UNLIMITED-WORKSPACE TELEOPERATION

A Thesis Submitted to
the Graduate School of Engineering and Science of
İzmir Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of

MASTER OF SCIENCE

in Mechanical Engineering

by
Osman Nuri ŞAHİN

December 2012
İZMİR

We approve the thesis of **Osman Nuri ŞAHİN**

Examining Committee Members:

<hr style="width:30%; margin-left:0;">

Assist. Prof. Dr. Mehmet İsmet Can DEDE

Department of Mechanical Engineering, İzmir Institute of Technology

<hr style="width:30%; margin-left:0;">

Assist. Prof. Dr. Erkin GEZGİN

Department of Mechatronic Engineering, İzmir Katip Çelebi University

<hr style="width:30%; margin-left:0;">

Dr. Gökhan KİPER

Department of Mechanical Engineering, İzmir Institute of Technology

7 December 2012

<hr style="width:30%; margin-left:0;">

Assist. Prof. Dr. Mehmet İsmet Can DEDE

Department of Mechanical Engineering, İzmir Institute of Technology

<hr style="width:30%; margin-left:0;">

Prof. Dr. Metin TANOĞLU	**Prof. Dr. R. Tuğrul SENGER**
Head of the Department of	Dean of the Graduate School of
Mechanical Engineering	Engineering and Sciences

ACKNOWLEDGEMENTS

ABSTRACT

UNLIMITED-WORKSPACE TELEOPERATION

Teleoperation is, in its brief description, operating a vehicle or a manipulator from a distance. Teleoperation is used to reduce mission cost, protect humans from accidents that can be occurred during the mission, and perform complex missions for tasks that take place in areas which are difficult to reach or dangerous for humans. Teleoperation is divided into two main categories as unilateral and bilateral teleoperation according to information flow. This flow can be configured to be in either one direction (only from master to slave) or two directions (from master to slave and from slave to master). In unlimited-workspace teleoperation, one of the types of bilateral teleoperation, mobile robots are controlled by the operator and environmental information is transferred from the mobile robot to the operator. Teleoperated vehicles can be used in a variety of missions in air, on ground and in water. Therefore, different constructional types of robots can be designed for the different types of missions.

This thesis aims to design and develop an unlimited-workspace teleoperation which includes an omnidirectional mobile robot as the slave system to be used in further researches. Initially, an omnidirectional mobile robot was manufactured and robot-operator interaction and efficient data transfer was provided with the established communication line. Wheel velocities were measured in real-time by Hall-effect sensors mounted on robot chassis to be integrated in controllers. A dynamic obstacle detection system, which is suitable for omnidirectional mobility, was developed and two obstacle avoidance algorithms (semi-autonomous and force reflecting) were created and tested. Distance information between the robot and the obstacles was collected by an array of sensors mounted on the robot. In the semi-autonomous teleoperation scenario, distance information is used to avoid obstacles autonomously and in the force-reflecting teleoperation scenario obstacles are informed to the user by sending back the artificially created forces acting on the slave robot. The test results indicate that obstacle avoidance performance of the developed vehicle with two algorithms is acceptable in all test scenarios. In addition, two control models were developed (kinematic and dynamic control) for the local controller of the slave robot. Also, kinematic controller was supported by gyroscope.

ÖZET

SINIRSIZ ÇALIŞMA ALANLI TELEOPERASYON

En basit tabiriyle bir manipülatörü veya mobil robotu uzaktan kontrol etmek anlamına gelen teleoperasyon, görev maliyetini düşürmek veya görev sırasında açığa çıkabilecek kazalardan insanları korumak amacıyla ve otonom robotlarla gerçekleştirilmesi güç görevlerde ve insanların ulaşamayacağı kadar uzak ve tehlikeli alanlardaki görevleri gerçekleştirmek için kullanılır. Teleoperasyon bilgi akışının yönüne göre, tek yönlü ve iki yönlü teleoperasyon olmak üzere iki ana guruba ayrılır. İki yönlü teleoperasyon çeşidi olan sınırsız çalışma alanlı teleoperasyonda görev operatör kontrolündeki mobil robotlarla (telerobot) gerçekleştirilmektedir. Telerobotlar havadaki karadaki ve denizdeki birçok görevde kullanılabilirler. Bundan dolayı, farklı alanlardaki farklı görevler için geliştirilmiş birçok robot mevcuttur.

Bu tezde, araştırmalarda kullanılabilecek limitsiz çalışma alanlı bir teleoperasyon sistemi geliştirilmesi amaçlanmaktadır. Ayrıca tasarlanan sistemde kullanılacak robotun her yöne bağımsız hareket edebilen bir mobil robot olması istenmiştir. Bundan dolayı teleoperasyon sistemini oluşturmadan önce bir mobil robot üretilmiş ve robot ile operatör arasındaki etkileşim karşılıklı olarak bir iletişim hattı üzerinden sağlanmıştır. Mobil robotun hareket kabiliyetine uygun bir dinamik engel algılama ve engellerden kaçınma sistemi geliştirilmiştir. Daha güvenli bir teleoperasyon için iki farklı engelden kaçınma algoritması oluşturulmuş ve test edilmiştir. Algoritmalar için gerekli olan robot ile engel arasındaki mesafe bilgisi robot üzerine yerleştirilen sensörlerden alınmış ve engel algılama testleri operatörün kullandığı güç geri bildirimi özellikli bir yönetme kolu ile gerçekleştirilmiştir ve engelden kaçınma testleri sonucunda geliştirilen mobil robotun engelden kaçınma potansiyeli uygun bulunmuştur. Ayrıca, dinamik ve kinematik olmak üzere iki farklı robot kontrolü tasarlanmıştır. Robotun kontrolü için gerekli olan tekerlek hız bilgisi robot şasisi üzerine yerleştirilmiş olan Hall sensörleri yardımıyla gerçek zamanlı olarak ölçülmüş ve kinematik kontrol jiroskop ile desteklenmiştir.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

T	Wheel traction force
L	Link length
ω	Angular velocity
τ	Torque
i	Current
f	Traction force of the vehicle on local coordinate frame
m	Mass
I	Moment of inertia
θ	Angle between local and global coordinate frames
V	Linear velocity
r	Radius of the wheel
v	Voltage
F_v	Virtual force
k	Spring coefficient
c	Damper coefficient
tr	Threshold
V_V	Virtual velocity
d	Distance between obstacle and vehicle
V_D	Desired velocity
V_T	Total velocity

CHAPTER 1

INTRODUCTION

Teleoperation is operating vehicles or manipulators from a distance. The first modern teleoperated robot was developed by R. Goertz and a group of researchers at the Argonne National Laboratory in 1950's and applied to the first nuclear reactor in the world. Since then, numerous teleoperated systems have been developed to achieve tasks in hazardous or unreachable environments such as nuclear reactor and space applications. Nowadays, teleoperated robots also are used in research, telesurgery, underwater applications and military.

A teleoperation system includes two subsystems and a communication line (Figure 1.1). One of these subsystems, called master, is used by an operator and it acquires the operator's demand. This demand can be in terms of motion and/or force. The other subsystem, called slave, is driven by operator's demands. Communication line provides data transfer between the master and the slave subsystems. In many cases, communication lines performance directly affects task performance. Reliable communication is required for especially critical tasks such as telesurgery.

(a)   (b)
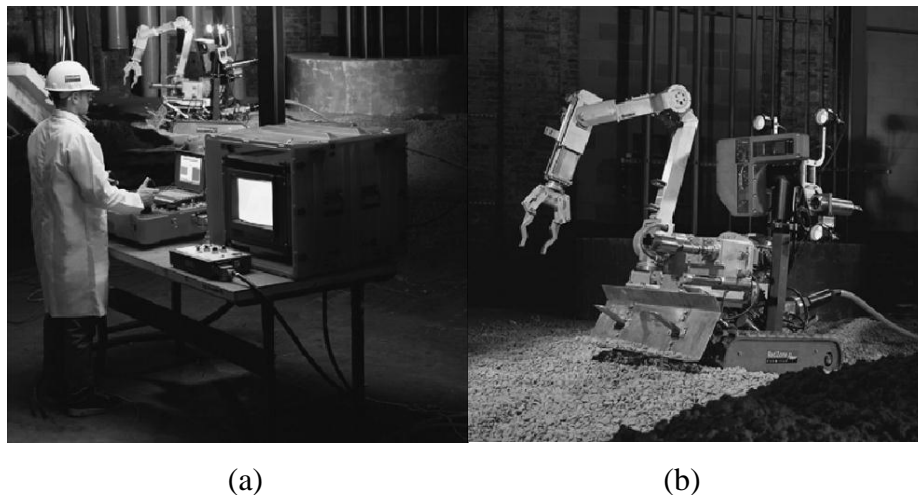
Figure 1.1. A teleoperation system: (a) Master subsystem, (b) Slave subsystem
(Source: Fong, et al. 2001)

Teleoperation systems are divided into two main categories as unilateral and bilateral teleoperation.

1.1. Unilateral Teleoperation

Unilateral teleoperation block diagram is shown in Figure 1.2. In unilateral teleoperation, slave subsystem is driven by the operator's demands (position, velocity, or force) which are acquired through master subsystem. Information flow is only from master to slave over the communication line. In this setting, there is no feedback from the slave to inform the operator about the slave environment conditions.
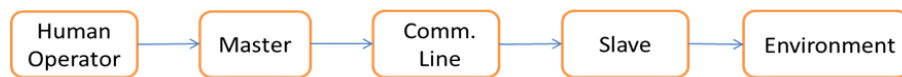
Figure 1.2. Unilateral teleoperation block diagram

1.2. Bilateral Teleoperation

In bilateral teleoperation, information flow is on two directions. Information flow is not only from master to slave but also it is from slave to master. Bilateral teleoperation system consists of slave and master subsystems and communication line which provides data transfer between two subsystems. In this system, the slave subsystem is the driven by operator's commands and the necessary feedback (visual, force, sound, position, temperature and radiation) can be sent from the slave to the master to inform operator about the environment or assist the operator while in operation (Figure 1.3). Bilateral teleoperation increases the effectiveness of human skills for more dexterous manipulation, so a bilateral control system improves maneuverability of teleoperation systems.
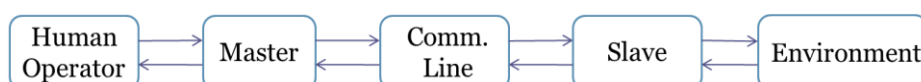
Figure 1.3. Bilateral teleoperation block diagram

According to workspace of the slave subsystem, bilateral teleoperation is divided into two categories: limited workspace teleoperation (telemanipulation) and unlimited-workspace teleoperation (vehicle teleoperation) (Dede, 2007).

1.2.1. Limited Workspace Teleoperation (Telemanipulation)

In a slave subsystem, if parallel or serial manipulators are used, this system is called limited workspace teleoperation or telemanipulation since these types of manipulators have limited workspaces.

Macro-micro manipulation in Kaneko, et al.'s work (1998) can be given as an example of the limited workspace teleoperation (Figure 1.4). Macro-micro teleoperation technology is an important part of constructing micro machines. In the macro-micro manipulation, the operator controls the slave robot in the micro world by operating the master robot in the macro world in order to assemble micro parts or perform any other works in the micro world by using operator's skills.
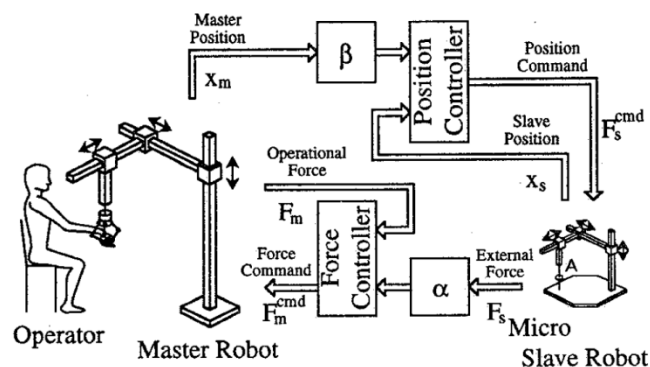
Figure 1.4. Macro-micro teleoperation
(Source: Kaneko, et al. 1998)

Telesurgery (Figure 1.5) can be given as another example of limited workspace teleoperation. Today, in the surgical area, telesurgery started to be used more and more. Telesurgery, also known as remote surgery, allows the surgeon to perform surgery on a human patient from remote operating room. One of the first surgical operations achieved by teleoperated robots is the Lindbergh operation (Butner, et al. 2003). In this operation, surgeon performed operation on patient, who was located in Strasburg France, from NY USA. Main advantages of the telesurgery are:

- Providing surgical care to patients who would otherwise go untreated
- Improving the overall quality of care by enabling expert surgeons to proliferate their skills more effectively
- Reducing the cost by eliminating unnecessary patient and surgeon travel. (Butner, et al. 2003)

(a)   (b)
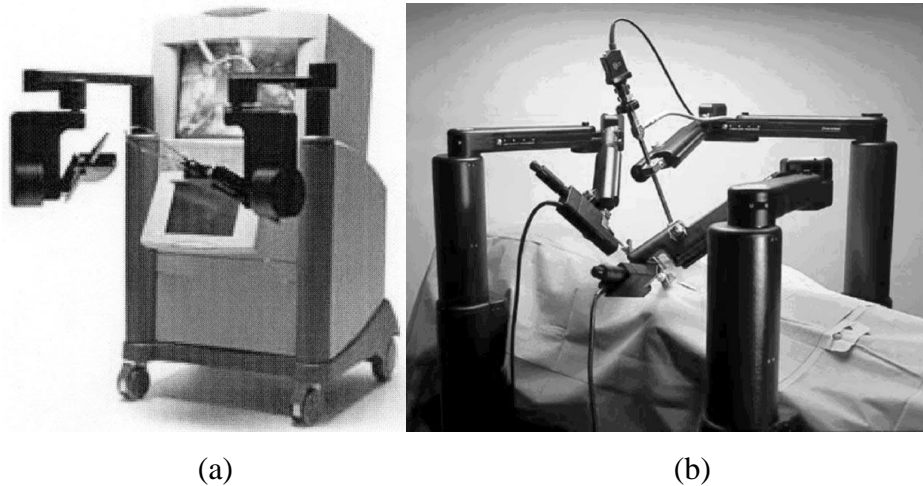
Figure 1.5. The Zeus robot: (a) Master Side (Surgeon's console) (b) Slave Side (Patient-Side Robot) (Source: Butner, et al. 2003)

1.2.2. Unlimited-workspace Teleoperation (Vehicle Teleoperation)

If mobile robots or any kind of mobile platforms, which have unlimited-workspace, are used as the slave subsystem, this teleoperation system is defined as the unlimited-workspace teleoperation or vehicle teleoperation.

Unlimited-workspace teleoperation firstly arose in the early 1900's. However, it was not widely used until 1970's. Nowadays, unlimited-workspace teleoperation systems are used in many different tasks on ground, in air and underwater. Vehicles used in different types of unlimited-workspace teleoperation applications are named as numerous terms such as unmanned air vehicle (UAV), remotely operated vehicle (ROV), unmanned ground vehicle (UGV).

4

1.2.2.1. Air Vehicles

Drones, also called Remotely Piloted Vehicles (RPV), are the first teleoperated air vehicles. Today, Unmanned Air Vehicles (UAVs) are the most common teleoperated air vehicles. United States tilt-rotor unmanned air vehicle, the Bell Eagle Eye, can be seen in Figure 1.6. UAVs are pilotless aircrafts which are controlled by human operator on the ground or master plane. Compared with manned air vehicles, UAVs are small weight and low cost air vehicles and they are suitable for dull, dirty and dangerous tasks due to its well concealment ability. In military, UAVs are used for reconnaissance and target identification tasks. Also, many UAVs have been used in combat (Fong, et al. 2001).

Figure 1.6. Bell Eagle Eye UAV
(Source: Cheng, et al. 2007)

1.2.2.2. Underwater Vehicles

Today, remotely operated vehicles (ROVs) perform tasks that take place in deeper environment or in environments that are risky for manned submersibles or drivers. Majority of the ROVs are used in oil-gas industry and others are used in survey, inspection, oceanography and marine salvage tasks. The main reason for using ROVs in the underwater operations is decreasing the mission costs and risk for human health.

A human-sized ROV with dual arm and control station, which is shown in Figure 1.7 on bottom right, were developed by Sakagami, et al. (2010). It is built to perform biological and geological research, and archaeological explorations in Lake Biwa, a lake in Japan (Sakagami, et al. 2010).
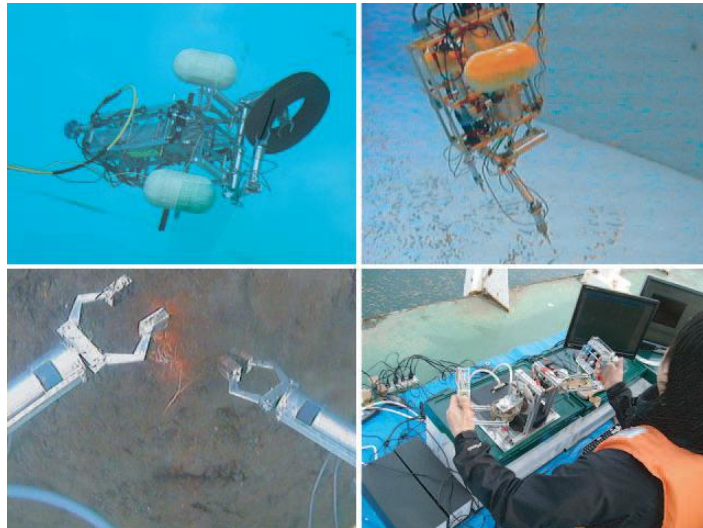
Figure 1.7. A human-sized ROV with a dual-manipulator system
(Source: Sakagami, et al. 2010)

1.2.2.3. Ground Vehicles

Teleoperated ground vehicles can be classified into three categories as unmanned ground vehicles (UGVs), exploration rovers and hazardous duty vehicles. Exploration rovers are ground vehicles developed for scientific purposes such as sample collection. Space robots can be given as an example of exploration rovers. Launching astronauts in space may be extremely difficult and dangerous in many situations. Therefore, autonomous space robots are employed in dangerous tasks and unknown territory instead of astronauts. However, they cannot accomplish all missions although they include high technology. Therefore, teleoperated space robots, which are operated from earth or space station by human operator, are used instead of autonomous space robots.

The first teleoperated space robots were Soviet Lunokhods (Figure 1.8). In the early 1970's, Soviet Union sent these robots to explore moon surface. After its achievement, National Aeronautics and Space Administration (NASA) has developed

various space vehicles. Today, space robots are an important part of NASA's Mars exploration duties (Fong, et al. 2001).

(a)   (b)
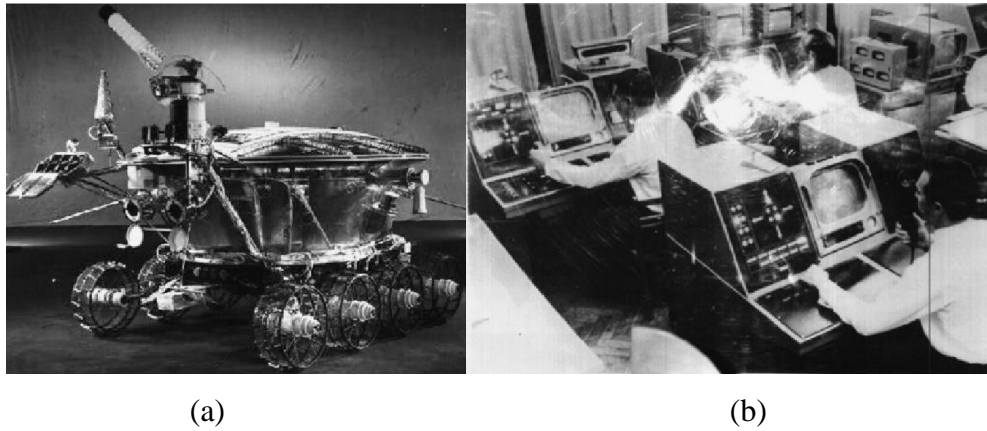
Figure 1.8. (a) Lunokhod I (Exploration Rover), (b) Control Station
(Source: Fong, et al. 2001)

Unmanned ground vehicles (UGVs) are used in various applications in military or civil areas, such as surveillance or rescue. Moreover, in military, UGVs are utilized in target identification tasks to increase the soldier's capabilities. UGVs can navigate in a wide variety of terrains. İZCİ, which is developed by ASELSAN, is an example of UGV (Figure 1.9).

Figure 1.9. Unmanned ground vehicle developed by ASELSAN
(Source: Aselsan 2012)

Hazardous duty vehicles are designed for hazardous duties which are extremely dangerous for human such as bomb disposal, assessment of nuclear reactors and mine rescue. In Figure 1.10, a bomb disposal robot developed by German Army is illustrated.

Figure 1.10. Bomb disposal robot of German Army
(Source: Wikipedia 2012)

Researches are still being conducted to improve performances of these robots while on duty, and the interaction between robot and human operator. Robot performance is directly related to how well robot moves. Thus, special wheels, which provide omnidirectional mobility, have been designed to increase robot moving capability.

1.3. Omnidirectional Mobility

Omnidirectional means an ability of a body to move instantaneously in any direction from any configuration. Ground vehicles operate on planar spaces such as warehouse or factory floors, wide variety of terrains, or roads. In two dimensional space, a rigid body, robot, has three degrees of freedom (DoF). Omnidirectional vehicles can be translated in two directions and rotated about the vertical axis in an uncoupled way. In contrast, conventional vehicles cannot control every degree of freedom independently.

Conventional wheels cannot move in a parallel direction to their axis. This non-holonomic constraint of the wheel inhibits vehicles from moving perpendicular to its drive direction. In two dimensional space, non-holonomic vehicles with car-like

Ackerman steering or differential drive system can reach every location and orientation, but it needs complex path planning and complicated maneuvers. Omnidirectional vehicles are more advantages than conventional vehicles. They have great maneuverability so they can follow complex trajectories. Moreover, omnidirectional vehicles can perform tasks in narrow environments that have static and dynamic obstacles, such as hospitals, warehouses, or factories.

Omnidirectional vehicles can be classified into two categories with respect to their special wheel design providing omnidirectional mobility: conventional wheel design and special wheel design.

Conventional wheels used in omnidirectional mobile platforms are caster and steering wheels (Figure 1.11). These types of wheels have larger load capacity and higher tolerance for ground irregularities than other special wheels. On the other hand, these wheels cannot provide completely omnidirectional mobility. Therefore, vehicles using steering wheels are also called pseudo-omnidirectional (Connette, et al. 2008). If the platform is required to move in another direction than driving direction, it can move only after motors reorient wheels on the desired direction.

(a)   (b)
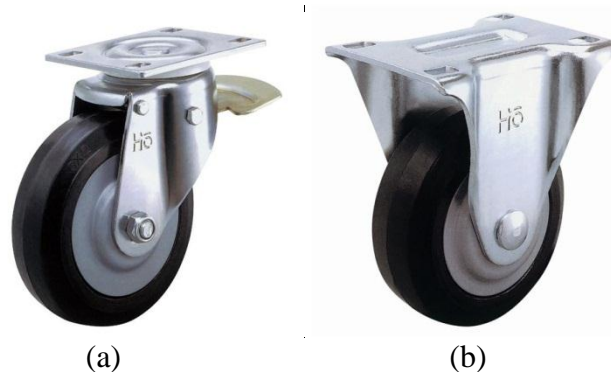
Figure 1.11. Conventional wheels; (a) Castor wheel, (b) Steering wheel
(Source: Condene Castor and Wheels 2012)

Special wheels used in omnidirectional vehicles include universal wheels, mecanum (Swedish) wheels and ball wheels (Figure 1.12). All the three wheels are based on a concept that achieves traction in one direction and allow passive motion in another. Universal wheels have small passive rollers located around outer diameter of the wheel. These wheels allow parallel motion of the wheel with respect to its rotation axis because rollers are mounted perpendicular to this axis. Mecanum wheel design is similar to universal wheels. Mecanum wheel rollers are mounted on outer diameter of

the wheel with 45°. To provide omnidirectional mobility, universal wheels are mounted on vehicles with square or triangle configuration with four or three wheels. However, in vehicles with mecanum wheels, four mecanum wheels must be attached on the vehicle like car wheels. Ball wheels (Figure 1.13) consist of an active roller ring driven by a motor and a ball that is capable of rotating about any axis instantaneously. Power is transmitted via friction from roller ring to ball.

(a)   (b)   (c)

Figure 1.12. Special wheels;   (a) universal wheel, (b) double universal   wheel, (c) mecanum wheel (Source: Vexrobotics 2012)
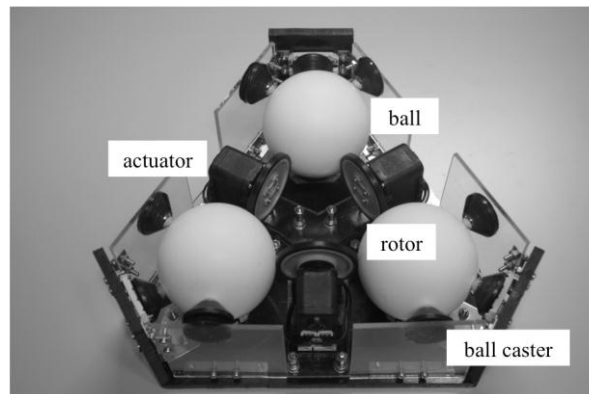
Figure 1.13. Omnidirectional vehicle with ball wheels
(Source: Ishida, *et al*. 2010)

1.4. Objective of the Thesis

The aim of this thesis is to design an unlimited-workspace teleoperation system to be utilized in teleoperation related research. As mentioned above, a teleoperation

system consists of two subsystems; master and slave. Therefore, another aim of the thesis is to develop an omnidirectional indoor ground vehicle as the slave subsystem.

The main problem of the omnidirectional vehicles that are configured with off-the-shelf omnidirectional wheels is that wheel slip occurs. Thus, omnidirectional vehicle unwillingly loses its orientation during motion due to the slip problem. In order to compensate for this, control of the vehicle is to be closed-loop control supported with gyro sensor attached on the mobile robot.

Also, the vehicle should have a reliable obstacle avoidance system to carry out teleoperation in unknown environments. A dynamic obstacle detection system, which is suitable for omnidirectional vehicles with reduced number of the sensors, and two different obstacle avoidance algorithms are proposed to resolve this problem. One of the algorithms is the semi-autonomous obstacle avoidance algorithm which is capable of avoiding collusion without sending any information to the operator about the motion of the robot while avoiding the obstacle. The other algorithm is force reflecting obstacle avoidance which transmits obstacle information to the host side. This information involves where and how far the obstacle is and is displayed through force reflecting joystick that is used by the operator.

1.5. Outline

In the following Chapter, firstly, omnidirectional ground vehicles and their usage areas are investigated and discussed. Secondly, force reflecting bilateral teleoperation systems in literature are explained. Then, obstacle avoidance techniques are presented. Finally, components used in mobile platforms; locomotion components, sensors which are utilized in obstacle avoidance systems and gathering information about environment or inner states of the robots, batteries, and software and hardware systems are explained.

In Chapter 3, firstly, design criteria of teleoperation system are listed. In this study, an omnidirectional indoor ground vehicle was designed as slave subsystem in order to execute an unlimited-workspace teleoperation system. Thus, possible designs for slave subsystem are discussed and compared. The selected parts for the slave subsystem are described. Also, provided communication line through Matlab xPC Target toolbox and master subsystem are described.

In the Chapter 4, Equations of motion of developed slave subsystem are deducted. After that, two control algorithms are created for developed system; kinematic control and dynamic control. Also, two obstacle avoidance algorithms are presented; semi-autonomous obstacle avoidance and force reflecting obstacle avoidance.

In the Chapter 5, the control and obstacle avoidance algorithms test results are presented and discussed. Finally, in Chapter 6, conclusions of the thesis are made and possible future works which are related with developed unlimited workspace teleoperation system are addressed.

CHAPTER 2

LITERATURE SURVEY

Unlimited-workspace teleoperation is divided into three categories with respect to the workspace of the vehicles; air vehicles, underwater vehicles and ground vehicles. In this thesis, the focus is on the teleoperation of ground vehicles, specifically omnidirectional indoor ground vehicles. In the literature, most of the research has been conducted to improve the performance of the omnidirectional robot and interaction between robot and environment or robot and human operator. The survey on these is presented in this Chapter.

Components of the system should be defined since one of the aims of this study is to develop an unlimited-workspace teleoperation system. Therefore, in this Chapter, necessary components of the slave subsystem to provide desired mobility and to acquire information about environment and inner states of the slave subsystem are explained. Literature survey is executed under four headlines:

- Omnidirectional indoor ground vehicles
- Force reflecting bilateral teleoperation
- Obstacle avoidance
- Parts of mobile robot systems

2.1. Omnidirectional Indoor Ground Vehicles

Many researchers have worked on wheeled or tracked mobile mechanisms to improve their teleoperated or autonomous robot mobility. Four wheeled car-like driving mechanisms or skid-steer mechanisms are the most common driving systems for the wheeled mobile robots. However, these types of robots cannot move efficiently in narrow environments, because their wheel mechanisms have non-holonomic constraints which prevents sideways movements. For better motion capability, omnidirectional wheel mechanisms have been developed. Omnidirectional mobile platforms can move any direction without changing wheel direction due to special architecture of omnidirectional wheels. Commercial omnidirectional wheels used in omnidirectional

vehicles can be classified into three categories; universal wheels, mecanum wheels and conventional wheels. In addition, many special wheel designs have been proposed to provide omnidirectional mobility.

In a study, adaptive dynamic motion controller for position control and trajectory tracking of the omnidirectional mobile robot equipped with four independent omnidirectional wheels equally spaced at 90 degrees from one to another was developed (Tsai, et al. 2010). In this work, universal wheels were used to provide omnidirectional mobility (Figure 2.1). Omnidirectional vehicles using universal wheels can be three or four wheeled configuration. However three wheeled systems may have stabilization problems because of the triangular ground contact while climbing ramps. Thus, four wheeled configuration was chosen in the study, although it has one extra degree of freedom (DoF). Proposed algorithm can be adapted in autonomous mobile robots such as home-care robots, tour-guided robots, reception robots or nursing-care robots.

(a)   (b)
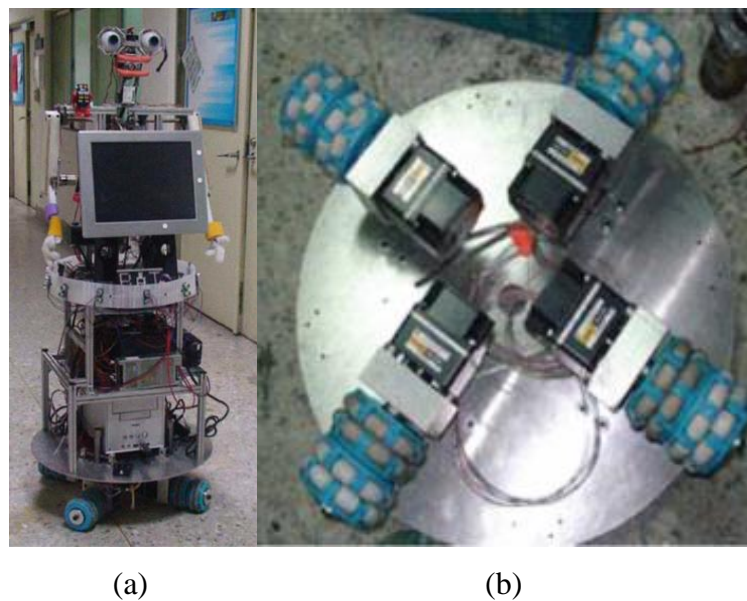
Figure 2.1. (a) Omnidirectional four-wheeled mobile robot, (b) Four wheels
            configuration (Source: Tsai, et al. 2010)

Another example of mobile robot with universal wheels is shown in Figure 2.2. In a similar research on this vehicle (Purwin, et al. 2005), an algorithm to calculate near-optimal minimum time trajectories was created. Vehicle dynamics, limited friction, and weight transfer are taken into account in the proposed algorithm. To prove the

efficiency of the algorithm, the algorithm was adapted to a real vehicle of the Cornell RoboCup system.

<div align="center">(a)   (b)</div>
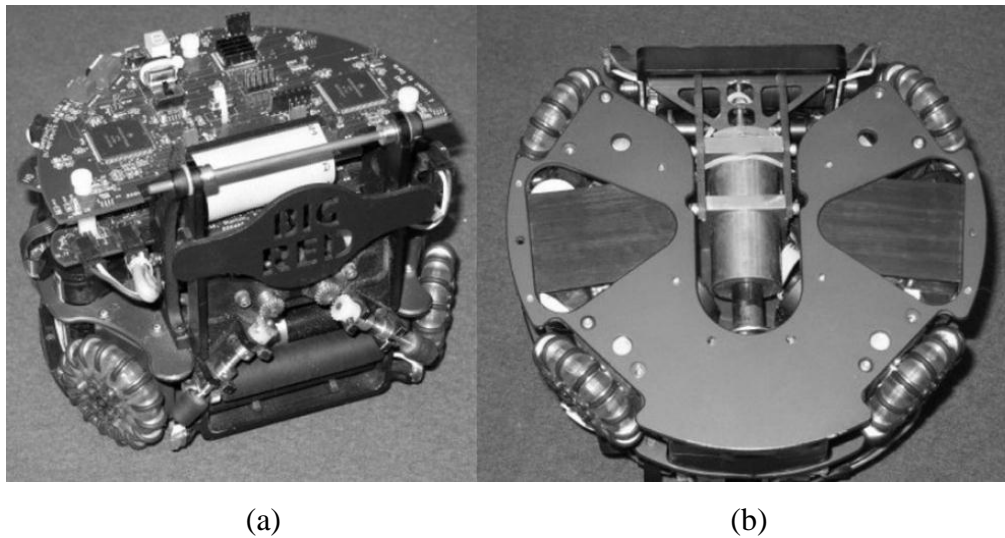
Figure 2.2. (a) Cornell RoboCup robot, (b) Robot wheelbase
(Source: Purwin, et al. 2005)

RoboCup is an international joint project to promote Artificial Intelligence (AI), robotics, and related fields. The main focus of the RoboCup competitions is the game of football/soccer, where the research goals concern cooperative multi-robot and multi-agent systems in dynamic adversarial environments. All robots in this league are fully autonomous. The games also serve as a great opportunity to educate and entertain the public around science and technology issues. Improved robot by Hibikino-Musashi team which is one of the member of RoboCup is illustrated in Figure 2.3. This robot has a three universal wheeled omnidirectional movement mechanism, omni-vision system and ball-kicking mechanism. Because it has omnidirectional universal wheels, it has reliable mobility and maneuverability (Takemura, et al. 2008).

In 1979, Bengt Ilon, who is a Swedish engineer, invented mecanum wheel which is also called Swedish wheel. The mecanum wheels have the freerolling sub-wheels positioned at an angle offset from the wheel rotation around its circumference. Due to its special characteristic, it can be moved forward and backward like traditional wheels. In addition, it enables sideway movements by small rollers attached outer diameter of the wheel. If two motions are combined, the mobile robot can move along any desired direction and orientation. (Han, et al. 2010)

Figure 2.3. Omnidirectional soccer robot developed by Hibikino-Musashi
(Source: Takemura, et al. 2008)

Mecanum wheels are very suitable for forklifts and wheelchairs. In the work of Han et al. (2009), mecanum wheels were used on a lifting system. Commercial mecanum wheels utilized in mobile robots consist of six rollers, so, new mecanum wheels were developed as suitable for lifting due to gap between rollers may cause instability of the control of the mobile robot. Mecanum wheels are shown in Figure 2.4.
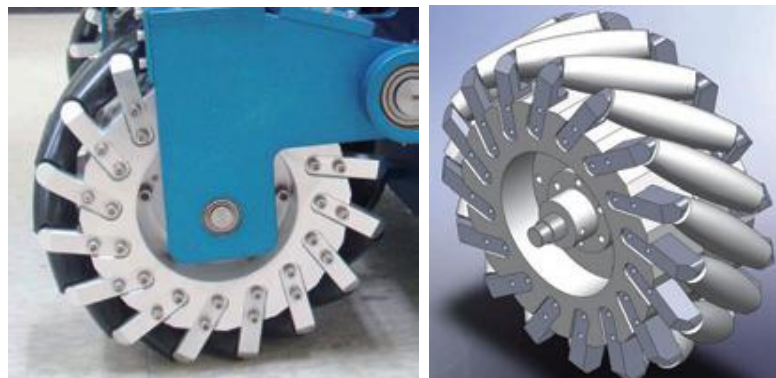
Figure 2.4. Mecanum wheel
(Source: Han, et al. 2009)

Developed lifting system with mecanum wheels is 552mm in width, 615mm in length and 935mm in height. Additionally, weight of the robot is about 90kg. The lifting robot and its six main parts are displayed in Figure 2.5. Mobile base has four suspension sets, 4 Mecanum wheels, 4 geared DC motors, and batteries. Driving control box has a DSP control board, 5 motor drivers, a laptop, a desktop 3 degrees of freedom haptic interface and a linear unit that has about 400mm stroke.
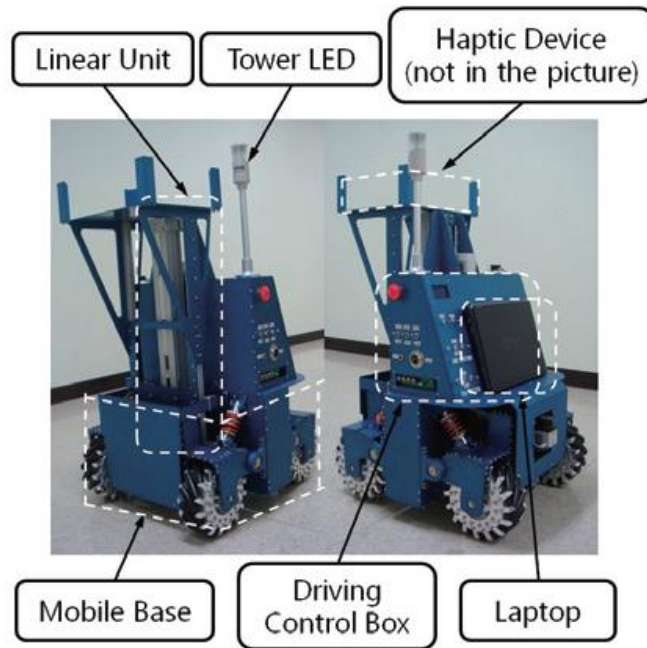
Figure 2.5. Lifting robot
(Source: Han, et al. 2009)

Wheelchairs provide action for elderly or handicapped people. Especially, the usage of electronic wheelchairs has been increased rapidly in order to support the mobility and activity without needing any people to thrust. On the other hand, commercial electronic wheelchairs used conventional wheels are not capable of moving efficiently in narrow and crowded environments. Therefore, wheelchairs should be improved and they must have effective maneuverability. To increase the movement performance of wheelchairs, omnidirectional wheels can be utilized to provide mobility.

An omnidirectional wheelchair, also called 4WD (four wheel drive) omnidirectional wheelchair, can be seen in Figure 2.6. This wheelchair is equipped with two omnidirectional wheels in the front and two normal wheels in rear. All four wheels are mounted on same side of the chair. To provide omnidirectional mobility, three motors are used: two motors for traction and one motor mounted on 4WD mechanism for rotating chair. Additionally, two parallel wheels on the rear of the wheelchair are off-centered from the steering axis. Hence, holonomic motion of the wheelchair is provided by controlling the two parallel drive wheels in such a way that the center of the chair base to translate to arbitrary direction from arbitrary configuration of the drive unit. (Wada, et al. 2008)
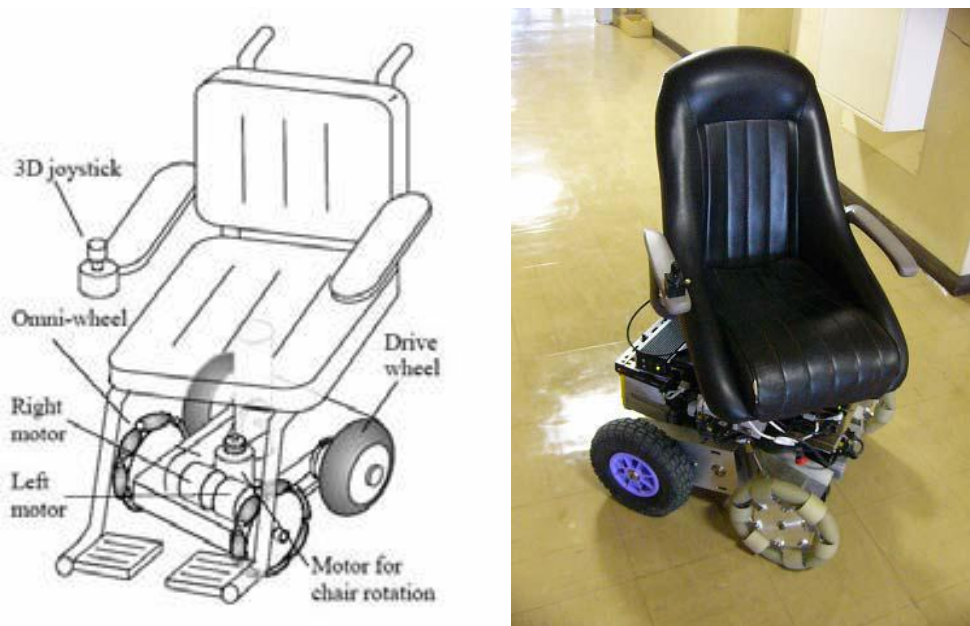
Figure 2.6. 4WD omnidirectional wheelchair
(Source: Wada, et al. 2008)

Also, omnidirectional mobility can be provided with conventional wheels; castor wheels and steering wheels. These wheels may be more appropriate than the special wheels due to its larger load capabilities and higher tolerance for ground irregularities. As an example of pseudo-omnidirectional vehicle AZIMUT-3 can be seen in Figure 2.7. AZIMUT-3 consists of eight actuators which provide locomotion; four for propulsion and four for steering the wheels, which can rotate 180 degrees around their steering axis. (Chamberland, *et al*, 2010)

Another example of vehicle with conventional wheels can be seen in Figure 2.8. Udengaard, et al. in 2007, from Massachusetts Institute of Technology, proposed an omnidirectional vehicle in rough terrains. Almost all vehicles used active split offset casters (ASOCs) are designed for flat or smooth terrains in indoor applications. In their work, they presented kinematic analysis of an omnidirectional mobile robot which is driven by ASOC integrated suspension system, along with an analysis of isotropy characteristics of the robot in rough terrain.
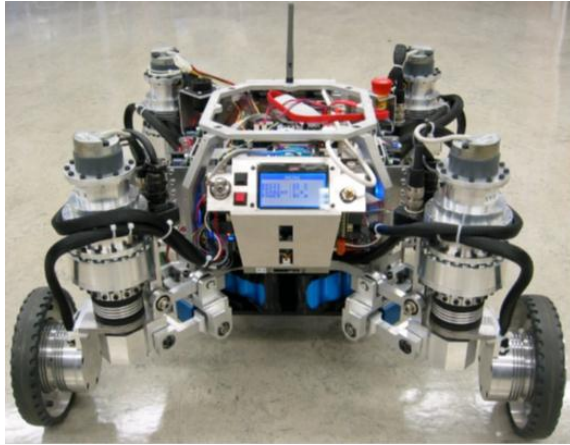
Figure 2.7. AZIMUT-3 pseudo-omnidirectional vehicle
(Source: Chamberland, et al. 2010)
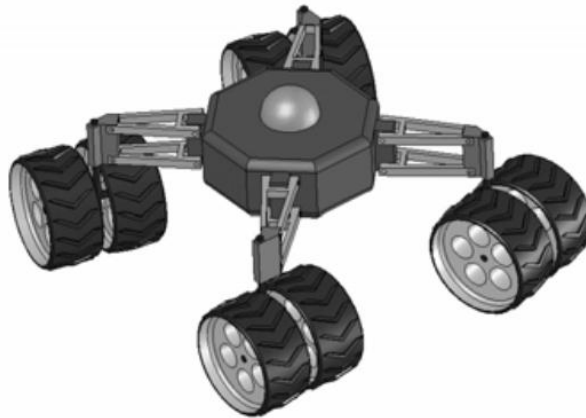
Figure 2.8. ASOC-driven omnidirectional mobile robot
(Source: Udengaard, et al. 2007)

Many researchers have worked to investigate the new concept omnidirectional wheels, because commercial omnidirectional wheels based on small passive rollers have some weak points. These wheels cause some problems while operating environments which include gaps and steps. These kinds of environments can be found excessively in hospitals, offices and houses. To deal with this problem, designed spherical wheel (omni-ball) can be seen in Figure 2.9. Thanks to this concept, while wheel rotates about wheel rotation axis, hemispheres allow passive motion on an axis which has an angle among active rotation axis. (Tadakuma, et al. 2007)
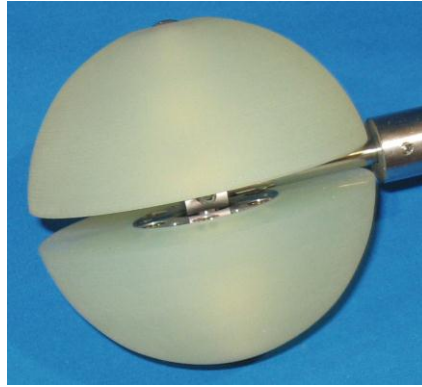
Figure 2.9. Spherical wheel ''Omni-ball''
(Source: Tadakuma, et al. 2007)

2.2. Force Reflecting Bilateral Teleoperation

Teleoperation systems provide human operator who uses master subsystem to perform successfully tasks with slave subsystem in remote environment. Teleoperated robots have been widely utilized to perform complex tasks in dirty and hazardous environments for humans such as military applications, mines, underwater and universe exploration tasks. To get information about environment such as temperature, humidity, or nearby objects, teleoperated robots have various kinds of cameras and sensors. In addition, teleoperation systems should have a reliable communication line to transfer information about the environment and control signals.

Only vision system consisting of a camera and a monitor is not enough to perceive whole environment because cameras are restricted in viewing angles or depth information. Therefore, user interfaces including multimodal information have been developed for better teleoperation. One of these, haptic interfaces with force feedback can be used in order to improve the teleoperation performance.

Aa mobile robot teleoperation was achieved through haptic interface by Diolaiti, et al. in 2002. In this research, authors considered several important aspects such as the non-holonomic constraint of the mobile robot, the need to detect the presence of obstacles by means of low-cost sensors, the stability of the overall system and the possible presence of communication time delays. Overview of the system can be seen in Figure 2.10. The data is collected by the sonar sensors mounted on pioneer mobile robot in order to build local map and detect obstacles on the surrounding of the robot. Virtual interaction force

between mobile robot and obstacle is computed by spring and damper model and this interaction force is sent by User Datagram Protocol (UDP). In addition, perception of the obstacles by operator is provided by PHANTOM haptic interface. In a similar study performed by Park et al. in 2003, six ultrasonic sensors to detect obstacles and measure distances and Wingman force joystick to transmit interaction forces were utilized.
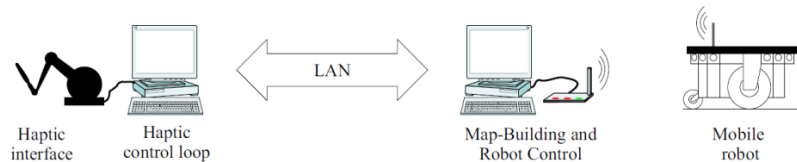
Figure 2.10. Overview of force reflecting teleoperation system
(Source: Diolaiti, et al. 2002)

Rösch, et al. in 2002, a joystick, which has force feedback feature, was used to perceive interaction between robot and obstacle. To measure interaction force among robot and obstacle, the authors used a force sensor. In addition, they proposed a communication line over the internet. developed teleoperation system is displayed in Figure 2.11.
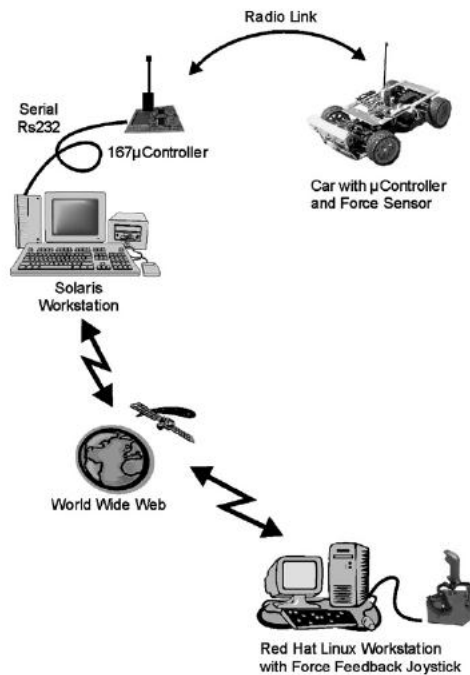
Figure 2.11. Developed teleoperation system
(Source: Rösch, et al. 2002)

In the study by Horan, et al. in 2007, a force reflecting teleoperation system with tracked outdoor mobile robot was developed. The control architecture of this system is shown in Figure 2.12. The interaction information among robot and obstacle was collected by sonar range finders mounted on front side of the robot. Thanks to this information, operator's feelings about environment were increased via a haptic interface.
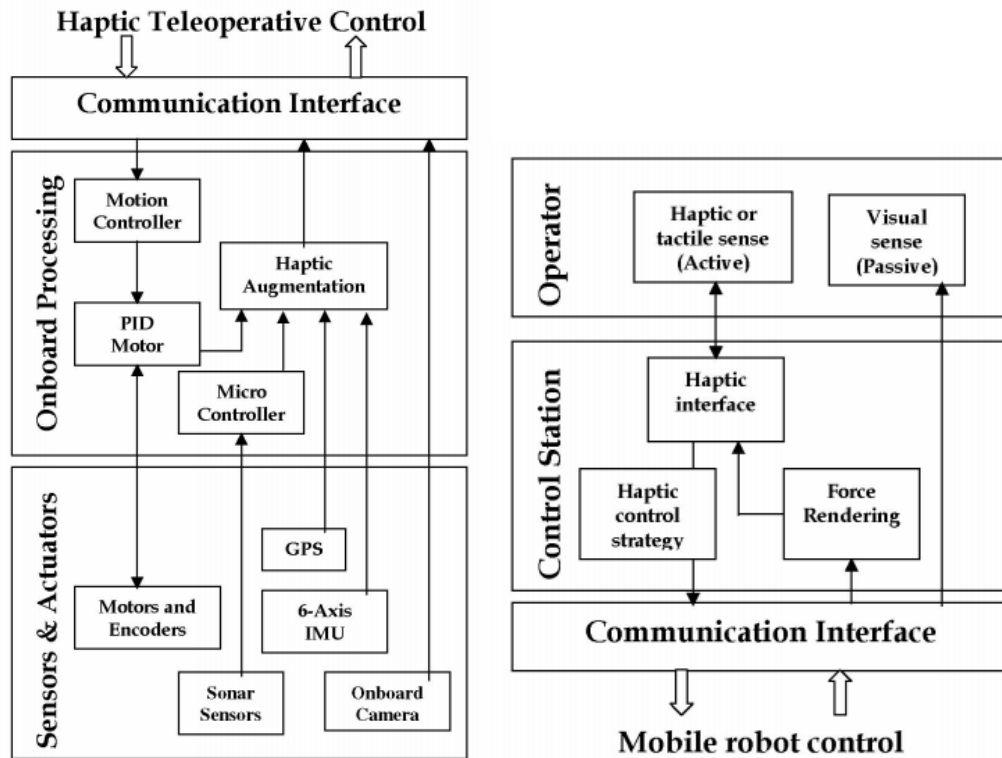
Figure 2.12. Left: Robot control architecture Right: Teleoperator control station system architecture (Source: Horan, et al. 2007)

2.3. Obstacle Avoidance

While developing a teleoperated or autonomous vehicle, one of the most important issues is to create a reliable obstacle avoidance algorithm. In many applications, robots get images from environment via camera and send these image data. According to the images, operator is informed about the location of the robot and objects near it. However, a camera mounted on one side of the robot is not enough to notice obstacles and unexpected situations. Therefore, mobile robots operating in

remote environments should have reliable obstacle avoidance algorithm. Therefore, many different types of sensors have been used. In addition, many researchers have investigated the different collision avoidance strategies.

In the work by Kim et al. in 2006, hybrid autonomous/teleoperated strategy for reliable outdoor navigation was developed (Figure 2.13). In this work, they used laser range finder to measure the distance from the mobile robot to the surrounding objects. This sensor is appropriate for non-holonomic vehicles. According to their improved strategy, if the robot controlled by operator faces with an obstacle or unexpected situations, the robot changes the control mode in four steps:

- The mobile robot sends a warning message to the teleoperator,
- It changes from teleoperation mode to autonomous mode,
- It automatically performs path planning and avoids the obstacles, and
- After avoiding the obstacles or the collision situation, it returns to teleoperation mode and the teleoperator has the control again.
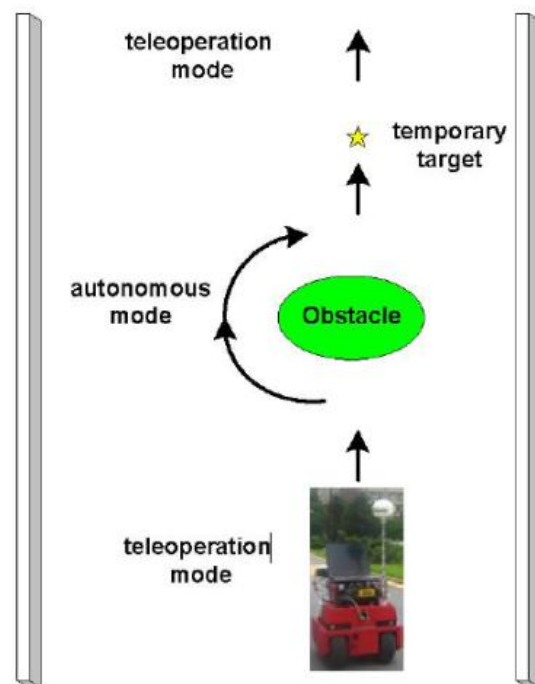
Figure 2.13. Improved obstacle avoidance strategy
(Source: Kim, et al. 2006)

In another work by Cho et al. in 2010, the virtual impedance method was used to avoid from dynamic and static obstacles. This method modifies general impedance for

mobile robot navigation with generated virtual impedance from obstacles. (Cho, et al. 2010) Figure 2.14 is shown relation among mobile robot goal position and obstacle which is modeled by spring and damper model. This calculated virtual impedance is used to reflect environmental situations via force feedback joystick. In addition, they used 16 ultrasonic sensors fixed on mobile robot with 22.5 degree to perceive obstacles. The virtual force (F_b) for both static and dynamic obstacles is calculated as;

$$F_b = \sum_{i=0}^{ns} F_{os(i)} + \sum_{i=0}^{nd} F_{od(i)} \tag{2.1}$$

where *ns* and *nd* are the numbers of static and dynamic obstacles, respectively, and virtual force for static obstacles (F_{os}) and virtual force for dynamic obstacle (F_{od}) are computed with using the following virtual impedance model;

$$F_{os(i)} = K_{s,i} \cdot X_{s(i)} + D_{s,i} \cdot \dot{X}_{s(i)}$$

$$= \begin{cases} K_{s,i} \cdot (p_0 - \|\vec{C}i\|)\vec{C}i, unit & \in when \; \|\vec{C}i\| < p_0 \\ -D_{s,i} \cdot (\Delta\|\vec{C}i\|)\vec{C}i, unit, & \\ 0 & otherwise \end{cases} \tag{2.2}$$
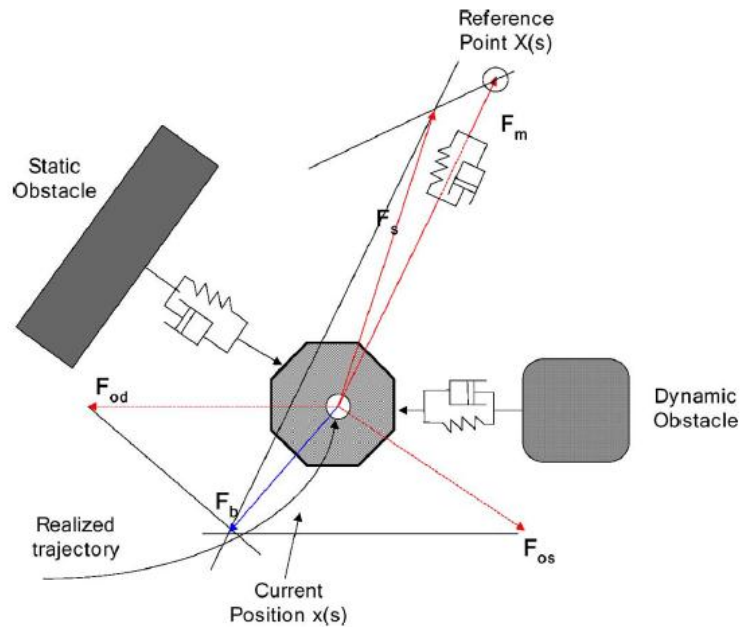
Figure 2.14. Virtual impedance model
(Source: Cho, et al. 2010)

where p_0 (85 cm) represents the threshold for collision avoidance and should be kept smaller than the sensible range of the ultrasonic sensors 3 m, the collision vector $\vec{C}_i$ is defined as a normal vector from an obstacle to the mobile robot, $\vec{C}_{i,\text{unit}}$ is its unit vector, and $\Delta\vec{C}_i$ is defined as the difference between the current and previous collision vectors. Also, $K_{s,i}$ is a spring coefficient and $D_{s,i}$ is a damper coefficient of the virtual impedance model. The value of F_{od} can be obtained by replacing "s" with "d". (Cho, et al. 2010)

In Figure 2.14, F_m is the traction force created by the operator via joystick. When the sensors measure distance between the mobile robot and obstacle or obstacles that is smaller than threshold, virtual impedance algorithm gives a virtual force vector (F_b). Then, mobile robot faces the F_s direction which is the combination of the traction force and the virtual force, until it avoids from obstacle.

In a similar work by Sohn, et al. in 2006, laser range finder (LRF) was used to measure the distance among the robot and the obstacle. In their algorithm, the robot changes its orientation according to the obstacle location. (Sohn, et al. 2006)

In another study by Chao et al. in 2009, an image processing approach for real-time target tracking and obstacle avoidance for mobile robot navigation in an indoor environment was proposed. In this study, the two CCD cameras were used to measure the relative distance of the target and obstacles from the mobile robot. Figure 2.15 shows triangulation method to measure distance with two cameras. (Chao, et al. 2009)
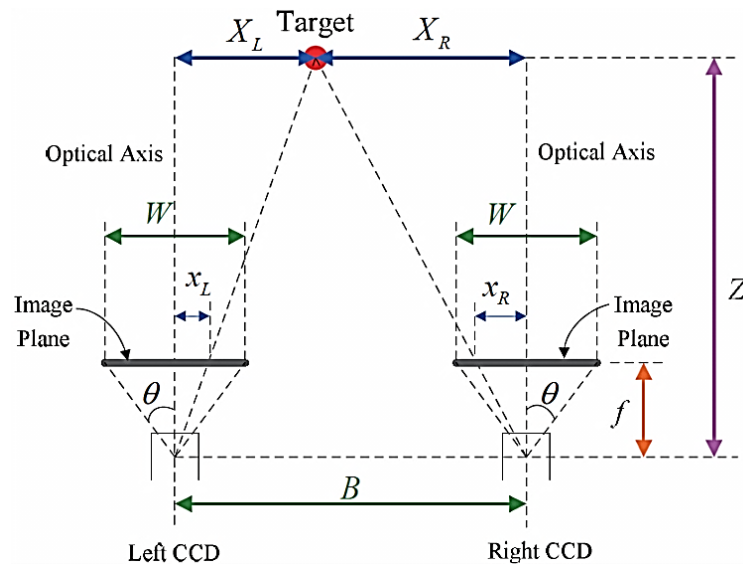
Figure 2.15. Triangulation of two parallel cameras
(Source: Chao, et al. 2009)

In the Figure 2.15 center of the two cameras are parallel, b is distance between cameras, X_L and X_R are the distances from the center of the cameras to target and x_L and x_R are projective length of the these distances on the image planes. The maximum visual angle of CDD is 2θ, f is the local length of the camera and W is the width that CCD projects on image plane. The distance between the normal axis of the cameras and the target (Z) can be calculated as:

$$Z = \frac{WB}{[2\,(x_R+x_L)\,\tan\theta]} \tag{2.3}$$

After measuring the distance between the obstacle and the robot, a control strategy is utilized according to this distance information and robot changes its orientation until it passes from the other side of the obstacle.

2.4. Mobile Robotic System Components

In this section, the components of the mobile robotic system are investigated under three headlines: motors, sensors and batteries.

2.4.1. Motors

Electric motors are components of machines converting electrical energy to mechanical energy. There are two kinds of electric motors: direct current (DC) motors and alternating current (AC) motors. AC motors are especially utilized for large machinery such as washers, cranes, and they are powered from an AC power line. AC motors are particularly used in industrial robots. They are not suitable for mobile robots, because power supplies of the mobile robots are especially DC batteries.

DC motors are commonly used in mobile robots because of their appropriate power need. Additionally, there are wide varieties of shapes and sizes such as brushed and brushless DC motors, stepper motors.

Direct usage of DC motors is not suitable in many applications, because DC motors usually runs at very high speed and low torque. To change these characteristics, DC motors have to be geared down. Many DC motors have their own gears in order to

reduce shaft rotation speed and increase motor torque. These compact motors, also called gearhead DC motors (Figure 2.16), are useful for small size robots. In addition, DC motors may have position encoders integrally connected.

Most DC motors consist of two electrical terminals. Motor spin in one direction is caused by applying a voltage across these two terminals. Spin in other direction is provided by reverse polarity voltage. In DC motors, the polarity of the voltage determines motor rotation direction and amplitude of the voltage determines motor speed.

Figure 2.16. Gearhead DC motor

On the other hand, stepper motors have more than two electrical terminals, often more than six or eight. The timing of the signals, which are applied from wires, determines the motor speed. The phase among the signals determines motor direction and a number of commands determines motor position.

Permanent magnet brushless motors (Figure 2.17) have permanent magnets which are mounted on the rotor and these magnets are encircled by electromagnets. Rotation of the armature is provided through current switch which is controlled by electronic controller.

Figure 2.17. Brushless DC motor
(Source: Iheartrobotics 2012)

Brushless motors are more powerful and efficient than the same sized brushed DC motors. Moreover, brushless DC motors are more reliable, silent and they have longer lifetime (no brush and commutator erosion) over brushed DC motors. These types of motors are suitable for many industrial applications, home electronics computer hardware and robotic and radio controlled vehicles.

Servo motors are widely used in robotic applications. The integrated circuit and potentiometer are utilized to provide closed-loop position control system. Servo motors are driven by pulse-width modulated signal (PWM). These motors have three specific colored wires emanating from the servo; the red one for power, the black wire for ground and the white one for PWM. Command protocol of the servo motor to adjust to desired position is illustrated in Figure 2.18.
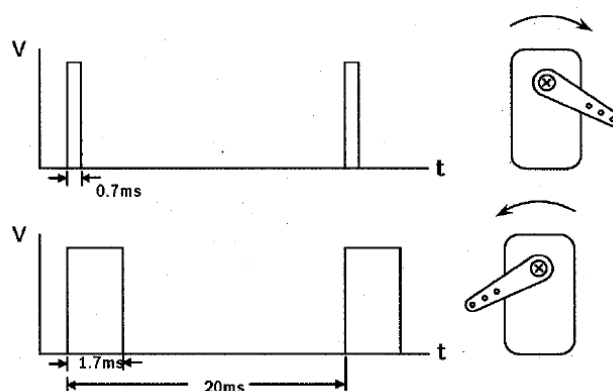
Figure 2.18. Command protocol of the servo to bring to designated position
(Source: Jones, et al.)

Servo motors are driven with pulses repeated at a specific period, notably set to 20 milliseconds (ms). The shaft position is determined by the width of the pulse. Usually, center position of the servo motor is reached with 1.3 ms wide pulses. The pulse widths vary from 0.7 ms to 1.7 ms. While pulse comes from 0.7 ms to 1.7 ms motor shaft position comes respectively from all the way to the right and all the way to the left. These kinds of servos are very useful for legged robots (Figure 2.19) or robot parts such as grippers and fingers. However, it is not used for continuous motion, due to its restricted motion capabilities.
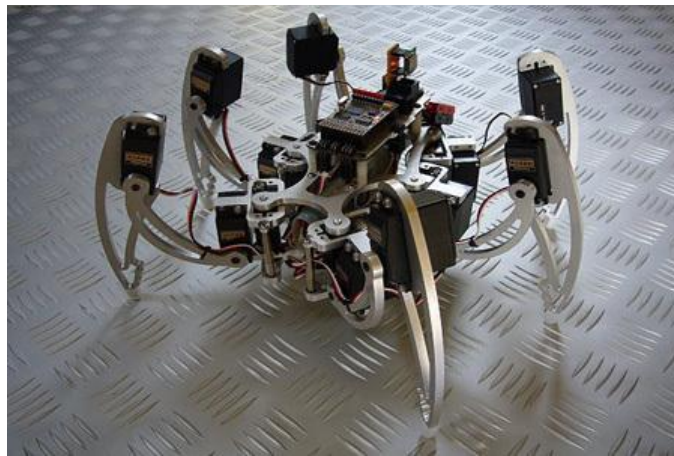
Figure 2.19. Legged robot which locomotion provided with servos
(Source: Techeblog 2012)

2.4.2. Sensors

In robotic applications, sensors are used to understand or be aware of robot environment. In fact, robots are limited by sensors and robot performance is directly related with the sensors on the robot. Sensors can perceive physical data and convert it into electrical signals. After that, these signals are read by microprocessors.

Most sensors utilized in mobile robot applications can be divided into four categories:

- Light sensors
- Sound sensors
- Odometry sensors
- Proprioceptive sensors

2.4.2.1. Light Sensors

In this section light sensors, which are used in mobile robotic applications, are investigated.

2.4.2.1.1. Photoresistors

Robot behaviors can be possible with light sensors such as detect obstacles and follow marks on the floor. Photoresistors are simple light sensors and easy to interface with a microprocessor. Photoresistors are simply variable resistors; the resistance is changed by a change in the light level.

2.4.2.1.2. Near-infrared Proximity Detectors

Many mobile robots are designed for following trajectories. These sensors are used to detect nearby objects of the robot. Unlike range finder, these sensors cannot provide actual distance to an object. They only sense whether there is something or not. The sensor perception ranges are smaller and the beam widths are narrower than sonar range finders.

The near-infrared proximity sensors are suitable for wall following applications. Wall following can be achieved by using two sensors; one pointed directly at the wall and the other one pointed at 45 degree or more forward (Figure 2.20). If either sensor detects any object, the robot arcs to the right until sensor B detects the wall and then robot moves forward. When sensor A detects any object, robot faces left.
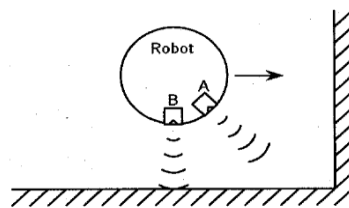
Figure 2.20. Wall following strategy with near-infrared proximity sensors
(Source: Jones, et al.)

2.4.2.1.3. Near-infrared Range Sensor

Proximity sensors are very popular, inexpensive and easy to use for simple mobile robot applications but they cannot give range information for nearby objects. Infrared sensors made by Sharp are useful for applications that need distance information between the robot and the object.

To calculate distance or perceive the objects in the sensors field of view, infrared range sensors use triangular and small linear Charge Coupled Device (CCD) array. Infrared sensors are capable of realizing this through emitting a pulse of infrared (IR) light with emitter. IR light travels out in the sensors field of view. If the IR light is not reflected from anywhere, this refers that there is no object. On the contrary, if IR light reflects from an object and returns to the detector, a triangle between point of reflection, the detector and the emitter is formed.

If the distance of the objects changes, triangle shape will also changes. The distance between the object and the sensor can be determined by calculating the angle between the reflected light coming back from the object and the sensor (Figure 2.21).
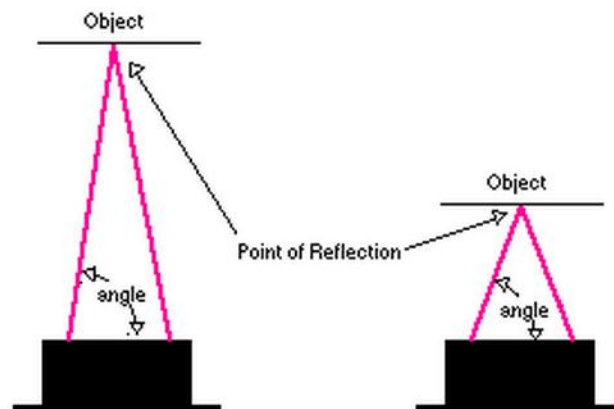
Figure 2.21. Different Angles with Different Distances
(Source: Acroname 2012)

2.4.2.2. Sound Sensors (Sonars)

The most common near infrared detectors only give proximity information, on the other hand, sonar transducers are used for providing distance information. This can

be possible by calculating the time between the initiation of a ping (pulse of sound) and the return of its echo (Figure 2.22). Then, the distance is calculated by round-trip time of the ping and speed of sound in the air.
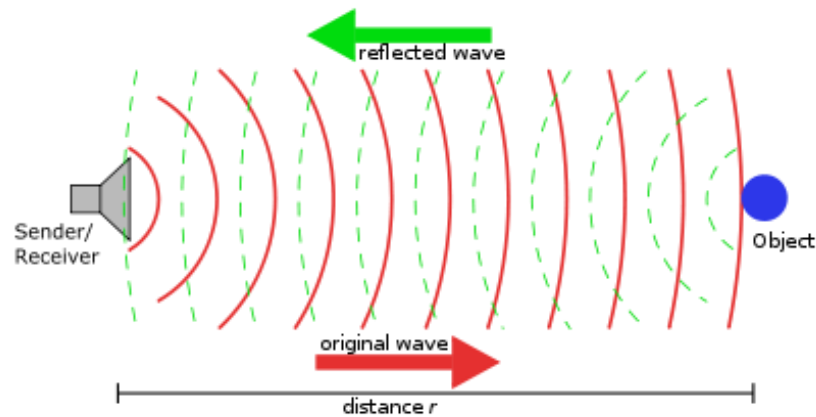
Figure 2.22. Principle of sonar
(Source: Wikipedia 2012)

2.4.2.3. Odometry Sensors

Making measurements of velocities of the wheels or other localization components are necessary to know about robot position and orientation in order to find where the robot is.

2.4.2.3.1. Shaft Encoders

Shaft encoder is a device which measures shaft speed of a motor (Figure 2.23). They are especially mounted on drive motor shaft from outside. There are two types of shaft encoders. One of these, called absolute encoders, deliver a signal which is a code form that corresponds to particular orientation of the shaft. The other type of encoder, called incremental encoder, produces a pulse train. The rate of pulse frequency indicates motor shaft velocity.

Instead of absolute position encoders, potentiometers can be used. Potentiometers are composing of a resistance and the amount resistance is changed for

each position of the shaft. Especially, in robot arms, absolute encoders and potentiometers are utilized for defining the position of the links.
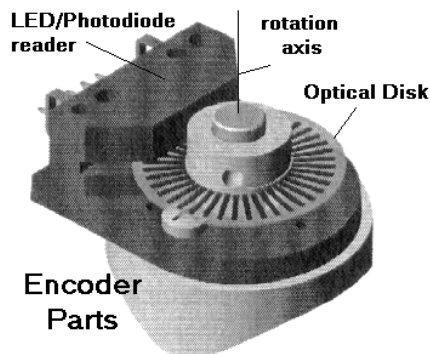
Figure 2.23. Shaft encoder
(Source: Bogan 2012)

Incremental encoders can be either assembled with motor or apart units. Incremental encoders consist of a spinning disk that has slots in it, near-infrared LED and phototransistor. The disk is assembled with motor shaft and spins with it. The near infrared sensor is mounted on one side of the disk and the phototransistor is on the other side. While disk is rotating the light created by LED is interrupted by slots on the disk. This discontinuous light is collected by the phototransistor and then pulse train is produced as output of the phototransistor. These pulses are counted by using microprocessor and by this way, rotational speed of the robot wheel can be calculated.

2.4.2.3.2. Gyroscope

To monitor robot orientation, gyroscopes (gyros) are very useful devices (Figure 2.24). Gyros can be mechanical or electronic. Mechanical gyros work with principle of conservation of angular momentum to keep one or more internal axes pointed in the same direction as the exterior of the gyroscope, the gyroscope case, translates and rotates. Therefore, by using a gyroscope, robot rotational speed and rotation angle with respect to a fixed coordinate system can be measured. Same as mechanical gyros, electronic gyros can create a signal related with the rate of rotation about a perpendicular axis with axis of the gyro. However, gyros cannot present absolute

orientation information. Electronic gyros can give an analog signal or pulse width modulated (PWM) signal based on the rate of rotation of the gyroscope case.

Figure 2.24. Gyroscope
(Source: Heliplane 2012)

2.4.2.3.3. Accelerometer

The basic accelerometer work can be explained with basic mass-spring-damper system (Figure 2.25). An spring suspend a proof mass and the damper controls ringing. Upon acceleration of the base frame, the spring must provide a force to make the proof mass keep up, and spring deflection is gotten as measure of acceleration. Therefore, these force-measuring instruments solve Newton's Equation to deduce acceleration;

$$F = m \cdot a \tag{2.3}$$

where ''m'' is the mass and ''a'' is acceleration of the sensor.
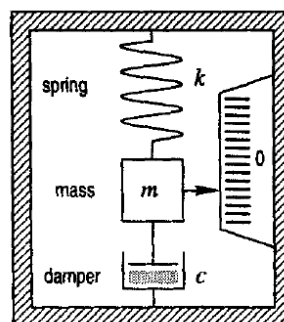
Figure 2.25. Basic accelerometer
(Source: Jones, et al.)

Although practical accelerometers vary in design and technology, they are all based on the Equation $F = m \cdot a$ in some way. They can be electromagnetic, vibrating string, gyro-pendulum, optical, piezoresistive, piezoelectric, capacitive.

2.4.2.4. Proprioceptive Sensors

Any sensors, which measure the internal state of the robot, are called proprioceptive sensor. These sensors give information about recharge of the batteries, motor temperature or overheating. Batteries are very important parts of the mobile robots because robot working time changes with changing of battery performance. Therefore, sensing robot battery voltage is a significant issue for the mobile robot applications.

By monitoring motor currents, robot can determine whether it is in impasse or not. If the robot collides with an obstacle such a wall, monitoring currents can be useful while any other sensors used for obstacle and collision avoidance, such as near-infrared sensors and sonar, are not working. While robot crashes with an object, wheels will not turn and current will go to a maximum value. In this situation, monitoring current can be used as a collusion detector.

Additionally, observing robot temperature keep robot parts from breakdown. Especially electronic parts may be damaged, if the temperature is high. Besides, electric motor performance and life dramatically decrease in high level of temperatures.

2.4.3. Batteries

Mobile robots need power source to perform the operation in the remote environment. The power source quantity must be enough to allow the robot to perform a useful amount of work. A constant voltage provided by power system is needed to ensure proper operation of the onboard electronic circuit.

Batteries are the most common energy source for mobile robots. A battery changes chemical energy into electrical energy on demand. There are different types of batteries and all types have complex variety of properties due to their different chemical nature.

While selecting a battery for the task, the following properties of the batteries can be considered:

- Rechargebility: Batteries can be rechargeable or not. While the battery selection, especially rechargeable batteries should be used as primary unit.
- Energy density: The maximum amount of energy per unit mass is known as energy density. Energy density is usually measured in units of Watt-hours/kilogram (Wh/kg). Alternatively, energy density can be measured in units of energy per unit volume.
- Capacity: The energy stored in a cell is known as battery capacity. Battery capacity unit can be shown as amp-hours or milliamp-hours. Energy density and the mass of the battery change the capacity.
- Voltage: Characteristic of the particular chemical reaction occurring in the battery causes the voltage produced by a single cell. Additionally, voltage of the battery can be changed depending on the state of charge of the cell.
- Internal resistance: Internal resistance of the battery limits the current. The internal resistance increases as the battery discharges.
- Discharge rate: Discharge rate refers to discharge time in units of current. Internal resistance of the battery limits maximum discharge rate.
- Shelf life: Batteries lose their charge even when there is no external load. Shelf life points out how quickly this occurs.
- Temperature dependence: Temperature affects most battery properties, especially, battery capacity and shelf life.

An ideal battery should have very high energy density, low internal resistance and maintain a constant voltage during discharge. Also, it should have maximum withstanding of the higher temperature, unlimited shelf life, rechargeable and low unit cost. However, there is no battery with all of these properties. Therefore, batteries should be selected by depending on the requirements of the task.

2.4.4. Microcontroller

Microcontrollers (Figure 2.26) are small computers which have processor core memory and programmable I/O peripherals and they are combined small size, low power consumption and computational abilities of a cheap microprocessor. Also, they comprise serial line for communicating with host computers or a terminal, A/D converters, timers for taking events or starting hardware and pulse counters. Therefore, microcontrollers simplify system design, and today, they are used for many embedded applications such as robotic systems, automobile engine control systems, implantable medical devices, remote controls, office machines, appliances, power tools and toys.

Figure 2.26. A PIC 18F8720 microcontroller in an 80-pin TQFP package
(Source: Wikipedia 2012)

2.4.5. Wireless Communication Systems

In this section wireless communication systems and their application areas are explained.

2.4.5.1. Wireless Personal Area Network (WPAN)

A wireless personal area network (WPAN) is a computer network designed for communication between electronic devices over wireless network technologies such as IrDA, Bluetooth, Wireless USB, Z-Wave, ZigBee, or even Body Area Network.

Containment distance of a WPAN differs from a few centimeters to a few meters. There are two types of wireless technologies executing for WPAN; Bluetooth and Infrared Data Association.

Bluetooth is a wireless technology standard for sending and receiving data over short distances by using short-wavelength radio transmissions in the ISM band from 2400–2480 MHz. Also, this technology is used for creating wireless personal area network with high level security. Devices, which have Bluetooth technology, can be fixed on other devices like mobile phones or mobile devices. A serial Bluetooth module used in robotic applications can be seen in Figure 2.27.

Figure 2.27. Serial Bluetooth module
(Source: Robotshop 2012)

The other WPAN standard, which is widely used in robotic applications, is ZigBee. It is a small, low-cost and low power standard based on an IEEE 802. These specifications allow that longer communication life with small batteries and widely deployed wireless control and monitoring applications. In Figure 2.28, a device used ZigBee standard can be seen.

Figure 2.28. The XBee module (ZigBee Mesh) developed by Sparkfun Electronic
(Source: Sparkfun 2012)

2.4.5.2. Wireless Local Area Network (WLAN)

A wireless local area network (WLAN) contacts two or more devices by using spread-spectrum or OFDM radio wireless distribution methods, and usually supplies a connection through an access point to the Internet. This provides great mobility to move around within a local containment area without disconnected to the network. WLANs are based on IEEE 802.11 standards. The notebook computer connected via radio waves to the wireless access point can be seen in Figure 2.29.

Figure 2.29. A wireless access point connected with a notebook computer through a wireless PC card (Source: Wikipedia 2012)

2.5. Conclusion

Literature survey was conducted under four headlines. Firstly, omnidirectional indoor ground vehicles, their wheels providing omnidirectional mobility and usage areas were investigated. Secondly, force-reflecting bilateral teleoperation systems that use haptic device or force feedback joystick were investigated. Thirdly, obstacle avoidance algorithms and sensors used in these algorithms were presented. Finally, robotic parts for mobility and acquiring information about environment or inner states of the robot were investigated. The survey results are evaluated and necessary parts are selected to configure the omnidirectional mobile platform as the slave systems. The text chapter is reserved for the methodology to develop the unlimited-workspace teleoperation system by utilizing the results of this chapter.

CHAPTER 3

METHODOLOGY

Before designing and developing unlimited-workspace teleoperation system, requirements of the system should be defined. Therefore, first of all, the design criteria of unlimited-workspace teleoperation system are identified. Then, the conceptual designs are described and compared with each other in order to select the most appropriate design with respect to the determined design criteria. After selecting the best design among the conceptual designs, components are selected and the first prototype is manufactured.

3.1. Design Criteria

The aim of the study is to design an unlimited-workspace teleoperation system. Additionally, the system should be suitable for force reflecting bilateral teleoperation. To realize a force reflecting bilateral teleoperation system, it needs a master device, which is a conventional force feedback joystick or a haptic device, to transmit environmental effects to a human operator. As the slave subsystem, omnidirectional mobile platform is selected, because it has great mobility than other conventional wheeled platforms (e.g. car-like Ackerman steering or differential drive system). All the desired characteristics of the unlimited-workspace teleoperation system are listed below:

- Developed mobile platform should be suitable for indoor applications

- Mobile platform should have omnidirectional motion

- Mobile platform should have fault tolerance in mechanism level

- Mobile platform should perceive environmental phenomena and transmit it to the master side

- Communication line should be a wireless communication system

- Information should be mutually provided among subsystems

3.2. Slave Subsystem

In this section, a conceptual design is selected to build the first prototype and the selected components of the slave subsystem are explained.

3.2.1. Conceptual Design

To provide omnidirectional mobility, two types of wheels can be used: conventional wheels (castor or steering) and special wheels (universal, mecanum or ball wheels). Comparisons of the omnidirectional wheels are listed in Table 3.1.

Table 3.1. Comparison of omnidirectional wheels

	Conventional Wheels	Special Wheels
Providing mobility	medium	high
Load capacity	high	low
Tolerance of ground irregularities	high	medium
Flexibility in congested environments	low	high
Traction performance	high	low

When considering the characteristics of the omnidirectional wheels, special wheels have more flexible motion capability than the conventional ones. Thus, universal wheel, which is a type of special wheel, is chosen to provide locomotion. In the literature, there are mainly two designs to provide omnidirectional mobility with universal wheels; triangular (Figure 3.1) and square configuration (Figure 3.2).
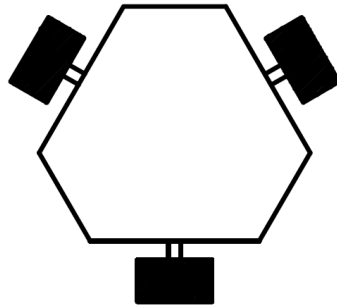
Figure 3.1. Triangular configuration of omnidirectional platform with universal wheels
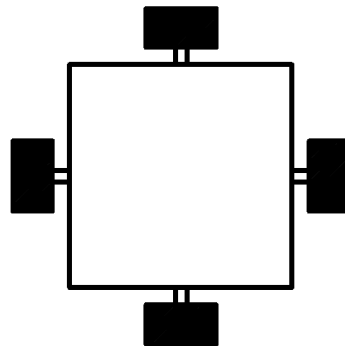
Figure 3.2. Square configuration of omnidirectional platform with universal wheels

Omnidirectional mobility can be provided with three universal wheels. This design is mainly used in soccer game robots in RoboCup. Its fault tolerance capability is limited, because if any wheel or motor is faulty during the task, it loses omnidirectional capability and it can carry out the task like a conventional two wheeled differential drive system. On the other hand, the four wheeled system has mechanism level redundancy (kinematically redundant), and it has greater fault tolerance capacity. If any motor or wheel is faulty, it performs the task with three wheels without losing omnidirectional mobility capability. Hence, square configuration is selected as it can increase the level of fault tolerance in the system.

Another important issue that affects performance of the robot is the shape of the mobile platform. It is difficult to avoid obstacles and navigate through narrow aisles with square shaped robot. On the other hand, cylindrical shaped robot has inherent advantage for these considerations and it is possible to achieve this with a simpler

algorithm. Hence, it can be concluded that cylindrical shape is the most suitable robot shape with respect to the design criteria set in this study. In the thesis, another focus is on creating obstacle avoidance algorithm so robot shape is chosen octagonal shape to adduct cylindrical shape for ease of manufacturing process while the vehicle can still easily navigate through obstacles and narrow passages (Figure 3.3).
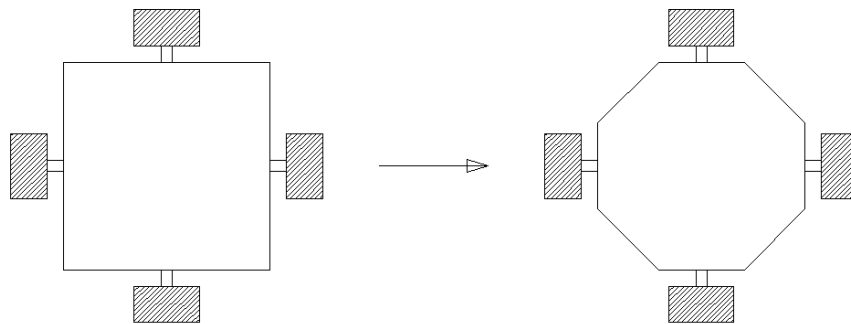
Figure 3.3. Left: Square shaped platform Right: Modified octagonal shaped platform

3.2.2. Locomotion Components

Transwheels produced by Kornylak Corp. are used (Figure 3.4) for configuring the locomotion system in the mobile platform. It has a special design with eight free rollers mounted at 90 degrees to the axis of rotation of the wheel around the outer diameter of the wheel. Combination of rotation of the wheel body and rollers movement enables the ability to move in any direction. These wheels are mainly used in conveyors but, they are also used in many applications such as mobile robots, multidirectional turn tables, and skate-wheels due to their special movement ability. However, in robotic applications, it causes slip problem because of its hard plastic nature. This fact results in traction problems. Specifications of Transwheels are listed below:

- Lightweight wheels
- High impact plastic
- Self-lubricating
- Low friction coating
- Stainless steel axles

- Washable even steam cleaning
- Corrosion resistant
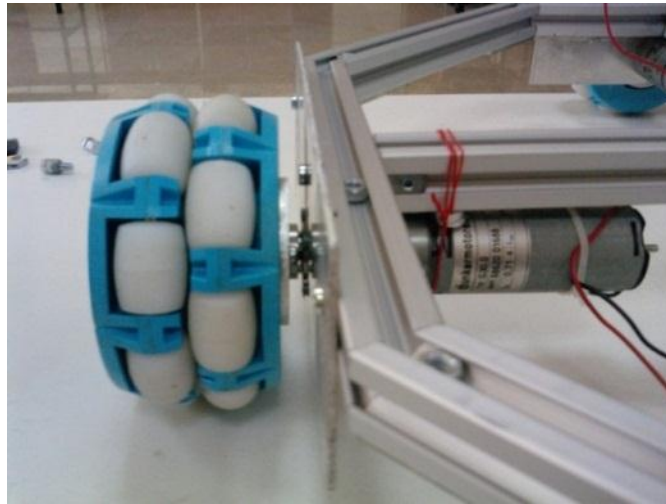
Figure 3.4. Transwheel

As the driving motors, Dunkermotoren 24V G 30.0 model brushed DC motors, which are integrated with planetary metal gearboxes (PLG32 model) whose reduction ratio are 20.5:1, are used. Specifications of the motor and gearbox are shown in Table 3.2 and Table 3.3.

Table 3.2. Specifications of the G 30.0 DC motor

Rated voltage	VDC	24
Continuous rated speed	rpm	3030
Continuous rated torque	Ncm	3
Continuous current	A	0,71
Starting torque	Ncm	12,1
Starting current	A	2.5
No load speed	rpm	4260
No load current	A	0.13
Rotor inertia	gcm	42.2
Weight of motor	g	240

Table 3.3. Specifications of PLG32 gearbox

Reduction Ratio		1/20.5
Efficiency		0.81
Number of Stages		2
Continuous Torque	Ncm	150
Weight of gearbox	g	0.18
Axial loads	N	30/100

The motor characteristic curves for 24V input voltage are displayed in the Figure 3.5. In the figure, the change of torque versus the current supplied is shown with black line. Torque can be calculated trough Equation (3.1), which is deduced from the diagram;

$$\tau = 4{,}96 * i \tag{3.1}$$

where τ is torque, i is current and 4.96 is the motor torque constant.
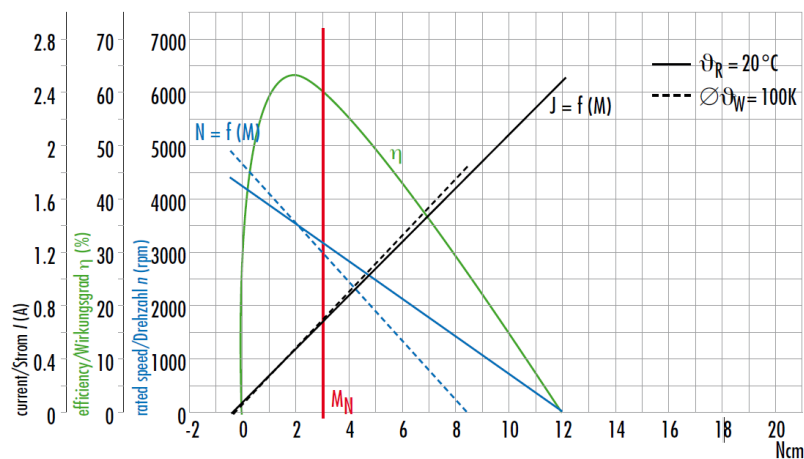
Figure 3.5. Characteristic diagram of G30.0 DC motor

Maxon ADS 50/10 (Figure 3.6) model DC motor amplifiers were selected to drive DC motors. This amplifier has four control modes:

- Speed control using tachometer signals
- Speed control using encoder signals

- IxR compensated speed control
- Torque or current control.

Speed control mode with using tachometer or encoder can be used when the motor is equipped with these sensors. The typical use of this case is in odometry related applications. In this study, during the tests, motors were driven with current and IxR modes to accommodate the use of kinematic and dynamic control. Specifications of the motor driver are listed in Table 3.4.

Table 3.4. Specifications of Maxon ADS 50/10 motor amplifier

Electrical data:	
Nominal supply voltage $+V_{cc}$	12 to 50 VDC
Absolute minimum supply voltage $+V_{cc\ min}$	11.4 VDC
Absolute maximum supply voltage $+V_{cc\ max}$	52.5 VDC
Max. output voltage	$0.9 \cdot V_{CC}$
Max. output current I_{max}	20 A
Continuous output current I_{cont}	10 A
Switching frequency	50 kHz
Efficiency	95 %
Band width current controller	2.5 kHz
Built-in motor choke	75 μH / 10 A
Inputs:	
Set value: -10 to +10 V (R_i = 20 kΩ)	-10 to +10 V (R_i = 20 kΩ)
Enable: +4 to + 50 VDC (R_i = 15 kΩ)	+4 to + 50 VDC (R_i = 15 kΩ)
Input voltage DC tacho "Tacho Input"	min. 2 VDC, max. 50 VDC (R_i = 14 kΩ)
Encoder signals "Channel A, A\, B, B\"	max. 100 kHz, TTL level
Output:	
Current monitor "Monitor I", short-circuit protected	-10 ...+10 VDC (R_O = 100 Ω)
Speed monitor "Monitor n", short-circuit protected	-10 ...+10 VDC (R_O = 100 Ω)

(Cont. on next page)

Table 3.4. (Cont.)

Status reading "READY" Open collector, short-circuit protected	max. 30 VDC ($I_L \leq 20$ mA)
Voltage outputs:	
Aux. voltage, short-circuit protected	+12 VDC, -12 VDC, max. 12 mA ($R_O = 1$ kΩ)
Encoder supply voltage	+5 VDC, max. 80 mA

Figure 3.6. Maxon ADS 50/10 motor amplifier

Connectors on the motor driver are illustrated in Figure 3.7. Connectors on the driver from one to five are to supply power for the motor. Motor voltage or current are defined by the set value input. The input voltage should be supplied from differential amplifier and the voltage range must be among -10V to +10V. Motor rotation direction is determined by two set value states;

- Positive set value: (+ Set Value) > (- Set Value) negative motor voltage or current motor shaft turns counter clockwise

- Negative set value: (+ Set Value) < (- Set Value) positive motor voltage or current motor shaft turns clockwise

If a voltage is given at "Enable", the servo amplifier switches the motor voltage to the winding connections. If the "Enable" input is not switched on or is connected to the Ground (Gnd), the power stage will be highly resistant and will be disabled.
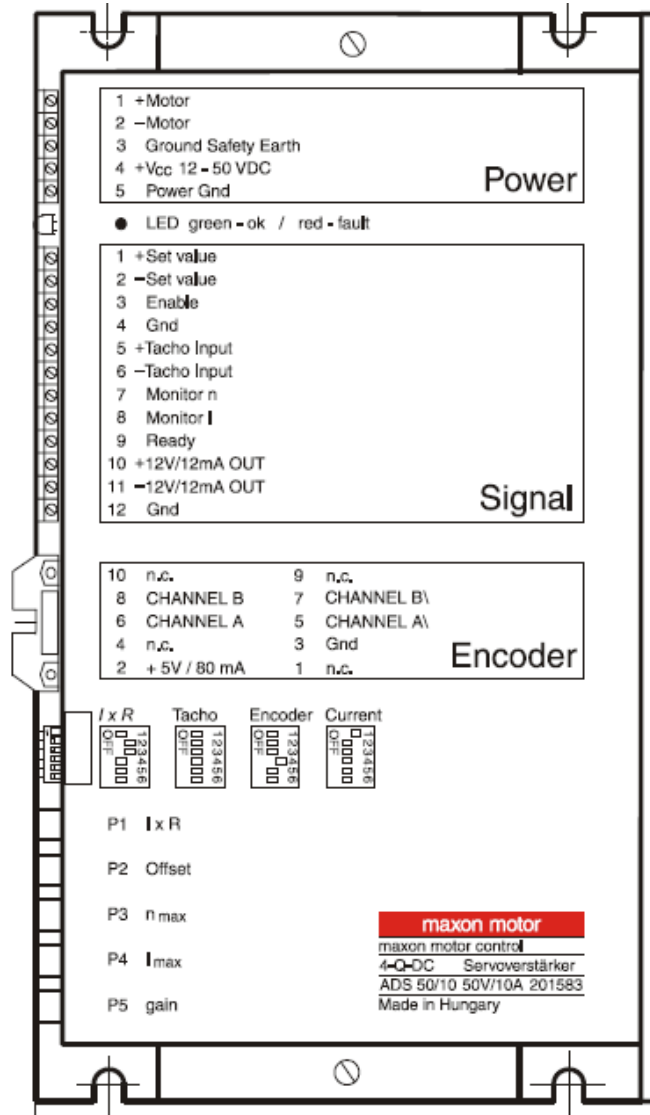
Figure 3.7. Connectors on the motor driver

3.2.3. Data Acquisition

In this study, Prometheus embedded PC/104 CPU (Figure 3.8) developed by Diamond System Corporation is used as data acquisition device. It integrates 3 separate circuits onto a single compact board; CPU, Ethernet and Analog I/O.

Figure 3.8. Prometheus data acquisition device

Prometheus conforms to the PC/104 standard which is an embedded standard based on the ISA and PCI buses and provides a compact, rugged mechanical design for embedded systems. PC/104 modules feature a pin and socket connection system in place of card edge connectors, as well as mounting holes in each corner. The result is an extremely rugged computer system fit for mobile and miniature applications. PC/104 modules stack together with 0.6″ spacing between boards (0.662″ pitch including the thickness of the PCB). Specifications of the Prometheus are listed below:

Processor Section:

- 486-DX2 processor running at 100MHz with co-processor
- Pentium class platform including burst-mode SDRAM and PCI-based IDE controller and USB
- 32MB SDRAM system memory
- 50MHz memory bus for improved performance
- 2MB 16-bit wide integrated flash memory for BIOS and user programs
- 8KB unified level 1 cache

I/O:

- 4 serial ports, 115.2kbaud max
- 2 ports 16550-compatible, 2 ports 16850-compatible with 128-byte FIFOs

- 2 full-featured powered USB ports

- 1 ECP-compatible parallel port

- Floppy drive connector

- IDE drive connector (44-pin version for notebook drives)

- Accepts solid-state flashdisk modules directly on board

- 100BaseT full-duplex PCI bus mastering Ethernet (100Mbps)

- IrDA port (requires external transceiver)

- PS/2 keyboard and mouse ports

- Speaker, LEDs

System Features:

- Plug and play BIOS with IDE autodetection, 32-bit IDE access, and LBA support

- Built-in fail-safe boot ROM for system recovery in case of BIOS corruption

- User-selectable COM2 terminal mode

- On-board lithium backup battery for real-time-clock and CMOS RAM

- ATX power switching capability

- Programmable watchdog timer

- Power surge monitor for fail-safe operation

- Zero wait-state capability for flash memory and PC/104 bus

- +5V-only operation

- Extended temperature range operation (-40 to +85oC)

- Cable-free operation when used with Diamond Systems' PNL-Z32 Panel I/O board

Analog Input:

- 16 single-ended / 8 differential inputs, 16-bit resolution

- 100KHz maximum aggregate A/D sampling rate

- Programmable input ranges/gains with maximum range of ±10V / 0-10V

- Both bipolar and unipolar input ranges

- 5 ppm/oC drift accuracy

- Internal and external A/D triggering

- 48-sample FIFO for reliable high-speed sampling and scan operation

Analog Output:

- 4 analog outputs, 12-bit resolution
- ±10V and 0-10V output ranges
- Simultaneous update
- Adjustable output range (optional)

Digital I/O:

- 24 programmable digital I/O, 3.3V and 5V logic compatible
- Enhanced output current capability: −8/+12mA max

Counter/Timers:

- 1 24-bit counter/timer for A/D sampling rate control
- 1 16-bit counter/timer for user counting and timing functions
- Programmable gate and count enable
- Internal and external clocking capability

Prometheus uses the PCI bus internally to connect the Ethernet circuit to the processor. It uses the ISA bus internally to connect serial ports 3 and 4, as well as the data acquisition circuit, to the processor. Only the ISA bus is brought out to expansion connectors for the connection of add-on boards. Prometheus is a PC/104 add-on board for analog I/O, digital I/O, counter/timer functions, serial ports, and power supply. Data acquisition scheme of the mobile platform and used ports and I/O connectors of the Prometheus are illustrated in Figure 3.9. First four digital I/O connectors are used for Hall-effect sensors and four analog outputs are used motor amplifier to drive motors. Also, first five analog inputs are utilized for infrared range finders and sixth analog input is utilized for gyroscope.
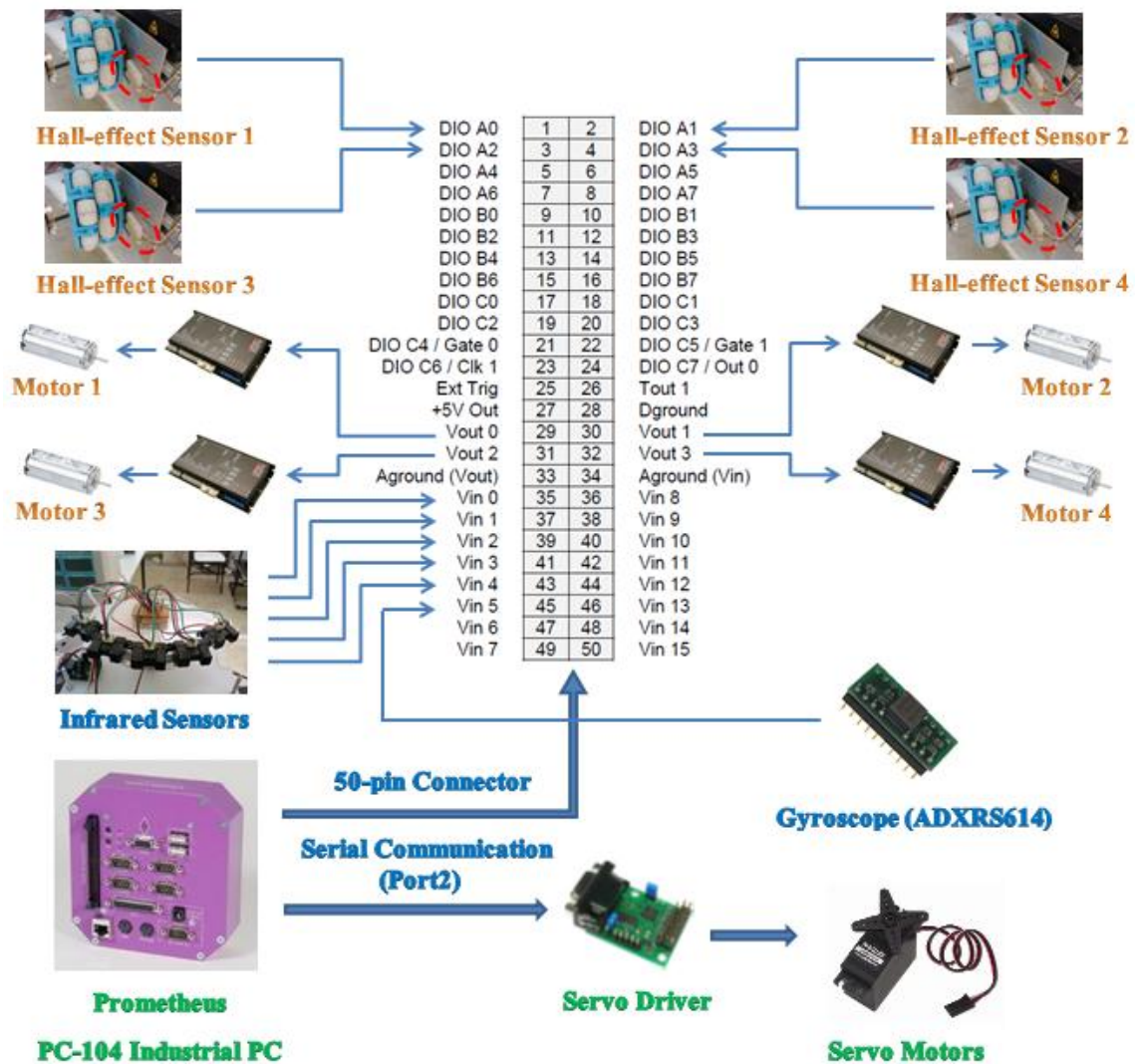
Figure 3.9. Data acquisition scheme of the slave subsystem

3.2.4. Obstacle Avoidance System

For obstacle detection, fixed sensors are generally used on the mobile platforms. In the most applications, density of the sensors on the front side of the mobile robot is more than the other sides. As a result of this, a lot of sensors are used in these applications and density of the sensors in each side is not homogeneous. This condition may be not a problem for platforms with differential drive system. However, this way of sensor distribution is not suitable for the omnidirectional vehicles, because the omnidirectional mobility ability allows moving every direction independently. Thus, in this thesis, a dynamic obstacle detection system, which dynamically changes its sensing

direction according to movement direction of the platform, is developed. Developed dynamic obstacle detection system consists of three components; two servo motors (1), which are mounted on same rotation axis to cover 360 degrees of rotation range, a servo driver (2) and five infrared sensors (3). The servo motors re-orient the array of sensors with respect to the direction of motion (Figure 3.10).
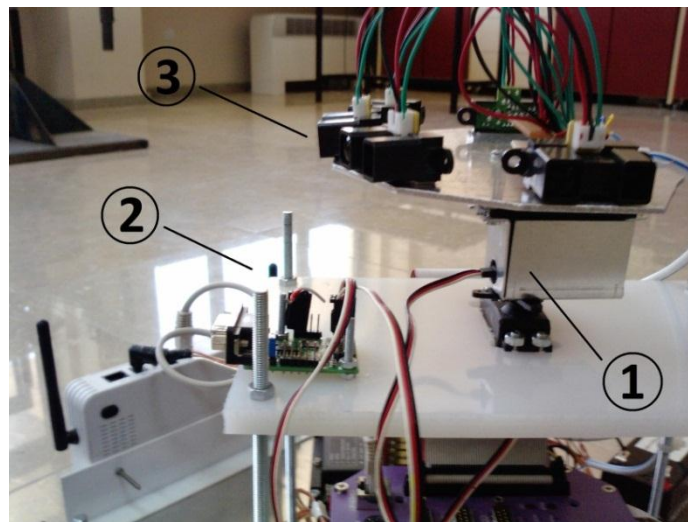
Figure 3.10. Dynamic obstacle detection system

3.2.4.1. Servo Motors

In the obstacle avoidance system, two hobby servo motors produced by Parallax, whose rotation range were 180°, were used. Other specifications of the servo motor are listed below;

- Power requirements: 4 to 6 VDC; Maximum current draw is 140 +/- 50 mA at 6 VDC when operating in no load conditions, 15 mA when in static state
- Communication: PWM, 0.75–2.25 ms high pulse, 20 ms intervals
- Dimensions approx. 2.2 x 0.8 x 1.6 in (5.58 x 1.9 x 40.6 cm) excluding servo horn
- Operating temperature range: 14 to 122 °F (-10 to +50 °C)

These servo motors are controlled through PWM and the position of the servo shaft is dependent on the duration of the pulse. In order to hold its position, the servo needs to receive a pulse every 20 ms. When pulse duration of duty cycle is 1ms, the

servo is positioned at left, when 1.5 ms, the servo is positioned at center and when 2 ms, the servo is positioned at right side. In Figure 3.11, a sample timing diagram for the center position of the Parallax standard servo is illustrated.
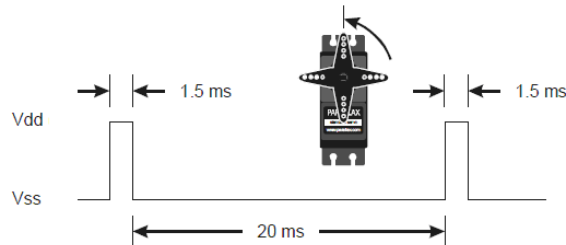
Figure 3.11. Sample timing diagram for the center position of the servo motor

3.2.4.2. Servo Motor Driver

Servo motors are driven through PWM signal, so in the obstacle avoidance system, a servo motor controller produced by Pololu (Figure 3.12) was utilized to control the position of the servos through generating PWM signals.

The servo motor controller requires a logic-level (0-5 V) serial input connected to the logic-level serial input or an RS-232-level serial input at the DB9 connector. The servo controller echoes all serial data out of the serial output pin, and the reset line can be brought low to reset the servo controller. In most applications, the reset input and serial output is left disconnected.
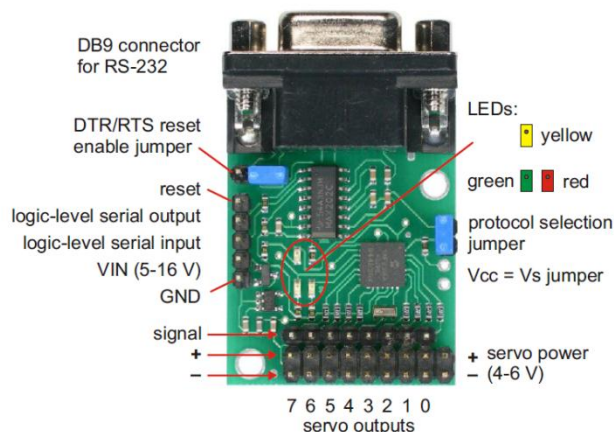
Figure 3.12. 8 servo driver produced by Pololu

Eight servos can be controlled with the servo controller through generating 8 independent servo control signals. The controller can generate pulses from 0,25 ms to 2,75 ms which is greater than the range of most servos. In addition, it allows for servo operating range of over 180 degrees.

The servo controller uses two communication protocols; Pololu mode and Mini SCC II mode. These protocols can be chosen with changing state of mode selection jumper.

- Pololu mode: Pololu protocol used for controlling multiple devices is active when jumper is open (default mode). In this mode servo controller can be on the same serial line as other devices. Additionally, this mode allows access to all of the special features of the servo controller, such as setting speeds, ranges and neutral settings.

- Mini SSC II Mode: This mode is set by placing the shorting block over the two jumper pins. This setting allows the servo controller to respond to the protocol used by the Mini SSC II servo controller. This is more simple protocol but it only allows the user to specify the desired servo positions in only one way.

Mini SSC II Mode was chosen to control servos, because it is simpler control mode and suitable for Matlab Simulink serial communication blocks.

3.2.4.2.1 Mini SSC II Mode

Baud rate range is approximately 500-10k baud. However, this mode only supports at 2400 or 9600 bauds. Thus, to use Pololu 8-servo controller with Mini SSC II mode, one of two baud rates must be chosen.

Servo controller needs a sequence of three bytes (Table 3.5) to set to servo position. The first byte is for active the controller and its value always must be 255. The second byte refers to the servo number and it can be 0-254. The third byte brings servo position at the desired position and its value is 0-254.

Table 3.5. Three byte command to servo position

Byte Number	1	2	3
	Start Byte = 0xFF	Servo number, 0x00-0xFE	Servo Position, 0x00-0xFE

This mode allows two different motion ranges. Servo numbers can be from 1 to 16. The numbers, which are from 1 to 8, restrict servo range within an approximately 90 degree. Other eight numbers from 9 to 16 allows the 180 degree range. For example, if the sending command sequence is [255, 2, 254], servo 2 will move in range of 90 degree. If the servo number is changed as 10, servo 2 works in 180 degree range. In the obstacle avoidance system, two servos mounted on same rotation axis in order to complete 360 degree rotation range were used. Thus, as the servo numbers are selected from 9 to 16, due to control servos in 180 degree range.

3.2.4.3 Sensors

In the study, to detect near the mobile platform, Sharp GP2Y0A02YK0F (Figure 3.13) infrared sensor which is composed of an integrated combination of PSD (position sensitive detector), IRED (infrared emitting diode) and signal processing circuit was used. It is not affected by environmental temperature and the operating durations. Specifications of the sensor are listed below:

- Distance measuring range : 20 to 150 cm
- Analog output type
- Size : $29.5 \times 13 \times 21.6$ mm
- Consumption current : 33 mA
- Supply voltage : 4.5 to 5.5 V
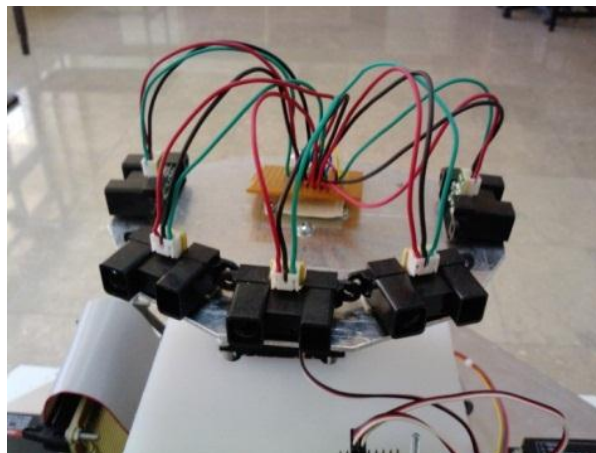
Figure 3.13. Sharp GP2Y0A02YK0F infrared sensor

These sensors use triangulation method to measure the distance among the object and the sensor. Detection distance is defined as the output voltage. The relation between output voltage and distance is shown in Figure 3.14.
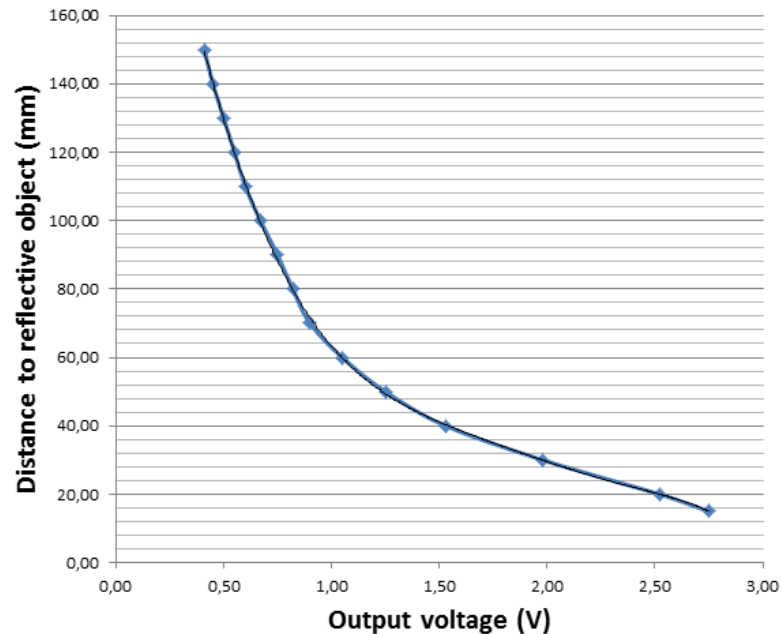
Figure 3.14. Output voltage related with the distance

3.2.5. Energy Supply

Energy supply is an important for mobile platforms since they perform the tasks in remote and mostly unreachable environments. Electrical energy supplies used in mobile applications can be classified into two main categories: rechargeable or non-rechargeable. In the thesis, rechargeable batteries are selected to be used to avoid battery costs. While selecting a battery for the project, three main properties of the batteries are considered:

- Voltage
- Capacity
- Dimensions

To estimate robots operation time, power consumptions of equipment on the mobile robot are required to know. Energy requirements of the components are listed in the Table 3.6.

Table 3.6. Power consumptions of electronic components

	Number	Voltage (V)	Current (A)	Power (W)
Motor	4	24	0.71	68 W
DAQ	1	5	1	5 W
Infrared Sensor	5	5	0.05	1,25 W
Camera				10 W
Modem				9 W
Servo Motor	2	6	0.14	1,68 W
Servo Driver	1	6	0.1	0,6 W
			Total =	98,53 W

Power supplies should be rated at 24V to drive motors while other electronic components should be supplied with 5V or 9V. Because of different voltage needs, power suppliers are separated for driving motors and for use of other electronic components. The two 12V-1,3Ah dry accumulators connected serial to provide 24V are used to drive motors (Figure 3.15) in spite of providing with one 24V accumulator, because, accumulator that mounted on one side of the platform can alter the center of mass of the platform. The two accumulators placed on each side provide balance in that respect. Estimation of operation time with these batteries is calculated as;
The entire specifications of battery pack are:

- 12V x 2 = 24 V
- 1,3 Ah
- 24V x 1,3 Ah = 31,2 Wh

Estimated operation time = 31,2Wh / 68W = 0,46h or 28 min.

Figure 3.15. 12V 1,3Ah dry accumulator

Other electronic components generally require 5V except the modem (9 V). Therefore, Ttec Plus MP3450 mobile battery which provides two different output voltage to be selected between 5-19V was used for supply power for electronic components. Specifications of this battery are listed below:

- Total battery capacity: 13800mAh / 50W (2300mAh, 6s1p)
- Battery type: Chargeable Lithium Polymer
- USB output port: 5V/ 1A
- Main output port: constant and selectable 8 section (5V, 6V, 7.5V, 9V, 12V, 14V, 16V, 19V)
- Output current: 5~9V/ 3A, 12~14V/ 3.5A, 16~19V/ 4A

3.2.6. Measuring Wheel Velocities

UGN 3113 Hall-effect sensors were mounted on chassis (Figure 3.16) to measure velocity of each wheel, because, there is no shaft encoder on the rear end of the motors. Eight Neodymium magnets (Figure 3.17) were attached on each wheel with an angle between them to be 45°. When Neodymium magnet passes across the Hall-effect sensor, the output signal is produced by the hall effect sensor raises. When the output is connected to a digital input, it produces digital ON signal at the instant when the magnet passes near the sensor. The time between two rises of the duty cycle gives 1/8

revolution time. Angular velocities of the wheels are calculated in real time with this revolution time information.
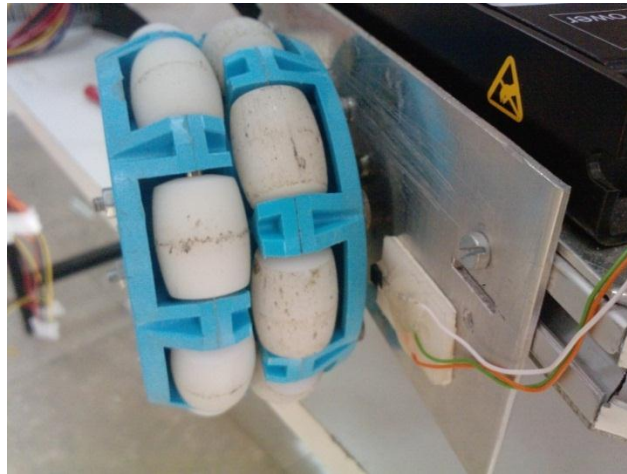
Figure 3.16. UGN 3113 hall-effect sensor

Figure 3.17. Neodymium magnets

3.3. Communication Line

In this thesis, communication line between the master and the slave subsystem is realized with Matlab/xPC target toolbox. Firstly, the connection between the host and target PC should be materialized to provide the communication line. Connection can be realized in two ways. The first is serial communication with RS232 cable, which is attached on serial ports on host and target PC. The second is network communication.

Network communication can be materialized with LAN, the Internet, or a direct connection using a crossover Ethernet cable. Both the host and target computers are connected to the network with Ethernet adapter cards using the TCP/IP protocol for communication. Network communication is more advantageous than the serial communication, because it has higher data transfer up to 100 Mbit/second and Longer containment area among host and target computer.

Because of these advantages, the network communication method is chosen for transferring data and wireless communication is achieved with using wireless modem which has router property in slave side and wireless adaptor in master side. In the beginning of the study, to execute connection between the host and target PC and to build developed control models at target side from host pc, target pc was booted with xPC target kernel for selected communication configurations. However, this connection method causes that robot gives slow responses for operators demands since building process of the model and connection among subsystems are performed over same line. Thus, the developed control models are embedded at slave side with xPC target embedded option and connection between subsystems are achieved with using UDP data transfer blocks in target and host models developed in Simulink. Process of embedding model is explained in Appendix B.

Figure 3.18 shows the UDP Send and Receive block parameters. Firstly, to provide communication among subsystems with UDP blocks, IP addresses should be adjusted correctly depending on chosen IP addresses. Then, port number is determined to match send and receive blocks mutually. After this, it will be ready to send and receive data between subsystems.
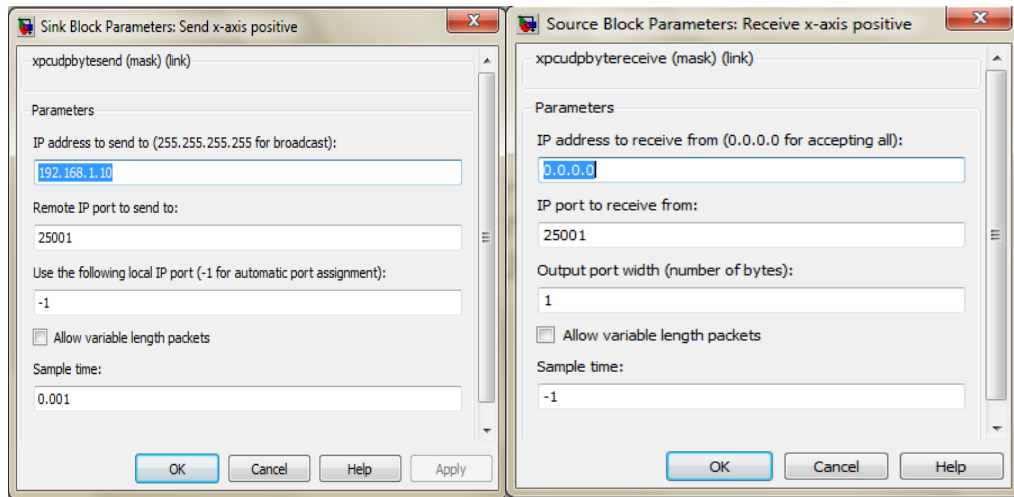
(a)   (b)

Figure 3.18. UDP Send and Receive block parameters; (a) UDP Send, (b) UDP Receive

3.4. Master Subsystem

Master subsystem acquires operator's demands and transmits feedbacks sent from slave side through a device that is handled by operator. In this study user interface was developed by using Matlab Simulink. Master subsystem consists of three parts; host PC (1), software (2), and master device (3) (Figure 3.19). One of aims of the study is to develop a force-reflecting bilateral teleoperation system. Therefore, force-reflecting joystick was used as hand controller to transmit force feedback information to the operator and to acquire operator's demands to drive the slave system.
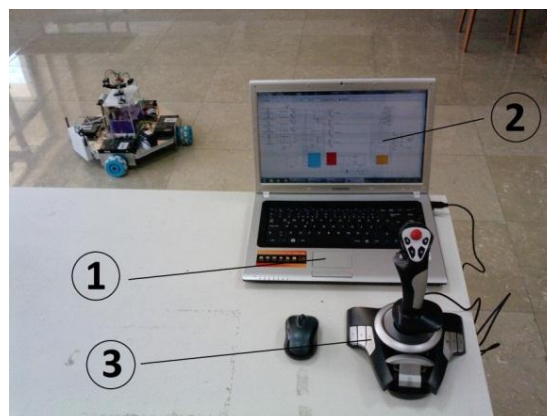
Figure 3.19. Master subsystem

As the force-reflecting joystick, Logitech Force 3D Pro force feedback was used (Figure 3.20). This joystick can acquire motion demand in three axes. This device is capable of providing force feedback in two axes. In addition, it has twelve buttons which can be utilized for controlling other components of the slave subsystem.

Figure 3.20. Force 3D Pro Joystick

3.5. Conclusion

In this Chapter, the unlimited-workspace teleoperation system development was explained. The three-DOF omnidirectional mobile robot that has different types of sensors to avoid obstacles and measure inner states of the mobile platform was developed as the slave subsystem. Additionally, communication line was provided to send sensory information from the slave subsystem to master and send demands from master to slave. The communication line was realized as a wireless communication system. In the master side, three-DOF force feedback joystick, which has force feedback property on two DOF, was used to acquire driving demands for slave subsystem and stimulate the operator for sensed obstacles on the path of the slave. On the other hand, conceptual design that has mechanism-level fault tolerance was selected as the mobile platform. In the next chapter information about the local controllers developed for the slave subsystem is introduced.

CHAPTER 4

CONTROL AND OBSTACLE AVOIDANCE ALGORITHMS

As described in Chapter 3, to realize unlimited-workspace teleoperation system, a prototype with four-wheeled square-configuration omnidirectional mobile platform was developed. Omnidirectional platforms allow motion in every direction at any orientation independently. They provide great mobility performance than the conventional mobile platforms. Additionally, it is easy to avoid obstacles and navigate through narrow aisles without getting trapped. A dynamic obstacle detection system appropriate for omnidirectional motion was developed to acquire information about the obstacles on the path of the robot.

To create control algorithms or to design desired tasks such as obstacle avoidance system, firstly, mechanical behavior of the platform must be understood mathematically. In this Chapter, firstly, the Equations of motion were formed for the developed platform. Then, two control algorithms to control the navigation of the platform and two obstacle avoidance algorithms were developed and modeled in Matlab Simulik. These algorithms were integrated to the developed teleoperation system.

4.1. Dynamic Equation of Motion

Overall system of square configuration omnidirectional vehicle is illustrated in Figure 4.1. It has four independent universal omnidirectional wheels mounted at 90 degrees with respect to each other around the vertical axis.
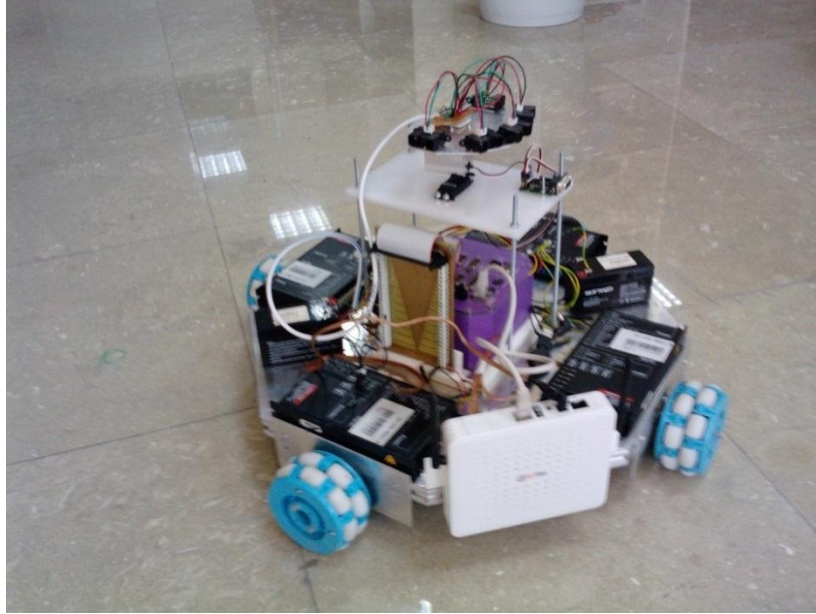
Figure 4.1. First prototype of four wheeled omnidirectional vehicle

Figure 4.2 represents free-body diagram of the platform motion with respect to the global coordinate frame. The global coordinate frame unit vectors are denoted with x_G and y_G. The local coordinate frame unit vectors of the vehicle are shown with x_m and y_m. The angle θ is the vehicle orientation, which is positive in the counter-clockwise direction and also shows the rotation of the local coordinate frame with respect to the global coordinate frame. m is total mass of the vehicle and I is the moment of inertia with respect to the center of gravity which is assumed to be at the center of the platform. T_i is traction force of the wheels where $i = 1$ to 4.

Some assumptions are made to simplify the Equation of motion. Firstly, the center of the vehicle is assumed to be at the same location with center of mass. The second assumption is that all link lengths, which are the distances between the center of mass and the center of wheels, are equal:

$$L_1 = L_2 = L_3 = L_4 = L \tag{4.1}$$

The last assumption to derive Equation of motion, the most important one is that vehicle moves on the ground without any slip.

Traction forces of the vehicle on the local coordinate frame are derived as:

$$f_x = T_4 - T_2 \tag{4.2}$$

$$f_y = T_3 - T_1 \qquad (4.3)$$

where the f_x is traction force along the local x axis, and f_y is traction force along the local y axis.
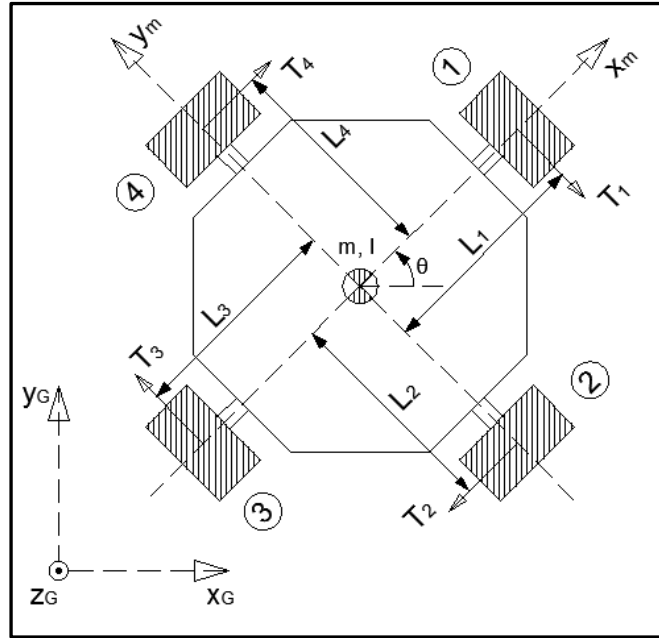
Figure 4.2. Geometry of square configuration platform

In addition, traction forces on the local coordinate frame are found from Newton's second law:

$$f_x = m \cdot \ddot{x}_m \qquad (4.4)$$

$$f_y = m \cdot \ddot{y}_m \qquad (4.5)$$

where m is mass of the platform, $\ddot{x}_m$ is acceleration along local x axis and $\ddot{y}_m$ is acceleration along local y axis.

Then Equations (4.2) and (4.3) are substituted into Equations (4.4) and (4.5) respectively:

$$m \cdot \ddot{x}_m = T_4 - T_2 \qquad (4.6)$$

$$m \cdot \ddot{y}_m = T_3 - T_1 \qquad (4.7)$$

Angular acceleration about local vertical axis is derived as:

$$I \cdot \ddot{\theta} = -L (T_1 + T_2 + T_3 + T_4) \qquad (4.8)$$

where I is the moment of inertia calculated about the center of gravity and $\ddot{\theta}$ is the angular acceleration about the vertical axis. Moment of inertia and mass properties of the vehicle is given in Appendix A.

Equations (4.6), (4.7) and (4.8) are the Equations of motion in local coordinate frame. To transform these Equations from the local coordinate frame to the global coordinate frame, transformation matrix $^{G}R_m$ is:

$$^{G}R_m = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad (4.9)$$

The dynamic Equations of motion in global coordinate frame are calculated as:

$$\begin{bmatrix} m * \ddot{x}_G \\ m * \ddot{y}_G \\ m * \ddot{z}_G \end{bmatrix} = {}^{G}R_m \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} \qquad (4.10)$$

Then, dynamic Equations of motion in global coordinate frame are:

$$m \cdot \ddot{x}_G = cos\theta (T_4 - T_2) - sin\theta (T_3 - T_1) \qquad (4.11)$$

$$m \cdot \ddot{y}_G = sin\theta (T_4 - T_2) + cos\theta (T_3 - T_1) \qquad (4.12)$$

$$I \cdot \ddot{\theta} = -L (T_1 + T_2 + T_3 + T_4) \qquad (4.13)$$

4.2. Control of the Platform

To control developed vehicle, desired velocity is generated via hand controller in the master side and sent slave side through communication line. Then, motors are driven

by control algorithm according to desired velocity in the slave side. To reach the desired velocity of the vehicle, the two control algorithms were designed to control the platform: kinematic control and dynamic control.

4.2.1. Kinematic Control

In this control algorithm, angular velocities of the each wheel are calculated in order to reach desired velocity of vehicle. Linear and angular velocities of four wheels are illustrated in Figure 4.3. Equations of the velocities of the robot according to global coordinate system are stated below:

$$V_x = V_1 . sin\theta - V_3 . sin\theta - V_2 . cos\theta + V_4 . cos\theta \qquad (4.14)$$

$$V_y = -V_1 . cos\theta + V_3 . cos\theta - V_2 . sin\theta + V_4 . sin\theta \qquad (4.15)$$

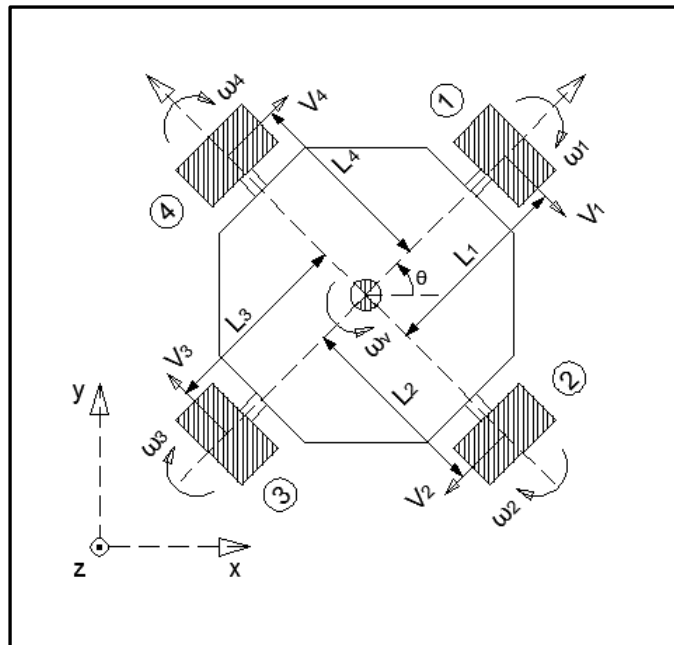$$\omega_v = -(V_1 + V_2 + V_3 + V_4)/L \qquad (4.16)$$

Figure 4.3. Angular and Linear velocities of the wheels

In the Equations 4.14, 4.15, 4.16, velocity of the robot on local x direction is shown with V_x, velocity of the robot on local y direction is shown with V_y and angular velocity on center of mass is shown with ω_v. Jacobian matrix from forward kinematic for velocity is:

$$\begin{bmatrix} V_x \\ V_y \\ -\omega_v.L \end{bmatrix} = J \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix} \tag{4.17}$$

$$J = \begin{bmatrix} sin\theta & -cos\theta & -sin\theta & cos\theta \\ -cos\theta & -sin\theta & cos\theta & sin\theta \\ 1 & 1 & 1 & 1 \end{bmatrix} \tag{4.18}$$

The robot was created as kinematically redundant robot as a result of using four omnidirectional wheels whereas only three omni-directional wheels are enough to provide holonomic motion in plane. It is not possible to find a single solution for velocities of the wheels according to global coordinate frame, because, there are three kinematic Equations of the robot according to global coordinate frame and four unknown wheel velocities which result in infinite number of solutions. However, infinite number of solutions can be used for optimization purposes. One method commonly used by researchers is for minimizing the common norm of joint motion (Golub, et al., 1965). Therefore, global velocity of the platform is related to the joint velocities with pseudo-inverse as given in Equation 4.19.

$$\begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix} = J^+ \begin{bmatrix} V_x \\ V_y \\ -\omega_V.L \end{bmatrix} \tag{4.19}$$

In Equation 4.19 $J^+ \in \mathbb{R}^{4x3}$ is the pseudo-inverse matrix. Right pseudo-inverse is used for this case because the rank of the Jacobian matrix is smaller than the column of the matrix (m<n). Pseudo-inverse of the Jacobian is calculated as shown in Equation 4.20 for the mobile platform considered in this thesis.

$$J^{+} = J^{T} (J J^{T})^{-1} = \begin{bmatrix} 0{,}5\ sin\theta & -0{,}5\ cos\theta & 0{,}25 \\ -0{,}5\ cos\theta & -0{,}5\ sin\theta & 0{,}25 \\ -0{,}5\ sin\theta & 0{,}5\ cos\ \theta & 0{,}25 \\ 0{,}5\ cos\theta & 0{,}5\ sin\theta & 0{,}25 \end{bmatrix} \qquad (4.20)$$

$$\begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix} = \begin{bmatrix} 0{,}5\ sin\theta & -0{,}5\ cos\theta & 0{,}25 \\ -0{,}5\ cos\theta & -0{,}5\ sin\theta & 0{,}25 \\ -0{,}5\ sin\theta & 0{,}5\ cos\ \theta & 0{,}25 \\ 0{,}5\ cos\theta & 0{,}5\ sin\theta & 0{,}25 \end{bmatrix} \begin{bmatrix} V_x \\ V_y \\ -\omega_V . L \end{bmatrix} \qquad (4.21)$$

Then linear velocities of wheels are:

$$V_1 = 0.5 . V_x . sin\theta - 0.5 . V_y . cos\theta - 0.25 . \omega_v . L \qquad (4.22)$$

$$V_2 = -0.5 . V_x . cos\theta - 0.5 . V_y . sin\theta - 0.25 . \omega_v . L \qquad (4.23)$$

$$V_3 = -0.5 . V_x . sin\theta + 0.5 . V_y . cos\theta - 0.25 . \omega_v . L \qquad (4.24)$$

$$V_4 = 0.5 . V_x . cos\theta + 0.5 . V_y . Sin\theta - 0.25 . \omega_v . L \qquad (4.25)$$

Angular velocities of each wheel are calculated as:

$$\omega_i = V_i / r \qquad ; i = 1, \dots, 4 \qquad (4.26)$$

where, V_i is linear velocities of each wheel, and ω_i is angular velocities of each wheel and r is radius of the wheel.

After calculating angular velocities of each wheel, motors are driven through the motor amplifiers (drivers) with voltage demands. Since there is a linear relation between input voltage and angular velocity of the DC motors, necessary voltages of wheels are calculated as:

$$v_i = (\omega_i \cdot 24) / \omega_{imax} \qquad (4.27)$$

where, v_i is necessary voltages of each wheel ($i = 1,\dots,4$) and ω_{imax} is maximum angular velocities of the motors running at 24V.

4.2.2. Dynamic Control

In the dynamic control model, the robot motion is deduced by the combination of wheel traction forces. These forces are provided by electric motors. Therefore, in this control algorithm, motor currents to drive motors are calculated through desired traction forces for each wheel in order to reach desired velocity of the platform. The second control scheme of the vehicle is illustrated in Figure 4.4.
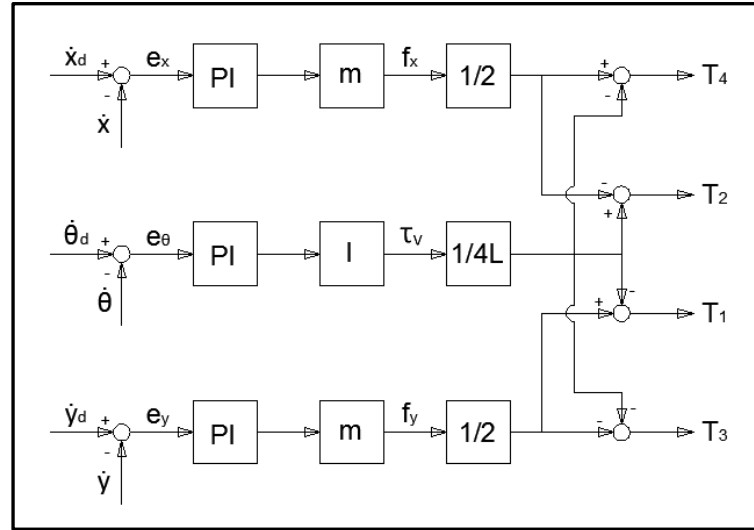
Figure 4.4. Control scheme of developed vehicle

In Figure 4.4, $\dot{x}_d$ and $\dot{y}_d$ indicates the desired velocities and $\dot{x}$ and $\dot{y}$ indicates the actual measured velocities on local coordinate frame. Errors, e_x and e_y, are calculated from the differences between these velocities. $\dot{\theta}_d$ is desired angular velocity and $\dot{\theta}$ is actual measured angular velocity. e_θ is the error between desired and actual angular velocities of the vehicle. Forces at local coordinate frame and torque at center of mass (τ_v) are calculated with Equations (4.28), (4.29) and (4.30) in order to follow the desired motion.

$$f_x = m\left(\left(Kp\left(\dot{x}_d - \dot{x}\right) + Ki \int \left(\dot{x}_d - \dot{x}\right) dt\right)\right) \qquad (4.28)$$

$$f_y = m\left(\left(Kp\left(\dot{y}_d - \dot{y}\right) + Ki \int \left(\dot{y}_d - \dot{y}\right) dt\right)\right) \qquad (4.29)$$

$$\tau_v \ = \ I\left(\left(Kp\left(\theta_d \ - \ \theta\right) \ + \ Ki\int\left(\theta_d \ - \ \theta\right)dt\right)\right) \qquad (4.30)$$

Wheel 2 and wheel 4 traction forces are calculated as:

$$T_4 \ = \ (f_x/2) \ - \ (\tau_v/4L) \qquad (4.31)$$

$$T_2 \ = \ -\,(f_x/2) \ - \ (\tau_v/4L) \qquad (4.32)$$

Wheel 1 and wheel 3 traction forces are calculated as:

$$T_1 \ = \ (f_y/2) \ - \ (\tau_v/4L) \qquad (4.33)$$

$$T_3 \ = \ -\,(f_y/2) \ - \ (\tau_v/4L) \qquad (4.34)$$

Relation between torque provided by the actuator and the desired traction force is illustrated in Figure 4.5. Necessary torques for each wheel to reach desired traction forces is calculated as:

$$\tau_i \ = \ T_i \cdot r \qquad (4.35)$$

where τ_i is necessary torque for each wheel ($i \ = \ 1,...,4$), T_i is desired traction forces and r is radius of wheel.
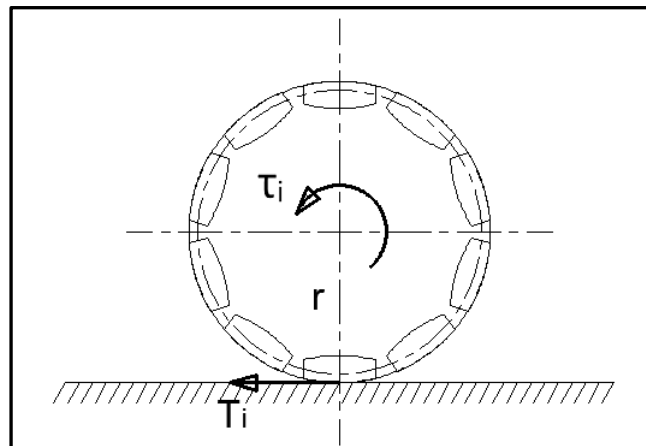

Figure 4.5. Wheel traction force

From Equations (4.35) and (3.1) necessary motor currents (i_i) are calculated as:

$$i_i = (T_i \cdot r) / 4.96 \tag{4.36}$$

4.3. Fault Tolerance Capacity of the Developed Robot

The omnidirectional mobile robot was designed by using four universal omni-directional wheels. However, planar motion can be accomplished with only three universal omni-directional wheels. Therefore, if any wheel is faulty in task, the robot can perform the task with remaining three wheels without losing omnidirectional mobility. Figure 4.6 represents a case when fourth wheel is faulty.
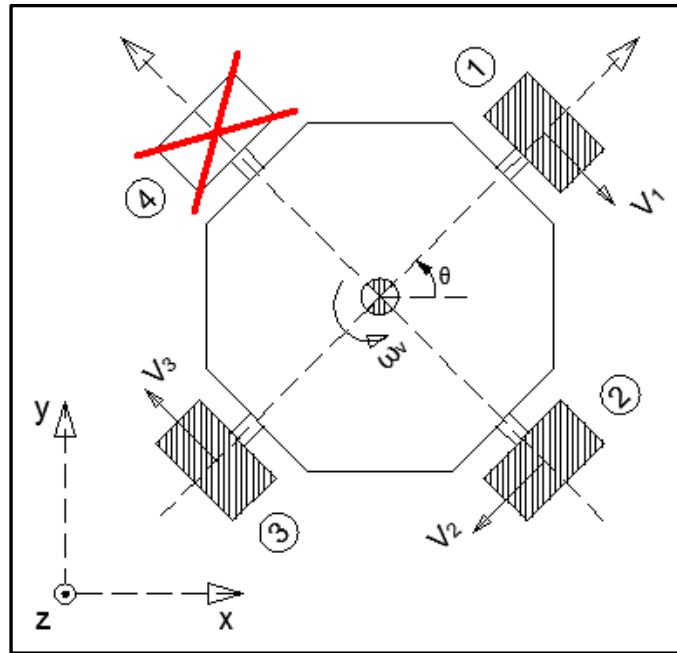

Figure 4.6. Developed robot when three wheels remain

The kinematic Equations of motion according to global coordinate frame for this case are:

$$V_x = V_1 . sin\theta - V_3 . sin\theta - V_2 . cos\theta \tag{4.37}$$

$$V_y = -V_1 . cos\theta + V_3 . cos\theta - V_2 . sin\theta \tag{4.38}$$

74

$$\omega_v = -(V_1 + V_2 + V_3)/L \tag{4.39}$$

Jacobian matrix from forward kinematic for velocity is:

$$\begin{bmatrix} V_x \\ V_y \\ -\omega_v.L \end{bmatrix} = J \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} \tag{4.40}$$

$$J = \begin{bmatrix} sin\theta & -cos\theta & -sin\theta \\ -cos\theta & -sin\theta & cos\theta \\ 1 & 1 & 1 \end{bmatrix} \tag{4.41}$$

Inverse of the jacobian matrix is used to obtain the linear velocities produced by the wheels.

$$\begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = J^{-1} \begin{bmatrix} V_x \\ V_y \\ -\omega_V.L \end{bmatrix} \tag{4.42}$$

$$J^{-1} = \begin{bmatrix} 0.5(sin\theta + cos\theta) & 0.5(sin\theta - cos\theta) & 0.5 \\ -cos\theta & -sin\theta & 0 \\ -0.5(sin\theta - cos\theta) & 0.5(sin\theta + cos\theta) & 0.5 \end{bmatrix} \tag{4.43}$$

Then linear velocities of wheels are:

$$V_1 = V_x.0.5.(sin\theta + cos\theta) + V_y.0.5.(sin\theta - cos\theta) + 0.5.\omega_v.L \tag{4.44}$$

$$V_2 = -V_x.cos\theta - V_y.sin\theta \tag{4.45}$$

$$V_3 = -V_x.0.5.(sin\theta - cos\theta) + V_y.0.5.(sin\theta + cos\theta) - \omega_v.L \tag{4.46}$$

The system can still move if two wheels are faulty acting as a differential drive system. One possibility is that the two wheels have parallel rotation axes fail. In this case, the robot can move with two wheels like differential drive system as shown in Figure 4.7.
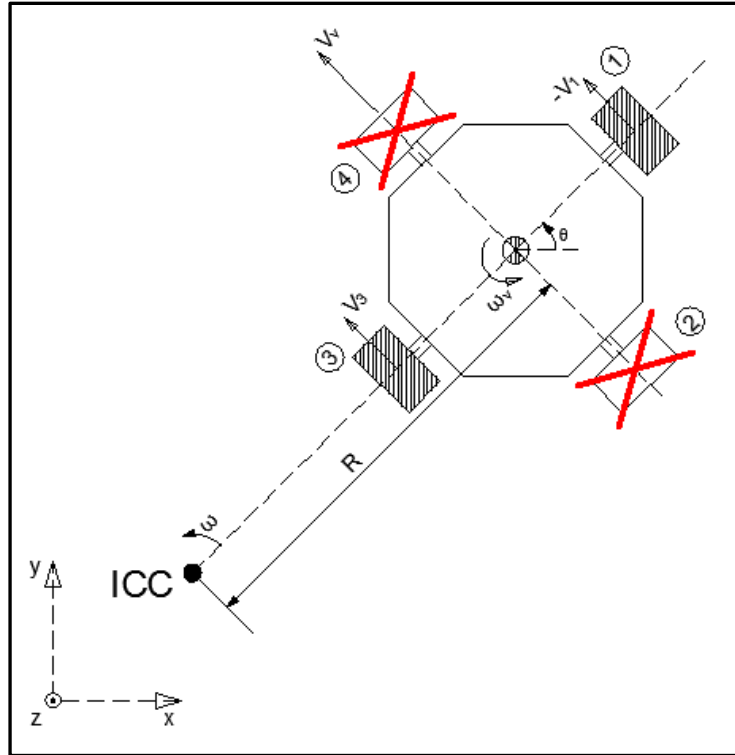
Figure 4.7. Two wheeled differential drive system

The rotation about a point, which is on the common axis of the wheels, is changed by wheel velocities. This point is known as Instantaneous Center of Curvature (ICC). While the robot rotates about ICC, wheels have same angular rate (ω).

$$\omega(R + L) = -V_1 \tag{4.47}$$

$$\omega(R - L) = V_3 \tag{4.48}$$

where R is distance between the ICC and the center of the vehicle. From Equations (4.47) and (4.48), ω and R can be calculated as:

$$R = L[(V_3 - V_1)/(-V_1 - V_3)] \tag{4.49}$$

$$\omega = (-V_1 - V_3)/2L \tag{4.50}$$

Linear velocity of the vehicle according to local coordinate frame is:

$$V_v = (V_3 - V_1)/2 \tag{4.51}$$

If the velocities of wheels are not equal at same or different directions, robot follows a curved trajectory and it rotates about any ICC.

The kinematic Equations of motion according to global coordinate frame for differential drive are:

$$V_x = V_v . sin\theta \tag{4.52}$$

$$V_y = V_v . cos\theta \tag{4.53}$$

$$\omega_v = -(V_1 + V_3)/L \tag{4.54}$$

Because, the differential drive system has non-holonomic constraint (it cannot move sidewise), it is not possible to find wheel velocities for given holonomic velocity demands according to global coordinate frame. However, there are two special cases for differential drive robots. If velocities of wheels are same value at same direction($V_3 = -V_1 = V_v$), rotation radius (R) is infinite and angular rate (ω) is zero. In this case the robot moves in a straight line. If velocities of wheels are same value at different directions($V_3 = V_1$), the robot rotates about the center of the vehicle and the location of the robot is not changed.

4.4. Implementation of Control Algorithm in Teleoperation System via Matlab Simulink

In this thesis, the control algorithms were modeled in Matlab Simulink environment. The teleoperation system developed in this study is illustrated in Figure 4.8.
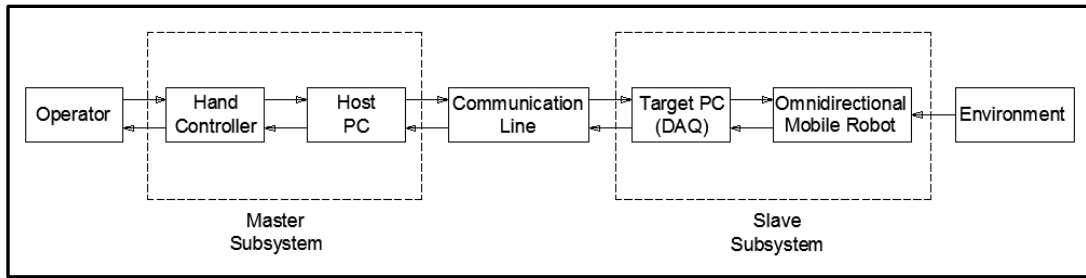
Figure 4.8. Teleoperation system scheme

In the master side, the desired velocities of the mobile platform ($\dot{x}_d$, $\dot{y}_d$) are generated by hand controller according to operators demands. The hand controller with respect to its position provides a signal that ranges between -1 to +1 V. To match signal created by hand controller and desired velocity, firstly, maximum velocity of the vehicle should be known. In order to calculate maximum velocity of the vehicle along one direction, the following features are required:

- Continuous rated speed of motor when 24V: 1804 rpm
- Reduction ratio of gearbox: 1/20.5
- Diameter of wheel: 10 cm
- Max. velocity of vehicle = (1804/20.5) · π · 10 = 2763 cm/min = 0.461 m/s

To reach 0.461 m/s, value of signal generated via the hand controller should be 1. Therefore, desired velocities at the signal form are calculated as:

$$Desired\ velocity\ signal\ value \ = \ (\dot{x}_d\ or\ \dot{y}_d) \ / \ 0.461 \qquad (4.55)$$

These desired velocity signals created by hand controller are double data type. However, the data transfer among subsystems is accomplished via UDP data transfer protocol and UDP blocks in Simulink allow sending and receiving only uint8 (unsigned integers of 8 bits) data. This data type contains all numbers from 0 to 255. The values must be non-negative. Therefore, before sending desired velocity signal, this must be converted the uint8 data type.

The first model of master subsystem created in Matlab Simulink was designed to only provide desired velocity information through the acquired position of the joystick

and to change the orientation of the vehicle through receiving inputs from two pushbuttons (two pushbuttons are used to differentiate the motion direction). However, it is not possible to provide different angular velocity demands of the vehicle with pushbuttons. Therefore, in the further stages of the study, the model of the master subsystem was replaced with a new model by using property of twisting handle of the hand controller. The final model of the master subsystem is demonstrated in Figure 4.9.
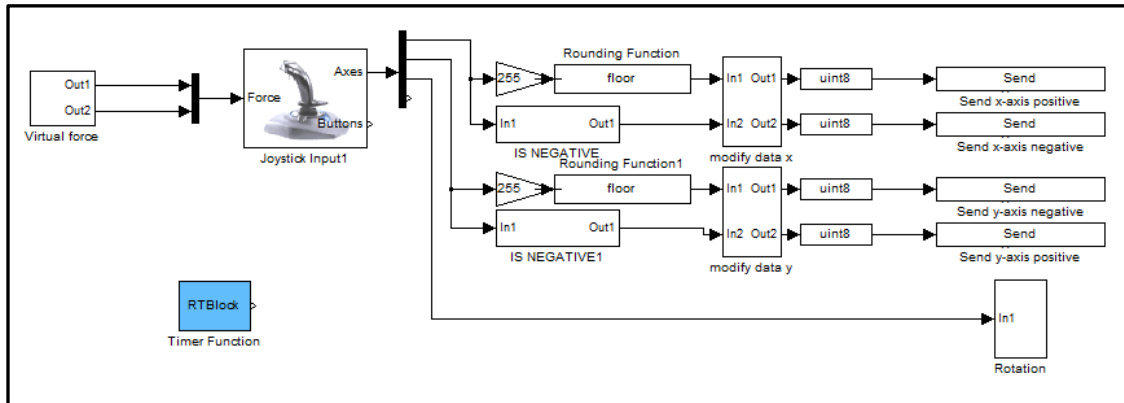
Figure 4.9. Master side of the teleoperation system

Kinematic control algorithm implemented with Matlab blocks is represented in Figure 4.10. In this algorithm, motors are driven by the motor drivers as they receive motor supply voltage demands through the Set Value port on the motor driver which is sent by the analog output ports of the DAQ card. Therefore, the desired velocities taken from the master subsystem via UDP protocol are processed in control algorithm and they are transformed set values to drive motors. In addition, this control model is used to test obstacle avoidance algorithms as described in the next sections.
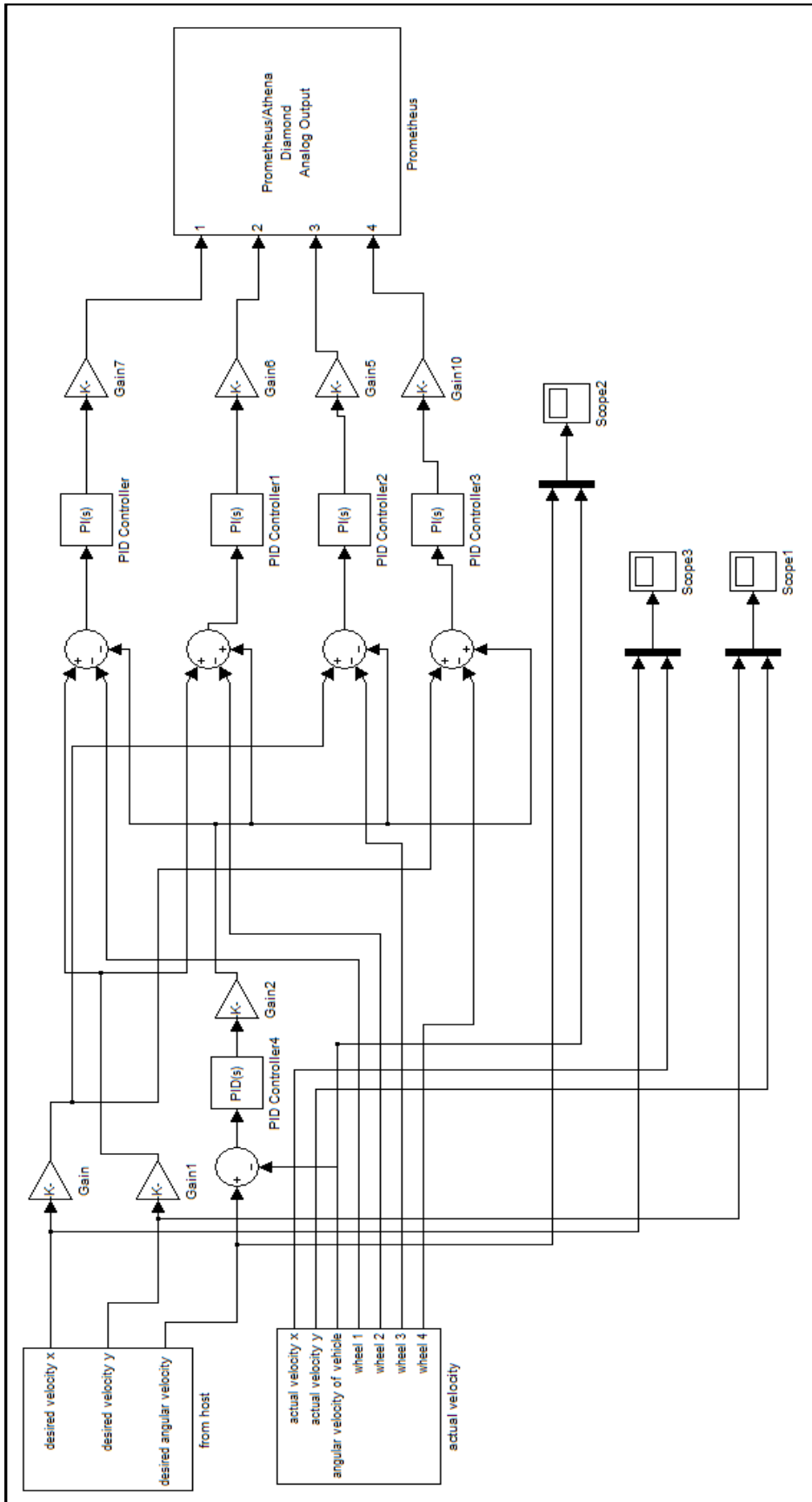
Figure 4.10. Slave side of the teleoperation system

4.5. Obstacle Avoidance Algorithms

In the dynamic obstacle avoidance system, described in Chapter 3, infrared range finders are utilized to perceive obstacles in the path of the mobile platform. Perceiving angle of these sensors is about 15 degrees. Therefore, to cover the critical area which is defined as the area between the maximum sensor range and maximum width of the robot minimum 5 sensors are required. Arrangement of the sensors (si ; $i = 1,...,4$) is shown in Figure 4.11a. In the Figure 4.11b, coverage areas of the sensors, which are represented with Si ; $i = 1,...,4$, are given.
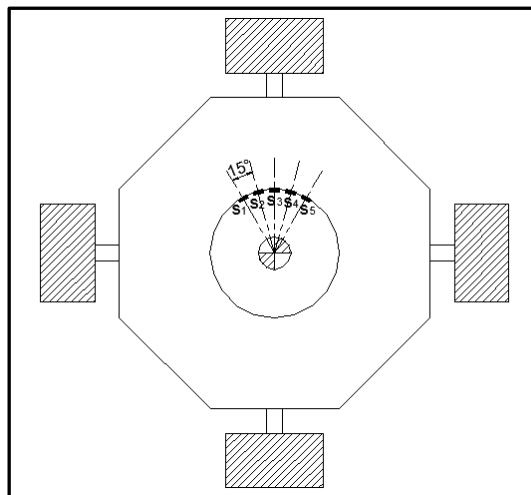
(a)
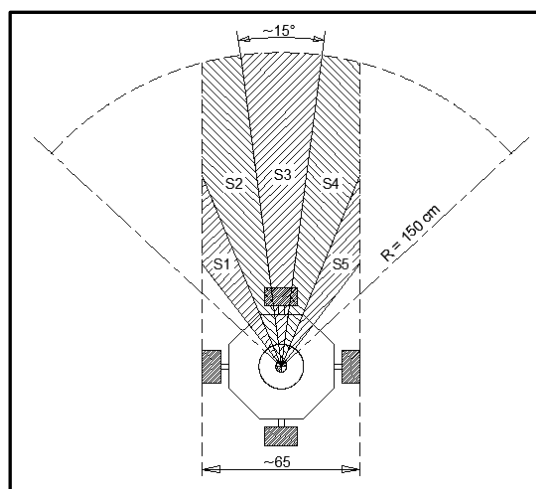
(b)

Figure 4.11. (a) Arrangement of infrared range finder, (b) Coverage areas of sensors

In this thesis two models are designed to avoid obstacles as:

- Force reflecting obstacle avoidance
- Semi-autonomous obstacle avoidance

4.5.1. Force-reflecting Obstacle Avoidance

In this model, if any object enters in one or more areas, virtual force, which is modeled as sprig model according to sensor information, is transferred from slave side to master side. Then this force is transmitted operator via force feedback joystick. By this way, the operator identifies the location and the distance of the obstacle. Force reflecting obstacle avoidance scheme is represented in Figure 4.12.
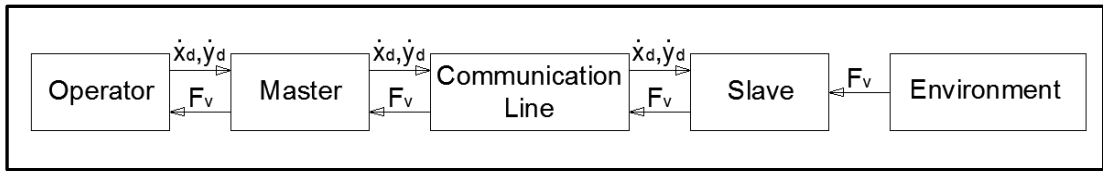
Figure 4.12. Force reflecting obstacle avoidance scheme

Virtual force (F_v) is calculated as:

$$F_v = k \ (tr - d)/tr \tag{4.56}$$

$$F_v = \begin{cases} k \ (tr - d)/T, & when \ d < tr \\ 0, & otherwise \end{cases} \tag{4.57}$$

where k is spring coefficient, d is distance between robot and obstacle and tr represents the threshold for obstacle avoidance. The maximum measuring distance of the infrared range finder (150 cm) is chosen as threshold.

4.5.2. Semi-autonomous Obstacle Avoidance

In the second model, virtual impedance model was adapted for kinematic control model (Figure 4.13). If the robot meets any obstacle while mobile robot is controlled by

the operator, the virtual velocity vector is created by obstacle avoidance algorithm, and then robot will move in the direction of total velocity (V_T) of desired (V_D) and virtual (V_V) velocities until avoiding the obstacle. After avoiding obstacle, virtual velocity will be zero and robot will keep moving in direction of desired velocity.
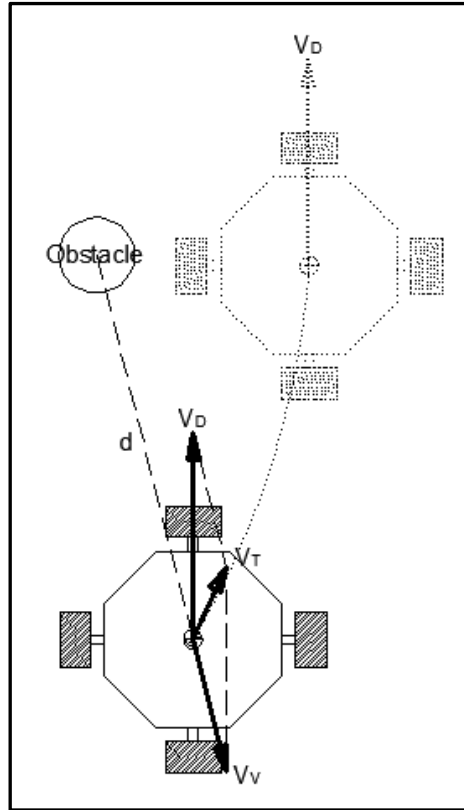
Figure 4.13. Adapted virtual impedance model for kinematic control model

Virtual velocity is calculated according to the relation between distance, which is between obstacle and robot, and desired velocity of the vehicle:

$$V_V = V_D \cdot (tr - d) / tr \qquad (4.58)$$

$$V_V = \begin{cases} V_D \cdot (tr - d)/tr & when\ d < tr \\ 0, Otherwise \end{cases} \qquad (4.59)$$

where, V_V is virtual velocity, V_D is desired velocity of the vehicle, d is measured distance between vehicle and obstacle, and tr is threshold for obstacle avoidance. If the

measured distance is smaller than the threshold, obstacle avoidance algorithm gives a virtual velocity; otherwise this velocity will be zero.

4.6. Conclusion

In this Chapter, firstly, Equations of motion of mobile platform were derived. According to Equations of motion and kinematic analysis two control models were designed. In the first model (kinematic control), motors are driven with voltage to reach desired velocity of the vehicle. The other control model (dynamic control) was created to drive motors in the current control mode to reach desired velocity of the vehicle.

Additionally, in this Chapter, two obstacle avoidance algorithms were developed: force-reflecting obstacle avoidance and semi-autonomous obstacle avoidance. When the robot is in the sensing range of any obstacles along its path, a virtual force is generated in the force-reflecting obstacle avoidance algorithm with respect to the distance between the obstacle and the robot. Then, this force is transmitted operator via force reflecting joystick. By this way, operator receives haptic information on where the obstacle is and how far the obstacle is. Operator, as he/she receives the information about the location of the obstacle, can avoid the obstacle by changing his/her input through the joystick. In the semi-autonomous obstacle avoidance algorithm, when there is an obstacle along the path of the robot, a virtual velocity vector is created depending on distance between robot to obstacle and desired velocity of the vehicle. Then, robot moves in the direction of total velocity vector of virtual and desired velocities until it avoids the obstacle.

These algorithms are implemented and test procedures are set to conduct evaluation tests of the algorithms. The results of the tests are presented in the next chapter.

CHAPTER 5

TEST RESULTS AND DISCUSSIONS

Semi-autonomous obstacle avoidance and force reflecting obstacle avoidance test procedure are explained in Section 5.1 while test results are presented in Section 5.1.1 and 5.1.2. In addition, the effect of using an external motion sensor, in this case, gyroscope, on the motion of the robot is also presented in Section 5.2.

5.1. Obstacle Avoidance Tests

In both obstacle avoidance algorithm tests, velocity demand to drive the robot on local $+y$ direction was sent from master side and one or more obstacles were placed on the left or right side of the robot. In semi-autonomous obstacle avoidance tests, path of the mobile robot was estimated using the marks (marked with red circle in Figure 5.1), which was formed with a permanent marker pen mounted on the robot, by referencing floor marbles that have the same dimensions. A cylindrical shaped dustbin whose diameter is 30.5 cm was used as an obstacle in all obstacle avoidance tests. Obstacle used in tests and the path of the mobile robot while avoiding obstacle in semi-autonomous tests can be seen in Figure 5.1.

In the force-reflecting obstacle avoidance tests, magnitude and direction of the virtual force generated by force reflection obstacle avoidance algorithm according to the obstacle direction and distance were observed.
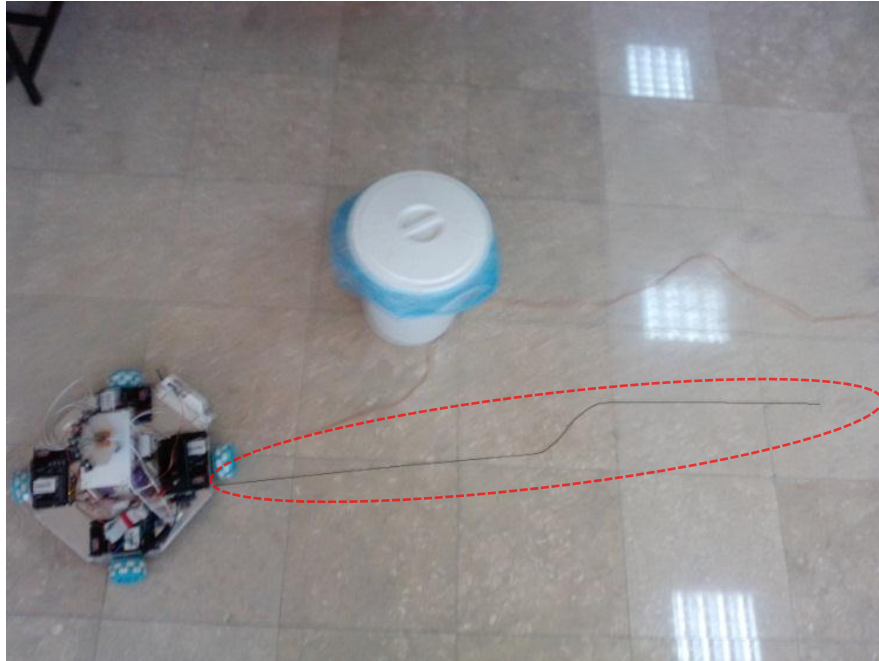
Figure 5.1. Semi-autonomous obstacle avoidance test

The slave controller sample time was chosen to be 0.01 ms during all tests, since servo controller do not allow smaller sample time rates than 0.01 ms.

5.1.1. Semi-Autonomous Obstacle Avoidance Tests

In the semi-autonomous obstacle avoidance (Figure 5.2), desired velocity (V_D) of the robot were generated in the master side according to the operator's demands and this velocity demand was sent from the master side to the slave side. Then, slave subsystem was driven according to desired velocity. If the obstacle avoidance system perceives an obstacle on motion direction of the robot, obstacle avoidance algorithm creates a virtual velocity vector (V_V). Then, the platform is driven with a total velocity (V_T) vector.
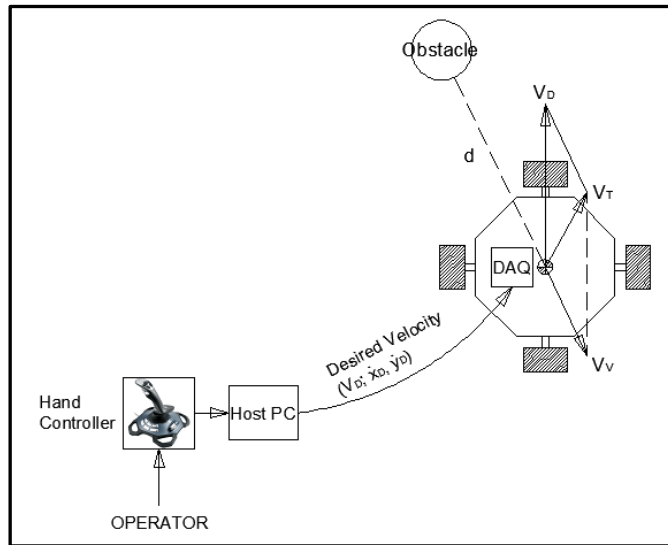
Figure 5.2. Semi-autonomous obstacle avoidance scheme

In the first test, an obstacle, which was placed on right side according to moving direction of the robot, was used. Path of the robot while robot avoids from obstacle can be seen in Figure 5.3. During the robot's forward motion, firstly, obstacle was appeared in the coverage area of the sensor 4 (S_4 represented in Figure 4.11), obstacle avoidance algorithm calculated a virtual velocity vector (V_V) according to distance between robot and obstacle, and magnitude of desired velocity of the vehicle. Then, velocity of the robot was decreased and platform slid a little bit to the left side. After that, obstacle appeared in the coverage area of the sensor 5 (S_5) and virtual velocity vector was adjusted according to the distance information collected from sensor 5. Then the robot slid left side again until the obstacle disappeared in the coverage range of any sensor.

The second test was performed with two obstacles. The first obstacle was placed on the right side of the robot according to the direction of motion and the second obstacle was placed on the right side of the robot after avoiding the first obstacle. Figure 5.4 shows tracked path of the robot while avoiding two obstacles. In this test, sensor 4 and 5 perceive the first obstacle and robot slided to the left side until the obstacle was out of the range of the sensors and then the robot moved along the desired direction again. Then, sensor 4 and 5 perceived the second obstacle and the robot repeated the same motion autonomously as it did for the first obstacle case. After avoiding the second obstacle, the robot kept moving along the desired direction.
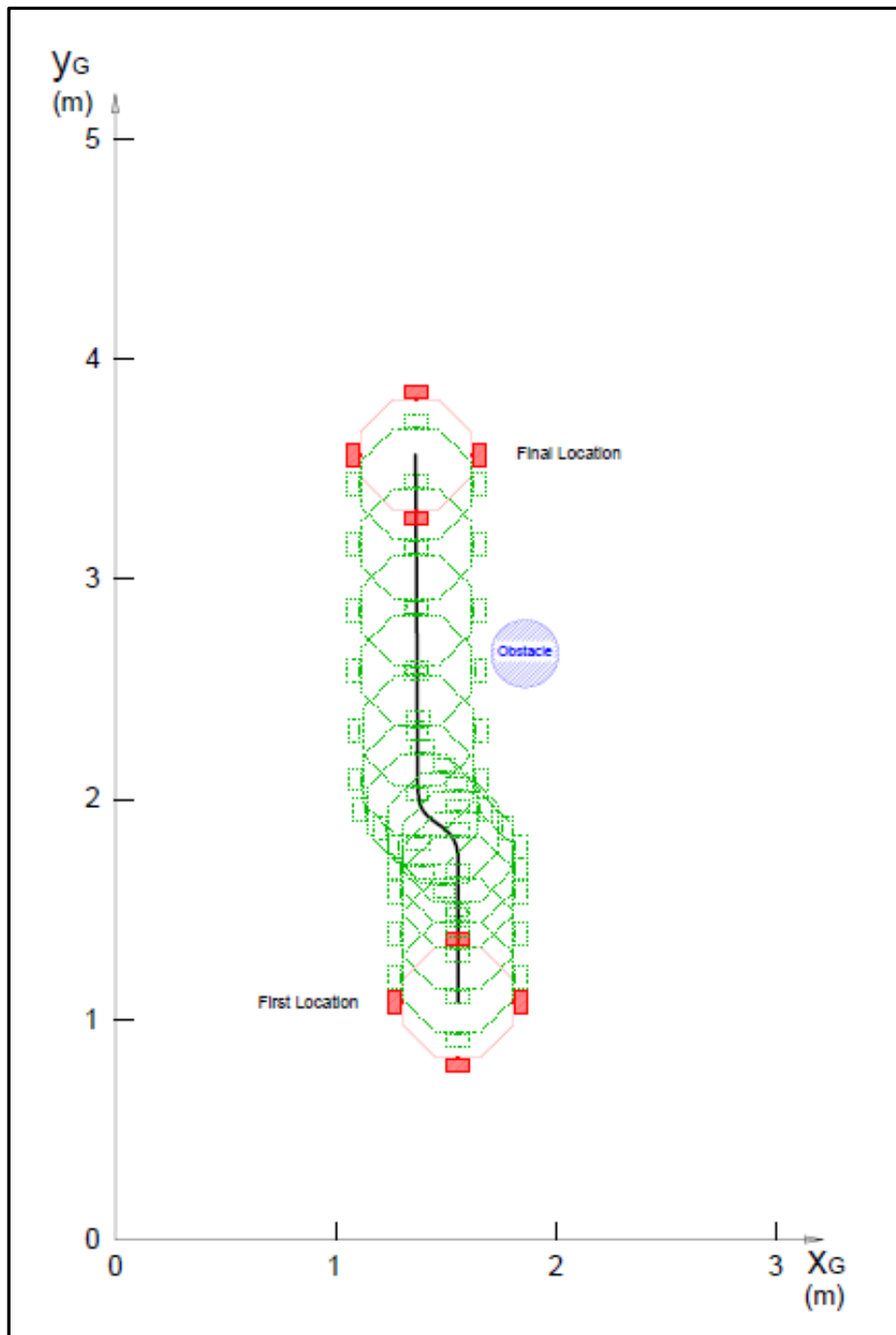
87

Figure 5.3. Tracked path of the robot during first test
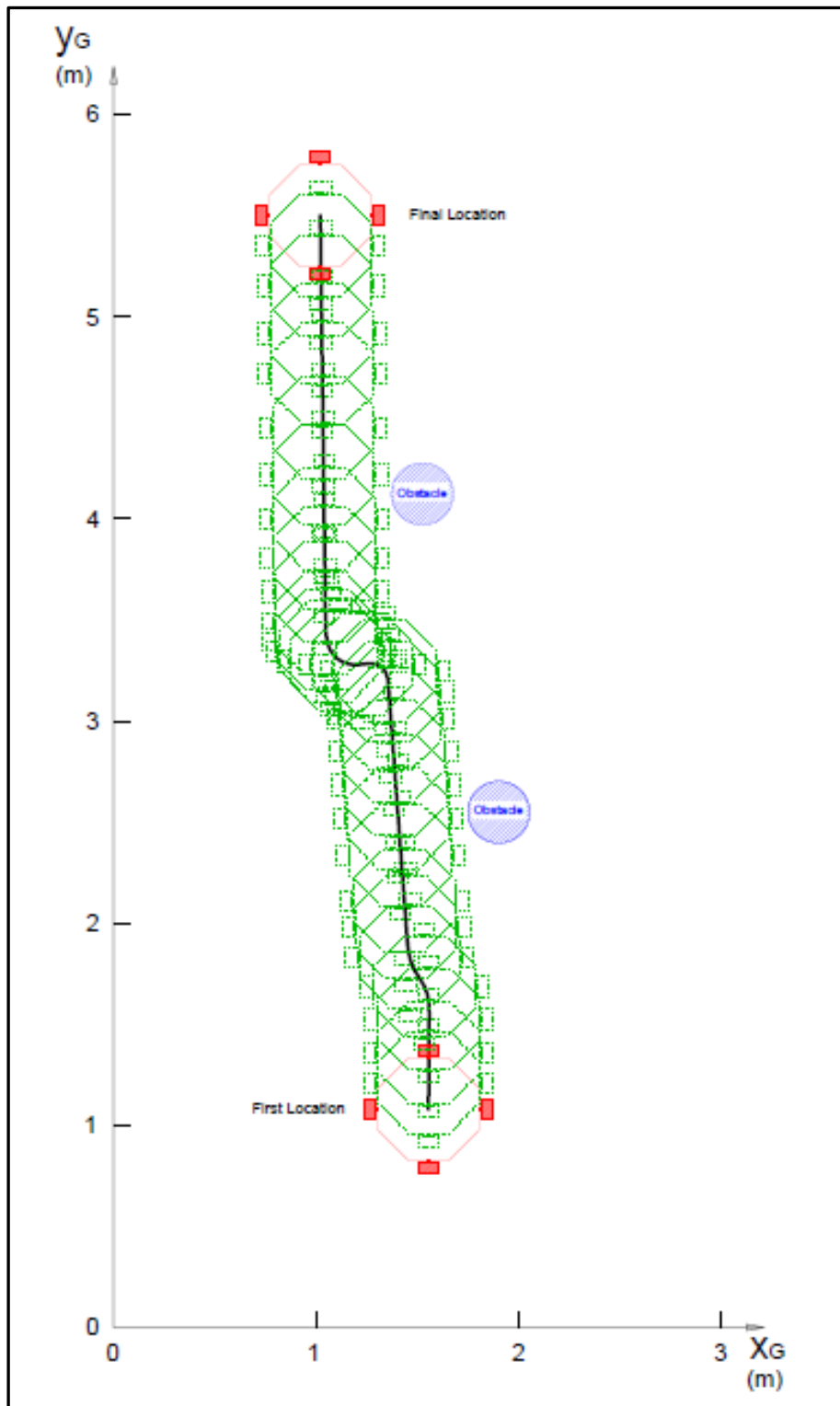
Figure 5.4. Tracked path of the robot during second test

In the third test (Figure 5.5), the first obstacle was placed on the right side of the robot with respect to the direction of motion of the robot and the second obstacle was placed on the left side of the robot after avoiding the first obstacle. While the robot was

moving forward, sensors 4 (S4) and 5 (S5) perceived the first obstacle and robot slid left side until avoiding first obstacle. Then, sensors 1 (S1) and 2 (S2) perceived the second obstacle and the robot slid right side until the obstacle was not in the sensing range of the sensors and the robot kept moving forward again.

Figure 5.5. Tracking path of the robot during third test

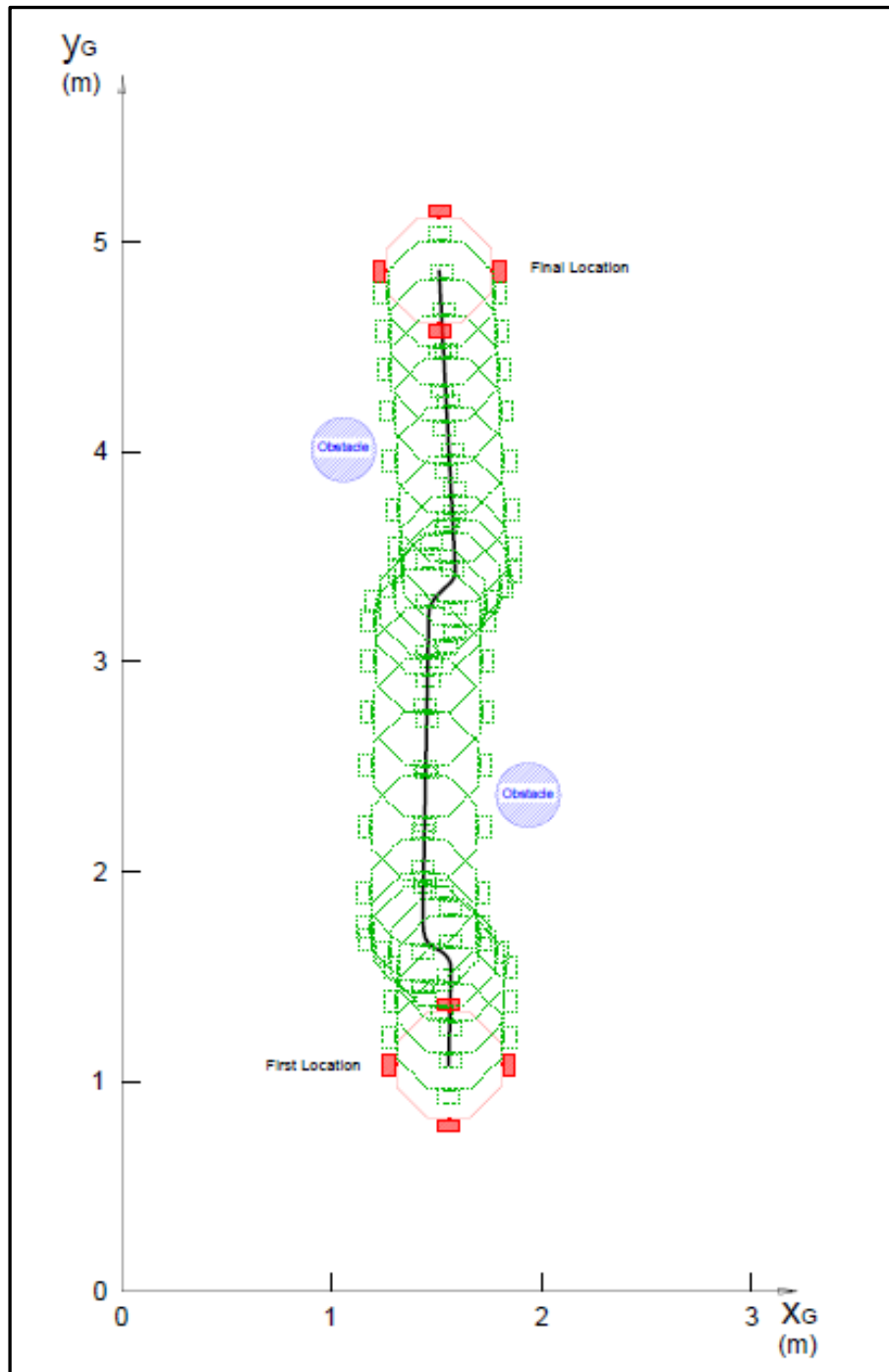## 5.1.2. Force-Reflecting Obstacle Avoidance Tests

These tests were conducted with kinematic control active as the main controller. Desired velocities were sent from the master side to the slave side. Then, the slave subsystem was driven according to desired velocity. In the force reflecting obstacle avoidance algorithm (Figure 5.6), if the robot encounters any obstacles along the moving direction, a virtual force, which is generated by algorithm according to distance information gathered from infrared sensors, is sent from the slave side to the master side. Then, this force is transmitted to the operator via a force-reflecting joystick. By this way, operator receives the location information of the obstacle through haptic feedback and changes the motion direction of the robot.
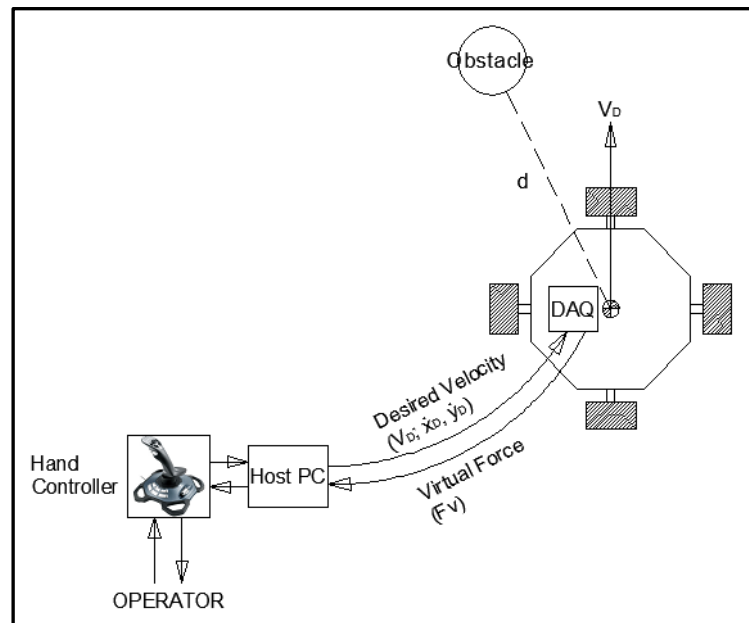
Figure 5.6. Force reflection obstacle avoidance scheme

Four tests were executed to evaluate force-reflecting obstacle avoidance algorithm performance. In the first test, the obstacle was placed on the right side of the robot. It was perceived by only sensor 5 when the robot was moving forward and virtual force was created by this sensory information. x and y components of the created virtual force in the first test are represented in Figures 5.7 and 5.8. As seen in Figures 5.7 and 5.8, virtual force is noisy since the signal, which is observed from the infrared range finder, is noisy. Therefore, 1 Hz low-pass filter was used due to filter noise of the

infrared range finder signal in the slave side so in the other force-reflecting obstacle avoidance test, more noiseless responses were observed for virtual force.

Figure 5.7. Virtual force component on x axis during first test
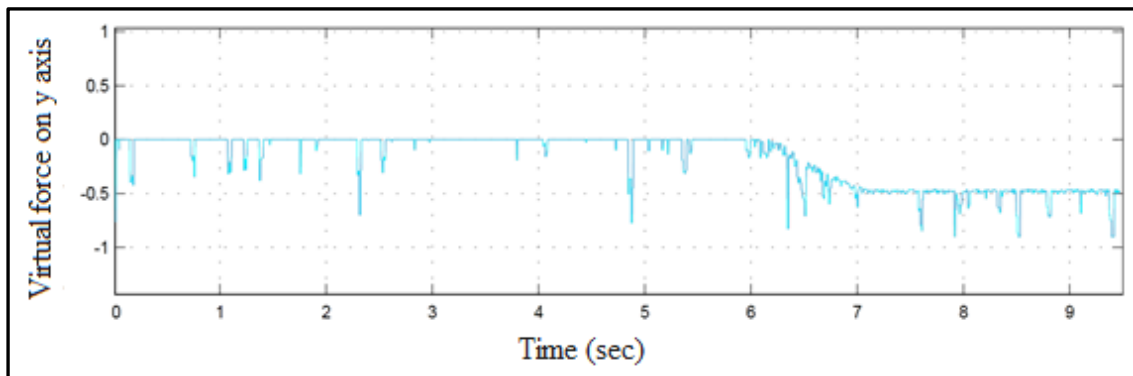
Figure 5.8. Virtual force component on y axis during first test

In the second test, while the robot was moving forward, sensor 4 and 5 perceived the obstacle and the virtual force was generated by algorithm according to the sensor information. Figure 5.9 and 5.10 represent virtual force components on x and y axes during second test. As seen in the Figure 5.9, firstly, sensor 4 perceived the obstacle and gave a virtual force. Then at the time about 4.75 second, sensor 5 perceived the obstacle with sensor 4 and component of the virtual force on x axis dramatically increased.

Figure 5.9. Virtual force component on x axis during second test
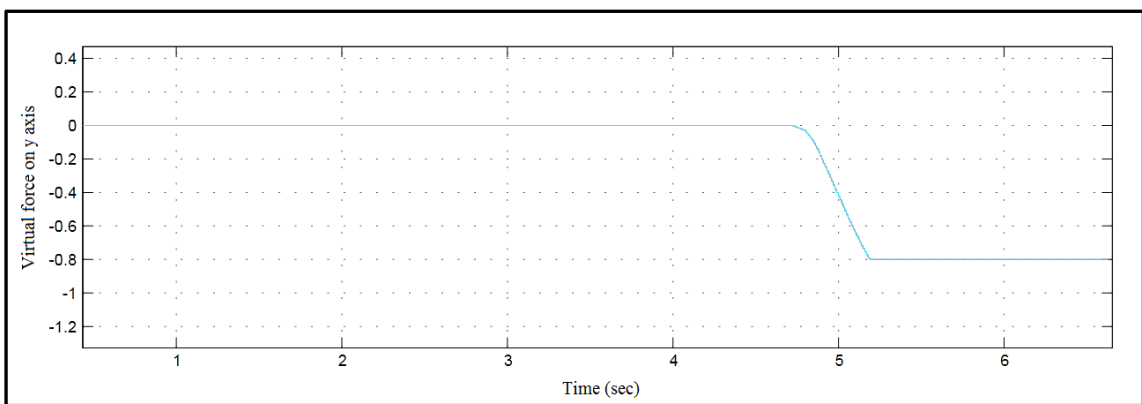
Figure 5.10. Virtual force component on y axis during second test

The same tests were repeated when obstacle was on the left side of the robot according to the motion direction of the robot. In these tests, direction of the component of the virtual force on *x* axis was changed, since sensors 1 and 2 perceived obstacle instead of sensor 4 and 5. Test results of these tests are illustrated in Figures 5.11, 5.12, 5.13, and 5.14. In the third test, only sensor 1 perceived the obstacle and the virtual force was created by this sensory information, while sensors 1 and 2 perceived obstacle together in fourth test. Therefore the component of virtual force on y axis in fourth test is higher than the third test.
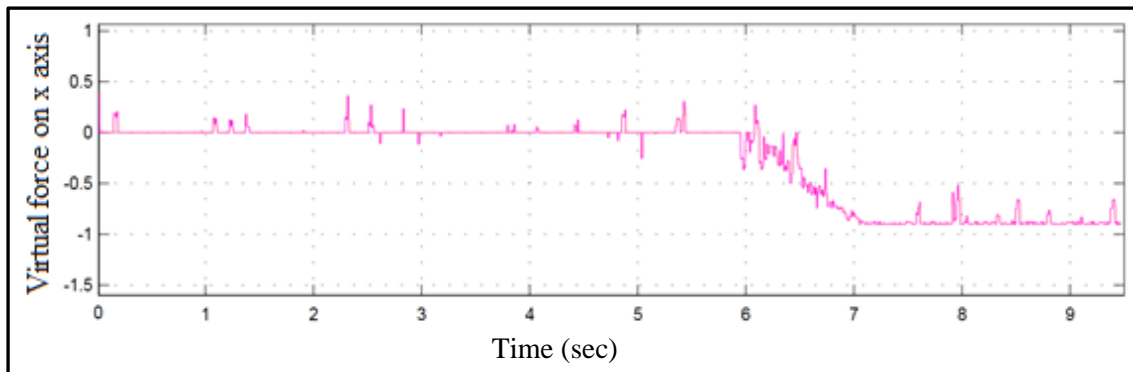
93
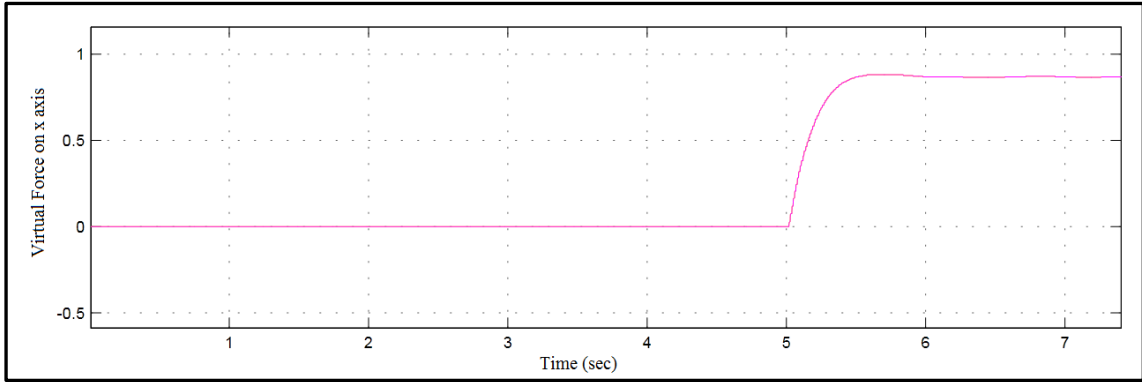
Figure 5.11. Virtual force component on x axis during third test
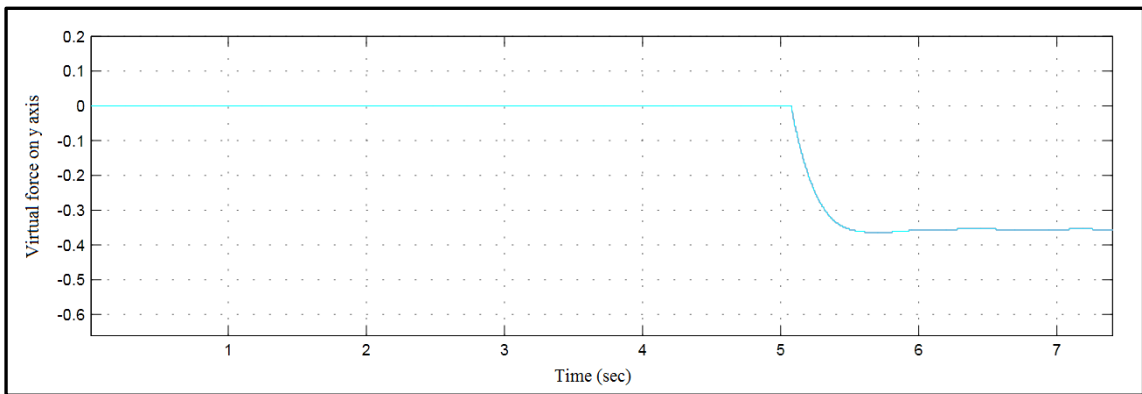
Figure 5.12. Virtual force component on y axis during third test
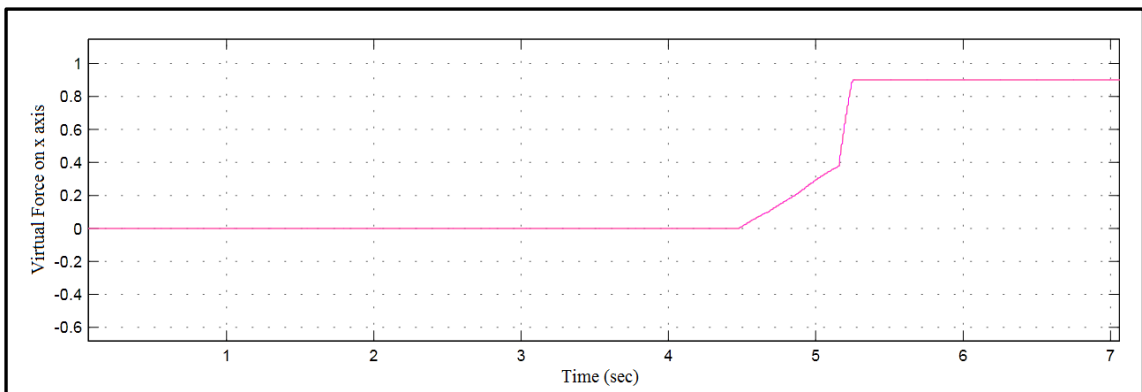
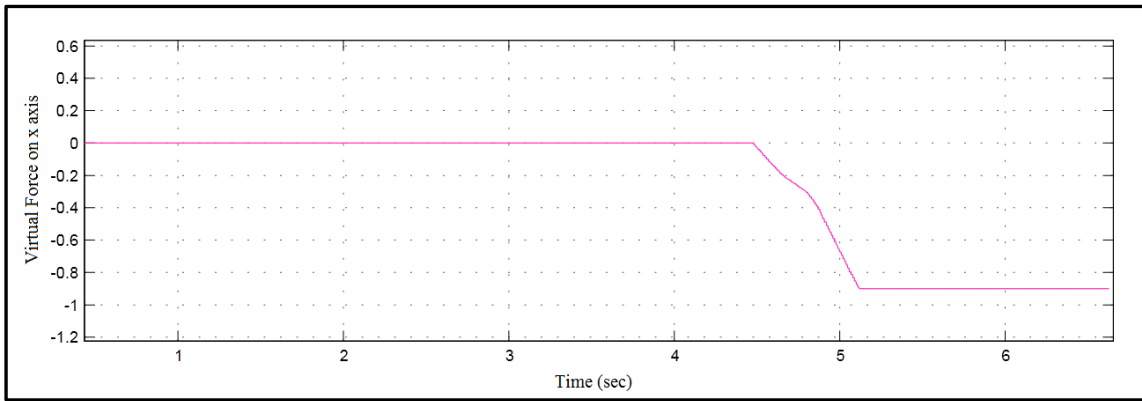

Figure 5.13. Virtual force component on x axis during fourth test
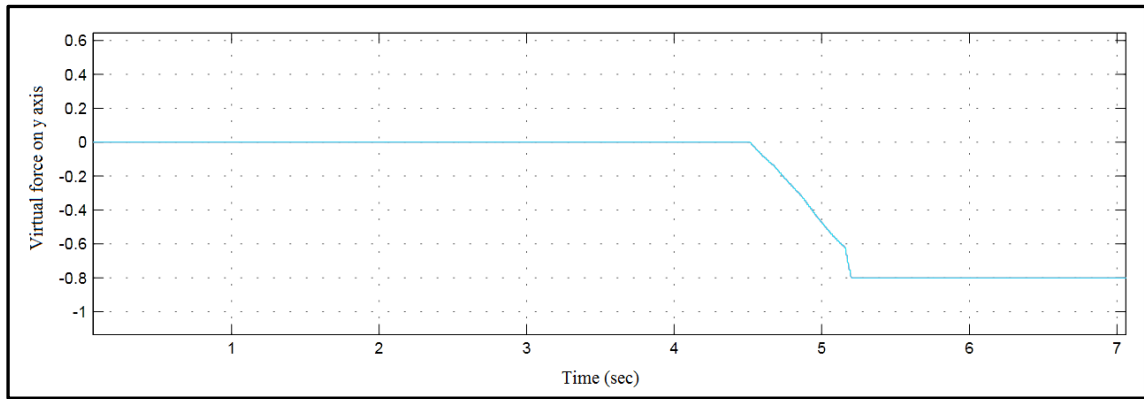
Figure 5.14. Virtual force component on y axis during fourth test

5.2. Gyroscope Test

Kinematic controller was supported by gyroscope since orientation of the robot can change during motion as it is observed in semi-autonomous tests. The reason is the slip of the wheels and constructional failures. Therefore, a test procedure is configured in order to show effectiveness of closing the loop of kinematic control on orientation by an external sensor, gyroscope.

Two tests were conducted that last for 20 seconds. In both tests, the linear velocity demand was set to 0.4 m/sec. In one of the tests, gyroscope was excluded from the control and the open-loop type of control algorithm was used. The result of this test is given in Figure 5.15 as marked in red. The test was repeated for the closed-loop control where the gyroscope was added in the loop as the external sensor. The result of the second test can also be observed in Figure 5.15 marked in blue. The green dotted line in Figure 5.15 is the given trajectory of the robot for this test.

At the end of the test without the gyroscope in the loop, robot trajectory was deviated to the left side by 45.4 cm and resulted in an orientation change of 8° (Figure 5.15). In the gyroscope enabled test, the deviation to the left side was limited to 5.5 cm and the orientation was modified with 0.6°. This clearly shows an improvement on the kinematic control when an external sensor is integrated to the control algorithm. Obviously, the result can be further improved by selecting more precise sensors or fine tuning the control gains.

Figure 5.15. Gyroscope test

CHAPTER 6

CONCLUSIONS

In this thesis, an unlimited-workspace teleoperation system was developed to be used in indoor mobile platform navigation and unlimited-workspace teleoperation research. Hence, an omnidirectional vehicle which can move in every dire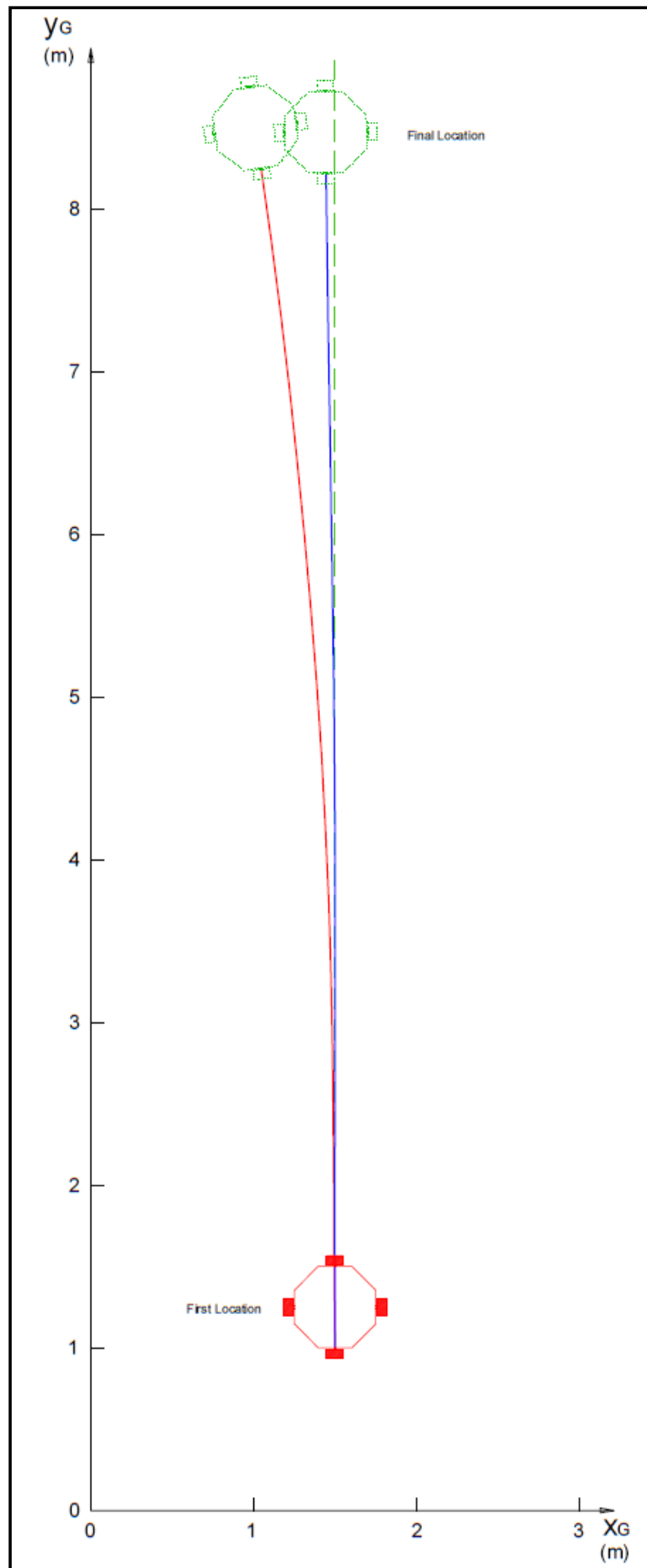ction independently was manufactured as the slave subsystem of the unlimited-workspace teleoperation. The two control algorithms; kinematic and dynamic control were developed and tested. Another goal of the thesis was that the system should be configured as a force-reflecting teleoperation system. Therefore, a force-reflecting obstacle avoidance algorithm was developed. Also a semi-autonomous obstacle avoidance algorithm was created and tested.

First of all, literature survey was presented under four headlines in Chapter 2; omnidirectional vehicles, force reflecting teleoperation, obstacle avoidance and components of the mobile robots. Omnidirectional vehicle examples and their usage areas, developed obstacle avoidance algorithms and environment perception techniques in literature were investigated. Additionally, force-reflecting teleoperation systems, and virtual impedance technique were explained. According to design criteria, hardware components to provide a wireless communication line, suitable sensors to perceive near objects and measure inner states of the vehicle were investigated.

In Chapter 3, design criteria of the unlimited-workspace teleoperation system were identified. According to design criteria, possible omnidirectional slave subsystem concepts were developed and compared. Then, the most suitable one with respect to the design criteria was selected. Depending on the selected slave subsystem, locomotion components were determined. Because of the omnidirectional mobility of the slave system, a dynamic obstacle avoidance system was developed and components of the system were explained. In addition, communication line and master subsystem that were used to configure the unlimited-workspace teleoperation were described. The system was configured so that the vehicle control algorithms developed in Matlab Simulink environment could be deployed to the system. This provides an open architecture for

further studies with this test system by implementing new control algorithms via Matlab Simulink.

Equation of motion of the developed vehicle was developed in Chapter 4, and then according to these Equations, two control algorithms are proposed; kinematic and dynamic control. Additionally, two obstacle avoidance systems were developed. One of them is the semi-autonomous obstacle avoidance algorithm in which the vehicle detects obstacles and avoids them autonomously. Since, desired unlimited-workspace teleoperation system is force-reflecting bilateral teleoperation system, second algorithm was designed as force-reflecting obstacle avoidance algorithm. In the second algorithm, operator feels the location of the near objects haptically through force feedback joystick and avoids them.

In Chapter 5, the proposed two obstacle avoidance algorithms were tested and test results were represented and discussed. As a result of semi-autonomous tests, the robot autonomously avoids the obstacles along motion direction of the robot, which is perceived by dynamic obstacle avoidance system. Also, force-reflecting obstacle avoidance algorithm gives accurate results according to distance between the robot and obstacle including the information on the direction of the obstacle which is according to local coordinate frame. In addition, kinematic and dynamic control algorithms were tested. However, dynamic control algorithm tests were not presented in this thesis since it produced unacceptable results due to the slip problem, manufacturing errors and the procedure in measuring the wheel speeds. Moreover, in Chapter 5, the addition of an external sensor on the kinematic control algorithm was tested. It is shown that closing the kinematic control loop with a gyroscope improved the performance of the vehicle in following orientation demands.

In the future studies, velocities of the wheels can be measured with encoders which can be attached on the rear end of the motors for consistent wheel velocity signals instead of velocity measuring method with Hall-effect sensors. Making use of the accurate measurement of wheel velocities, dynamic control algorithm tests can be repeated by using velocity information integrated from the accelerometer measurements. Also, in force reflecting obstacle avoidance algorithm, virtual force is modeled to be created by a virtual spring between the obstacle and the vehicle. This would produce the same amount of force feedback for different velocities of the vehicle. This may be problematic in the cases when the vehicle approaches the obstacle with faster speeds. Hence, in the future work, force-reflecting obstacle avoidance algorithm

can be modified to create the force feedback information by a virtual spring and damper model.

REFERENCES

Almeida A. T. de, Khatib O. (Editors), 1998, Autonomous Robotic Systems, Lecture Notes in Control and Information Sciences 236.

Acroname (2012), Acroname website, (Accession date: 06.11.2012), http://www.acroname.com/robotics/info/articles/sharp/sharp.html

Aselsan (2012), Aselsan website, (Accession date: 06.11.2012), http://www.aselsan.com.tr/

Bogan (2012), (Accession date: 06.11.2012), http://www.bogan.ca/astro/telescopes/digtcrcl.html

Butner S.E., Ghodoussi M., 2003, Transforming a Surgical Robot for Human Telesurgery, IEEE Transactions on Robotics and Automation, Vol. 19, No. 5, October 2003, pp. 818-824.

Byun K.-S., Kim S.-J., Song J.-B., 2002, Design of a Four-wheeled Omnidirectional Mobile Robot with Variable Wheel Arrangement Mechanism, International Conference on Robotics and Automation Washington, DC, pp. 720-725.

Chamberland S., Beaudry É., Clavien L., Kabanza F., Michaud F., Lauriay M., 2010, Motion Planning for an Omnidirectional Robot with Steering Constraints, The IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan, pp. 4305-4310.

Chao C.-H., Hsueh B.-Y., Hsiao M.-Y., Tsai S.-H., Li T.-H. S., 2009, Real-Time Target Tracking and Obstacle Avoidance for Mobile Robots using Two Cameras, ICROS-SICE International Joint Conference, pp. 4347-4352.

Chen Y.-S., Juang J.-G., 2009, Intelligent Obstacle Avoidance Control Strategy for Wheeled Mobile Robot, ICROS-SICE International Joint Conference, pp. 3199-3204.

Cheng S.-W., 2008, Rapid Deployment UAV, Aerospace Conference, IEEE, pp. 1-8.

Cho S. K., Jin H. Z., Lee J. M., Yao B., 2010, Teleoperation of a Mobile Robot Using a Force-Reflection Joystick With Sensing Mechanism of Rotating Magnetic Field, IEEE/ASME Transactions On Mechatronics, Vol. 15, No. 1, pp. 17-26.

Coldene Castor and Wheels (2012), Coldene Castor and Wheels website, Accession date: 06.11.2012, http://www.coldenecastorsandwheels.co.uk

Connette C. P., Pott A., Hagele M., Verl A., 2008, Control of an Pseudo-omnidirectional, Non-holonomic, Mobile Robot based on an ICM Representation in Spherical Coordinates, The 47th IEEE Conference on Decision and Control, Cancun, Mexico, pp. 4976-4983.

Dede M. İ. C., 2007, Adaptive Fault-Tolerant Teleoperation, Florida International University.

Diolaiti N., Melchiorri C., 2002, Teleoperation of a Mobile Robot through Haptic Feedback, IEEE Int. Workshop on Haptic Virtual Environments and Their Applications Ottawa, Ontario, Canada, pp. 67-72.

Doroftei I., Grosu V., Spinu V., 2008, Design and Control of an Omni-Directional Mobile Robot, Novel Algorithms and Techniques in Telecommunications, Automation and Industrial Electronics, pp. 105–110.

Ferland F., Clavien L., Fremy J., L´etourneau D., Michaud F., Lauriay M., 2010, Teleoperation of AZIMUT-3, an Omnidirectional Non-Holonomic Platform with Steerable Wheels, The IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan, pp. 2515-2516.

Fong T., Thorpe C., 2001, Vehicle Teleoperation Interfaces, Autonomous Robots 11, pp. 9–18.

Golub G.H., Kahan W., Calculating The Singular Values And Pseudo-Inverse Of A Matrix, SIAM J. Numer. Anal. 2, Ser. B (1965), pp. 205–224.

Goris K., 2004 – 2005, Autonomous Mobile Robot Mechanical Design, Vrije Universideit Brussel.

Grange S., Fong T., Baur C., 2000, Effective Vehicle Teleoperation on the World Wide Web, IEEE International Conference on Robotics & Automation, San Francisco, CA, pp. 2007-2012.

Iheartrobotics (2012), (Accession date: 06.11.2012), http://www.iheartrobotics.com/2009/06/brushless-motor-internals.html

Ishida S., Miyamoto H., 2010, Ball Wheel Drive Mechanism for Holonomic Omnidirectional Vehicle, World Automation Congress.

Habib M. K., Baudoin Y., 2010, Robot-Assisted Risky Intervention, Search, Rescue and Environmental Surveillance, International Journal of Advanced Robotic Systems, Vol. 7, No. 1, pp. 1-8.

Han K.-L., Choi O.-K., Kim J., Kim H., Lee J. S., 2009, Design and Control of Mobile Robot with Mecanum Wheel, ICROS-SICE International Joint Conference, Fukuoka International Congress Center, Japan, pp. 2932-2937.

Han K.-L., Kim H., Lee J.S., 2010, The Sources of Position Errors of Omni-directional Mobile Robot with Mecanum Wheel, IEEE, pp. 581-586.

Han K.-L., Choi O. K., Lee I., Hwang I., Lee J. S., Choi S., 2008, Design and Control of Omni-Directional Mobile Robot for Mobile Haptic Interface, Control, Automation and Systems, Seoul, Korea, pp. 1290-1295.

Hanly E.J., Broderick T.J., 2005, Telerobotic Surgery, Operative Techniques in General Surgery, pp. 170 – 181.

Heliplane (2012), (Accession date: 06.11.2012), http://www.heliplane.net/2010/08/hype-professional-piezo-gyro/

Hokayem P.F., Spong M.W., 2006, Bilateral teleoperation: An historical survey, Automatica 42, pp. 2035-2057.

Holland J. M., 2004, Designing Autonomous Mobile Robots, Elsevier.

Horan B., Creighton D., Nahavandi S., Jamshidi M., Bilateral Haptic Teleoperation of an Articulated Track Mobile Robot, IEEE.

Jin T. S., Lee J. M., Hashimoto H., 2004, Internet-based Obstacle Avoidance of Mobile Robot Using a Force-reflection, IEEE/RSJ International Conference on Intelligent Robots and Systems, Sendai, Japan, pp. 3418-3423.

Jones, et al. J. L., Flynn A. M., Seiger B. A., Mobile Robots Inspiration to Implementation, Second Edition.

Kaneko K., Tokashiki H., Tanie K., Komoriya K., 1998, Macro-Micro Bilateral Teleoperation Based on Operational Force Feedforward, Proceedings of the 1998 IEEE/RSJ Intl. Conference on Intelligent Robots and Systems Victoria, B.C., Canada, pp. 1761-1769.

Kim S.-H., Rohl C.-W., Kang S.-C., Park M.-Y., 2006, A Hybrid Autonomous / Teleoperated Strategy for Reliable Mobile Robot Outdoor Navigation, SICE-ICASE International Joint Conference, Busan, Korea, pp. 3120-3125.

Lauria M., Nadeau I., Lepage P., Morin Y, Giguere P., Gagnon F., Letourneau D., Michaud F., 2006, Design and Control of a Four Steered Wheeled Mobile Robot, IEEE, pp. 4020-4025.

Lauria M., Michaud F., Legault M.-A., Létourneau D., Rétornaz P., Nadeau I., Lepage P., Morin Y., Gagnon F., Giguère P., Frémy J., Clavien L., 2008, Elastic Locomotion of a Four Steered Mobile Robot, 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems Acropolis Convention Center, Nice, France, pp. 2721-2722.

Lee S., Sukhatme G. S., Kim G. J., Park C.-M., 2002, Haptic Control of a Mobile Robot: A User Study, IEEE/RSJ Int. Conference on Intelligent Robots and Systems, EPFL, Lausanne, Switzerland, pp. 2867-2874.

Liu J., Sun L., Chen T., Huang X., Zhao C., 2005, Competitive Multi-robot Teleoperation, IEEE International Conference on Robotics and Automation Barcelona, Spain, pp. 75-80.

Maaref H., Barret C., 2002, Sensor-based navigation of a mobile robot in an indoor environment, Robotics and Autonomous Systems 38, pp 1–18.

Minguez J., Montano L., 2004, Nearness Diagram (ND) Navigation: Collision Avoidance in Troublesome Scenarios, IEEE Transactions on Robotics and Automation, Vol. 20, No. 1, pp. 45-59.

Mohebbi A., Safaee S., Keshmiri M., Keshmiri M., Mohebbi S., 2010, Design, Simulation and Manufacturing of a Tracked Surveillance Unmanned Ground Vehicle, IEEE International Conference on Robotics and Biomimetics, Tianjin, China, pp. 1268-1275.

National Instruments Application Note 007, Data Acquisition (DAQ) Fundamentals, http://physweb.bgu.ac.il/COURSES/SignalNoise/data_aquisition_fundamental.pdf.

Ota J., Arai T., Yoshida E., Kurabayashi D., Mori T., Real Time Planning Method for Multiple Mobile Robots, IEEE, pp. 406-411.

Pin F. G., Killough S. M., 1994, 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.

Park J. B., Lee B. H., Kim M. S., 2003, Remote Control of a Mobile Robot Using Distance-Based Reflective Force, IEEE International Conference on Robotics & Automation, Taipei, Taiwan, pp. 3415-3420.

Park S., Seo C., Kim J.-P., Ryu J., 2011, Robustly stable rate-mode bilateral teleoperation using an energy-bounding approach, Mechatronics 21, pp. 176–184.

Passenberg C., Peer A., Buss M., 2010, A survey of environment-, operator-, and task-adapted controllers for teleoperation systems, Mechatronics 20, pp. 787–801.

Purwin O., D'Andrea R., 2006, Trajectory Generation and Control for Four Wheeled Omnidirectional Vehicles, Robotics and Autonomous Systems 54, pp. 13–22.

Roboshop (2012), Roboshop website, (Accession date: 06.11.2012), http://www.robotshop.com/dfrobot-serial-bluetooth-module-3.html

Rösch O. J., Schilling K., Roth H., 2002, Haptic interfaces for the remote control of mobile robots, Control Engineering Practice 10, pp. 1309–1313.

Sakagami N., Shibata M., Hashizume H., Hagiwara Y., Ishimaru K., Ueda T., Saitou T., Fujita K., Kawamura S., Inoue T., Onishi H., Murakami S., 2010, Development of a Human-Sized ROV with Dual-Arm, IEEE.

Sohn W.-J., Hong K.-S., 2006, Moving Obstacle Avoidance Using a LRF Sensor, SICE-ICASE International Joint Conference, Busan, Korea, pp. 5957-5862.

Sparkfun (2012), Sparkfun website, (Accession date: 06.11.2012), https://www.sparkfun.com/products/11217

Tadakuma K., Tadakuma R., 2007, Mechanical Design of "Omni-Ball": Spherical Wheel for Holonomic Omnidirectional Motion, IEEE Conference on Automation Science and Engineering Scottsdale, AZ, USA, pp. 788-794.

Tadakuma K., Tadakuma R., Berengeres J., 2007, Development of Holonomic Omnidirectional Vehicle with "Omni-Ball": Spherical Wheels, IEEE/RSJ International Conference on Intelligent Robots and Systems San Diego, CA, USA, pp. 33-39.

Takemura Y., Sanada A., Ichinose T., Nakano Y., Nassiraei A.A.F., Azeura K., Kitazumi Y., Ogawa Y., Godler I., Ishii K., Miyamoto H., 2007, Development of "Hibikino-Musashi" omni-directional mobile robot, International Congress Series 1301, pp. 201–205.

Takemura Y., Ogawa Y., Nassiraei A.A.F., Sanada A., Kitazumi Y., Godler I., Ishii K., Miyamoto H., 2008, System Design Concept Based on Omni-Directional Mobility, Safety and Modularity for an Autonomous Mobile Soccer Robot, Journal of Bionic Engineering Suppl., pp. 121–129.

Techeblog (2012), (Accession date: 06.11.2012), http://www.techeblog.com/elephant/photo.phtml?post_key=146432&photo_key=7656

Tsai C.-C., Wu Z.-R., Wang Z.-C., Hisu M.-F., Wang Z.-C., 2010, Adaptive Dynamic Motion Controller Design for a Four-Wheeled Omnidirectional Mobile Robot, International Conference of Systems Science and Engineering, pp. 233-238.

Udengaard M., Iagnemma K., 2007, Kinematic Analysis and Control of an Omnidirectional Mobile Robot in Rough Terrain, IEEE/RSJ International Conference on Intelligent Robots and Systems San Diego, CA, USA, pp. 795-800.

Wada M., 2008, A 4WD Omnidirectional Wheelchair with a Chair Tilting System, IEEE International Conference on Mechatronics and Automation, pp. 21-26.

Wada M., 2008, Mechanism and Control of a 4WD Robotic Platform for Omnidirectional Wheelchairs, The IEEE/RSJ International Conference on Intelligent Robots and Systems, St. Louis, USA, pp. 4855-4862.

Watanabe K., Control of an Omnidirectional Mobile Robot, 1998 Second International Conference on Knowledge-Based Intelligent Electronic Systems, Adelaide, Australia, pp. 51-60.

Wikipedia (2012), (Accession date: 06.11.2012), http://en.wikipedia.org/wiki/Bomb_disposal

Wikipedia (2012), (Accession date: 06.11.2012), http://en.wikipedia.org/wiki/Sonar

Wikipedia (2012), (Accession date: 06.11.2012),
    http://en.wikipedia.org/wiki/Microcontroller

Wikipedia (2012), (Accession date: 06.11.2012),
    http://en.wikipedia.org/wiki/File:Wireless_network.jpg

Wu K., 2004, Realtime Control of a Mobile Robot Using Matlab, The University of Applied Science Hamburg.

Xu Z., Ma L., Schilling K., 2009, Passive bilateral teleoperation of a car-like mobile robot, 17th Mediterranean Conference on Control & Automation, Makedonia Palace, Thessaloniki, Greece, pp. 790-796.

Vexrobotics (2012), Vexrobotics website, (Accession date: 06.11.2012),
    http://www.vexrobotics.com

Yoon W.-K., Goshozono T., Kawabe H., Kinami M., Tsumaki Y., Uchiyama M., Oda M., Doi T., 2004, Model-Based Space Robot Teleoperation of ETS-VII Manipulator, IEEE Transactions On Robotics and Automation, Vol. 20, No. 3, pp. 602-612.

Zhang L., ZhiXin C., Jia W., 2009, A Networked Teleoperation System for Mobile Robot with Wireless Serial Communication, IEEE.

Zhi-hua Q., Yi-bo L., Shao-peng K., Qiong Z., 2008, Design of UAV Telepresence and Simulation Platform based on VR, International Conference on Cyberworlds, pp. 520-524.

Zhu C., Oda M., Luo X., Watanabe H., Yan Y., 2009, Platform Development of an Omnidirectional Mobile Robot for the Elderly's Walking Support and the Caregiver's Power Assistance, International Conference on Robotics and Biomimetics, Guilin, China, pp. 1900-1905.

APPENDIX A

MASS PROPERTIES OF THE DEVELOPET ROBOT

Developed slave subsystem in this study was modeled in Solidworks (Figure A.1). Also, all components of the system were modeled according to real dimentions and materials. Moment of inertia of the vehicle utilized in dynamic control model was taken from Solidworks Mass Properties window. It is represented in Figure A.2.
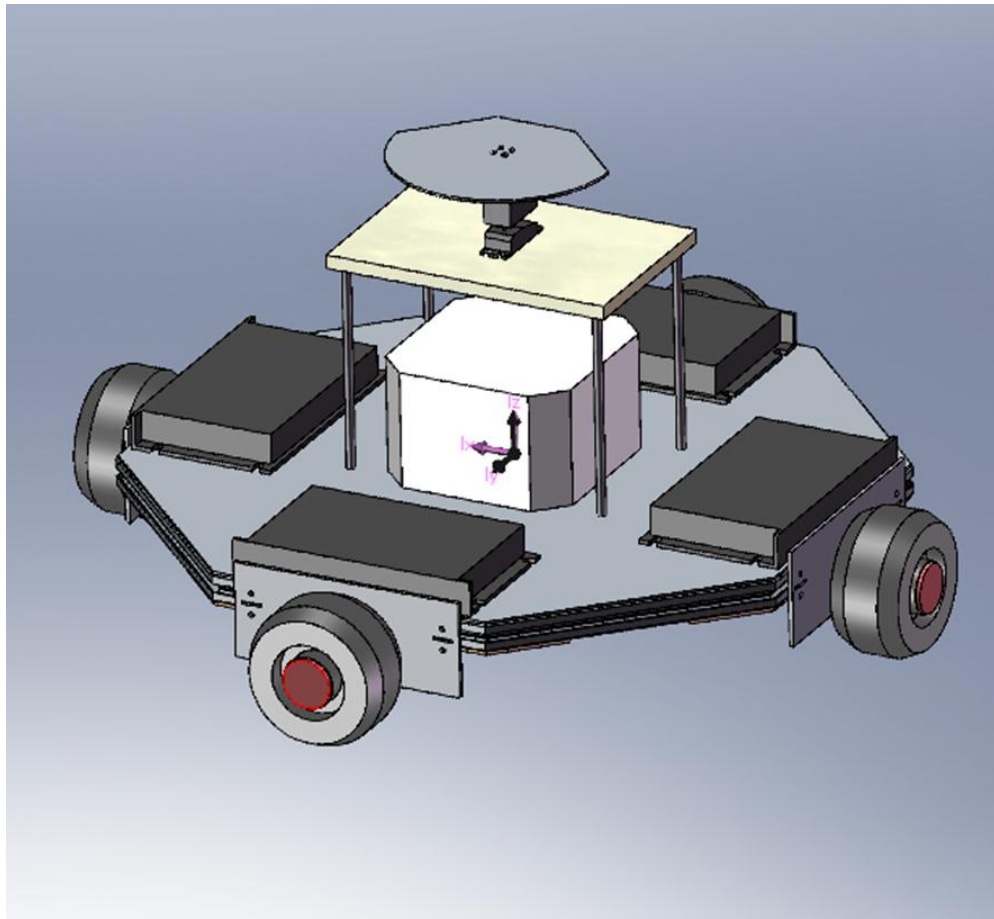

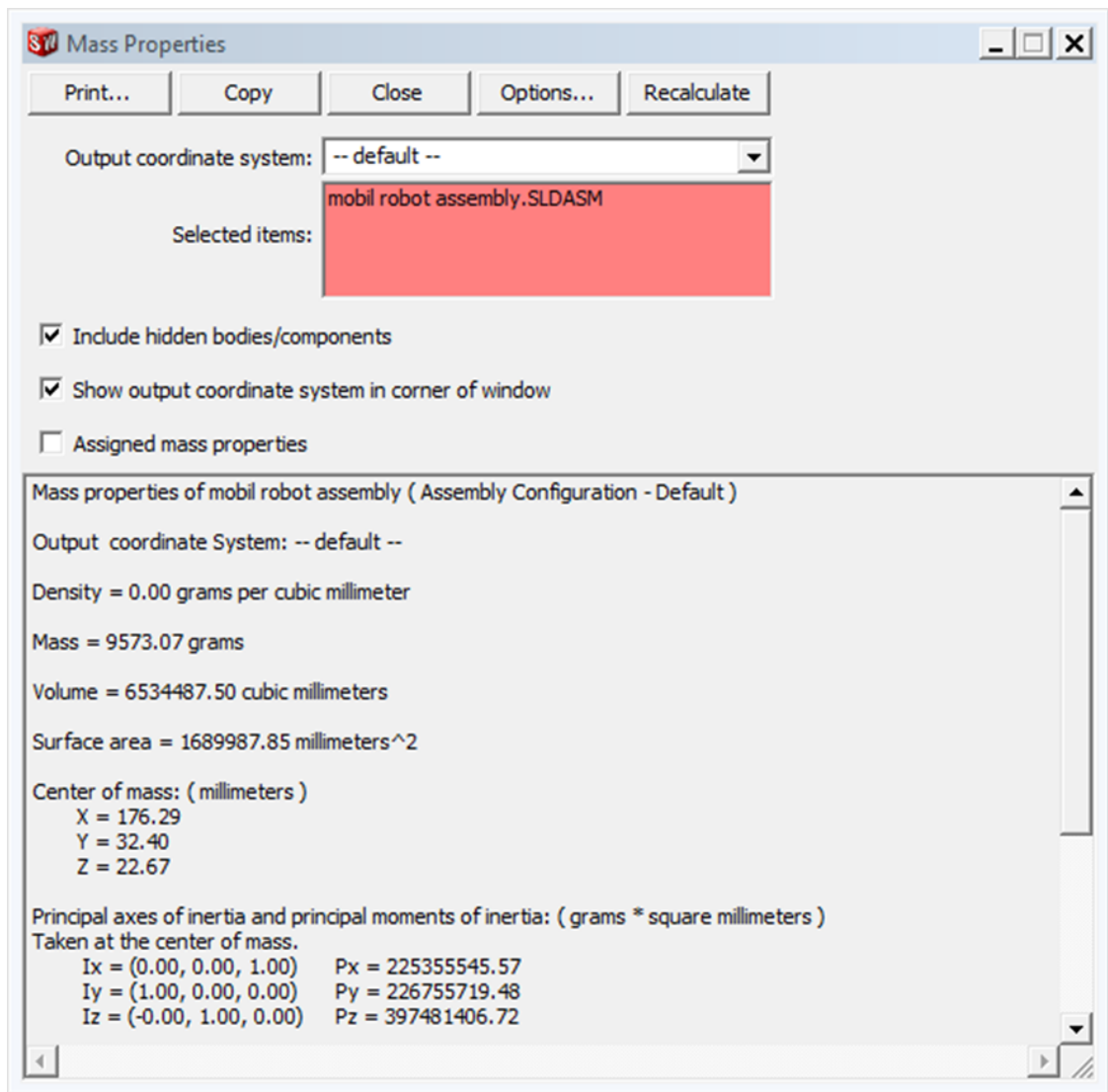Figure A.1. CAD model of developed mobile robot

Figure A.2. Mass properties of the developed slave subsystem

APPENDIX B

SIMULINK EMBEDDED OPTION

In this thesis, control and obstacle avoidance algorithms were modeled with Matlab Simulink and they were embedded on slave subsystem through xPC Target Embedded Option. It allows deploying stand-alone applications on the target PC independent of the host PC.

To embed control and obstacle avoidance algorithms, firstly, standalone mode must be enabled under node of target PC configuration in xPC Target Explorer hierarchy pane (Figure B.1). Target Explorer is called in Command Window by entering "xpcexplr".
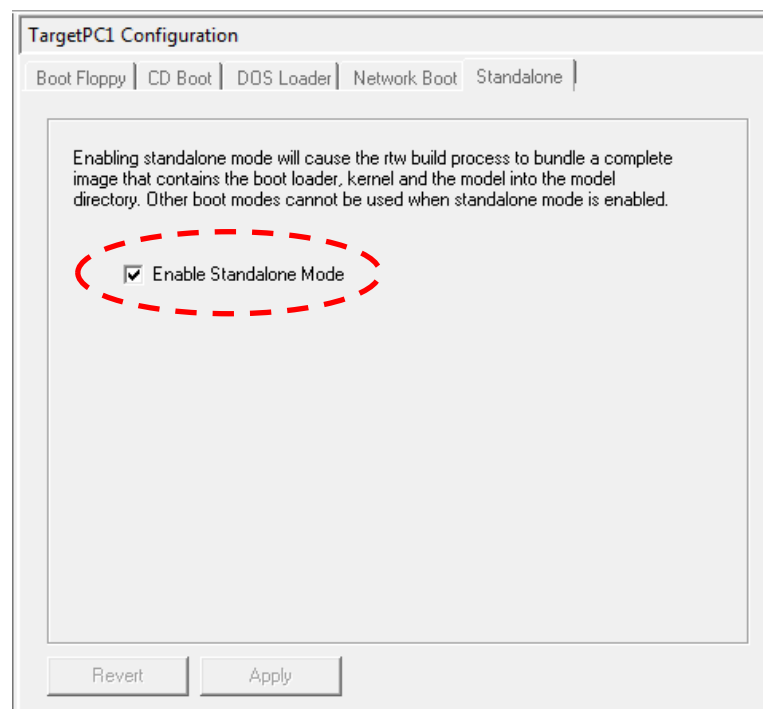

Figure B.1. TargetPC Configuration

After that, building process of the Simulink creates a directory whose name is same with model name in the current working folder on the host computer. It contains three files:

- *.rtb — This file contains the xPC Target kernel. It also contains, as applicable, options such as serial or TCP/IP communications and the IP address of the target PC.

- xpcboot.com — This file executes loads and executes the *.rtb file.

- autoexec.bat — xPC Target version of this file that calls the xpcboot.com executable to boot the xPC Target kernel.

Then, in the xPC target explorer window, the target PC must be connected and these three files must be copied on the target PC hard drive. The target PC automatically runs created algorithms at its first startup.