertia matrix of the full system is given by

$$I = T^T M T = T_1^T M_1 T_1 + T_2^T M_2 T_2 + \dots T_7^T M_7 T_7 \quad (4.26)$$

We define the contribution of rear wheels alone to the inertia matrix by  $I_m$  as

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

then

$$I_m = \begin{pmatrix} I_w + (\rho\delta)^2 H + m_w r^2 & -2(\rho\delta)^2 H \\ -2(\rho\delta)^2 H & I + (\rho\delta)^2 H + m_w r^2 \end{pmatrix} \quad (4.27)$$

where  $M_i \equiv \begin{pmatrix} \tilde{I}_w & 0 \\ 0 & m_w \mathbf{1} \end{pmatrix}$  and  $\tilde{I}_w \equiv \begin{pmatrix} I_w & 0 & 0 \\ 0 & H & \\ 0 & 0 & H \end{pmatrix}$ . Matrix  $\tilde{I}_w$  is the  $3 \times 3$  moment

of inertia matrix of the wheel in co-ordinate frame  $\{i, j, k\}$ , mass of the motorized wheels is  $m_w$  and  $m\mathbf{1}$  is the  $3 \times 3$  identity matrix. If the mass of the platform is

$m_p$  and its moment of inertia about vector  $k$  is  $I_p$ , then as derived in [18]

$$I_3 = T_3^T M_3 T_3 = I_p(\rho\delta)^2 \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} + m_p r^2 \begin{pmatrix} (1/4) + \gamma^2 & (1/4) - \gamma^2 \\ (1/4) - \gamma^2 & (1/4) + \gamma^2 \end{pmatrix} \quad (4.28)$$

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

$$\begin{aligned} I_c = \sum_{i=4,6} T_i^T M_i T_i &= (m_c r^2 / 4) \begin{pmatrix} 6F_{11}^2 + \bar{F}_{21}^2 & 6F_{11}F_{12} + \bar{F}_{21}\bar{F}_{22} \\ 6F_{11}F_{12} + \bar{F}_{21}\bar{F}_{22} & 6F_{12}^2 \bar{F}_2^2 \end{pmatrix} \\ &\quad + \begin{pmatrix} 6G_{11}^2 + \bar{G}_{21}^2 & 6G_{11}G_{12} + \bar{G}_{21}\bar{G}_{22} \\ 6G_{11}G_{12} + \bar{G}_{21}\bar{G}_{22} & 6G_{12}^2 \bar{G}_2^2 \end{pmatrix} \end{aligned} \quad (4.29)$$

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

#### 4.2.2.2 Matrix of Convective Inertia term $C$

The matrix of convective inertia terms of equation 4.9 can be broken down into two parts,  $T^T M \dot{T}$  and  $T^T W M T$ . As given in equations 4.14, 4.15 and 4.16  $T_1, T_2, T_3$  of the rear wheels and the platform is constant. Therefore  $T^T M \dot{T} = 0$ . The generalized inertia matrix too is constant for the rear wheels and the platform, so the vector  $I_i \omega_i$  is parallel to vector  $\omega_i, i = 1..3 \rightarrow \omega \times I\omega = 0 \rightarrow T^T W M T = 0$ . This shows that contribution of the rear wheels and the platform to the convective inertia term is zero. Moreover we have considered the mass of the brackets to be zero, so they also do not contribute to the convective inertia term. Hence

$$C = T^T M \dot{T} + T^T W M T = \sum_{i=4,6} T_i^T M_i \dot{T}_i + \sum_{i=4,6} T_i^T W_i M_i T_i \quad (4.30)$$

The expression for the first term is found by using equations 4.19, 4.20, 4.22 and 4.25. The terms  $\dot{F}_{ij}$  and  $\dot{G}_{ij}$  denote the derivatives of the elements of the matrix  $F$

and  $G$  defined in 4.19, 4.20 . To find  $\dot{T}_4$  and  $\dot{T}_6$  we have used the fact  $\dot{e}_4 = \omega_4 \times e_4$  and  $\dot{e}_4 = \omega_6 \times e_6$ .

$$T^T M \dot{T} = (m_c r^2 / 4) \left[ \begin{pmatrix} 6F_{11}\dot{F}_{11} + \bar{F}_{21}\dot{F}_{21} & 6F_{11}\dot{F}_{12} + \bar{F}_{21}\dot{F}_{22} \\ 6\dot{F}_{11}F_{12} + \dot{F}_{21}\bar{F}_{22} & 6F_{12}^2\bar{F}_{22} \end{pmatrix} \right. \\ \left. + \begin{pmatrix} 6G_{11}\dot{G}_{11} + \bar{G}_{21}\dot{G}_{21} & 6G_{11}\dot{G}_{12} + \bar{G}_{21}\dot{G}_{22} \\ 6\dot{G}_{11}G_{12} + \dot{G}_{21}\bar{G}_{22} & 6G_{12}^2\bar{G}_{22} \end{pmatrix} \right] \quad (4.31)$$

The second term of equation 4.30 i.e,  $\sum_{i=4,6} T_i^T W_i M_i T_i$  evaluates to zero, as shown below. Consider castor wheel (body #4). Using equation 4.22 and  $W$  defined in the equation 4.3 we get,

$$T_4^T W_4 M_4 T_4 = [\Theta_4, C_4] \begin{pmatrix} \Omega_4 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} I_4 & 0 \\ 0 & m_4 \mathbf{1} \end{pmatrix} \begin{pmatrix} \Theta_4 \\ C_4 \end{pmatrix} = \Theta_4 \Omega_4 I_4 \Theta_4 \quad (4.32)$$

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  $\Theta_4$  from equation 4.22 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 cros product matrix of  $\omega_4$  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 4.32, we get

$$T_4^T W_4 M_4 T_4 = 0, \Rightarrow T^T W M T = 0 \quad (4.33)$$

So the matrix of convective inertia term  $C$  of equation 4.9 is evaluated as

$$C = (m_c r^2 / 4) \left[ \begin{pmatrix} 6F_{11}\dot{F}_{11} + \bar{F}_{21}\dot{F}_{21} & 6F_{11}\dot{F}_{12} + \bar{F}_{21}\dot{F}_{22} \\ 6\dot{F}_{11}F_{12} + \dot{F}_{21}\bar{F}_{22} & 6F_{12}^2\bar{F}_{22} \end{pmatrix} + \begin{pmatrix} 6G_{11}\dot{G}_{11} + \bar{G}_{21}\dot{G}_{21} & 6G_{11}\dot{G}_{12} + \bar{G}_{21}\dot{G}_{22} \\ 6\dot{G}_{11}G_{12} + \dot{G}_{21}\bar{G}_{22} & 6G_{12}^2\bar{G}_{22} \end{pmatrix} \right] \quad (4.34)$$

All the components of equation 4.9 have now been evaluated, except the  $\tau$ . These are simply the torques exerted by the actuated wheels. This completes the dynamic model of the WMR with generalized caster wheel configuration.

## 4.3 Special cases

### 4.3.1 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 4.19 and 4.20 become a diagonal matrix. Therefore, the first and second equation of 4.19 and 4.20 get divided by  $d_2$  and  $r$ , respectively. The resulting equations relating the unactuated joint rate to actuated joint rates are similar to those reported by [20],[18].

### 4.3.2 Under Actuated Case ( $d_2 = 0$ )

It can be seen from equation 4.19 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.

Another solution is to put an extra actuator to control the bracket motion by controlling  $\psi_i$ . As in the case of Ackerman steering mechanism, where the steering wheel controls the orientation of the front passive wheels of a car.

## 4.4 Simulation

The mobile manipulator proposed in chapter 3 was modelled as a differential drive 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, (point  $O_3$ ), is required to trace a circle of radius 5m. As shown in the figure 4.4.  $\beta$  is the angle between the line joining point  $O_3$  and the origin  $O$  with respect to  $X - axis$ . The function  $\beta(t)$  is defined such that the ful circle is completed in 60Sec.

$$\beta(t) = \frac{20\pi}{60^3}t^3 - \frac{30\pi}{60^4}t^4 + \frac{12\pi}{60^5}t^5 \quad (4.35)$$

Moreover the velocity and acceleration of the robot is zero at the beginning ( $t=0$ ) and end of travel ( $t=60$ ). The initial pose of the vehicle is parallel to the  $Y - axis$  i.e.  $\beta = 0$  shown by dotted line in figure 4.4.

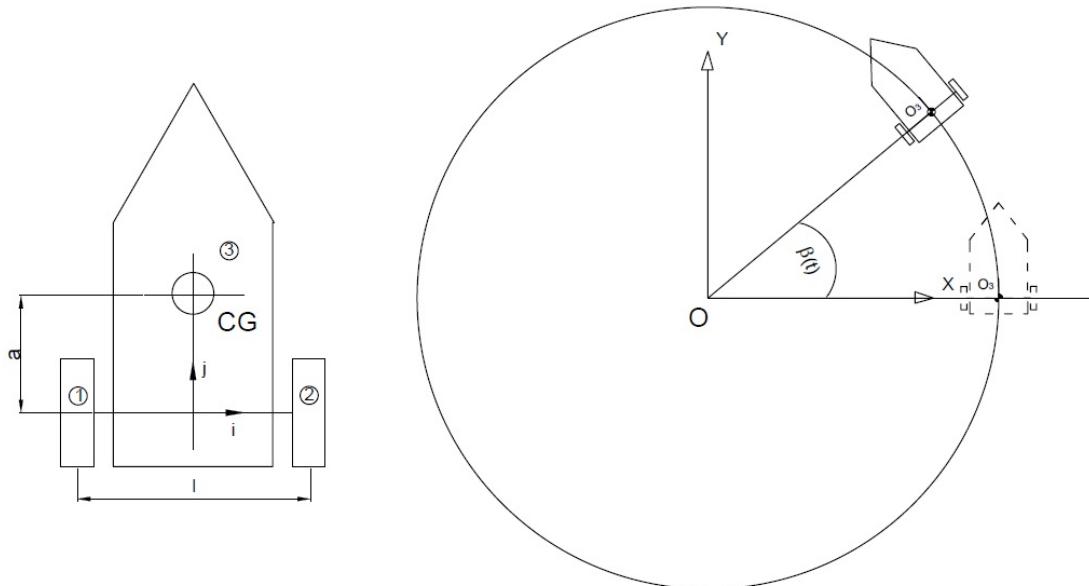


Figure 4.4: Path Traced by Robot

#### 4.4.1 Inverse Dynamics

Using the model inverse kinematic, the wheel velocity and acceleration are determined using pseudo Inverse The wheel angle, velocity and acceleration were used in the dynamics equation to calculate the torque required by each motor. The results are plotted in figure 4.5.

The equation of dynamic model is given in equation 4.36, where the actuated joints are  $\theta_a(t) = (\theta_l, \theta_r)$  are the rear left and right wheel rotation angles. The dynamic and kinematic parameters used in the simulation is listed in table 4.1. The rear wheels ( $i = 1, 2$ ) and the platform  $i = 3$ , twist  $t_i^T = T_i\theta_a$ , is given in equations 4.14, 4.15 and 4.16. The dynamic equation is of the vehicle on slope is given as

$$T^T M T \ddot{\theta}_a = -T^T(M \dot{T} + W M T)\dot{\theta}_a + T^T(w^J + w^G)$$

where  $T = (T_1^T T_2^T T_3^T)^T$ ,  $M = \text{diag}(M_1, M_2, M_3)$

$$T^T M T = I_1 + I_2 + I_m + I_3 = I_m + I_3 \quad (4.36)$$

$$W = \text{diag}(W_1, W_2, W_3), \quad w^G = g \sin \alpha, \quad \alpha = 10^\circ$$

$w^G$ ,  $w^J$  and  $\alpha$  represent the gravitational force acting along the inclined the external torque/Force 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 4.27 and 4.28. As stated earlier the convective inertia term is zeros a the inertia matrices are constant. The motor torque are then given a

$$T_1^T w_1^j = \tau_1, \quad T_2^T w_2^j = \tau_2 \quad (4.37)$$

$$T_3^T w^g = \begin{pmatrix} \rho \delta k & r(\lambda i + (1/2)j) \\ -\rho \delta k & r(-\lambda i + (1/2)j) \end{pmatrix} \begin{pmatrix} 0 \\ -j g \sin \alpha \end{pmatrix} = \begin{pmatrix} -\frac{1}{2} r g \sin \alpha \\ -\frac{1}{2} r g \sin \alpha \end{pmatrix} \quad (4.38)$$

The torques required at the wheels of the vehicle to move up a spiral ramp of slope  $10^\circ$  and radius 5m is given figure 4.5.

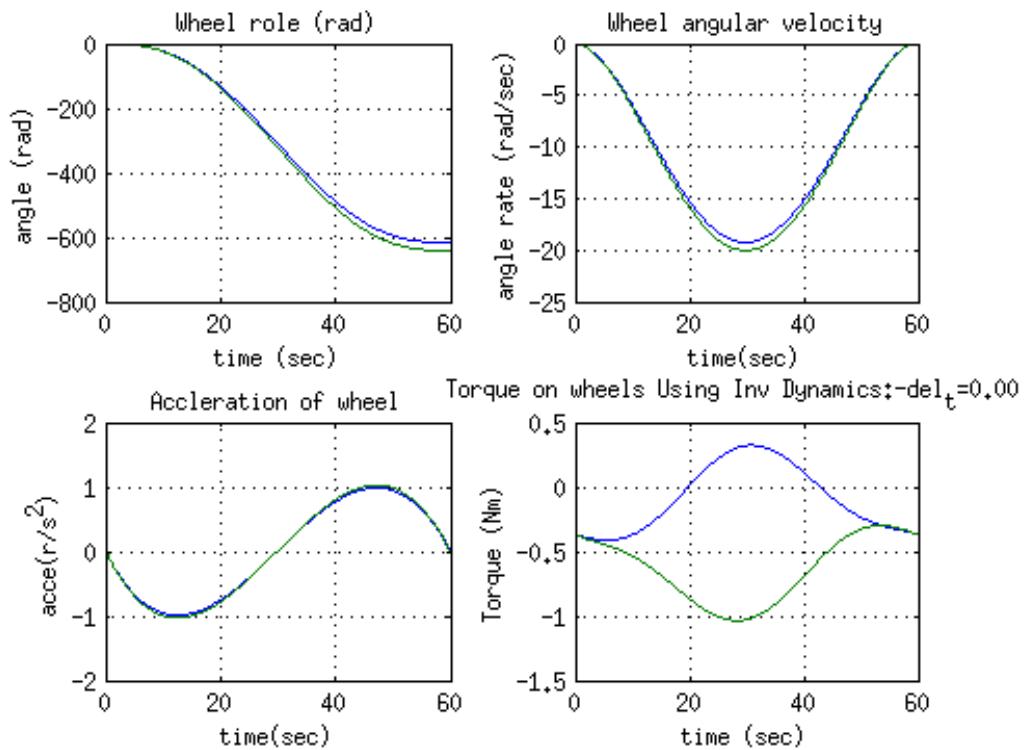


Figure 4.5: Inverse dynamics of the mobile robot

Table 4.1: Dynamic & kinematic parameters

<i>Part Name</i>	<i>Property</i>	<i>Value</i>
Rear Wheels	mass	300g
	Moment of Inertia	diag(242, 242, 465)kg mm <sup>2</sup>
Base Frame	mass	70Kg
	Moment Of Inertia	$\begin{pmatrix} 1.18 & 0.01 & -0.05 \\ 0.01 & 1.28 & 0.08 \\ -0.05 & 0.08 & 0.53 \end{pmatrix} Kg - m^2$
l	length	400mm
r	wheel radius	100mm
a	see figure	

## 4.5 Summary

In this chapter the dynamic equation was derived for the most general form of caster wheel configuration. Even though only two caster wheel configuration was considered the same formulation can be extended to any number of caster 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  $d_2 = 0$ .



# Chapter 5

## Control of Mobile Manipulator

In this chapter, the control architecture of the teleoperated mobile robot is presented. The user interface for teleoperation is discussed. The control algorithm running on the mobile robot 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.

### 5.1 Control Architecture and Hardware

The mobile manipulator explained in chapter 3 was planned to be teleoperated over a wireless network. The control block diagram and architecture are shown in Figure 5.1. It has a remote control station which is the interface for the operator

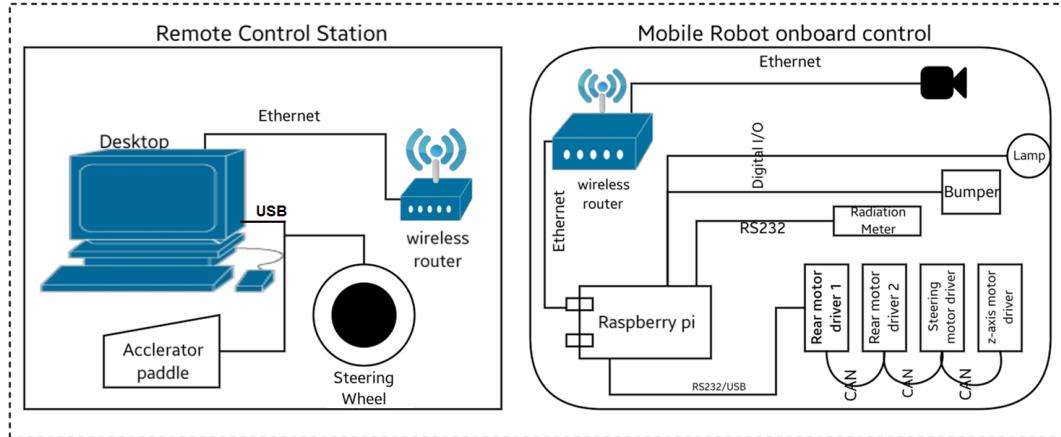


Figure 5.1: Control architecture

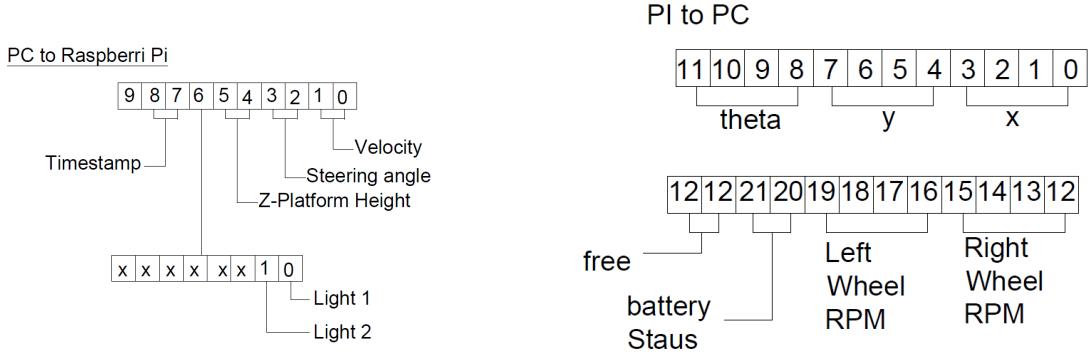


Figure 5.2: Data from PC to the robot    Figure 5.3: Data from the robot to PC

and a local onboard controller of the mobile robot. They communicate over a dedicated wireless network. The remote station sends data packet every in 50 millisecond (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 5.2. The onboard controller of the mobile robot replies with a data packet consisting of the X, Y position and orientation  $\theta$  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 5.3.

### 5.1.1 Local Onboard Controller

The onboard computer which is Raspberry Pi running raspbian (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 [79]. 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 digram of the

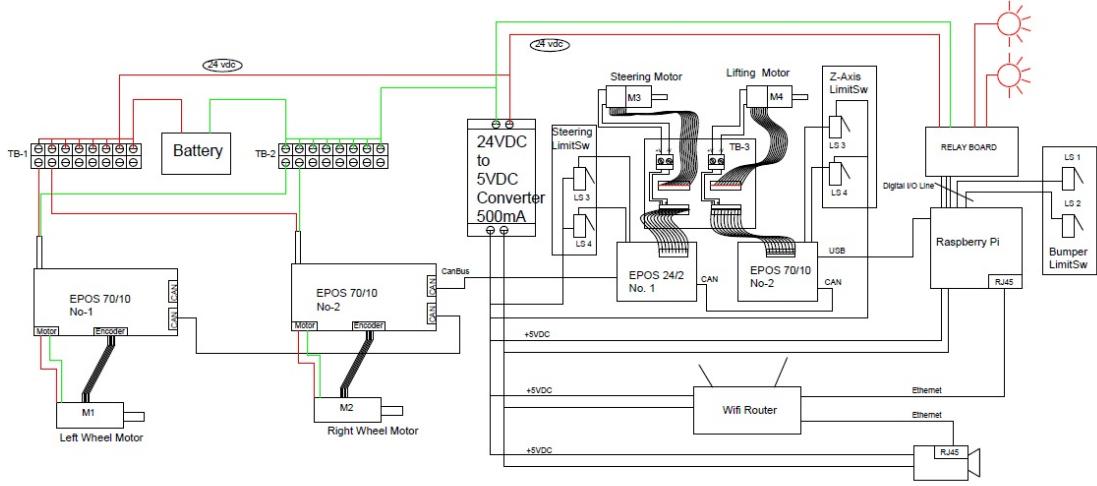


Figure 5.4: Wiring diagram of the WMR

robot is given in Figure 5.4. 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 [80]. 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 paddle's position, i.e. in position-velocity mode. This can be given by the following equation:

$$\begin{pmatrix} V \\ \theta_s \end{pmatrix} = \begin{pmatrix} K_v & 0 \\ 0 & K_s \end{pmatrix} \begin{pmatrix} \tilde{X}_p \\ \tilde{\Theta}_s \end{pmatrix} \quad (5.1)$$

where  $\tilde{X}_p$  is the displacement of paddle,  $\tilde{\Theta}_s$  is the twist of steering wheel,  $K_v$  and  $K_s$  are the proportionality constants,  $V$  is the velocity of point  $O_3$  Figure 5.10 and  $\theta_s$  is the displacement of the steer motor. These proportionality constants are derived as below based on extreme limits and listed in Table 5.1.

Table 5.1: Proportionality constant table

Robot parameters	range	Joy-Stick Parameter	range	parameter Value
$\theta_s$	-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	$K_v = 2$

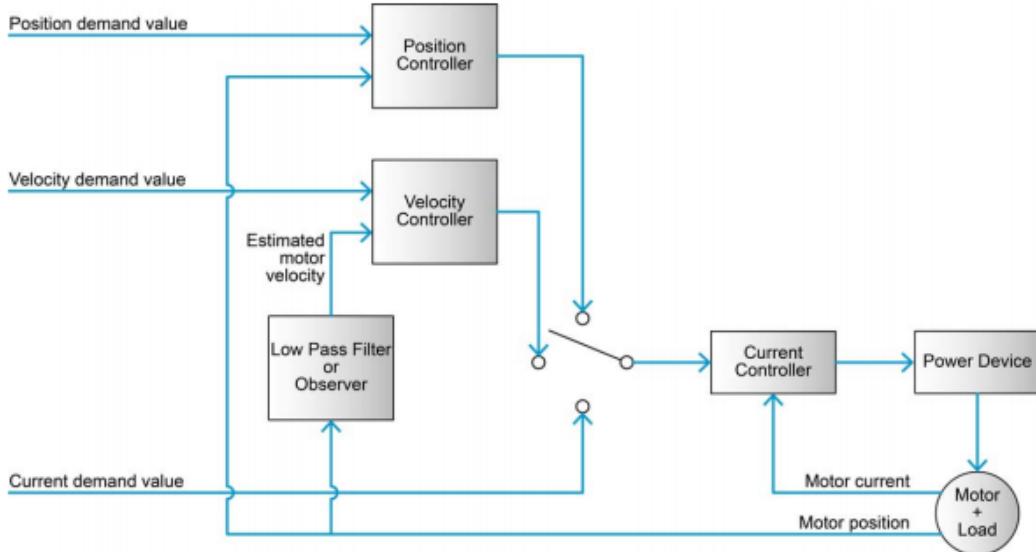


Figure 5.5: Block diagram of EPOS4 controller

### 5.1.2 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 [81], [82] is shown in Figure 5.5. 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 5.6 is a Proportional Integrator (PI) controller, running at 25KHz and the transfer function of the PI block is given as 5.2.

$$C(s) = K_p + \frac{K_I}{s} \quad (5.2)$$

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 5.7. The velocity controller is a PI controller with velocity and acceleration feed-forward. The transfer function

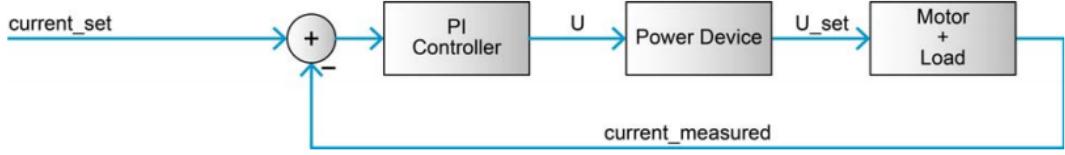


Figure 5.6: Current control block

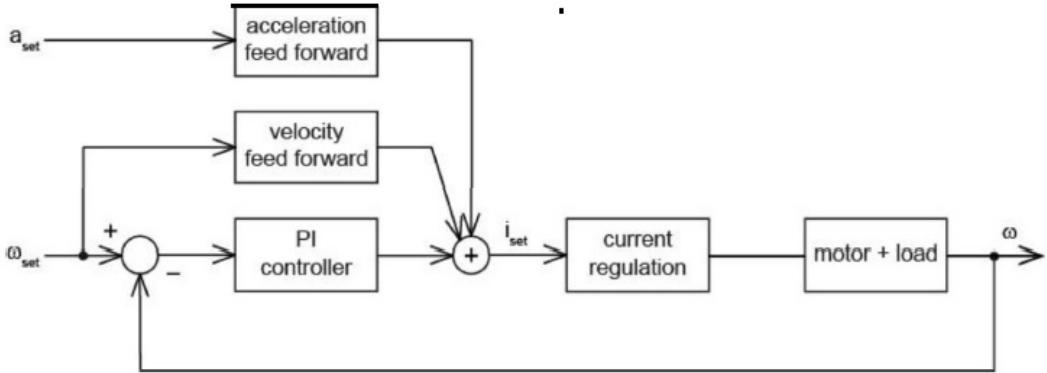


Figure 5.7: Velocity control bolck

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

$$V(s) = K_{p\omega} + \frac{K_{I\omega}}{s} \quad (5.3)$$

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 [82] respectively. Velocity is estimated from differentiation of the position data, the low-pass filter in Figure 5.5 eliminates noise due to differentiation. The transfer function  $H(s)$  for the low-pass filter is given by

$$H(s) = \frac{1}{1 + \frac{K_{p\omega}}{48K_{I\omega}}} \quad (5.4)$$

The gain values used for each rear motor in the velocity control mode is listed in Table 5.2. No acceleration or velocity feedforward was used. These parameters were determined by auto tuning.

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

Table 5.2: Parameters for left and right rear wheel motor controllers

Gain Parameter	Right motor Value	Left motor Value	Unit
$K_p$	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	$\frac{rad}{mA}$

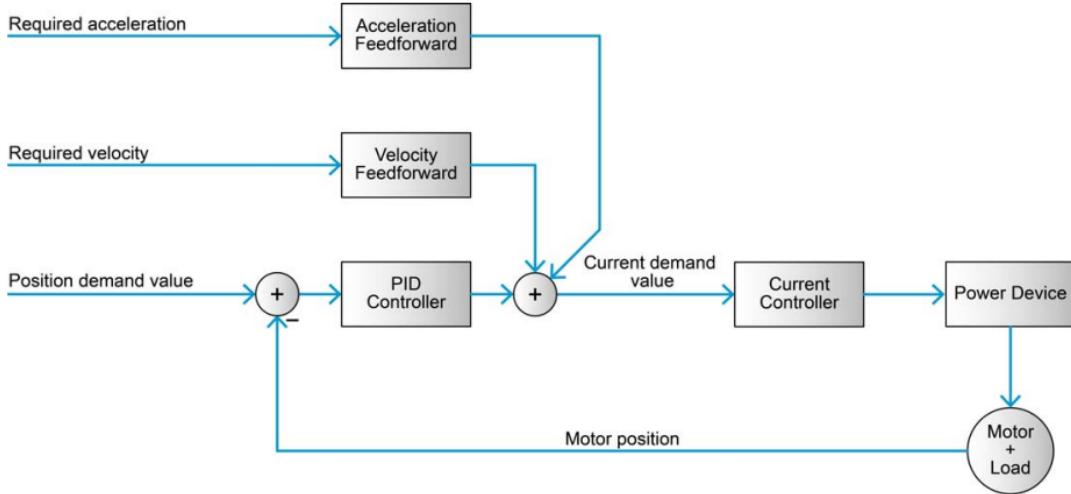


Figure 5.8: Position control bolck

in Figure 5.8. It is a PID controller with transfer function given as

$$P(s) = K_{PP} + K_{IP}s + \frac{K_{DP}s}{1 + \frac{K_{DP}}{10K_{PP}}s} \quad (5.5)$$

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 5.3.

## 5.2 The control algorithm for the WMR

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

Table 5.3: Parameters of steering motor controller

Gain Parameter	Value	Unit
$K_p$	537	$\frac{mV}{A}$
$K_I$	307	$\frac{mA}{A \cdot mS}$
$K_{PP}$	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	$\frac{mA \cdot sec^2}{rad}$

steer angle information. They were suitably scaled to get command velocity  $V$  in mm/sec, and steering angle  $\theta_s$  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 5.9

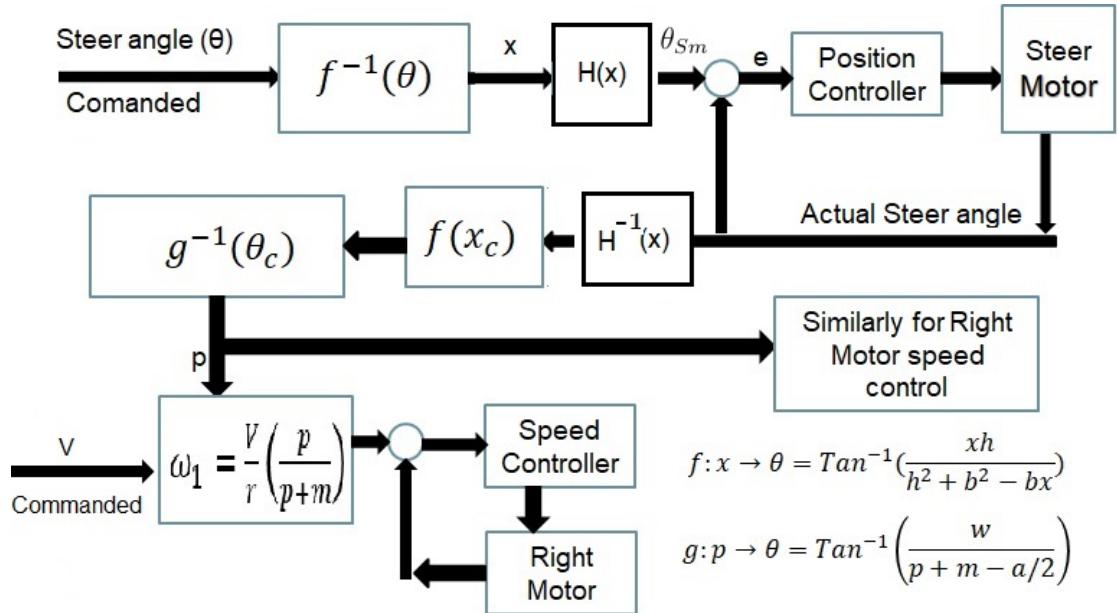


Figure 5.9: Block diagram of WMR controller

1. Calculate the setpoint for steering motor  $\theta_{SM}$  based on  $\theta_s$ .
2. Read the current steering angle  $\phi_{ic}$ .
3. Calculate the velocity setpoints  $\omega_i$  and  $\omega_o$  of rear wheels based on the  $V$  and  $\phi_{ic}$ .

4. Command setpoints  $\omega_i$ ,  $\omega_o$  and  $\theta_{SM}$  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, particularly, in case of large change in commanded  $v$  and  $\theta_s$ . Each block in Figure 5.9 is discussed next.

It may be noted that the  $\theta_s$  always refers to the steer angle of the inner front wheel  $\phi_i$  and  $V$  refers to  $O_r$  as shown in figure 5.10, i.e.

$$\theta_s = \phi_i \quad (5.6)$$

The set point of the steering motor at the output of gear box  $\theta_{SM}$  is given by equations 5.7 and 5.8 based on the geometry of Davis steering gear [67].

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

or

$$f(\phi_i) : x = \frac{\tan \phi_i(h^2 + b^2)}{h + b \tan \phi_i} \quad (5.7)$$

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 5.11. Therefore, the steering motor angle,  $\theta_{Sm}$ , is given below as

$$H(x) : \theta_{Sm} = x \frac{360}{\pi D_p} \quad (5.8)$$

Next, the equation relating current steer angle  $\phi_i$  and rear wheel set point

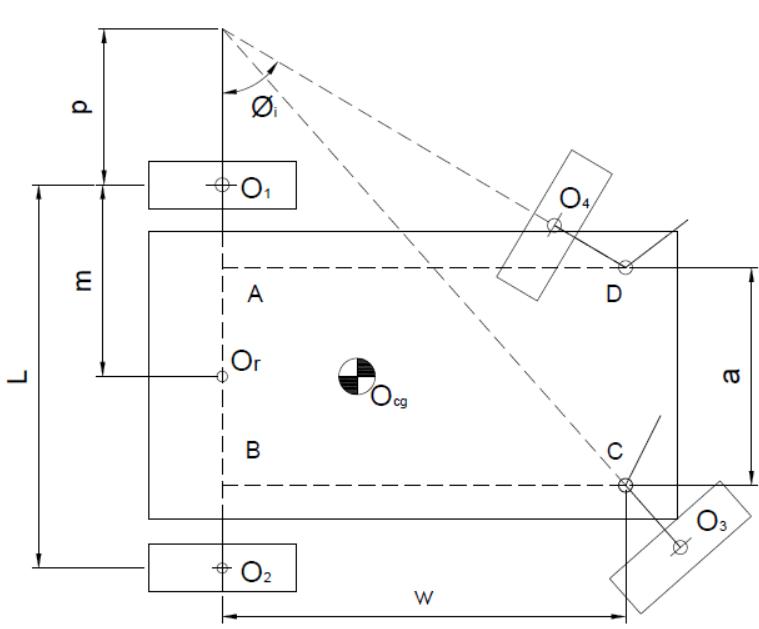


Figure 5.10: Ackerman Steering Condition

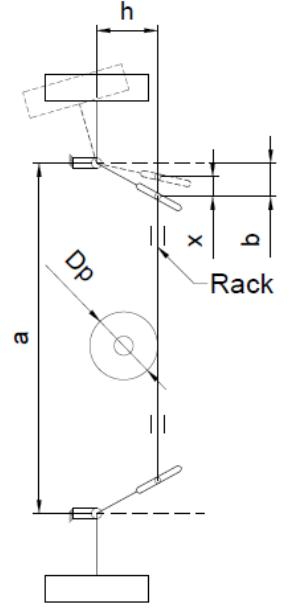


Figure 5.11: Davis Steering Gear

velocities  $\omega_{RS}$  and  $\omega_{LS}$  is presented. From the geometry of Figure 5.10, one gets

$$\begin{aligned} \tan \phi_i &= \frac{\bar{BD}}{\bar{OB}} = \frac{w}{p + m - a/2} \\ g : \phi_i \rightarrow p, \quad p &= \frac{w}{\tan \phi_c} - m + \frac{a}{2} \end{aligned} \quad (5.9)$$

Now using Equations 4.10 and 4.11 in equations relating the right and left wheel velocities to the WMR platform angular velocity  $\omega_3$  and velocity,  $V$ , of the reference point  $O_r$ , presented below

$$\dot{O}_r = \dot{O}_i + \omega_3 \times (O_r - O_i)$$

$$\dot{O}_r = \dot{O}_o + \omega_3 \times (O_r - O_o)$$

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 5.9, the setpoints for the rear wheels are as follows

$$\omega_i = \frac{V}{r} \frac{\left(\frac{w}{\tan \theta_o} - m + \frac{a}{2}\right)}{\left(\frac{w}{\tan \theta_o} - m + \frac{a}{2} + m\right)} \quad (5.10)$$

$$\omega_o = \frac{V}{r} \frac{\left(\frac{w}{\tan \theta_o} - m + \frac{a}{2}\right) + 2m}{\left(\frac{w}{\tan \theta_o} - m + \frac{a}{2} + m\right)} \quad (5.11)$$

### 5.2.1 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 was used/ 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  $\omega_3$  from current wheel velocities  $\omega_1$  and  $\omega_2$  using equations 4.11 and 4.12 with  $a = 0$ .
2. integrate  $V$  and  $\omega_3$  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 [11]

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\beta} \end{pmatrix} = \begin{pmatrix} \cos \beta & 0 \\ \sin \beta & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} V \\ \omega_3 \end{pmatrix} \quad (5.12)$$

Equation 5.12 is numerically integrated for time  $\delta t$  using the following expressions:

$$x(i+1) = x(i) + \delta t V(i) \cos \beta(i); \quad (5.13)$$

$$y(i+1) = y(i) + \delta t V(i) \sin \beta(i); \quad (5.14)$$

$$\beta(i+1) = \beta(i) + \delta t \omega_3(i); \quad (5.15)$$

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.

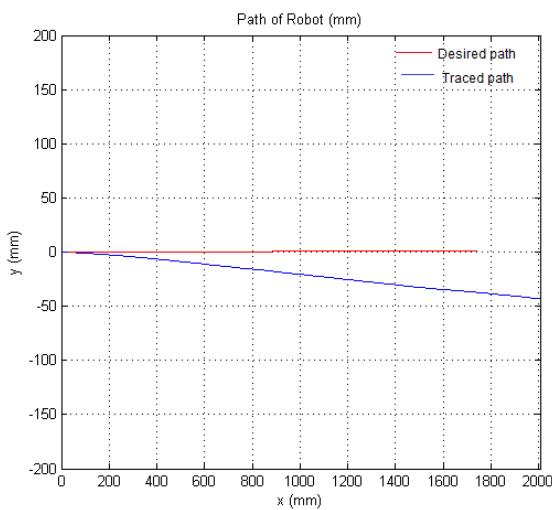


Figure 5.12: Tracing a line

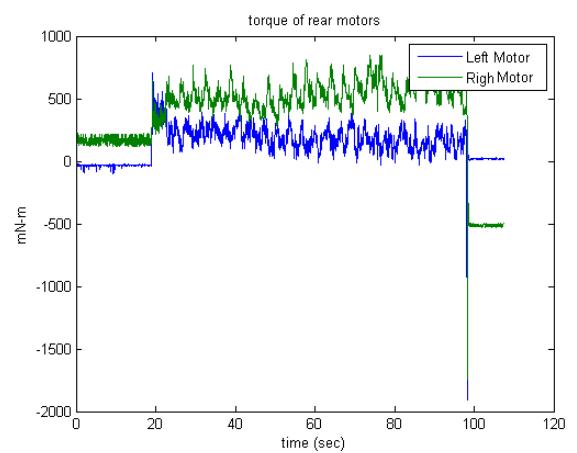


Figure 5.13: Motor torque

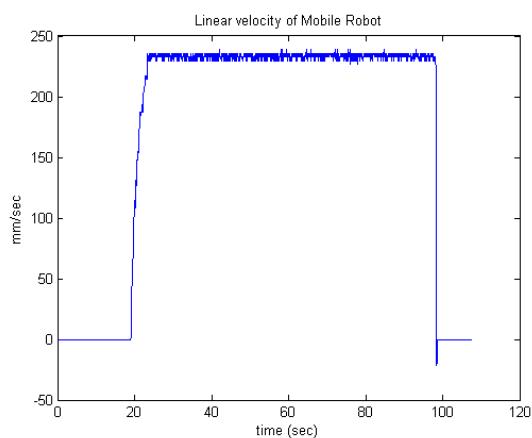


Figure 5.14: Linear velocity

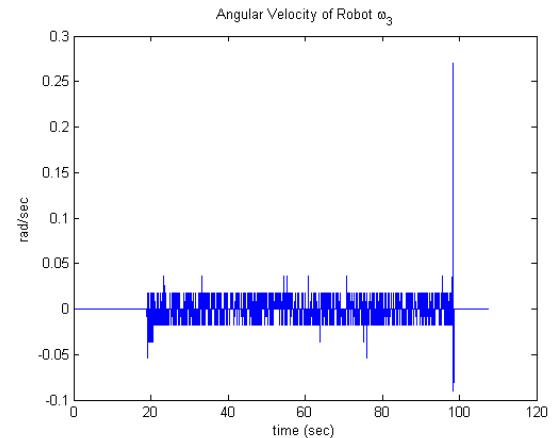


Figure 5.15: Angular velocity

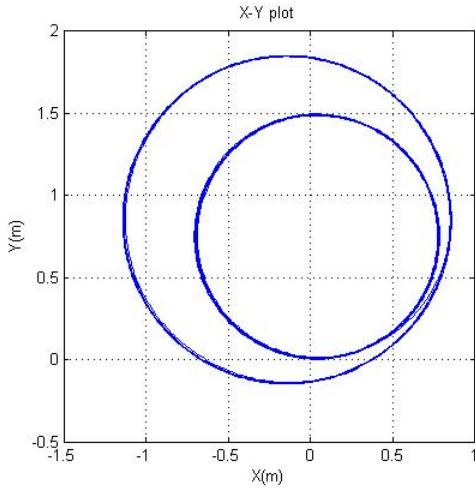


Figure 5.16: Tracing a circle

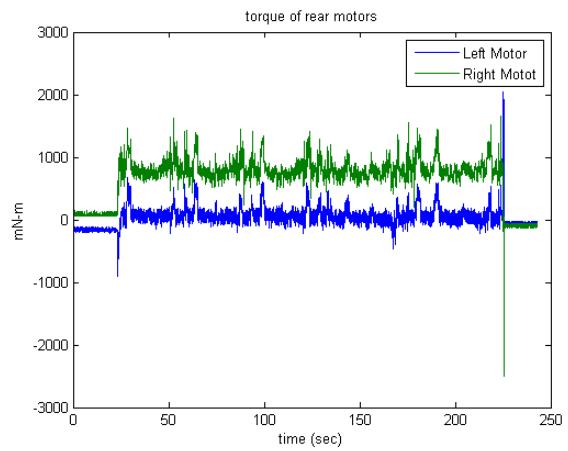


Figure 5.17: Motor torque

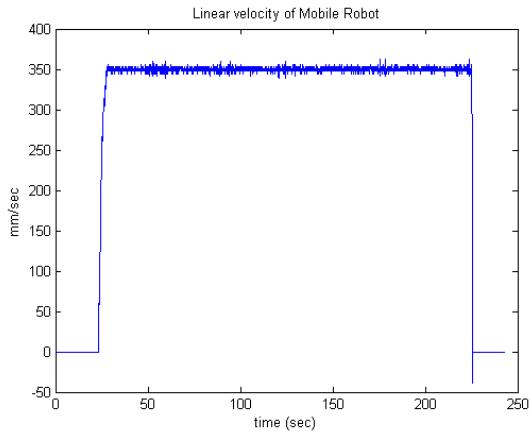


Figure 5.18: Linear velocity

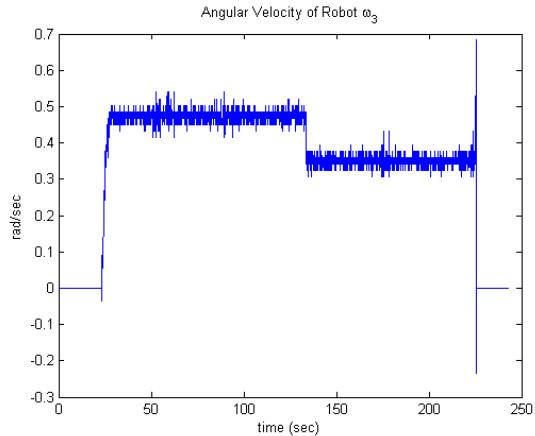


Figure 5.19: Angular velocity

As seen in the graph of Figure 5.12 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 of 300 mm for 15000 mm long path was calculated. This clearly indicates slip in the wheels. The torque curves also show that one wheel is more loaded than the other this is expected as the battery weight was on one side of the robot.

### 5.2.2 The 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.

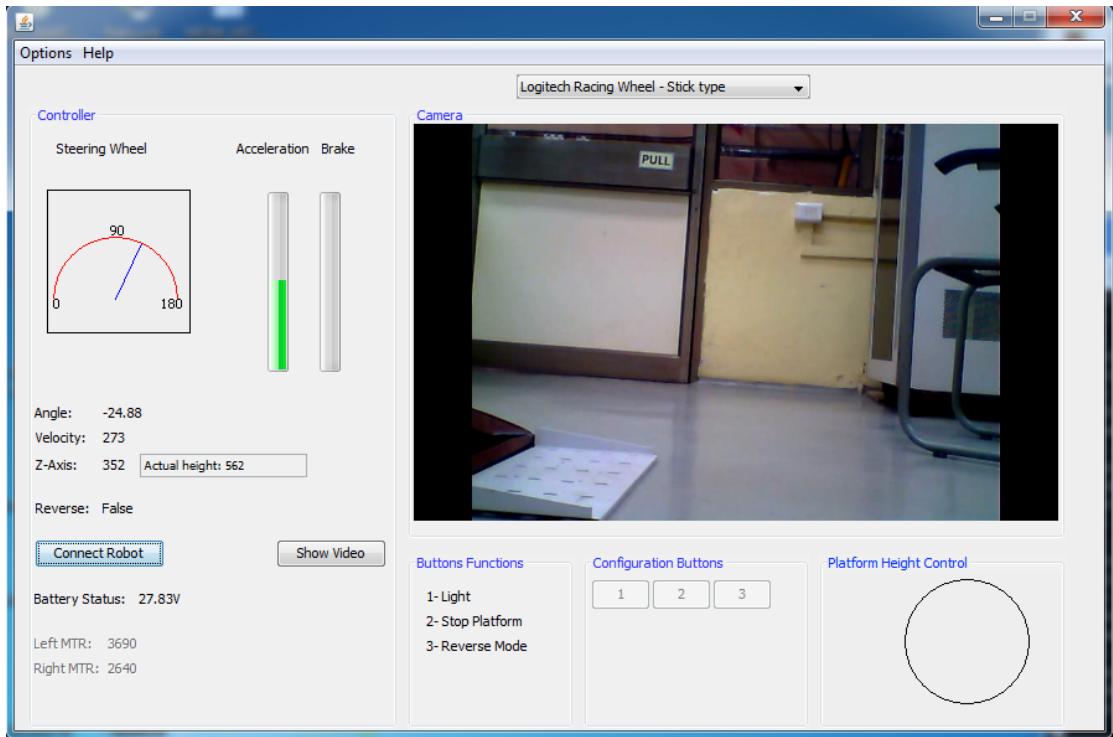


Figure 5.20: User interface for teleoperation

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 5.20 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.

### 5.3 Summary

In this chapter, the control architecture of the mobile robot 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 6

## Simulation of Tele-operation

In this chapter, simulation of a teleoperated mobile robot is presented. In tele-operation, the human operators observe a remote scene through camera(s), and manipulate the local steering wheel and accelerator paddle, as illustrated in figure 6.1 . 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 [46] and references therein. This chapter simulates the teleoperation both without and with time delay in transmission network.

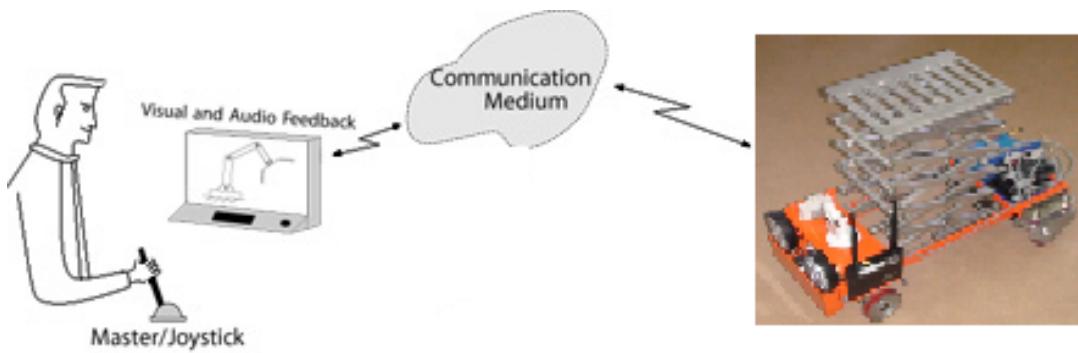


Figure 6.1: Teleoperation architecture

## 6.1 Modeling of Mobile Platform

The standard kinematic model, as described in [11], 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 3 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:

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} r_w/2 & r_w/2 \\ 1/b & -1/b \end{pmatrix} \begin{pmatrix} \dot{\phi}_L \\ \dot{\phi}_R \end{pmatrix} \quad (6.1)$$

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  $\delta$ . 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

$$\begin{pmatrix} \dot{\phi}_R(t) \\ \dot{\phi}_L(t) \end{pmatrix} = \begin{pmatrix} u_r(t - \delta) \\ u_l(t - \delta) \end{pmatrix} \quad (6.2)$$

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 complete loop.

## 6.2 Model of Human Operator

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

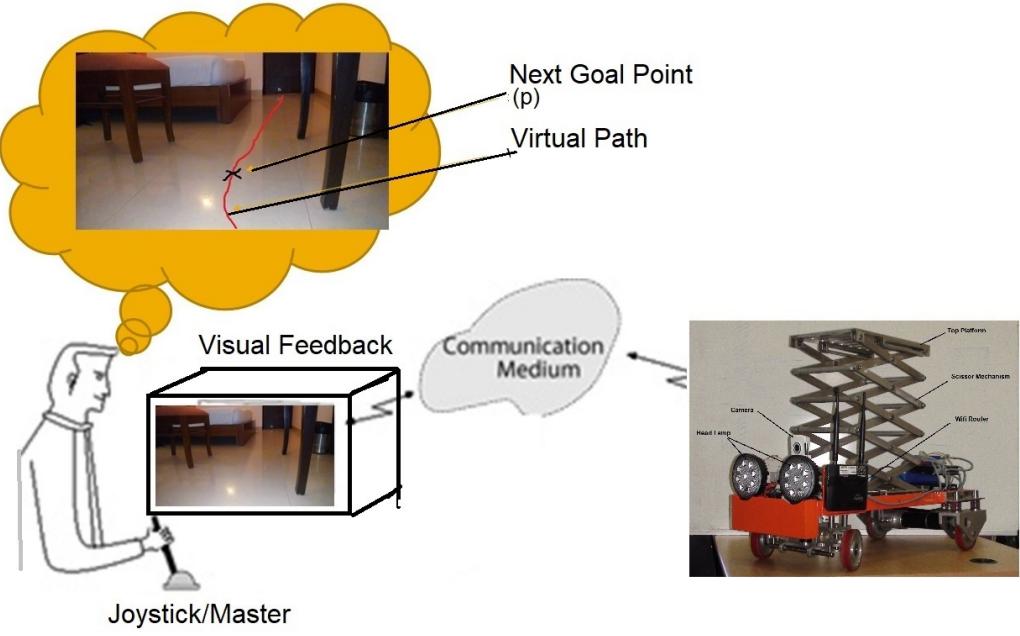


Figure 6.2: Assumed driving strategy

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 6.2. 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 [21].

The mathematical model for the pure pursuit method of path following can be derived as given in Figure 6.3. As shown in Figure 6.3, 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 teleoperated robot will depend on the field of view of camera at the remote location and the obstacles present in the remote environment.

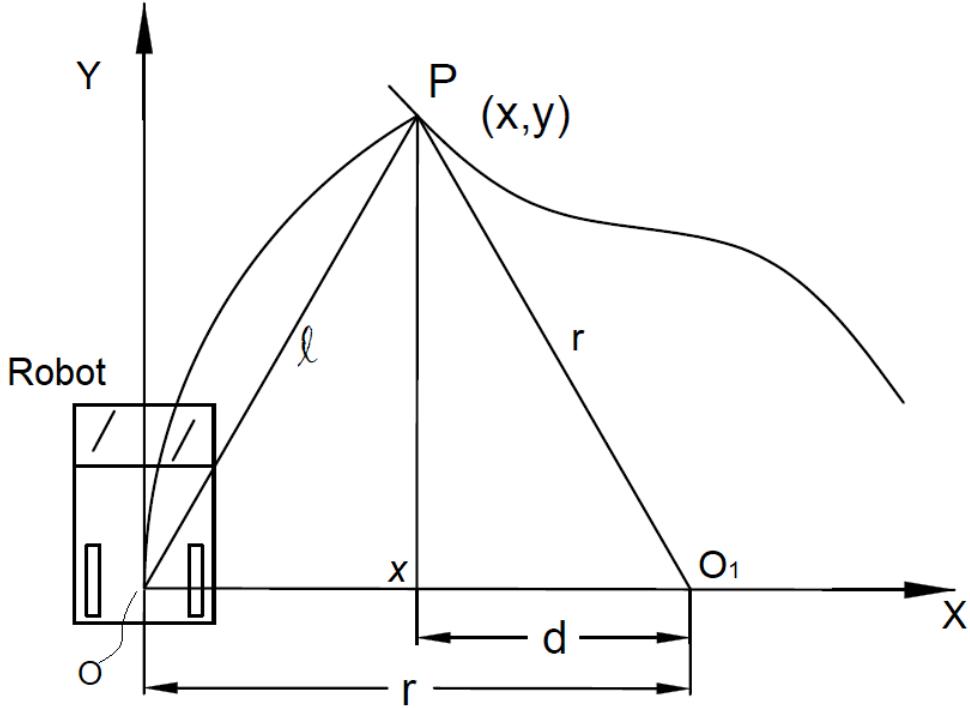


Figure 6.3: Geometry of Pure Pursuit

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

$$x^2 + y^2 = l^2, \quad d = r - x \quad (6.3)$$

Similarly, from triangle  $p, x, o_1$  we get

$$d^2 + y^2 = r^2 \Rightarrow (r - x)^2 + y^2 = r^2 \Rightarrow x^2 + y^2 - 2rx = 0 \quad (6.4)$$

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

$$2rx = l^2 \Rightarrow r = \frac{l^2}{2x} \quad (6.5)$$

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 6.3,  $\theta = 90^\circ$ . We then get

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ r_w/2 & r_w/2 \\ b/2 & -b/2 \end{pmatrix} \begin{pmatrix} \dot{\phi}_L \\ \dot{\phi}_R \end{pmatrix} \Rightarrow \begin{pmatrix} \dot{\phi}_L \\ \dot{\phi}_R \end{pmatrix} = \begin{pmatrix} 1/r_w & 1/r_w \\ 1/b & -1/b \end{pmatrix} \begin{pmatrix} v \\ \dot{\theta} \end{pmatrix} \quad (6.6)$$

The operator station sends  $v$  and  $\dot{\theta}$  as the command over the communication network to the robot. The Figure 7.5 shows the simulation strategy

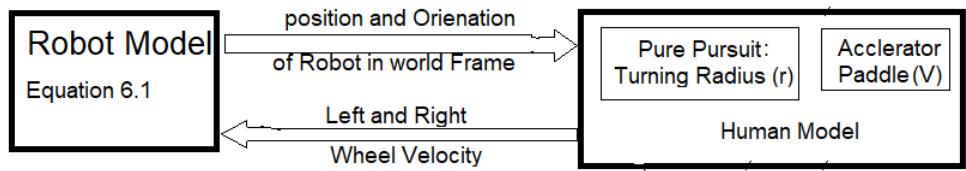


Figure 6.4: Simulation Scheme

## 6.3 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 6.5, 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.

### 6.3.1 The 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

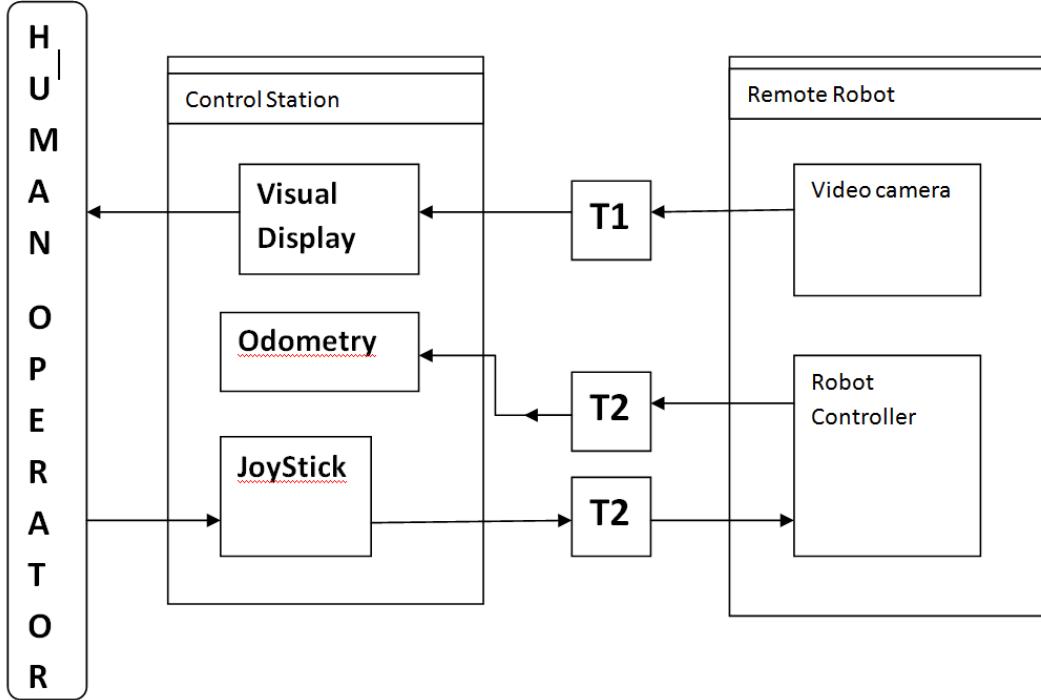


Figure 6.5: Block diagram for teleoperation

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 equation 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 6.6. Figures 6.7 and 6.8 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,  $\delta$  in Equation 6.2.

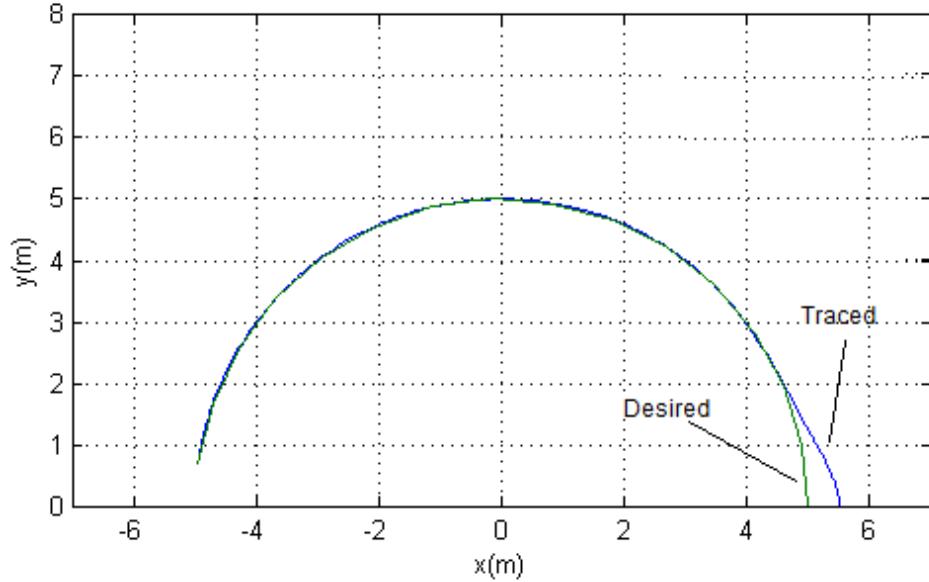


Figure 6.6: Simulation with no time delay in either direction

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

## 6.4 Summary

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

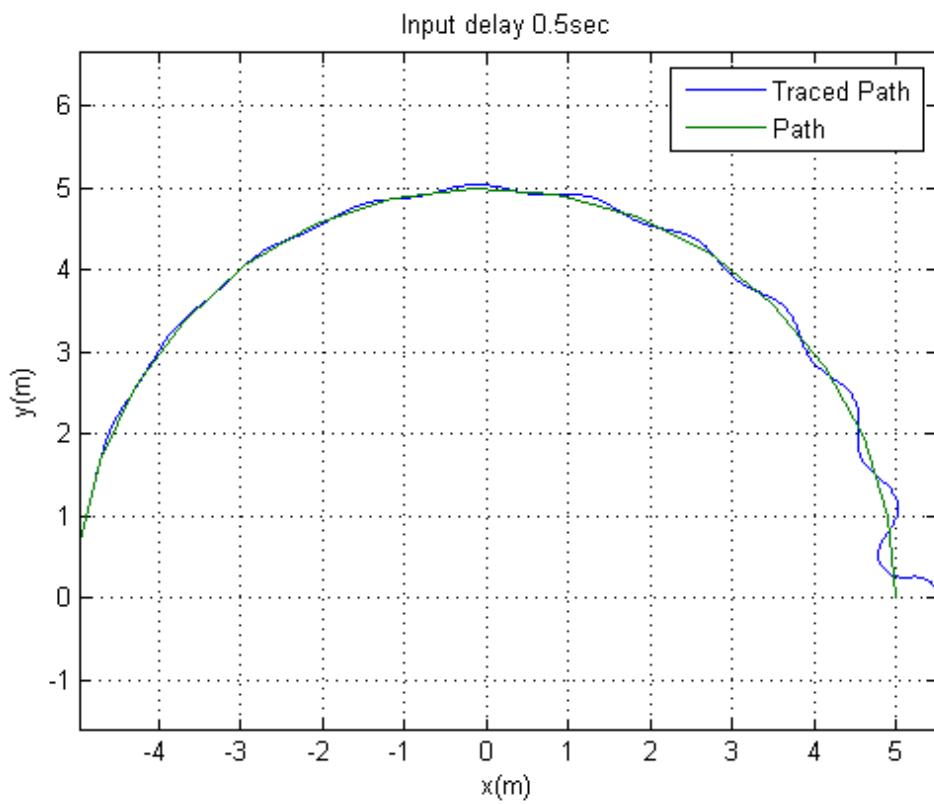


Figure 6.7: Simulation with time delay  $h_1 = .5sec$  and  $h_2 = 0$

between the local and remote station. With large time delay the system becomes unstable.

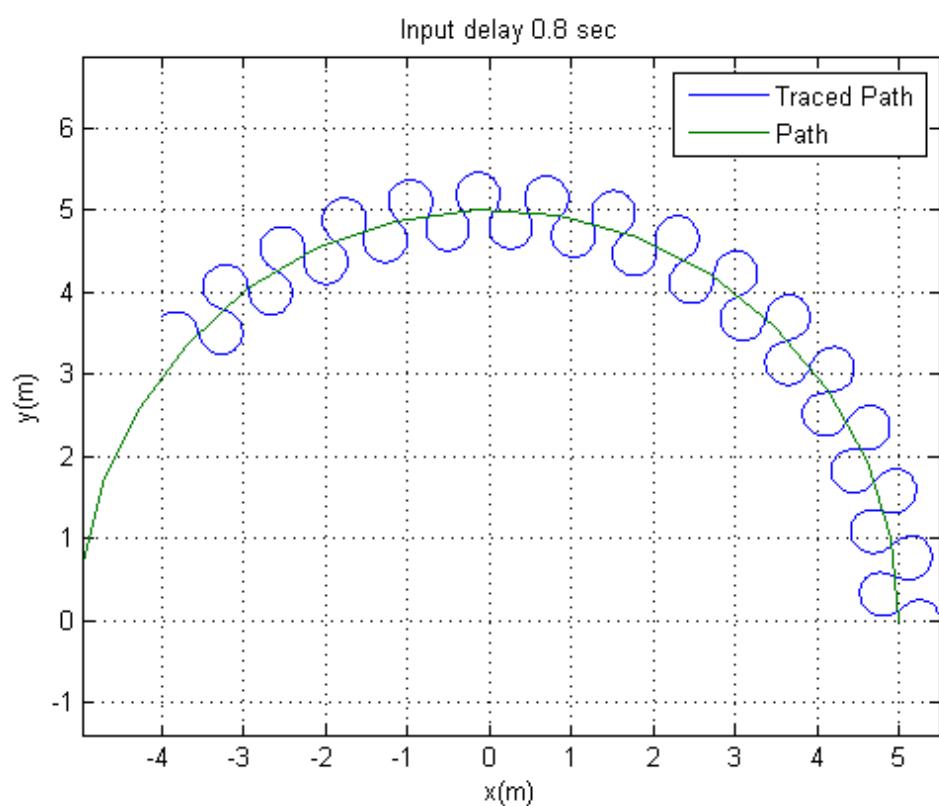


Figure 6.8: Simulation with time delay  $h_1 = .8sec$  and  $h_2 = 0$



# Chapter 7

## Time Delay Compensation and Predictive Display

In the last chapter it was shown using simulation results that time delay between the remote and local station leads the system towards instability. In this chapter we first present predictive model controller for time delayed teleoperation. We then highlight the issues of time delay on human performance. We then present our RGB-Depth sensor based predictive display implemented in our system to alleviate the problem of time delay in visual feedback.

### 7.1 Model Based Time Delay Compensation

As shown in the last chapter, simulation of time delays system with pure pursuit model for human controller in stability starts with as the delay in feedback i.e input delay to the human model increase. In paper by Ollero [1] stability of pure pursuit with input delay was presented. The details of which is presented in Appendix B. In view of the above theoretical analysis and the simulation results presented in the last chapter its required to design a stabilizing controller to take care of large delays i.e 0.8 sec. One such design based on model based predictive control is presented next.

One of the earliest predictor based controller for Linear system with time delay

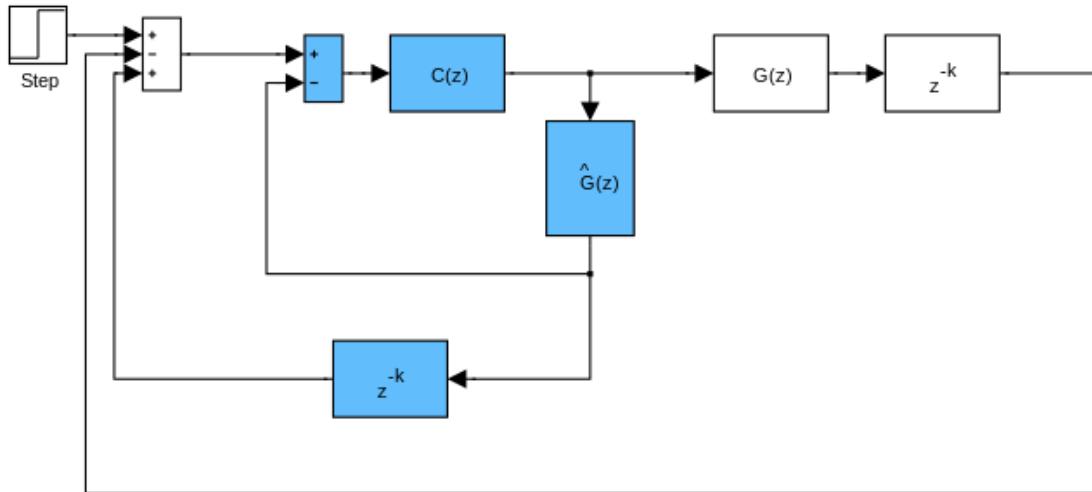


Figure 7.1: Smith Predictor

was proposed by Smith [83] called the Smith Predictor or Smith Controller. The schematic of the smith controller is shown in figure 7.1. There are two loops the inner and the outer loop. Where  $G(z)$  is the plant,  $C(z)$  is the stabilizing controller for plant with out delay,  $\hat{G}(z)$  is the model of the plant. As can be seen from the figure 7.1 that 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 our case the plant is a mobile robot. The block diagram shown in Figure 7.3 depicts the architecture of the time delay system applicable to the current system. The teleoperation over wireless network for our system resulted in a delay of  $T_1 = 500\ ms$ , or update frequency of 2Hz, for the video feedback from robot camera. This is due to the large amount of data being transmitted. The Figure 7.2 shows the measured delay in video link. The frequency of odometric and control data exchange between the two stations was at the rate of 20Hz, ie a delay  $T_2 = 50ms$ . This loop runs was independent of the video feedback link. It may be noted that  $T_2 \ll T_1$  for this system.

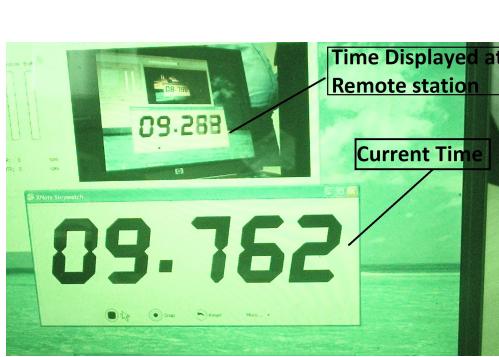


Figure 7.2: Delay measurement

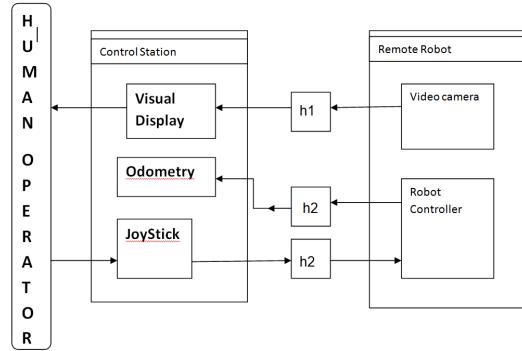


Figure 7.3: Block diagram of time delay system

### 7.1.1 Proposed Controller

The proposed control strategy to mitigate the effect of time delay is to predict the current position based on the dynamic model of the mobile robot from the last known position from the delayed video image. As shown in Figure 7.4 let us assume that we are at time  $t + \delta$ , but the latest pose data of the robot available is that of at time  $t$ . We predict the present pose i.e at time  $t + \delta$  based on the dynamic model of the robot derived in Chapter 4, Equation 4.9, and presented here again for convenience.

$$I(\theta)\ddot{\theta} = C(\theta, \dot{\theta})\dot{\theta} + \tau \quad (7.1)$$

It may be noted that the dynamic model uses torque  $\tau$  as its input. Whereas the local station which uses pure pursuit for simulation of human action, generates Linear  $v$  and Angular  $\dot{\theta}$  velocities of the robot as shown in Figure 7.5. This is taken care by first converting  $V \equiv \dot{\phi}_3$  and  $\dot{\theta} \equiv \omega_3$  to rear wheel velocities using Equations 4.11 and 4.12. The rear wheel velocities are then used in PI controller given Chapter 5 of the rear wheel motors to generate the  $\tau$  for Equation 7.1.

This predicted position is given to the pure pursuit algorithm to generate required control outputs for the remote robot. In the simulation it is assumed that this control input reaches the remote robot instantaneously, as  $T_1 \gg T_2$ . As discussed in the Chapter 6 Kinematic model given by Equation 6.1 is used to

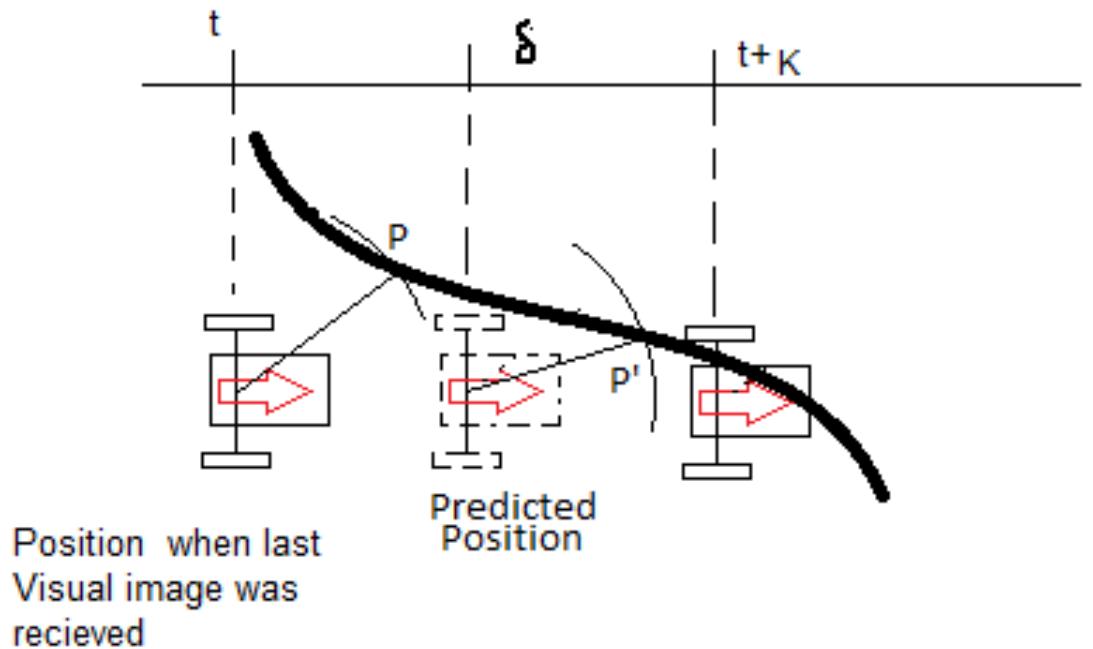


Figure 7.4: Smith predictor applied to mobile robot

simulate remote robot.

### 7.1.2 Simulation algorithm and results

The simulation scheme is shown in block diagram shown in Figure 7.5. The algorithm is explained below.

- 1: Convert the path from global coordinate system (CS) to Robots Local Coordinate System based current pose  $(x, y, \theta)$  of 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  $\omega$  using turning radius ( $r$ ) and given Linear Velocity  $V$
- 5: Command Robot  $\omega$  and  $V$
- 6: **if** new pose of robot available from remote station **then**
- 7:     Update robot pose  $(x, y, \theta)$

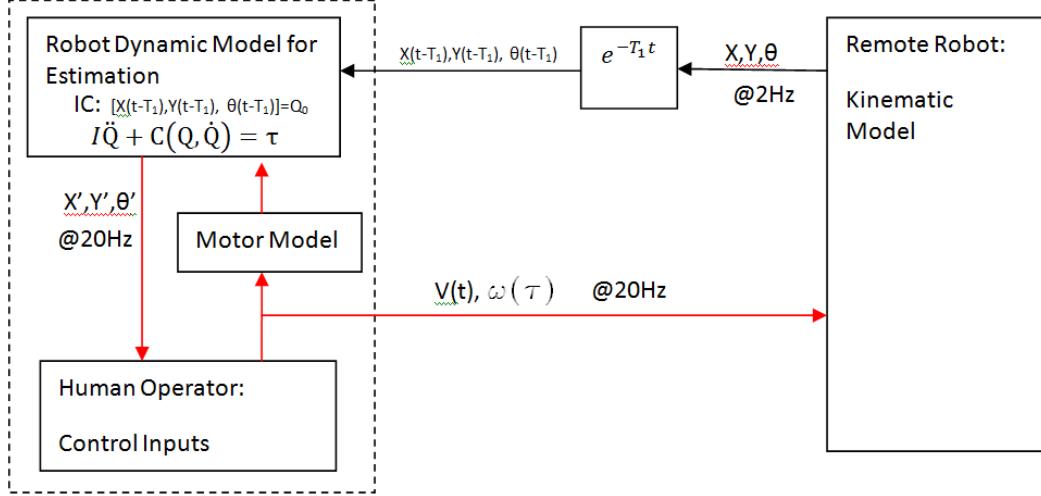


Figure 7.5: Simulation block Diagram

- 8: **else**
- 9:     Calculate the predicted pose of the robot based on commanded given in Step 4, using dynamic model of robot given in Equation 7.1.
- 10:    Update robot pose ( $x, y, \theta$ )
- 11: **end if**
- 12: **Goto Step 1**

Simulation was carried using Matlab. Differential equation solver "*Ode24*" was used to solve equation 6.1 and 7.1. 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 0.5m/s. The initial position of the robot was (4.5,0.0).

The robot's motion under feedback delay of 0.5 and 0.8 sec i.e.  $T_1 = 0.5sec$  and  $T_1 = 0.8sec$ , is shown in Figures 7.6 and 7.7 respectively. It is seen that oscillation are no more visible at 0.5sec delay and with the delay of 0.8 sec. The system shows no instability.

The above result shows that model based prediction of robot 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 performance, as the operator tends

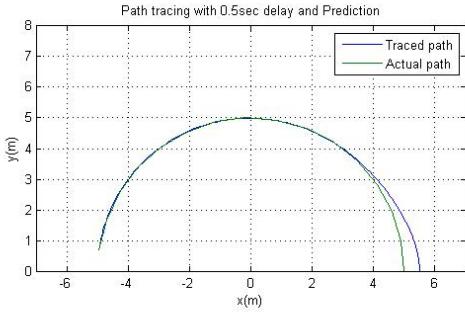


Figure 7.6: Time delay  $T_1 = .5\text{sec}$  and  $T_2 = 0$

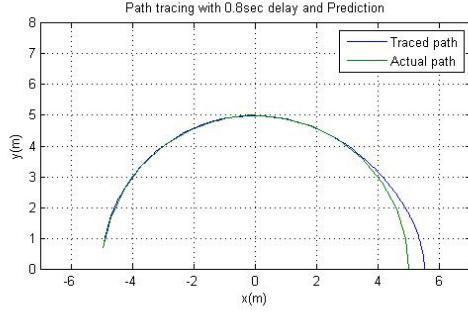


Figure 7.7: Time delay  $T_1 = .8\text{sec}$  and  $T_2 = 0$

...

to adopts a wait and watch policy to see the affect of his control action. In the next section, we present an implementation and adaptation of above discussed *model based prediction method* for visual display feedback.

## 7.2 Predictive Display

As indicated in chapter on literature survey, the delay of visual feedback leads to inefficient and unstable performance of teleoperation system. There are two major methods to over come this time delay induced problem in teleoperation. One is to use *supervisory control* and the other is *predictive display*. We adopt the second methodology ie, the Predictive Display, in our teleoperation architecture. This is because it is more intuitive to human operator, 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 [84]. In this a local model of the remote scene is used to predict and render the remote scene in response to operator 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 sensor such as monocular camera fixed on the wall, camera mounted on the robot arm, fusion



Figure 7.8: Kinect Sensor from Microsoft

of Lidar scanner and RGB camera, etc as has been discussed in literature survey. Our approach is to use low cost Kinect Sensor from Microsoft Inc. to generate 3-D model of the remote environment and use kinematic model of the mobile robot to predict the motion of actual robot in this 3-D model to generate delay free estimated image of the remote environment to the operator.

### 7.2.1 Depth Sensor Kinect

Kinect sensor from microsoft is shown in Figure 7.8. 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 the 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. 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 [85].

The *RGB camera* delivers three basic colour components of the video. The camera operates at 30 Hz, and can offer images at  $640 \times 480$  pixels with 8-bit per channel. The *3-D Depth Sensor* create a depth map, which provides the distance information between an object and the camera. The sensor has a practical ranging limit of 0.8m-3.5m distance, and outputs video at a frame rate of 30 frames with the resolution of  $640 \times 480$  pixels. The angular field of view is  $57^\circ$  horizontally and  $43^\circ$

vertically. Number of different open source software libraries are available which includes OpenNI [86] , Microsoft Kinect SDK [87] and OpenKinect [88] to access the Kinect sensor data. We used OpenNi as it is compatable with RTabMap an Open Source Software used for reconstruction of 3-D point cloud model of remote environment.

### 7.2.2 Onboard Data processing and transmission

In our teleoperation structure the Kinect sensor mounted on the mobile robot is 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 the manufacturer. It therefore generates two stream the colour and depth data stream. It is possible to send the two raw data streams over the network combine without any processing at the robot side. This puts less stresses on the onboard compute 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. Where the known transformation between the two cameras is 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 literature, The word "depth" in the *depth registration* indicates that the final image data is with respect to the depth camera frame. We adopt the second method in our architecture. Where the robot transmits registered depth data, i.e. the RGB and the Depth (RGB-D) value associated with each pixel to the control station over network.

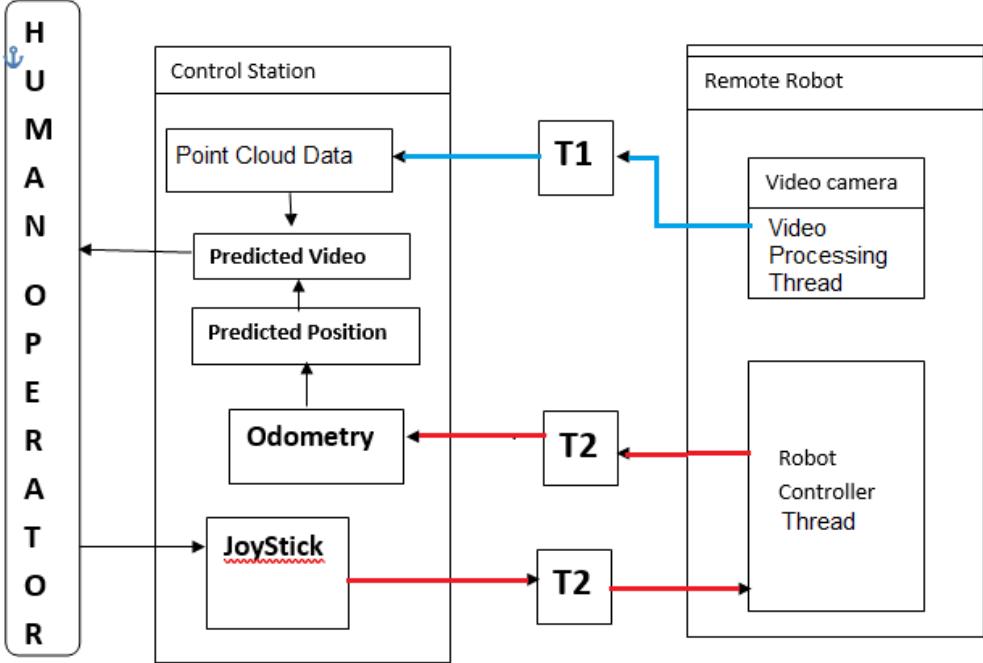


Figure 7.9: Predictive display architecture

### 7.2.3 3-D Reconstruction at operator station

The registered depth data received at the operator station is processed using the RTabMap library [89] and 3 Pint Cloud data is generated. The details of the working of RTabMap can be found in [90] and [91]. Point cloud map is a set of points in 3-D space derived using the camera model and the depth 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 two frames of data is created by matching the common feature in the two frame. The RTabMap library thus outputs a set of 3-D point along with there RGB color. The coordinates of the points are always w.r.t to camera coordinate frame. It may be noted that the registered depth image that has arrived is  $T_1$  unit of time delayed. So the local station always has a 3-D map w.r.t to the robot at  $(t - T_1)$  sec.

### 7.2.4 Remote scene extrapolation

The visual data present in the current frame is the view that the robot has seen  $T_1$  second earlier, let this data be denoted by  $P(t - T_1) = \text{Set}\{P_i\}$ . This point  $P_i$ , has associated with it coordinate  $Px_i, Py_i, Pz_i$  and color data  $c_i$ . In order to predict the current scene that the robot might be seeing we need to estimate the current position of the robot. The teleoperation architecture shown in Figure 7.9 is used in our system to accomplish this. As 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 is 20ms, shown in Figure 7.9 by red line. It may thus be assumed that the using the odometric Equation 5.13, 5.14 and 5.15, the current position of the robot is always known to the operator station. This is performed by Odometric Block in Figure 7.9 by calculating Equations 5.13, 5.14 and 5.15 from time  $t - T_2$  to  $t$ , with

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

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

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

This is 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 Vehicle coordinate at point  $O_r$  in Figure 5.10 and the camera frame is used to calculate the change in pose of the camera. Since the points  $P_i$  are in the camera frame there

current coordinates  $P'x_i, P'y_i, P'z_i$  is arrived at by using

$$\begin{pmatrix} P'x_i \\ P'y_i \\ P'z_i \end{pmatrix} = T_r^c \begin{pmatrix} Px_i \\ Py_i \\ Pz_i \end{pmatrix} \quad (7.2)$$

This new location of the points are now projected to the screen of the local operator. Thus giving him the estimated current view from the robot.



Figure 7.10: PCD at time T



Figure 7.11: Predicted Scene

The view available at the operator station at time  $t$  is shown in Figure 7.10 which  $T_1$  sec old. Based on this old data the predicted scene is shown in Figure 7.11. The operator sees Figure 7.11 instead of Figure 7.10.

### 7.3 Result and conclusion

With this predictive display model the operator was able to move the robot without wait and see strategy adopted earlier.

### 7.4 Summary

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

to project a synthesised image of the 3-D model of the remote environment, from the estimated location to the operator.

# **Chapter 8**

## **Conclusions**

### **8.1 Thesis Summary**

This thesis describes the design and development of a customized mobile manipulator developed at Bhabha Atomic Research Centre, 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 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 was achieved within the required space. 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 equation 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 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 on-board camera was available. The limited bandwidth resulted in the delay of the video. Chapter 6 dealt with the simulation of a teleoperation of a 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 matches 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 us to teleoperate the mobile robot in an unknown environment.

In summary, a mobile manipulator was sucessfully concived, designed assem-

bled and controlled remotely for the purpose for which it was intended, i.e, to map the radiation of a given area.

## 8.2 Current status

A prototype of mobile manipulator and its teleoperation 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 <https://youtu.be/B5OngfLQjo4>. The system will shortly be deployed at Variable Energy Cyclotron Centre (VECC) at Kolkata, India for the same purpose.

## 8.3 Future Scope

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

- (i) Use of inertial sensors such as gyroscope and accelerometer on the mobile robot in combination with wheel odometer for more accurate pose estimation.
- (ii) Pose estimation using encoder mounted on the front passive wheels for slip detection.
- (iii) Semi-automated navigation using intermediate goal point.
- (iv) Texture mapping of predicted visual image instead 3-D point cloud model currently used.
- (v) Some kind of force feedback to the operator for estimation of obstacle location in the remote environment.



# Bibliography

- [1] 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.
- [2] 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.
- [3] 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.
- [4] 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.
- [5] 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.
- [6] 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.
- [7] 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.
- [8] 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.
- [9] 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.

- [10] 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.
- [11] 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.
- [12] 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.
- [13] N. A. M. Hootsmans *et al.*, *The motion control manipulators on mobile vehicles*. PhD thesis, Massachusetts Institute of Technology, 1992.
- [14] 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.
- [15] 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.
- [16] 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.
- [17] 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.
- [18] J. Angeles, *Fundamentals of robotic mechanical systems: theory, methods, and algorithms*, vol. 124. Springer Science & Business Media, 2013.
- [19] 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.
- [20] 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.
- [21] R. C. Coulter, “Implementation of the pure pursuit path tracking algorithm,” tech. rep., Carnegie-Mellon UNIV Pittsburgh PA Robotics INST, 1992.
- [22] M. J. Barton, “Controller development and implementation for path planning and following in an autonomous urban vehicle,” *Undergraduate thesis, University of Sydney*, 2001.
- [23] 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.

- [24] 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.
- [25] 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.
- [26] H. L. Martin and D. P. Kuban, *Teleoperated robotics in hostile environments*. Society of Manufacturing Engineers, 1985.
- [27] J. Vertut and P. Coiffet, “Teleoperations and robotics: Evolution and development,âĂĲrobot technology,âĂĲ vol. 3a,” 1986.
- [28] 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, pp. 295–300, Citeseer, 1994.
- [29] S. Grange, F. Conti, P. Helmer, P. Rouiller, and C. Baur, “Overview of the delta haptic device,” tech. rep., 2001.
- [30] 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.
- [31] 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.
- [32] 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.
- [33] 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.
- [34] 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.
- [35] G. Hirzinger, “Robots in space-a survey,” *Advanced Robotics*, vol. 9, no. 6, pp. 625–651, 1994.
- [36] 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.
- [37] C. Flatau, “Sm-229: a new compact servo master-slave manipulator,” in *Proceedings of the 25th conference on remote systems technology*, 1977.

- [38] D. A. Lawrence, “Stability and transparency in bilateral teleoperation,” *IEEE transactions on robotics and automation*, vol. 9, no. 5, pp. 624–637, 1993.
- [39] P. F. Hokayem and M. W. Spong, “Bilateral teleoperation: An historical survey,” *Automatica*, vol. 42, no. 12, pp. 2035–2057, 2006.
- [40] 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.
- [41] 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.
- [42] G. Niemeyer and J.-J. Slotine, “Stable adaptive teleoperation,” *IEEE Journal of oceanic engineering*, vol. 16, no. 1, pp. 152–162, 1991.
- [43] 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.
- [44] 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.
- [45] M. Krstic, *Delay compensation for nonlinear, adaptive, and PDE systems*. Springer, 2009.
- [46] 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.
- [47] 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.
- [48] P. K. Pook and D. H. Ballard, “Teleassistance: Contextual guidance for autonomous manipulation,” in *AAAI*, pp. 1291–1296, 1994.
- [49] 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.
- [50] 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.
- [51] 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.

- [52] 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.
- [53] 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.
- [54] 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.
- [55] 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.
- [56] 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.
- [57] T. Burkert, J. Leupold, and G. Passig, “A photorealistic predictive display,” *Presence: Teleoperators & Virtual Environments*, vol. 13, no. 1, pp. 22–43, 2004.
- [58] 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.
- [59] 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.
- [60] 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.
- [61] 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.
- [62] 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.
- [63] 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.

- [64] 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.
- [65] 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.
- [66] J. Y. Wong, *Theory of ground vehicles*. John Wiley & Sons, 2008.
- [67] T. Bevan, *Theory of Machines, Third Edition*. Delhi, India: CBS Publishers & Distributors, 1984.
- [68] *Motor catalogue 2014*. Switzerland: Maxon Motors, 2017.
- [69] 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.
- [70] K. P. Valavanis and G. J. Vachtsevanos, *Handbook of unmanned aerial vehicles*. Springer Publishing Company, Incorporated, 2014.
- [71] J.-B. Mouret, *Evolutionary Adaptation in Natural and Artificial Systems*. PhD thesis, Université Pierre et Marie Curie, 2015.
- [72] G. Campion and W. Chung, “Wheeled robots,” in *Springer Handbook of Robotics*, pp. 391–410, Springer, 2008.
- [73] 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.
- [74] 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.
- [75] 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.
- [76] 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.
- [77] 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.
- [78] S. Saha, *Introduction to robotics*. Tata McGraw-Hill, New Delhi, 2010.

- [79] *EPOS Application Note: RS232 to CANopen Getway*. Maxon Motors, 2012.
- [80] 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.
- [81] M. Z. Zeroual, “maxon auto tuning,” *iNaCoMM2013,IIT Roorkee, India*, 2013.
- [82] *EPOS4 Application Note: Positioning Controllers*. Maxon Motors, 2012.
- [83] O. J. Smith, “A controller to overcome dead time,” *ISA J.*, vol. 6, pp. 28–33, 1959.
- [84] S. T. B, “Space teleoperation through time delay: review and prognosis,” *IEEE Transactions on Robotics and Automation*, vol. 9, no. 5, October 1993.
- [85] J. Geng, “Structured-light 3d surface imaging: a tutorial,” *Advances in Optics and Photonics*, vol. 3, no. 2, pp. 128–160, 2011.
- [86] “Openni [online].” <http://www.openni.org/>. Accessed: 2018-09-30.
- [87] “Microsoft kinect sdk [online].” <https://www.microsoft.com/en-hk/download/details.aspx?id=44561>. Accessed: 2018-09-30.
- [88] “Openkinect [online].” <https://github.com/OpenKinect/libfreenect/>. Accessed: 2018-09-30.
- [89] “Rtabmap [online].” <https://github.com/introlab/rtabmap>. Accessed: 2018-09-30.
- [90] 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*.
- [91] 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**

## **Simulation Time Delay Tele-operation**

### **A.1 Measurement of Time Delay in Video Feed-back**

The experimental method used to find the delay in video stream coming from the remote station is described here.



# Appendix B

## Stability of Pure Pursuit under time delay

The kinematic model shown in Equation B.1 and first order model for steering dynamics for the vehicle was assumed.

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

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

where,  $(x', y', \theta)$  is the posture of robot in coordinate system attached to the robot,  $V$  is the longitudinal velocity,  $\gamma$  is the angular velocity,  $T$  time constant of steering dynamics and  $\gamma_R$  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 6.5 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 form of 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  $\tau$  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  $J_\tau$  is Jacobian with respect to  $\tau$ .

Based on location of roots of the Characteristic Quasi-Polynomial  $q(s) = \det[sI - J - J_\tau e^{s\tau}]$ , results for different input delay a limiting look ahead distance for stable path tracking was presented for a circular path. His results are represented here for in Figure B.1.

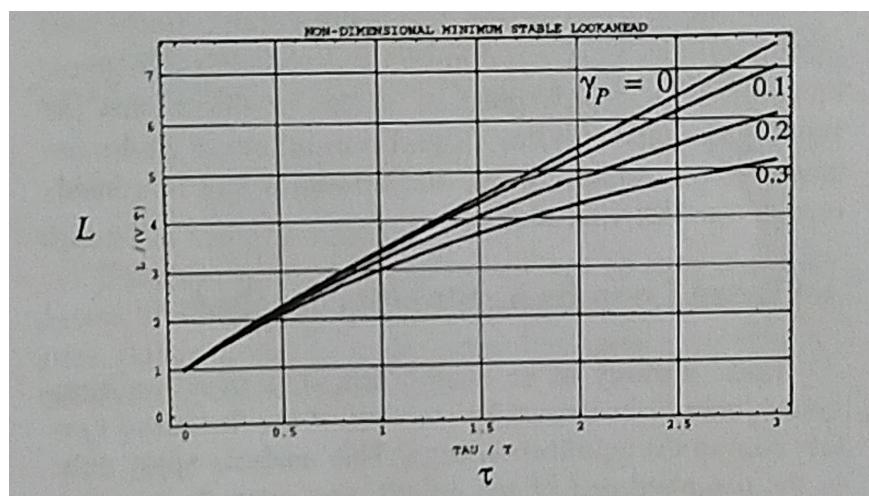


Figure B.1: Results obtained by Ollero [1]



# **Publications from the Thesis**

Research papers published/presented/under preparation are listed below

1.



## **Brief Bio-data of the Author**