RENDEZVOUS OF HETEROGENEOUS ROBOTS WITH LIMITED SENSING CAPABILITIES

A Project Report

submitted by

ANIKETH ANIL MORE

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

MASTER OF TECHNOLOGY

DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY MADRAS.

MAY 2014

THESIS CERTIFICATE

This is to certify that the thesis titled RENDEZVOUS OF HETEROGENEOUS ROBOTS WITH LIMITED SENSING CAPABILITIES, submitted by **ANIKETH ANIL MORE**, to the Indian Institute of Technology, Madras, for the award of the degree of **MASTER OF TECHNOLOGY**, is a bona fide record of the research work done by him under our supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Prof.K.SRIDHARAN
Project Guide
Professor
Dept. of Electrical Engineering
IIT-Madras, 600 036

Place: Chennai

Date: 05th May 2014

ACKNOWLEDGMENT

I take this opportunity to express my gratitude to my guide Prof. Dr. K. Sridharan sir, for his exemplary guidance, monitoring and constructive suggestions throughout the course of this project. He always made sure that everything that is necessary for the project be it equipment or knowledge was readily available. His sense of timing and his vast area of expertise have inspired me in so many ways. It was because of this motivation that I was able to perform tasks which I had thought were impossible to perform. It has been a wonderful experience working under him. I would also like to thank P. Prabhakar Rao sir, for his expertise in hardware. His experience has save me a lot of time.

My grateful thanks are also extended to P. Anand sir, Technical Superintendent,IE lab for his continuous help during this project. He has provided us with many components necessary that were otherwise very difficult to obtain. I would also like to thank Bhaskar and Lokesh for maintaining a proper record of their project. Their project was a benchmark for me to start with. I would like to thank XMOS Semiconductor (India) Pvt Ltd for providing Motor Control Board, XK-1A and XC-1A. Working with these boards gave me an insight of the high performance microprocessors and helped me understand the technology available in the industries today.
I would also thank my lab mates Vikram, Tahiyah and Srinivas for giving me keys to the lab at graveyard hours so that I could work in lab.I would like to thank my friend B.A Baji for supporting me throughout the project.

I would especially like to thank my team mate S.V.V.Raju for his continuous help in the project. Finally, I wish to thank my parents and my sister for their support and encouragement throughout my study.

Abstract

The robotic future is heading towards robots collaborating to perform a task. The major requirement to co-ordinate a task is to position the robots. When the numbers of robots collaborating are more it becomes necessary for right combination of robots to find themselves at right position in order to perform the task efficiently. Most of the systems present today widely use GPS for localization. Though there are a lot of advantages of GPS and other such positioning systems, there are few disadvantages like the reachability, dependence on a global system, cost, etc.

This project helps two heterogeneous robots to find each other and position themselves at a given point without using satellite or any other global positioning systems. The robots are brought together by using the simplest possible sensors. This reduces the complexity and the cost required in achieving this very important task. The sensors used in this project are rigorously tested in order to understand the behaviour of the sensor in different conditions and environment. The proposed algorithm has been carefully designed in order to utilize the on board sensors to the best extent and is tested on two robots attempting to achieve rendezvous. The algorithm has been successfully implemented on a biped and a mobile robot.

Contents

List of Figures

Chapter 1

INTRODUCTION

1.1 Motivation

The major purpose of robots is to replace the humans in hazardous places like mines. It is essential for the robots to coordinate with each other for doing tasks in these places

In this report, we consider a particular aspect of multi-robot environment exploration: how to get a pair of robots to meet one another initially in an unknown environment if they do not know each others starting positions. The problem of robot rendezvous is a key step in collaborative exploration. A rendezvous is a meeting between two or more agents at an appointed place and time, for example, when two people meet at a familiar restaurant. The ability to meet facilitates localization, allows collaborative map exploration and has a plethora of other advantages, but most importantly allows communication. Most existing hardware agents are only capable of communication over short distances. Environmental geometry, wireless transmission technology, power considerations and atmospheric conditions (or water conditions for underwater agents) all contribute to fairly short communication limits.

In the absence of sophisticated satellite receivers or high power devices, a common constraint for successful communication is maintaining line of sight between agents, a constraint that is rarely satisfied in the real world. However, multi-agent robot systems for the majority of real-life applications enjoy substantial speed gains only with some level of communication. This project deals with bringing together two different types of robots, biped a two limbed robot and a mobile robot. The main objective of the project is to build upon a strategy which is simple and effective to help the robots meet at a rendezvous point autonomously.

The objective of this project is to obtain rendezvous point between biped and mobile robot with limited sensing capabilities. Here, the sensors are used only to find the relative angle between the two robots and a distance measurement sensor to detect obstacles. The sensors implemented here are IR sensors for locating the direction of target robot. But the range of IR sensors is limited by constraints like range, surrounding environment. If the target is within the range of visibility, then the robot executes the defined convergence algorithm for getting close to the target robot. There is possibility that the target would not be in the visibility region of the robot. Thereby, the robot should make the target be in its visibility range and once the robot gets within the range the convergence algorithm will make sure that it never escapes from the visibility range.

1.2 Contribution of the project

In this project, we investigated the convergence of two heterogeneous robots to a point with minimal sensing capabilities. This convergence has been achieved without the help of coordinates of the positions of the robots and without any information regarding the initial position of the robots.

- Rendezvous problem has been achieved without the help of co-ordinates between mobile robot and a biped.
- The convergence of two robots to a point when there are no obstacles has been achieved.
- The convergence of two robots to a point has been achieved in the presence of obstacles and other reflecting mediums.
- Experiments have been carried out on the variations of the transceiver signals due to reflection and designed algorithm to reduce its effect.

1.3 Organization of the project

- Chapter 1 gives the introduction to the project.
- Chapter 2 gives insight on the rendezvous problem.
- Chapter 3 explains about the proposed algorithm for the rendezvous problem.

- Chapter 4 explains about the details of robots and sensors that have been used.
- Chapter 5 explains the experiments conducted on IR transceiver sensors.
- Chapter 6 explains about the convergence of two robots in absence of obstacles.
- Chapter 7 explains about the strategies implemented for convergence of robots avoiding obstacles overcoming the reflection problem.
- Chapter 8 gives the conclusion of the project.

Chapter 2

LITERATURE SURVEY

The problem of rendezvous is not a new one; there exists a body of research in the optimization and operations research community involving search problems. Rendezvous is a particular variant of the search problem, similar to games with mobile hiders, called princess and monster games[SG95]. There are many variants of the rendezvous problem itself, involving distinguishable or indistinguishable agents and collaborating or interfering agents. The environment may have focal points, or may be completely homogeneous.

There are a number of differences between the highly theoretical approach most prior work takes and the approach used here. In our work, the environment is unknown, and one of the key problems is to obtain convergence of two robots with the concomitant problems of noise and other limiting constraints.

Research has been done on multi-agent rendezvous without position coordinates by Jingjin Yu in 2012[YLL12]. Dubins car vehicles equipped with sensors are used as agents which obtained rendezvous without ever obtaining coordinate data or performing state estimation. A n-dimension rendezvous problem was approached via proximity graphs which is closely related to cyclic pursuit or the n bug problem. Their work is similar to our work in terms of minimal use of sensors and obtaining rendezvous, thereby minimizing the cost. Other than this, other similarity with our work is the quantized control without the need for perfect coordinate information of the robots.

For Sensing of other robots location has been done by IR transceiver which is combination of few transmitters and receivers.Similar sensor has been designed by G.Bemes and F.Blames [GF02] with which the distance between two robots has been measured. That sensor works base on the light intensity back scattered from the objects and is able to measure distances up to 1 m.

Some work has been carried on selection of optimal rendezvous points which would lead to more efficient exploration which allows robot to re-plan when one of them has unexpected obstacles in its path[dHCV10]. This work has meet this requirement by tracking other robot instantaneously

A distributed algorithm for converging autonomous mobile robots with limited visibility toward a single point has been designed by Hideki Ando, Yoshinobu Oasa and Ichiro Suzuki[AOSY99] where next position of the robots is entirely determined from the positions of the robots that it can see at that moment. Our work is similar this point as algorithm that has been used will only consider the present information of the robot and wont store any information about other robots previous positions.

Chapter 3

PROPOSED ALGORITHM FOR RENDEZVOUS PROBLEM

The algorithm designed to achieve the rendezvous between the robots is independent of the level of reflection of the surroundings. The obstacles treated to be in repulsive potential filed, which is handled by the ultrasonic sensor and the transceiver helps in handling the attractive potential field. One of the robots is given a map to the rendezvous point the other just follows the robot to the point. The Robots in doing so may not take the same path to rendezvous depending upon the environmental conditions.

The transceiver pair are subjected to attract each other till the robots come closer than 2 feet of each other. Once the distance between the robots or the transceiver pairs is less than 2 ft, the field changes to neutral, i.e the robots come to halt. This happens due to the crowding of IR signals when the transceiver pair are close to each other. The threshold value obtained in adaptation mode helps in detecting this crowded phenomenon.

3.1 Algorithm

3.1.1 Algorithm for rendezvous without obstacles

The algorithm for convergence of two robots without any obstacles in between is as follows:

1. Initialize N and M(N>M) number of samples for adaptation mode and action mode respectively.

2. Start sampling sensor outputs for adaptation mode and update the direction variables.

3. At the end of N samples, the threshold for the action mode is calculated and the required tolerance is added. The direction variables are reset to zero.

4. Once the threshold value is generated, the action mode is initialized.

5. The sensor outputs are sampled M times and the direction variables are updated.

6. The direction variables are compared with the threshold and corresponding direction is decided.

3.1.2 Algorithm for rendezvous in presence of obstacles

The algorithm for convergence of two robots in presence of obstacles in between is as follows:

1. Initialize N and M(N>M) number of samples for adaptation mode and action mode respectively.

2. Start sampling sensor outputs for adaptation mode and update the direction variables.

3. At the end of N samples, the threshold for the action mode is calculated using the threshold equation. The direction variables are reset to zero.

4. Once the threshold value is generated, the action mode is initialized.

5. The sensor outputs are sampled M times and the direction variables are updated.

6. The direction variables are compared with the threshold and corresponding direction is decided.

7. The obstacle sensor output is checked and hence the course of action is taken (moving forward, right or left)

3.2 Different modes of algorithm

The algorithm has been divided into two modes :

1. Adaptation mode.
2. Action mode.

3.2.1 Adaptation mode

During adaptation mode, the robot is stationary. Only the Transceiver pair is switched on. During this mode, the sensor data is collected in order to obtain the threshold value for the action mode. This mode helps the robot decide an appropriate threshold based on the lighting and reflective conditions of the environment, hence the name adaptation mode. The following steps describe adaptation mode.

1. Initialize N (number of samples required)
2. sample the direction sensor outputs "N" times at a sampling rate of more than 20 Hz.After each sample check for the minimum value at the output pins.
3. Add all the minimum values at the end of "N" samples and average them over N.
4. Calculate the threshold value using the threshold equation.

Threshold equation

The threshold value generated during the adaptation mode plays an important role in avoiding false decision making by the robots due to reflection. The threshold value also helps in eliminating the need of an extra sensor to stop the mobile robot from moving once it has reached the rendezvous point or when the robots are close to each other.

The threshold value is decided by the threshold equation. The threshold equation is designed by studying the properties of the IR transceivers. The equation is as shown below,

Threshold value = $(\Phi * \epsilon + \Phi)/2$

Φ= average of set of least values
ϵ = multiplying factor (0,1)

Φ is calculated during the first half of adaptation mode, where the robot is stationary and keeps receiving signals N times per iteration. The minimum value obtained from each iteration is summed up and is averaged over the number of iteration, this generates the value of Φ.

ϵ is obtained experimentally and is confined to values between 0 and 1. The value of ϵ decides how close the robots will stop from each other. If the value of ϵ is more than required the robots will collide and if the value is lesser, the convergence will not take place. Hence it is necessary that the value of ϵ is determined accurately. In the experiments described in this report the value of ϵ is set to be 0.6

3.2.2 Action mode

In action mode the robot is in motion. This mode helps in tackling obstacles and reflections caused by shiny surfaces present closed to the robot. It constantly checks the direction with transceiver pair and compares it with the threshold obtained in adaptation mode. If the direction sensor value is less than the threshold value, the obstacle sensor output is checked and hence corresponding action is taken thereby reaching one step closer to the rendezvous point. The following steps describe action mode.

1. Initialize M (number of samples required)

2. sample the direction sensor outputs "N" times at a sampling rate of more than 20 Hz. After each sample check for the minimum value at the output pins.

3. Compare the minimum value with threshold.
   If obstacle sensor detects an obstacle, The direction is false and thereby check the next minimum value, else the direction is true

Chapter 4

DETAILS OF ROBOTS AND SENSORS USED

Two heterogeneous robots i.e., one biped and one mobile robot have been used conducting experiment on rendezvous problem. A heterogeneous robot refers to the robots of different kinds. Here, the motion characteristics of the biped and the mobile robot are different from each other.

4.1 Mobile robot

The designed mobile robot is of a heavy platform driven by two brush less D.C motors. These D.C. motors are rated as 12 Volts, 2.7 A and 26 W.
This mobile robot has two wheels at the rear side of it which are controlled by the two motors. There is a dummy wheel fixed at the front portion of the robot so as to provide sufficient balance to it.
Its motion is dependent on the rotation of two rear wheels.

Table 4.1: Motion of mobile robot

Sl no.	Right wheel	Left wheel	Robot motion
1	Clockwise	Clockwise	Move forward
2	Clockwise	Counter Clockwise	Turn left
3	Counter Clockwise	Clockwise	Turn right
4	Counter Clockwise	Counter Clockwise	Move backward

4.1.1 Motor control

The control of motor mainly includes the controlling of speed of motor and changing the direction of rotation through control signals from micro controller. This is done by controlling the voltage given to the motor. But, the motor is rated at 12 Volts and carries large amount of current that a micro Controller cannot drive the motors hence, a H bridge circuit has been used to drive the motors which would control the voltage fed to the motor. The output levels of this H bridge can be controlled by the PWM signals thereby controlling the motors. Care should be taken that the motors rotate at same speed by adjusting the duty cycles of corresponding PWM channels.

4.1.2 Xmos Motor Control Board

XMOS XC-1A is a low-cost development board based on the XMOS XS1-G4 device. It includes a single G4 device, 4Mbits SPI FLASH memory, 16 user-configurable LEDs, four push buttons, a speaker, JTAG and serial interfaces, four expansion areas suitable for IDC headers and a through-hole prototyping area for connecting external components. It consist of three microprocessors with clocks speeds of 250MHz and 100MHz. High clock speeds and multi-thread processing increases processing speed. The availability of large number of pins for interfacing is also a major advantage of XC-1A over boards similar to it.
XK-MC-LVM2 is a motor control unit built by XMOS, the architecture of the processor used in this unit is similar to that of XC-1A. The control board is equipped with a high power motor control board to power two DC motors.

Since, the project deals with mechanical systems and the objective of the project is to keep the sensing capabilities of the robots as low as possible. Most of the key features like the speed of the processor would not be utilized to its maximum extent. Hence, a combination of 'ATMEGA32' micro controller and 'Dual VNH5019' motor control boards were used in the project.

4.1.3 Motor driver board

For the control of the motors on the mobile robot, 'Dual VNH5019 motor driver board' has been used which is able to control the motion of two motors simultaneously through pwm signals generated by ATMEGA32 micro controller. The motor driver board is able to withstand currents up to 14 A current.

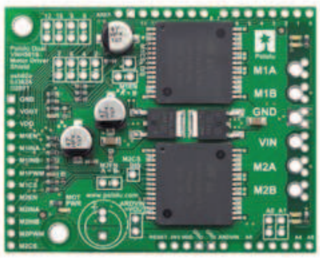

Figure 4.1: Pololu Dual VNH5019 Motor driver Board

4.1.4 Control of motors

The motors can be controlled through the input terminals IN1A,IN1B,IN2A,IN2B which decides the direction of rotation of motors.M1PWM and M2PWM are for the control of speed of the motors. If IN1A is high an IN1B is low, then the motor1 rotates in the clockwise direction and if IN1A is low and IN1B is high, then the motor 1 rotates in counter clockwise direction. The same logic holds for the motor 2 rotation with IN2A an IN2B pins. The speed of the motor can be varied by changing the duty cycle of the PWM that is being fed to the motor driver board. The time period of pulse is kept constant and the ON-Time is varied. The operating frequency of the pwm should be within the limit of 2 Khz to 20 Khz. The optimum frequency for controlling a DC motor lies between 10 KHz to 15 KHz.

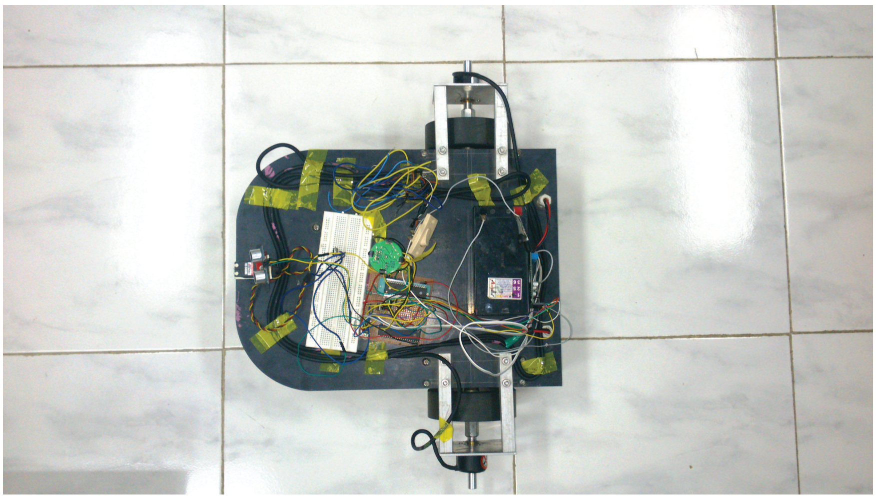

Figure 4.2: Mobile robot equipped with Sensors

4.2 Biped

A biped is a two limbed robot. The motion of the robot replicates the walking of a human being. The robot is built using an aluminium structure. Four servo motors placed at the waist and the foot of the biped act as the actuators. Motors are controlled by ATMEGA32 micro controller. Transceiver is mounted on the top of the biped. A 7.4 volt Lithium-Polymer battery powers the entire setup.

4.2.1 Biped gait algorithm- walking of biped

Walking of biped mainly involves in six steps.

1. Right bend out
2. Left leg forward
3. Right bend in
4. Left bend out
5. Right leg forward
6. Left bend in

The code for Right Bend out makes the biped to lean towards right and this; makes the left leg lose contact from ground and to be in air so that it can move. The code for left leg forward makes the left leg of the biped to move forward by a prefixed angle. The code for Right Bend in makes the Biped to regain the normal position. Usually this is used after bending out. The left bend in, out and right forward are similar to the above codes but in opposite direction. Execution of this algorithm will make the biped make one complete step forward.

4.3 Encoder

Two rotary encoders has been fixed on each side of the motor wheels of the mobile robot. Encoder used here is of incremental type of model E40H12-2000-3-N-5.This has six terminals of A,B,Z as output phases, one +5 V supply wire, one ground terminal and one shielded wire. Figure 4.4 shows the parts used to attach encoder to the wheels of mobile robot. "A" is the encoder used in the project, "B" is a M10 size coupling nut with with a length of 25

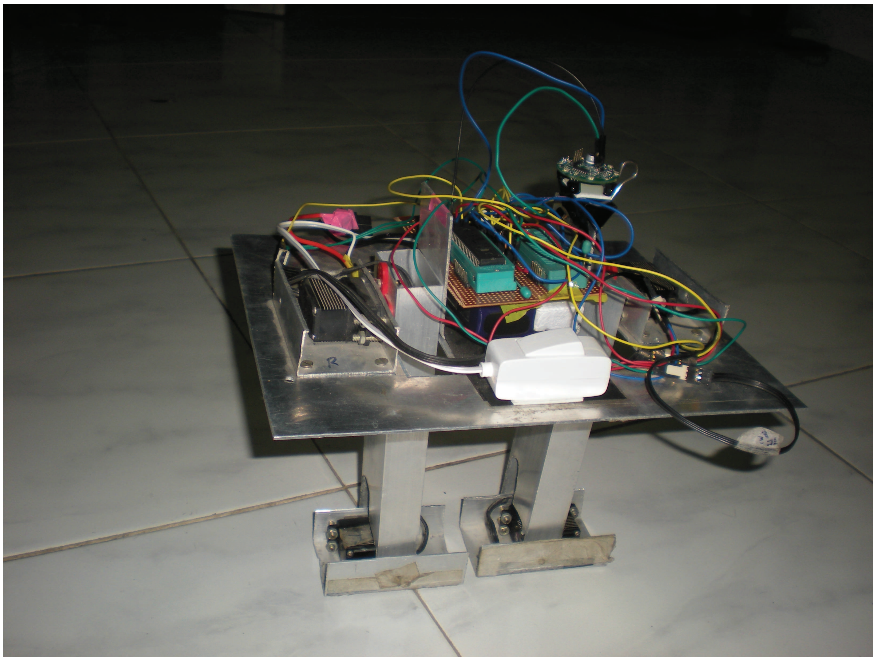

Figure 4.3: Biped with circuit board and transceiver placed on it

mm. The coupling shaft is of length 90 mm. The coupling nut connects the coupling shaft and the motor wheel.
This encoder generates 2000 pulses for one revolution and the output type is NPN open collector output.
In the NPN open collector type encoder, the output can't be taken directly from the terminals. There should be a load circuit connected across the output phase and the +V supply terminal. For this encoder, the maximum load current is given as 30 mA. The load should be chosen according to it. 470 Ω has been used at each phase so that the load current will be within the limits.

4.3.1 Working

This rotary encoder gives 2000 pulses per rotation from each phase. The output phases A,B are 90^o to each other as shown in the figure below.
The position of the wheel can be tracked by counting the number of pulses generated at the output terminals which has been programmed to count by ATMEGA32 micro controller. The direction of the rotation of the wheels can be tracked by differentiating the sequence of the pulses from A,B.

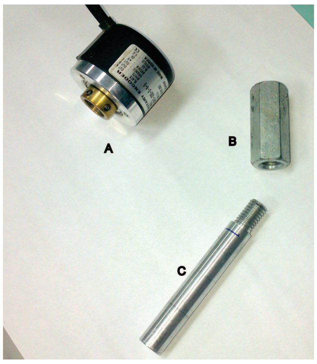

Figure 4.4: A : Encoder, B : coupling nut, C : coupling shaft

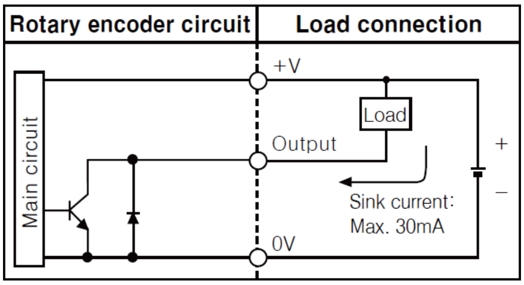

Figure 4.5: NPN- Open collector type terminal connection

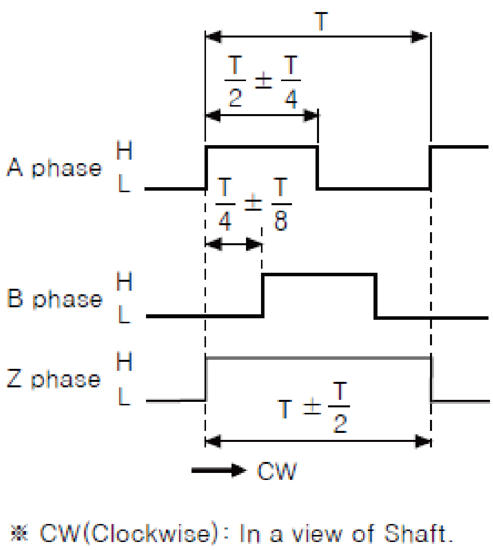

Figure 4.6: Waveforms of output terminals A,B and Z

The direction of the rotation is clockwise if the OUT B phase is logic high at the instant of rising edge of the OUT A phase and the direction is counter clockwise if its logic low.

If the OUTA,OUTB phases doesn't change for specific period of time, it is considered to be at halt.

The rotation of the wheels is counted by incrementing the variables M1,M2 which are for the left and right motors respectively. The values of M1,M2 are equal, when both wheels are moving at same speed. This implies that the mobile robot is moving in forward direction. If the value of M1 is greater than that of value of M2, then the robot is turning right and if M2 is greater, robot is turning towards its left.

4.3.2 Implementation as Feedback

The necessity for a feedback loop is due to the unpredictability of the motors used in mobile robots. There could be cases when the MCU controlling the motors of mobile robot give out signals to rotate one of the motors but due to some discrepancies the wheel does not respond to the control command. Hence, feedback loop is essential for a robust control. The output from the encoder is fed back to the micro controller which controls the wheel movement of the mobile robot. Thereby, updating the wheel rotation status to the MCU controlling the wheels of mobile robot.

A counter keeps counting the pulses generated by the encoders, the difference in the values of these pulses gives us the direction of movement of the robot. For example if "P" and "Q" are the number of pulses count of the two encoders mounted on the mobile robot. If the difference between "P" and "Q" is zero then the mobile robot is moving straight. If either "P" or "Q" is zero and the non zero value is equal to some "X" where "X" is the constant value set for a 90 degree turn , the mobile robot has made a turn.

4.4 Transceiver

The transceiver has four IR sensors and six transmitters connected along the circumference of a circuit board. These six emitters and four IR receivers which transmit and receive IR radiations simultaneously. The output from the IR sensors can be seen through the four terminals marked N,E,W,S which resembles the four directions. There are four LEDs connected to the sensors which will glow if that particular sensor is receiving signal. Apart from the four output pins, it has three pins which are +,- and E pins. the +,- pins are given to +12 Volts D.C and ground respectively. E pin is enable pin which should be kept high while using the receivers.

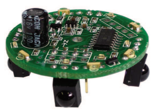

Figure 4.7: IR tranceiver

4.4.1 Working

The IR transceiver basically works as a compass showing the direction of source of IR signals that it would receive. If there is some IR source within the receiving range of the sensor, then the sensor corresponding to the direction of source receives the signal and it gives output high at the corresponding pin. The refresh output rate is 20 Hz. For example, let us assume the IR source in north direction. Then, the IR which is in direction 'N' will receive signal and sends logic high at Pin 'N' and the indicating LED corresponding to 'N' will glow indicating the reception of signal.

Chapter 5

IR TRANSCEIVER SENSOR ANALYSIS

In this chapter, experiments are conducted on the IR transceiver for understanding the behaviour of sensors in an environment with reflective surfaces. The experiments help us understand the reflection pattern in IR signals emitted by the transceiver pair.

5.1 Experimental setup for the analysis

There are two transceivers in use, one mounted on mobile robot and other on Biped. Both robots are kept away from each other but within the range of these sensors. The transceiver unit placed on biped acts as a transmitter and the unit arranged on the mobile robot acts as a receiver.

The sensors are placed at a height of 20cms from ground. The transceiver unit placed on Biped is placed upright for maximum signal emission. Whereas, the transceiver placed on the mobile robot acts as a receiver and hence is placed upside down . The orientations of the sensors are decided based on the intention of receiving and transmitting maximum amount of sensor data. Since the output refresh rate of IR transceiver is 20Hz,i.e the output changes every 50 milliseconds. Therefore, the output should be sampled at less than 25 milliseconds interval (according to sampling theorem).

Experiments are conducted on the IR transceiver by placing them at different positions and environments. Several experiments were conducted keeping obstacles on either side,creating a passage alongside the transceiver pair(the setup results in maximum reflection condition). The output of the transceivers were sampled at 20 milliseconds.A set of 2000 samples of the received signal is considered and recorded for each position for analysing the behaviour of the IR sensor. The results were displayed on four 7-segment display. The experimental data collected for various positions and environment are enlisted below.

1. Three boxes kept on either side of path.

   The received signals are shown in table: 5.1

Figure 5.1: All three boxes are kept

Table 5.1: Three Boxes on both sides

Sl no.	North	East	South	West
1	199	174	64	38
2	186	173	31	30
3	187	173	85	62
4	159	180	58	49
5	196	177	49	15
6	179	188	83	68
7	196	199	89	46
8	192	186	94	12
9	179	184	87	20
10	169	175	119	78

2. Boxes 2 and 5 are removed.

   The received signals are shown in table:5.2

Figure 5.2: boxes 1,3 are kept

Table 5.2: With Boxes 1 and 3

Sl no.	North	East	South	West
1	189	185	123	57
2	139	189	130	15
3	149	184	112	13
4	199	189	111	45
5	194	190	133	61
6	191	197	152	77
7	192	197	116	99
8	199	182	132	22
9	198	169	103	38
10	199	192	143	43

3. Boxes 3 and 4 removed.

   The received signals are shown in table: 5.3

Figure 5.3: Boxes 2,3 are kept

Table 5.3: Only Boxes 2,3 on both sides

Sl no.	North	East	South	West
1	175	145	113	67
2	178	174	130	18
3	149	173	120	24
4	129	157	120	64
5	148	171	103	00
6	165	156	123	11
7	170	176	131	31
8	185	152	97	58
9	192	165	113	91
10	192	169	105	75
11	196	190	140	18

4. Boxes 1 and 6 removed.

   The received signals are shown in table: 5.4

Table 5.4: Only boxes 1,2 on both sides

Sl no.	North	East	South	West
1	197	193	159	22
2	194	159	125	67
3	198	156	149	99
4	197	177	156	52
5	189	176	48	16
6	196	192	57	60
7	198	167	38	96
8	179	195	17	49
9	189	174	35	81
10	199	175	38	87

5. All boxes removed.

Table 5.5: All Boxes are removed

Sl no.	North	East	South	West
1	196	187	88	26
2	145	178	165	82
3	195	163	151	61
4	169	188	179	31
5	157	177	183	99
6	177	182	172	66
7	196	181	190	25
8	199	189	177	95
9	189	173	179	56
10	199	164	169	98

6. One of the transceiver unit is placed at the corner of a closed narrow room

Table 5.6: At the corner of a closed room

Sl no.	North	East	South	West
1	175	194	168	45
2	186	192	125	34
3	184	191	143	22
4	189	193	121	27
5	192	194	95	44
6	187	192	101	35
7	178	192	72	71
8	187	196	107	51
9	194	196	179	37
10	197	193	82	64

5.2 Conclusion from the experiments

1. In almost all cases the direction obtained by the sensor was the actual direction in which the other transceiver was placed.

2. There are two types of reflection problem.

   - The two transceivers "A" and "B", The IR signal from "A" gets reflected by the nearby objects before reaching "B" hence, reporting a false direction.
   - if "A" is very close to an object with a reflective surface. In this case the IR radiations from "A" gets reflected back to the receiver sensor thereby reporting a false direction.

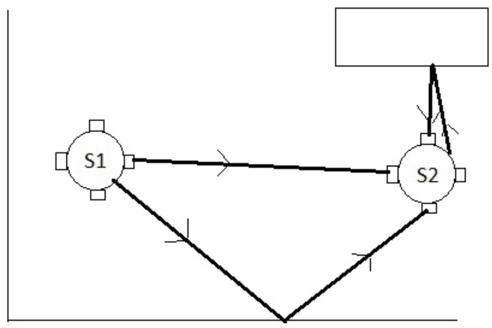

Figure 5.4: Reflection pattern

3. The transceivers have a minimum range of 1 feet, below which the sensors have a very high error factor.

4. The maximum range of the transceiver is 26 feet.

5. The maximum fluctuation in readings was noted when all six boxes were placed. There were many false direction report in this case.

6. An offset in the sensor values were noted when there was a change in the ambient lights. The graph given below helps us understand the effect of light on the sensor output. The iteration number 6 to 11 show an offset in the sensor output values.In the figure A,B,C,D represent sensor outputs from North, East, South and West respectively.

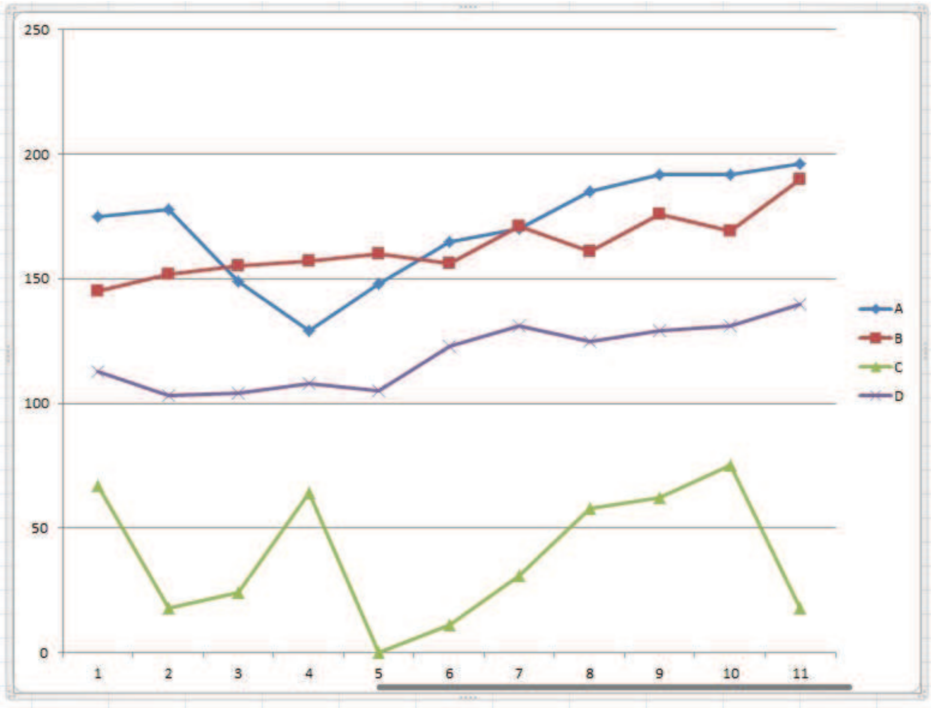

Figure 5.5: Effect of light on the sensor output

Chapter 6

RENDEZVOUS EXPERIMENTS IN AN ENVIRONMENT WITHOUT OBSTACLES

This chapter discusses the tactics involved in reaching the rendezvous point by the two robots in and environment without obstacles. The experimental results and data from the previous chapter have been carefully analysed to design an algorithm for rendezvous of two heterogeneous robots with limited sensing capabilities.

The algorithm for rendezvous without obstacles has been discussed in chapter 3. The algorithm was converted to C-program and implemented on the robots using a micro-controller unit as an interface. The algorithm was tested in a closed room where, the biped was fed with the map to the rendezvous point and the mobile robot had to completely depend on the IR signals coming from the biped. This experiment makes use of the IR transceiver pair as a lone sensing element, i.e there are no other sensors used during the experiment.

The following sections describe different cases of the working of the algorithm when implemented on the robots in different environments. Figure 6.1 shows the figures and labels used in the experimental figures used for explaining the working of algorithm.

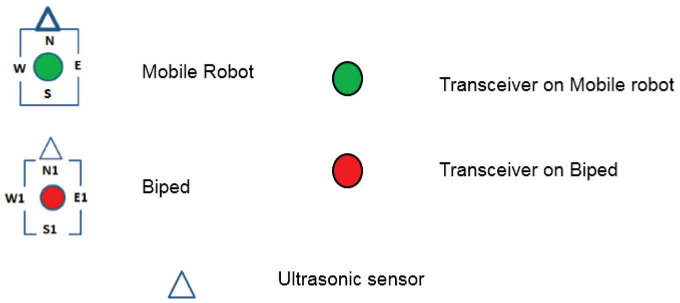

Figure 6.1: Legends

6.1 Case 1

Case 1 shows how the mobile robot follows the biped from position "A" to the rendezvous point(position "D"). Initially, at position "A" the robots start

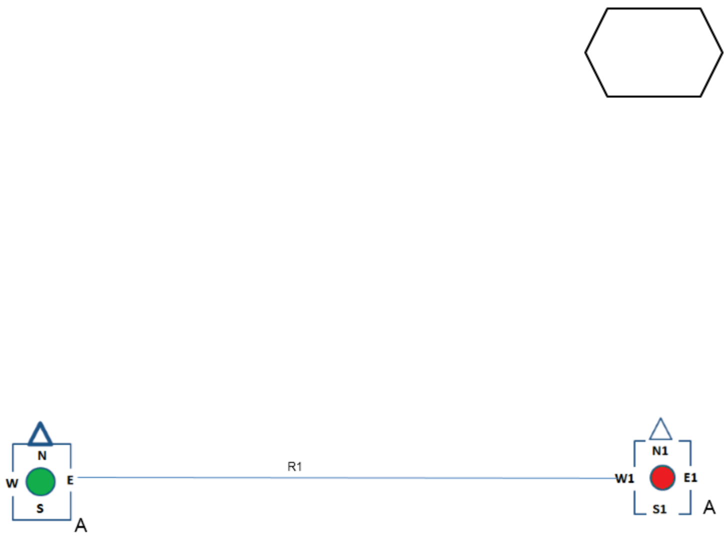

Figure 6.2: case 1 without obstacles

with entering into adaptation mode, where the IR transceiver pair get themselves adjusted or adapted to the lighting and reflection conditions. During adaptation mode the MCU(micro-controller unit) calculates the threshold value for the action mode. When the bots enter into action mode the decision is taken based on the maximum number of signal registered among the directional pins on the IR transceiver unit of the mobile robot. It can be seen in figure 6.2 that the maximum number of signals (R1) are received by pin "E". The action mode checks the value of the signal received at pin "E" to the threshold value and if the value is less than the threshold value the mobile robot turns right. In the case where the value is more than the threshold value the robot re-engages the action mode till a value less than threshold is obtained.

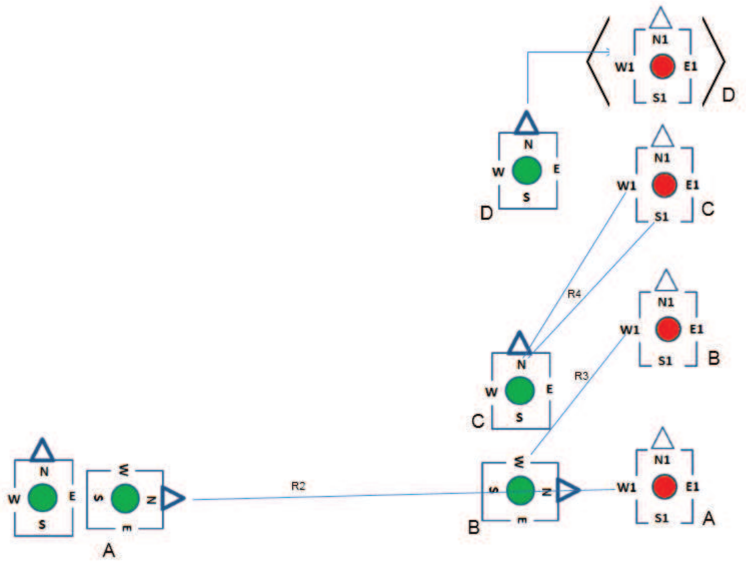

Figure 6.3: case 1 without obstacles position 2

Figure 6.3 shows the remaining steps in reaching point "D" the maximum signal by the mobile robot has been shown by lines R2 and R3 at positions B and C. Once the mobile robot reaches point D it no longer moves in any direction since the algorithm is designed such that threshold value is never compromised when the robots are next to each other. Hence the robots reach rendezvous point with the use of just one sensor.

6.2 Case 2

This case shows how the mobile robot reacts to when the amount of signals received at 2 pins are equal.
As shown in Figure 6.4 the signal from the biped is received at almost 45 degree angle between "N" and "E" pins. In such cases even a slight change in the number of times the signal is received on a particular pin results in taking the decision. This is a special case as the mobile robot never moves

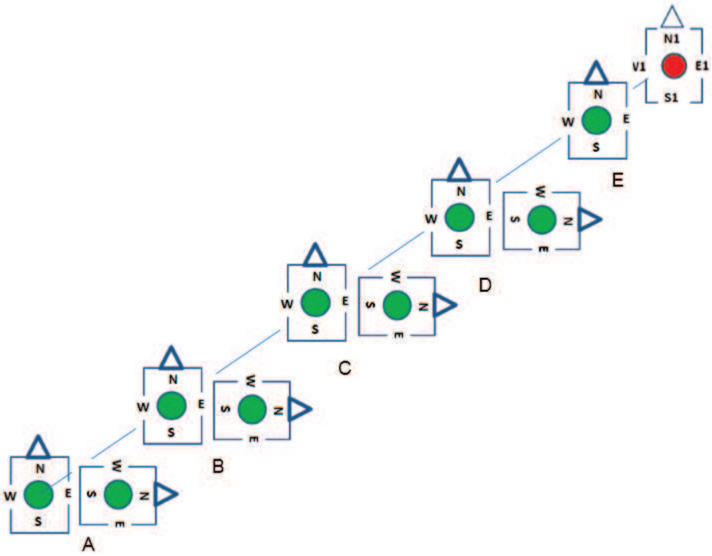

Figure 6.4: case 2 without obstacles position 2

in a straight path. It reaches the rendezvous point by continuously taking left and right turns till it reaches the rendezvous point.

Chapter 7

RENDEZVOUS EXPERIMENTS WITH OBSTACLES IN THE ENVIRONMENT

This chapter briefly explains the working of the rendezvous algorithm with obstacles on a biped and a mobile robot.
Mobile robot is represented as robot 1 with its transceivers are N,E,W and S with green coloured circle on it. The biped is represented as robot 2 with its transceivers as N1,E1,W1 and S1 with red coloured circle on it. The triangle placed in front of the robot is the ultrasonic sensor mounted on the servo motor which can rotate freely over 180^o

7.1 Case 1

7.1.1 Position 1

In this position,the two robots i.e., robot 1 and robot 2 shown in the figure **??** initially adapt themselves to the surroundings during adaptation mode. The adaptation mode helps the on board sensors to adapt to the lightings, surrounding reflection patterns and hence generates a threshold value for the action mode. Figure 7.1 shows the IR signal distribution. As seen in the figure, the E pin receives maximum number of signals L1 and L2. The ultrasonic sensor hence turns towards direction E and checks for any possible obstacles. Since there are no obstacles, the robot turns towards east

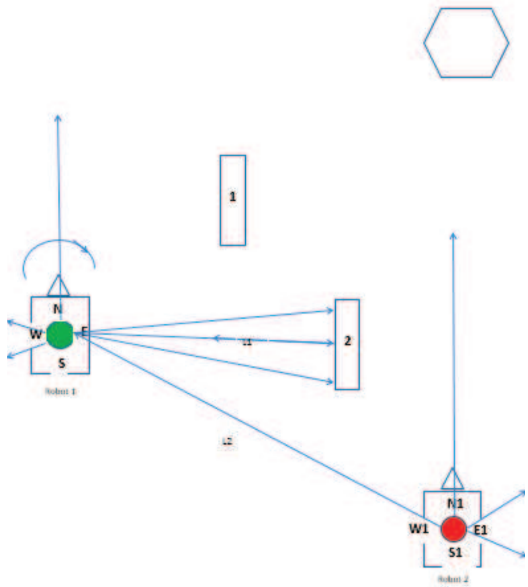

Figure 7.1: Initial position of robots

7.1.2 Position 2

Figure 7.2 indicates the next iteration of action mode. As seen from the figure 7.2, the maximum number of signals are received at N the number of times the signals collected is compared with the threshold value obtained in adaptation mode, the presence of obstacle is checked by the ultrasonic sensor, by changing the direction of the sensor to N. Since there are no obstacles present, the robot moves in north direction.

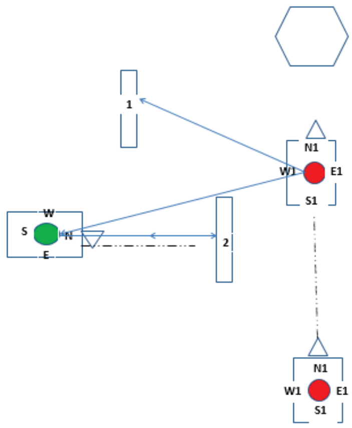

Figure 7.2: 2nd Position of Case 1

7.1.3 Position 3

Figure 7.3 explains what happens when there are signals which are received as a result of reflection from a surface. As seen in the figure, the number of signals received at W and N are almost equal. The ultrasonic sensor recognizes the obstacle in the direction of N hence informing the robot that the signal received from direction N is false, The sensor hence checks for the next best signal, which in this case is coming from W thereby deciding the next action to be turning in the west direction.

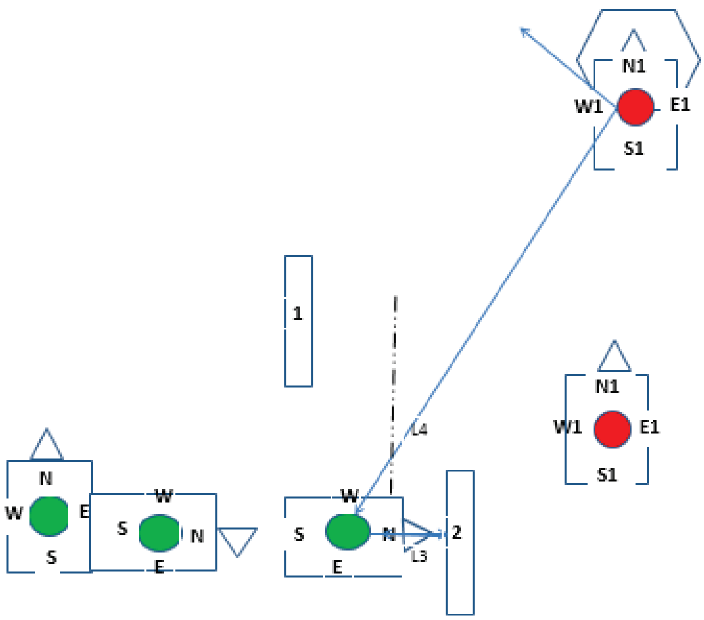

Figure 7.3: 3rd Position of Case 1

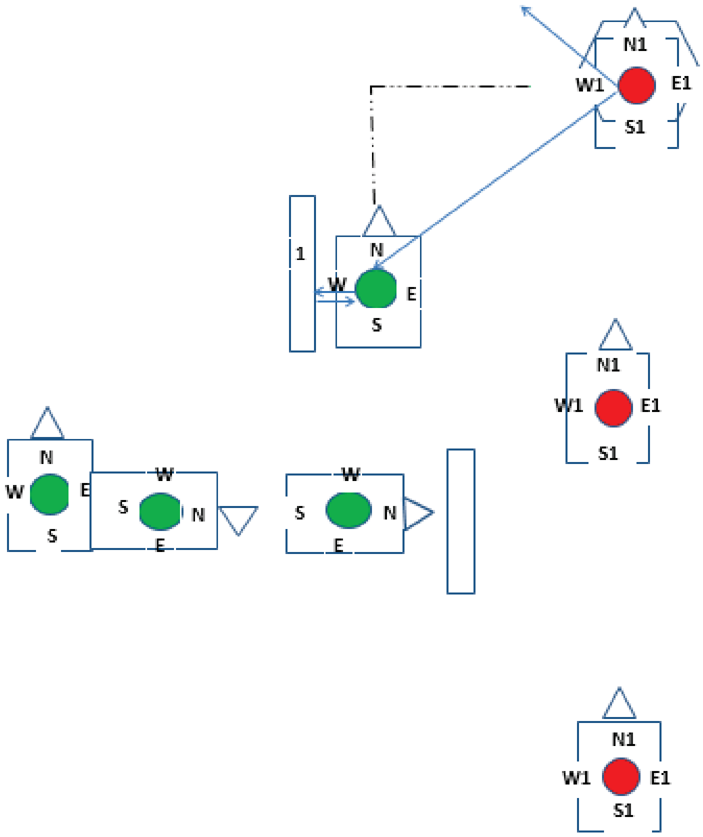

Figure 7.4: 4th Position of Case 1

Position 4

Figure 7.4 explains the remaining steps in reaching the rendezvous point. The steps involved are similar to the ones explained in figure **??** and figure 7.2 .

7.2 Case 2

In this case, initially robots 1,2 are in the corners of the room as shown in fig 7.5. Transceiver of robot 1 receives maximum number of signals at 'E'. So, the ultrasonic sensor turns to the right side of the robot 1. As there is no obstacle in that path, robot 1 turns right and now after getting clearance from ultrasonic sensor. After the robot moves few steps forward, the 'W' of the transceiver receives maximum signal. The ultrasonic sensor checks the

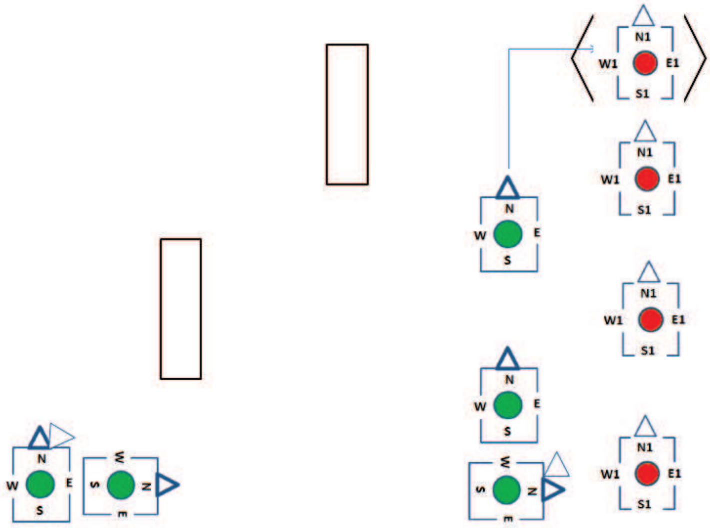

Figure 7.5: Case 2

path in that direction and as there is no obstacle, robot 1 takes a left turn. Now, 'N' of the robot 1 receives the maximum signal. It sends signal to the ultrasonic sensor to check that path and as the path is clear, it proceeds further till it reaches near the target. As the robot 1 reaches very nearer to the robot 2 which has already reached the target space and the distance between the two robots is less than the limit. So, robot 1 stops.

7.3 Case 3

In this case, the robots are initially kept close to each other as shown in figure 7.6. So, if the robots are very close to each other, mobile robot will be in halt mode. As the biped is designed to move towards the target, it moves forward. The mobile robot waits till the biped moves to a certain distance.
At this position, biped moved to certain distance and now, the mobile robot tracks the biped and follows it. The 'N' of the micro controller receives the maximum signal, the mobile robot moves forward after checking for obstacles using ultrasonic sensor. After its moves for few steps, it detects obstacle in front and now, it checks for path clearance towards its right side as 'E' receives the greater signal next to 'N'. As the path is clear, it turns right as shown in figure 7.7 and figure 7.8.
The figure 7.8 and figure 7.9 are similar to case 1 of this chapter.

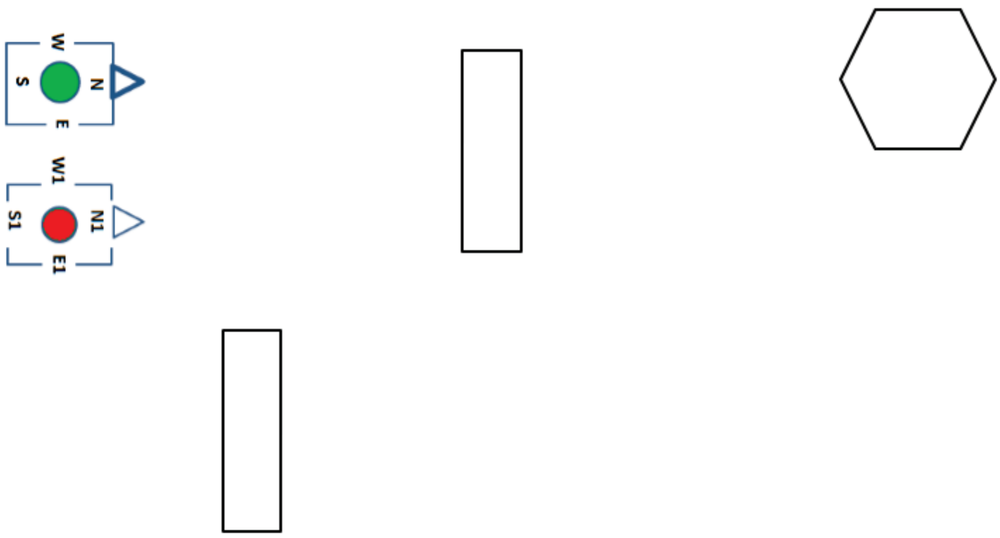

Figure 7.6: 1st Position of Case 5

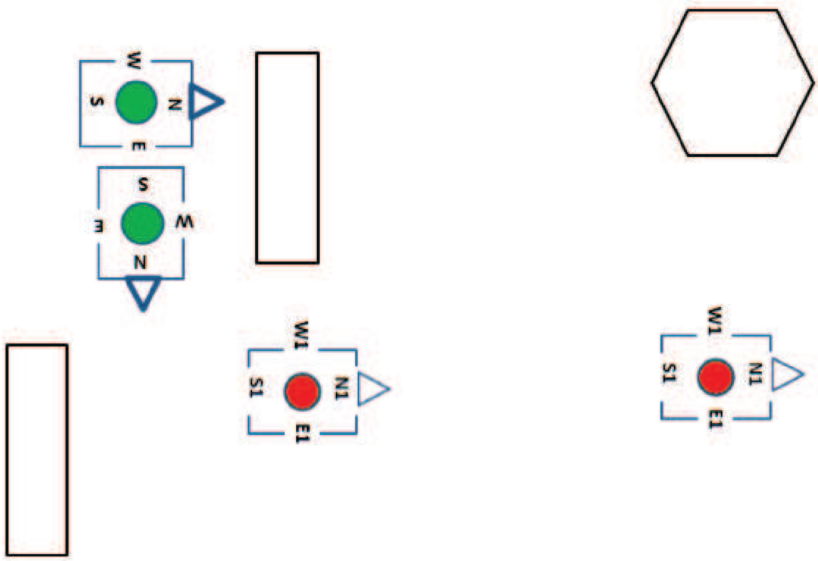

Figure 7.7: 2nd Position of Case 5

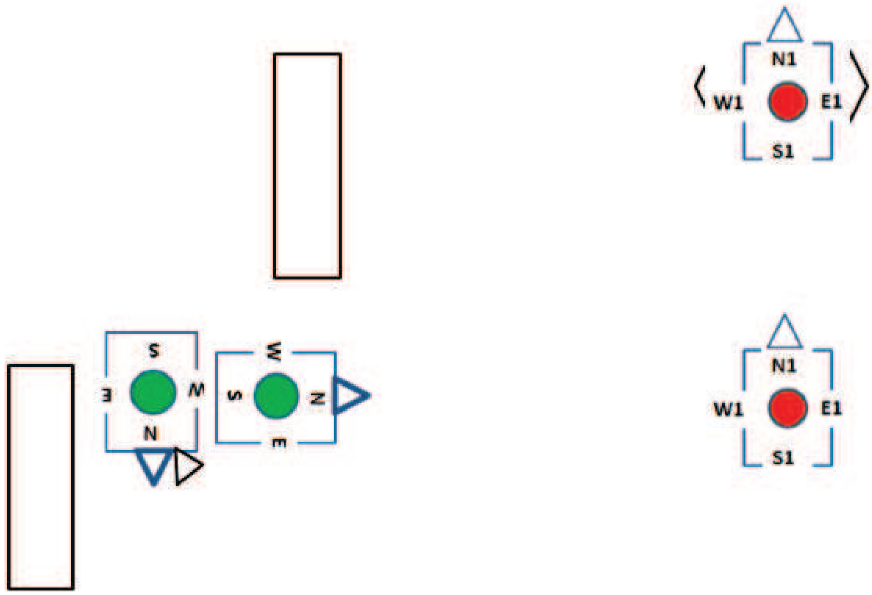

Figure 7.8: 3rd Position of Case 5

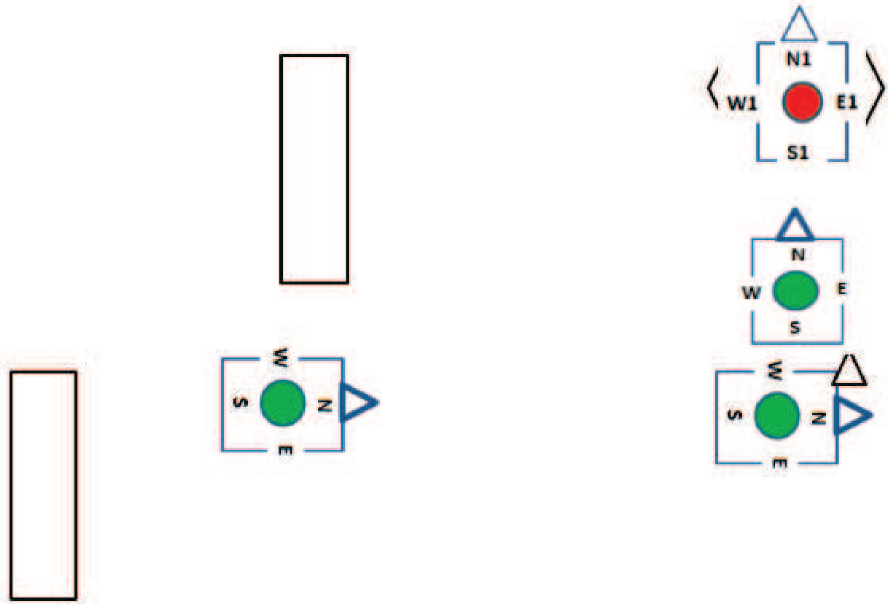

Figure 7.9: 4th Position of Case 5

Chapter 8

CONCLUSION

Rendezvous point has been reached by two robots with minimum sensor support one transceiver is used on each robot which is used for knowing the direction of robot. We have tried to make maximum use of the IR transceiver. We have created a generalized algorithm, i.e This algorithm can be used for any movable robot which can move straight and take left and right turns. We were able to achieve the task using the least number of sensors.

If the two robots comes near by less than distance of 2 feet between each other. The variation of output of the sensor is random and is found to never go less than the pre-obtained threshold value. So, this behaviour can be treated by the robot as the target as next to it. As per the designed algorithm, when the robot comes to this state, it stops as it indicates that the target area has been reached. Thus, avoiding the need of another sensor for detecting the final target place.

8.1 Future Work

1. Image processing can be implemented to track the robot and study the surroundings.
2. Artificial intelligence can be implemented to make the robot itself decide the rendezvous points to meet.
3. Biped that has been used can be replaced with a fast moving biped so that its speed is comparable to the speed of the mobile robot.
4. This project can be extended with more than two robots with slight change in the operation of the sensors.

5. Communication through RF transmitter and receiver can be employed so that robots can exchange information and commands to each other.

Bibliography

[AOSY99] H. Ando, Y. Oasa, I. Suzuki, and M. Yamashita. Distributed memoryless point convergence algorithm for mobile robots with limited visibility. *Robotics and Automation, IEEE Transactions on, 15(5):818828*, October 1999.

[dHCV10] Julian de Hoog, Stephen Cameron, and Arnoud Visser. Selection of rendezvous points for multi-robot exploration in dynamic environments. *IEEE TRANSACTIONS ON AUTOMATIC CONTROL*, 2010.

[GF02] G.Bemes and F.Blames. Using infrared sensors for distance measurement in mobile robots. *Robotics and Autonomous Systems*, page 455461, 2002.

[SG95] Alpern S and Shmuel G. Rendezvous search on the line with distinguishable players. *SIAM Journal on Control and Optimization*, 1995.

[YLL12] Jingjin Yu, Steven M. LaValle, and Daniel Liberzon. Rendezvous without coordinates. *IEEE Transactions on Automatic Control*, 57, 2012.

Appendix A

Codes on ATMEGA32 micro controller

A.1 Code used in ATMEGA32 micro controller for Motor Control

```
//WinAVR code for the designed algorithm implemented on mobile robot

#include <avr/io.h>
#include <util/delay.h>
#include <math.h>
#define US_PORT PORTA
#define  US_PIN   PINA
#define US_DDR    DDRA
#define US_POS PA0
#define US_POS1 PA1
#define US_POS2 PA2
#define US_POS3 PA3
#define US_POS5 PA4



void forward() //north --->  both forward
{

 PORTB = 0b00100010;
```

```
 _delay_ms(150);
 stop();

}
void backward() //south ---> both backward
{
 PORTB = 0b00010100;
}

void right() //right turn
{

 PORTB = 0b00100000;
 _delay_ms(135);
 stop();
}
void left() //left turn
{

 PORTB = 0b00000010;
  _delay_ms(150);
  stop();

}
void stop() //stop
{
 PORTB = 0b00000000;
 _delay_ms(50);
}

int main(void)
{


  int klp=0,i,a=0,b=0,c=0,k=0,m4,flag,avgdis=0,min1=0,min2=0,avg1=0,avg2=0,set;
DDRB |=0xFF;
 DDRD |=0xFF;
```

```
TCCR1A |= 1<<WGM11 | 1<<COM1A1 | 1<<COM1A0  | 1<<COM1B1 | 1<<COM1B0;
TCCR1B |= 1<<WGM12 | 1<<WGM13  | 1<<CS10;
ICR1 = 99; // for 10 KHz

TCCR0 |= 1<<WGM00 | 1<<WGM01 | 1<<CS01 |1<<CS00 | 0<<COM00 | 1<<COM01;
TCCR2 |= 1<<WGM20 | 1<<WGM21 | 1<<CS20  |1<<CS21  | 0<<COM20 | 1<<COM21;


OCR2 = 55;//Servo motor control
OCR1A= 55;//motor1------->Right motor
OCR1B= 60;//motor2------->Left Motor


  PORTB =0b00000000;//*
  PORTD =0b00000001;//*


  for (set=1;set<6;set++)
{

a=0;
 b=0;
 c=0;
 k=0;

 for(i=0;i<199;i++)//sampling the sensor outputs every 20ms
  {

 _delay_ms(20);
   if(US_PIN & (1<<US_POS))
   {
a=a+1;
   }
   if(US_PIN & (1<<US_POS1))
   {
    b=b+1;
   }
   if(US_PIN & (1<<US_POS2))
   {
    c=c+1;
   }
```

```
   if(US_PIN & (1<<US_POS3))
   {
    k=k+1;
   }
  }

  if(b<a)
  {
  min1=b;
  min2=a;

  }
  else
  {
  min1=a;
  min2=b;

  }

  if(c<min1)
  {
  min1=c;

  }
  else
  {
  min1=min1;
  min2=min2;
  }

  avg1=avg1+min1;
  avg2=avg1/set;

  }
m4=avg2*0.6;//adding tolerance to average value
avg2=avg2+m4;
avg2=avg2/2;

////////////////End of Adaptation mode/////

///////////////////Action mode Initialised//
```

```
while(1)
{
a=0;
 b=0;
 c=0;
 k=0;

 for(i=0;i<99;i++)
  {

 _delay_ms(20);
   if(US_PIN & (1<<US_POS))
   {
a=a+1;
   }
   if(US_PIN & (1<<US_POS1))
   {
    b=b+1;
   }
   if(US_PIN & (1<<US_POS2))
   {
    c=c+1;
   }
   if(US_PIN & (1<<US_POS3))
   {
    k=k+1;
   }
  }


  if(b<a)
  {
  min1=b;
  min2=a;
  flag=2;

  }
  else
  {
  min1=a;
```

```
  min2=b;
  flag=1;

  }
   if (c<min2)
  {
  min1=min1;
  min2=c;

  }

  if(c<min1)
  {
  min2 = min1;
  min1=c;
  flag=3;
  }

  ////decision making loop///

if (flag=1)
{

if(a<avg2)
{
 OCR2=80;//command to control servo motor
 _delay_ms(200);

if(US_PIN & (1<<US_POS5))
 {
  right();
  }
  else if (min2=b)
 {
  forward();
 }

 else if (min2=c)
 {
  left();
 }
```

```
}
}

if (flag=2)
{

if(b<avg2)
{
 OCR2=55;
 _delay_ms(200);

if(US_PIN & (1<<US_POS5))
 {
  forward();
  }
  else if (min2=a)
 {
  right();
 }

 else if (min2=c)
 {
  left();
 }
}
}

if (flag=3)
{

if(c<avg2)
{
 OCR2=19;
 _delay_ms(200);

if(US_PIN & (1<<US_POS5))
 {
  left();
  }
  else if (min2=a)
 {
```

```
  right();
 }

 else if (min2=b)
 {
  forward();
 }
}
}


}
/////////end of Action mode/////
}
```

A.2 Code used in ATMEGA32 micro controller for Motor Control for Encoder

```
////WinAVR code for encoder

#include <avr/io.h>
#include <avr/interrupt.h>
#include <math.h>
#include <util/delay.h>
#define US_PORT PORTA
#define  US_PIN   PINA
#define US_DDR    DDRA
#define US_POS PA0
#define US_POS1 PA1
#define US_POS2 PA2
#define US_POS3 PA3
#define US_POS5 PA4


int main(void)
{

 DDRB =0xFF;
 DDRD |= 0xF0;
```

```
 DDRC |= 0xFF;
 int a=0,b=0,c,i,m1=91,m2=92;


while(1)
{

if(US_PIN & (1<<US_POS3)
{
_delay_us(18);
for(i=0;i<1000;i++)////counting the number of pulses
{
_delay_us(30);
   if(US_PIN & (1<<US_POS))
   {
    a=a+1;
   }
   if(US_PIN & (1<<US_POS1))
   {
    b=b+1;
   }
 }
 c=a-b;
 /////deducing the course of movement and feeding it back to MCU controlling bip
if(c<20)
{
if(c>0)
{
PORTB &= 0x00;
PORTC |= 0x01;
 display(0110);
 a=0;
 b=0;
}
}
 if(a>800)
{
PORTB &= 0x00;
PORTC |= 0x02;
display(1000);
```

```
}
  if(b>800)
   {
   PORTB &= 0x00;
PORTC |= 0x04;
   display(0001);
   }

   a=0;
   b=0;
   c=0;

   }
   else
   {
   PORTB &= 0x00;
   }
}


 }
 ///////displaying the action on an 7 segment LED display

int display(int d)

{

int number,i,a,b,c,e,j=0,pl,st;


a=d%10;
b=(d/10)%10;
c=(d/100)%10;
e=(d/1000);


for(st=0;st<19999;st++)
{
PORTD &=0x00;
j=0;
```

```
pl=0;
for(i=0;i<4;i++)
{

j=j+1;


if(j==1)
{
number=e;
PORTD |=0x01 ;
}
else if(j==2)
{
number=c;
PORTD |= 0x02;
}
else if(j==3)
{
number=b;
PORTD |= 0x04;
}
else if(j==4)
{
number=a;
PORTD |= 0x08;
}

switch(number)
{
case 0: PORTB |=0x3F;break;
case 1: PORTB |=0x06;break;
case 2: PORTB |=0x5B;break;
case 3: PORTB |=0x4F;break;
case 4: PORTB |=0x66;break;
case 5: PORTB |=0x6D;break;
case 6: PORTB |=0x7D;break;
case 7: PORTB |=0x07;break;
case 8: PORTB |=0x7F;break;
case 9: PORTB |=0x6F;break;
case 91: PORTB |=0x5F;break;
```

```
case 92: PORTB |=0x7C;break;
case 93: PORTB |=0x58;break;
case 94: PORTB |=0x75;break;
}


PORTB &=0x00;
PORTD &=0x00;
}

}

}
/////end of encoder code
```

A.3 Code used in ATMEGA32 micro controller for Biped

```
/////////WinAVR code for Biped



#include <avr/io.h>
#include <util/delay.h>
#define US_PORT PORTA
#define  AM_PIN   PINA
#define AM_DDR    DDRA
#define AM_POS PA1
//#define AM_POS1 PA3

//OCR1A--19th pin--brown colour--right foot--neutral at 'right'-inverting mode--
//OCR1B--18th pin--black colour--left foot---neutral at 'left'--inverting mode--
//OCR0---4th pin---brown colour--right top---neutral at 'rt'----non inverting mo
//OCR2---21st pin--black colour--left top----neutral at 'lt'----non inverting mo


void rightbendout(int rf, int lf)
{
```

```
int i,j;
for (i=0; i<=100; i++)
{
_delay_ms(10);
if (i<80) j=2*i; else j=160;
OCR1A = ICR1 - (rf-(8*i));
OCR1B = ICR1 - (lf-(10*j));
}
}

void rightbendin(int rf, int lf)
{
int i,j;
for (i=99; i>=1; i--)
{
_delay_ms(10);
if (i<80) j=2*i; else j=160;
OCR1A = ICR1 - (rf-(8*i));
OCR1B = ICR1 - (lf-(10*j));
}
}

void leftbendout(int rf, int lf)
{
int i,j;
for (i=0; i<=110; i++)
{
_delay_ms(10);
if (i<80) j=2*i; else j=160;
OCR1A = ICR1 - (rf+(10*j));
OCR1B = ICR1 - (lf+(10*i));
}
}

void leftbendin(int rf, int lf)
{
int i,j;
for (i=109; i>=1; i--)
{
_delay_ms(10);
if (i<80) j=2*i; else j=160;
```

```
OCR1A = ICR1 - (rf+(10*j));
OCR1B = ICR1 - (lf+(10*i));
}
}
//right leg forward
void rightstep(int lt, int rt)
{
int i;
for (i=0; i<=20; i++)
{
_delay_ms(10);
OCR0 = (rt+i*0.025);
OCR2 = (lt+i*0.1);
}
}
//left leg forward
void leftstep(int lt, int rt)
{
int i;
for (i=0; i<=25; i++)
{
_delay_ms(10);
OCR0 = (rt-i*0.1);
OCR2 = (lt-i*0.1);
}
}

void forwardwalk(int rt, int lt, int rf, int lf)
{
rightbendout(rf,lf);
leftstep(lt,rt);
rightbendin(rf,lf);

leftbendout(rf,lf);
rightstep(lt,rt);
leftbendin(rf,lf);
}
void leftturn(int rt, int lt, int rf, int lf)
{
leftbendout(rf,lf);
rightstep(lt,rt);
```

```
leftbendin(rf,lf);
OCR0=rt;
OCR2=lt;
_delay_ms(20);
}

int main(void)
{
int lf,rf,lt,rt,jklo;
DDRB |= 0xFF;
lf=5990;  rf=5920; lt=95; rt=90;
DDRD |= 0xFF;
//generation of PWMs
TCCR1A |= 1<<WGM11 | 1<<COM1A1 | 1<<COM1A0  | 1<<COM1B1 | 1<<COM1B0;
TCCR1B |= 1<<WGM12 | 1<<WGM13  | 1<<CS10;
ICR1 = 15999;
TCCR0 |= 1<<WGM00 | 1<<WGM01 | 1<<CS01 | 1<<CS00 | 0<<COM00 | 1<<COM01;
TCCR2 |= 1<<WGM20 | 1<<WGM21 | 1<<CS22  | 0<<COM20 | 1<<COM21;

OCR1A=ICR1 - rf;
OCR1B=ICR1 - lf;
OCR0=rt;
OCR2=lt;
///////////////////////////////////////////////////////Main///////////////////
_delay_ms(200);

DDRA = 0xFD;
DDRC=0xFF;


////////////enter the location of the rendezvous point here/////////////////////

for(jklo=0;jklo<40;jklo++)
{

forwardwalk(rt, lt, rf, lf);
}

}
```