DEVELOPMENT OF GENERAL SEARCH BASED PATH FOLLOWER IN REAL TIME ENVIRONMENT

A Thesis submitted in partial fulfillment of the Requirements for the degree of

Master of Technology

In

Industrial Design

By

Mounika Vummaneni

Roll No. : 213ID1357

Under the Guidance of

Dr. B.B.V.L. Deepak

Department of Industrial Design

National Institute of Technology Rourkela

Rourkela, Odisha, 769 008, India

May 2015

DEVELOPMENT OF GENERAL SEARCH BASED PATH FOLLOWER IN REAL TIME ENVIRONMENT

A Thesis submitted in partial fulfillment of the Requirements for the degree of

Master of Technology

In

Industrial Design

By

Mounika Vummaneni

Roll No. : 213ID1357

Under the Guidance of

Dr. B.B.V.L. Deepak

Department of Industrial Design

National Institute of Technology Rourkela

Rourkela, Odisha, 769 008, India

May 2015

Dedicated to…

My Dear Friends

My parents and my brother

Declaration

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgement has been made in the text.

Mounika Vummaneni

Date:

N.I.T. Rourkela

Department of Industrial Design

NATIONAL INSTITUTE OF TECHNOLOGY, ROURKELA

ORISSA, INDIA – 769008

CERTIFICATE

This is to certify that the thesis entitled "***Development Of General Search Based Path Follower In Real Time Environment***", being submitted by **MOUNIKA VUMMANENI**, Roll No. **213ID1357,** to the National Institute of Technology, Rourkela for the award of the degree of ***Master of Technology*** in Industrial Design, is a bona fide record of research work carried out by her under my supervision and guidance.

The candidate has fulfilled all the prescribed requirements.

The thesis, which is based on candidate's own work, has not been submitted elsewhere for the award of a degree.

In my opinion, the thesis is of the standard required for the award of degree of Master of Technology in Industrial Design.

To the best of my knowledge, she bears a good moral character and decent behavior.

Supervisor

Dr. B.B.V.L. DEEPAK,

Assistant Professor, Department of Industrial Design

NATIONAL INSTITUTE OF TECHNOLOGY

Rourkela-769 008 (INDIA)

Acknowledgement

I am mainly indebted to my guide Dr. B B V L Deepak for his continuous support, encouragement, and help throughout the course work. I express my deep regard to him for the successful completion of this work. The blessing, help and guidance given by him from time to time made it possible for me to complete the work in stipulated time. He has guided me at every stage of the work and explained patiently as and when a problem arose.

I am thankful to Prof. B.B.Biswal, Head of the Department of Industrial Design, National Institute of Technology, Rourkela, for providing me facilities to carry out my thesis work in the Department of Industrial Design.

I express my sincere gratitude to all the faculty members of Department of Industrial Design, NIT Rourkela for their affection and support.

I am thankful to all the staff members of Department of Industrial Design, National Institute of Technology, Rourkela for their support.

I render my respect to all my family members and my well-wishers for giving me mental support and inspiration for carrying out my research work.

I thank all my friends who have extended their cooperation and suggestions at various steps in completion of this thesis.

Mounika Vummaneni

Abstract

Mobile robots are being used for various industrial, medical, research and other applications to perform the various tasks accurately and efficiently. Path planning of such mobile robots plays a prominent role in performing these tasks. This paper deals with the path planning of mobile robots in a predefined structured environment. In this case the environment chosen is the roadmap of NIT Rourkela obtained from Google maps as reference. An Unmanned Ground Vehicle (UGV) is developed and programmed so as to move autonomously from an indicated source location to the defined destination in the given map following the most optimal path among the available paths. The source and destination points are the different departments, academic blocks in NIT Rourkela campus map. In this case we use a two wheeled mobile robot consisting of IR sensors is used to verify the validation of the proposed algorithm. The vehicle receives the details of the surrounding environment through sensors and processes this data to aid in the safe and accurate navigation. A linear search based algorithm is implemented on the autonomous robot to generate shortest paths in the NIT Rourkela campus map generated. The algorithm is similar to that of the right wall follower algorithm, Dijkstra algorithm etc. used in maze solving robots but in this case the paths treaded are not stored in the memory and the vehicle does not check the available paths to choose the shortest one, but chooses them with the aid of the information provided by the sensors. The coordinates of source and goal positions plays a prominent role in deciding the particular path at the branching node. This method saves the time and cost of following all the available paths to check if it is the shortest one. The results are verified with the simulations performed using MATLAB. Moreover experiments were performed on the developed model in the scaled version of NIT Rourkela campus map printed on a banner to compare and verify the results.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

Origin of the Work

Problem Statement

Objectives

Thesis Overview

1. INTRODUCTION

A Robot can be defined as a reprogrammable, multifunctional manipulator designed for the movement of material, parts, tools, or specialized devices through various programmed motions for the performance of a variety of tasks. It is an electro-mechanical device which finds its uses in all aspects of our life. It performs various tasks that may be human controlled or automated. With the advancement in technology, especially in the field of robotics, mobile robots are used for transportation, processing, assembling etc. in industries, for performing complicated surgeries and working in bio hazardous environments. They are used for search and rescue operations during natural disasters, accidents and used in other areas where the environment is impractical or fatal for the human beings to work.

1.1. Origin of Work

With the advancement in the field of robotics, robots are being used in almost all the fields of prominence. Robots are being used to perform surgeries, for operations such as machining, painting etc. in industries, for gaming applications and entertainment, for research in different fields, and in areas where the environment and working conditions are hazardous and dangerous or impossible for humans to operate. For performing these applications accurately and efficiently proper path planning and navigation techniques must be used to suit the task it performs. In recent decades a huge number path planning algorithms are developed to navigate a robot in static or dynamic environment when it is having a clutter of obstacles along the path. There have been different local and global path planning methods for the navigation of robot in such environments. Most of these methods have their own advantages and limitations. But when it comes to path planning in a particular environment where the robot has to move only along the given path and has to choose among the available paths there are very few methods available to perform the task in such situations. Hence an algorithm is developed for fulfilling this purpose.

1.2.Problem Statement

The robot must be able to move in the given environment from the source position to destination and along a given path for some applications. For example when robots are used for transportation in industries or rescue operations it has to move along a predefined paved path. When following the path sometimes it is imperative to choose a path among the multiple available paths at the particular node or branching point. Most of the available path planning algorithms deals with obstacle avoidance in a static or dynamic environment. We also have different maze solving algorithms to determine the path when multiple paths are available, but in this type the robot follows along the paths depending on the type of algorithm used and this is stored in the controller memory. This data from memory is compared with at every possible routine and the shortest among the paths is chosen. But to use this algorithm in real time applications is not plausible because of the time and cost considerations. Therefore it is imperative to have a path planning method to follow when the robot has to move along a prescribed path in a given environment from source to destination position.

1.3.Objectives

The main aim of the proposed research work is to provide an approach for designing a path planning methodology for UGV in a complex environment with multiple branching paths to be chosen from, in order to reach the destination.We have to develop and program an Unmanned Ground Vehicle (UGV) so that it moves autonomously from an indicated source location to the defined destination in the given map following the most optimal path. The following are the major objectives of research:

- Develop an algorithm for choosing the shortest path from multiple branching paths.
- Develop and study the environment where the path planning is to be done. Here the environment chosen is roadmap of NIT Rourkela has been chosen for the path planning environment.
- To analyze various paths and locations using simulation results in MATLAB.
- Check the results in the real time environment with the wheeled robot.

1.4.Thesis Overview

This dissertation has been arranged in the following order

- *Chapter 2* provides with the background of the different types of navigation techniques and motion control of the robot. It provides with the literature survey on various types of path planning techniques and different types of robots and their motion control strategy.

- *Chapter 3* deals with the kinematic analysis of the robot. In this section different types of robots like differential robot etc are studied and kinematic analysis of each type of wheels present in the system is presented along with the governing mathematical models.

- *Chapter 4* deals with methodology used for the path planning in the given environment. The linear search algorithm used for the path planning is explained in this section. It also presents with the simulation results obtained by applying this algorithm in MATLAB software.

- *Chapter 5* provides with the details of the experimental setup used for checking the performance and accuracy of the proposed algorithm. Here the details and specifications of all the individual parts that are used in the robot are clearly described. Here the experimental setup is explained and the corresponding results are presented.

- *Chapter 6* provides with the experimental results and the simulation results. It gives the performance of the robot using given methodology.

- *Chapter 7* gives the conclusions obtained from the entire work done and analysis of results obtained. Future scope of the work is also given in this section.

Chapter 2

RELATED WORK

Mobile Robot Locomotion

Navigation of Mobile Robot

2. RELATED WORK

The most common problems associated with the navigation of mobile robot in a given environment include the structural stability of the robot, navigation technique used, motion control of the robot etc**.** There have been a lot of research and discussions along the decades regarding the locomotion and navigation of a mobile robot. There have various algorithms for navigation of robots depending on the type of environment, sensors used, application of the robot etc.

2.1. Mobile Robot Locomotion

This section deals with the basics and the previous work that is related to solving mobile robot navigation problem. A lot of research works have been published in numerous journals and during conferences pertaining to the area of navigation problem of a mobile robot. There are a huge number of techniques to solve this problem but any single technique can be used for all the applications and in all the cases. Hence some of the major techniques are briefly explained in this section.

2.1.1. Overview of motion planning concepts

The most important characteristic feature of a motion planner is problem specific. It means the applicability and usefulness of the motion planner being used depends on the properties of the robot that solves the given task. After studying the complete information about the task to be performed and the type robot system being used, the algorithm can be developed to solve the particular problem.

The different aspects that are to be considered while planning the motion control scheme of mobile robot are shown in Fig.2.1. below. The following are the aspects of motion planning problem for autonomous mobile robot/robots:

Robot locomotive ability (most popular mechanism for robot motion)

Perception (obtaining the information about the robots environment conditions)

Localization (determines the robot position with respect to its surrounding environment)

Cognition (decision-making and execution of the decisions to complete the assigned task)

Motion Control (executing the path in real time environment)

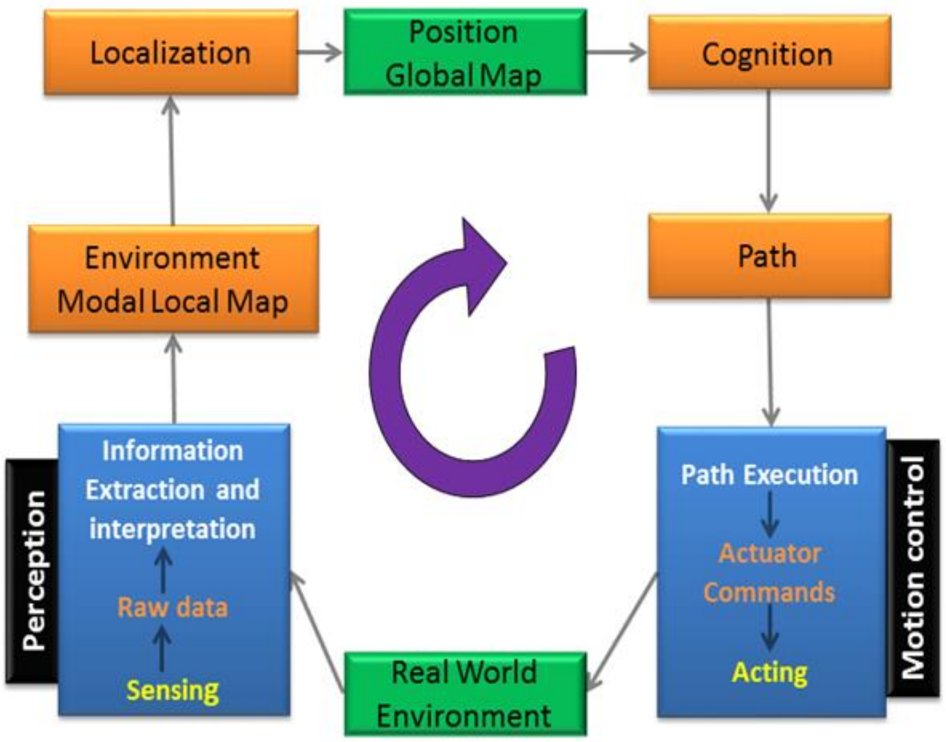

Fig.2.1: Control scheme for path execution of autonomous mobile robot.

2.1.2. Wheel locomotion

In order for the movement of a mobile robot from one given location to another location some locomotion mechanisms are required. So selection of the locomotion mechanism while designing a mobile robot depends on important aspects that define the possible number of ways to move. Most of the locomotion mechanisms described in [1-4] for a mobile robot, such as walk (human), slide (snake), run (4legged animal), jump (animals like kangaroo) etc. are inspired biologically. The selection of locomotion mechanism by studying the locomotive aspects of different biological system is more prone to be successful because they provide deep insights into the movement in various types of environments. Some of the reasons that make replication of biological systems in manmade structures difficult are listed below:

Mechanical complexity (individual fabrication of each part)

Biological energy storage systems used by animals (manmade hydraulic activating system)

Mathematical complexity etc.

Because of such limitations, though there are numerous locomotion techniques based on biological systems the simplest biological locomotion mechanism with lest complicated structure are widely being used.

Because of the mechanical complexity in legged locomotion due to the requirement of higher degrees of freedom, wheeled locomotion is preferred over legged locomotion. In order to overcome this difficulty active powered wheels can be used in the mobile robot. Wheeled type of locomotion mechanism is rather simple and is most suitable for flat ground type environments. The only limitation with this type of mechanism is that when the environment (surface) is too soft that it results in slippery motion due to rolling friction (between contact point of wheel & surface). Several algorithms and motion control techniques are proposed in [5] for controlling the motion of a mobile robot.

The relation between the robot chassis and the wheels attached to it can easily be developed by considering the mobile robot as a planar rigid body [6]. The wheels of a mobile robot can be mainly classified into five types depending upon the geometrical constraints of wheels. Then for each type of wheel, the analysis of the structure of kinematic and dynamic models for each type of wheel has been performed in [7].

2.2. Navigation of Mobile Robot

A fundamental approach for formulating and solving the path planning problem is the configuration space (c-space) approach [8]. The central idea of this approach is the representation robot as a single point. Thus, the c-space of the mobile robot path planning problem is reduced to a 2-dimensional problem. Most of the Path planning algorithms are based on configuration space or c-space representations like Voronoi diagram, generalized cones [9]. This representation can also be seen being used in quad-tree [10] and vertex graph, where the C-space is filled with data structures that represent the orientations and position of objects and robots in the workspace area that is a combination of both the free space regions and restricted regions with obstacles or mazes.

The cell decomposition approach computes the c-space of the mobile robot, decomposes the resulting space into cells and then searches for a route in the free space cell graph. Grid method [11, 12] is a popular cell decomposition approach where grids are used to generate the map of the environment.

There are various types of algorithms to find and change the data structures that are utilised to contain the maps and the boundaries of the environment/work space area. Some of the basic types of algorithms that use this kind of methodology are graph search algorithm that consists of breadth first search algorithm that scans the graph using a prioritised queue system to find the shortest collision free path (A* search algorithm [13], modified A* search algorithm [14], genetic algorithm that is an optimized version of these path planning algorithms, potential field algorithm [15] and roadmap algorithm [16]. [17] Describes paths or roadmaps of variable path lengths that are dependent upon the number of polygonal vertices obstacles. Previous research [18] presents with a different algorithm type that can complete the same task (i.e. with a varied instruction set) in less or more time, effort, space etc. Wall follower algorithms, djikstra and flood algorithms are analysed and their efficiency is determined [19]. It is hampering to deal with pure reactive navigation systems, so by using immune network theory we can convert an earlier reactive controller to a connectionist device whose connections are evolved during robot navigation [20].

A tactical path planning algorithm to follow ridges or valleys when traversing across a 3D terrain is proposed in [21]. This helps in improving the surveillance of an unmanned vehicle by providing maximum observability when traversing along the ridges of a terrain. This can also be used to provide maximum covertness when navigating the valleys. Here a 3D triangular mesh of the terrain of interest is given as the input. The algorithm uses research in the field of computer graphics and computer vision for identifying ridge-valley features of the terrain. Finally these features of terrain are referred as obstacles in artificial potential field algorithm.

A mathematical formula can be developed basing on the artificial immune system path planning methodology. It uses the basics of human immune system. An offline path planning algorithm can be used for the navigation of mobile robot in maze environments as described in [22]. The robot can be induced with a learning incentive by correlating the sensory information and predefined robotic actions. By using this slight modification to immune system algorithm the path planning effectiveness and accuracy can be improved as in [23].

Chapter 3

KINEMATIC ANALYSIS OF MOBILE ROBOT

Definition of Robot Kinematics

Introduction of Robot Kinematics

Kinematic Analysis of Wheel Mobile Robot

Wheel Kinematic Analysis

3.1. Definition of Robot Kinematics

Robot kinematics is defined as the accumulation of multi degree freedom of kinematic chains that forms the configuration of the robotic system. In a robotic system, numerous links are connected together to form the geometry. The study of such geometry is considered as the robot kinematics. The configuration of the mobile robot is studied using non-linear equations and kinematic analysis is done with the help of these equations.

3.2. Introduction of Robot Kinematics

In present day a wheeled mobile robot is widely used in almost every field including military, industrial, and agricultural where the working environment is such that human beings cannot work efficiently or properly or it is impossible to work. When considering such applications the main focus is on the aspect of path planning. So it is imperative to have a proper control mechanism for the mobile robot.

Different types of kinematic mechanisms are present. Therefore, it is important to study the different types of wheeled mobile robot mechanisms such as fixed wheel mobile robot, steered wheel mobile robot, castor wheel mobile robot and so on. By considering the restraint to robot mobility that is affected by several kinematic parameters, we can derive the kinematic model expressions. Then this equation can be implemented on the mobile robot to find the steering angle and type of robot configuration.

We require the following parameters to control the mobile robot movement:

- Robot Kinematic or dynamic model.
- Interaction model between wheel and the robot.
- Definition of the necessary motion.
- Position and Speed control.
- Control law that satisfies the requirements.

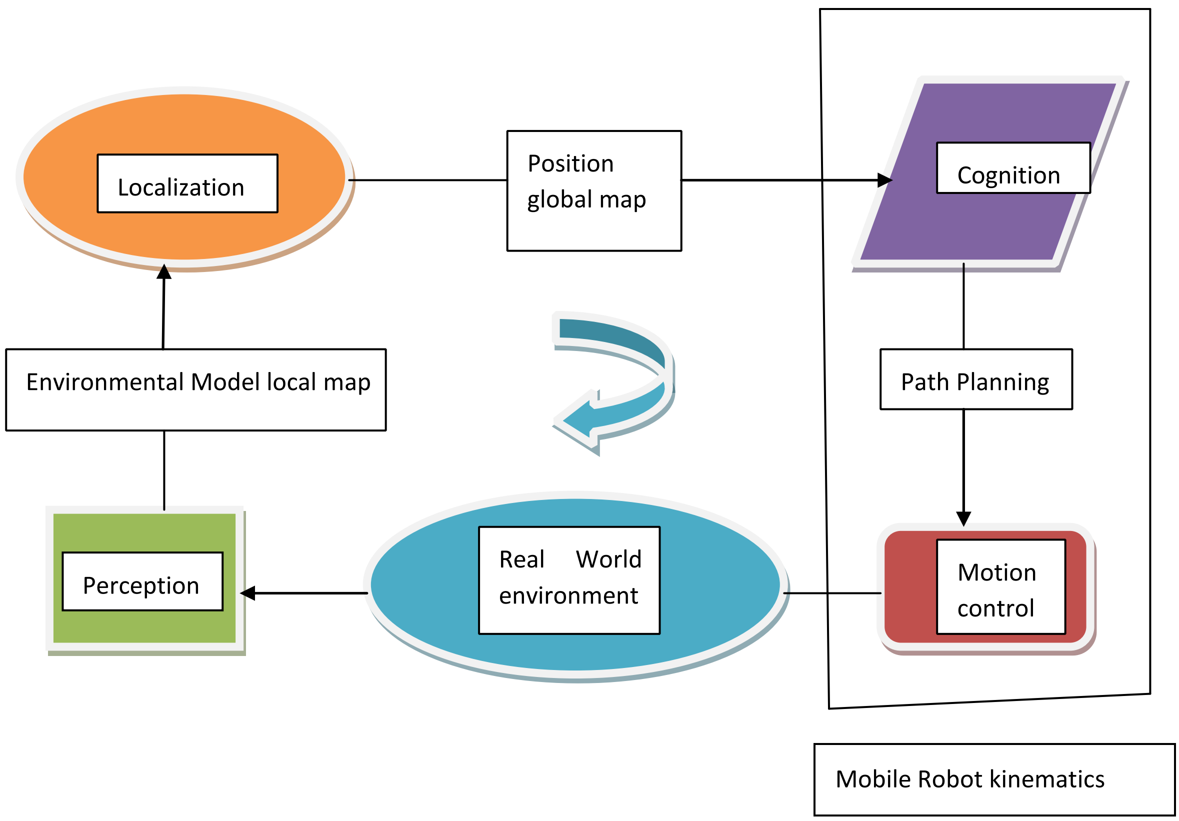

Fig. 3.1: Control Mechanism for mobile robot navigation.

3.3. Kinematic Analysis of Wheel Mobile Robot

3.3.1. Mobile robot position

Consider the kinematic model of an autonomous wheeled robot on a leveled surface as shown in the fig.3.2.

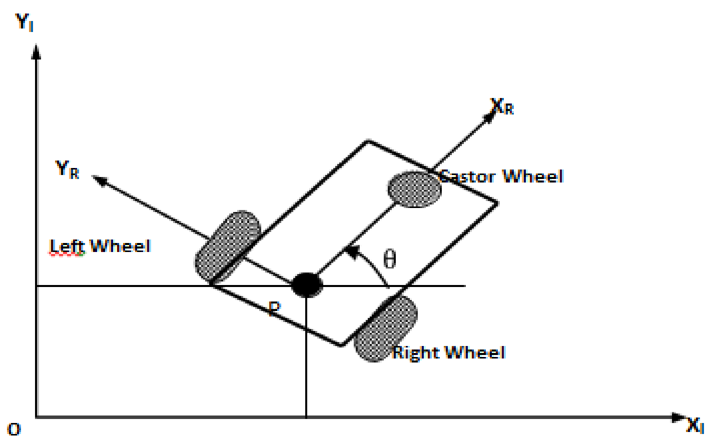

Fig. 3.2: Position of mobile robot in plane.

Where,

(OX_IY_I) - Base Frame

(OX_RY_R) - Moving Frame

θ -Steering angle

The mobile robot shown above in fig.3.2.has three wheels out of which one wheel is castor wheel that is attached to one side of the chassis and remaining two wheels are non-deformable wheels that are attached to another side. The whole setup moves in a horizontal plane. The position δ of the wheeled robot is defined in the global coordinates by x, y and θ. Point P is the orientation of mobile robot that is represented by (x, y) and θ.

$$\delta = \begin{matrix} x \\ y \\ \theta \end{matrix} \tag{3.1}$$

To find out the position of the robot it is important to find the robot movement along the world orientation structure axes and the robot movement along the robot local orientation structure axes. The orientation of (OX_IY_I) with respect to the robot frame (OX_RY_R) can be expressed by using the orthogonal rotation matrix which is given by

$$R(\theta) = \begin{matrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{matrix} \tag{3.2}$$

The above matrix used to map the movement in global reference frame to action in terms of robot frame.

$$\delta_R = R(\theta)\delta_I \tag{3.3}$$

3.4. Wheel Kinematic Analysis

For performing this analysis following assumptions are to be made:

- Robot moves only on a planar surface.
- The Guidance axis is perpendicular to the floor.
- Wheel rotation does not have any slippery problem.

- Mobile robots do not have any flexible parts as they make the system more complicated and handling becomes difficult.

- During small amount of time, the direction maintained constant and vehicle moves from one point to another point follows the circumferential arc.

3.4.1. Fixed standard wheel

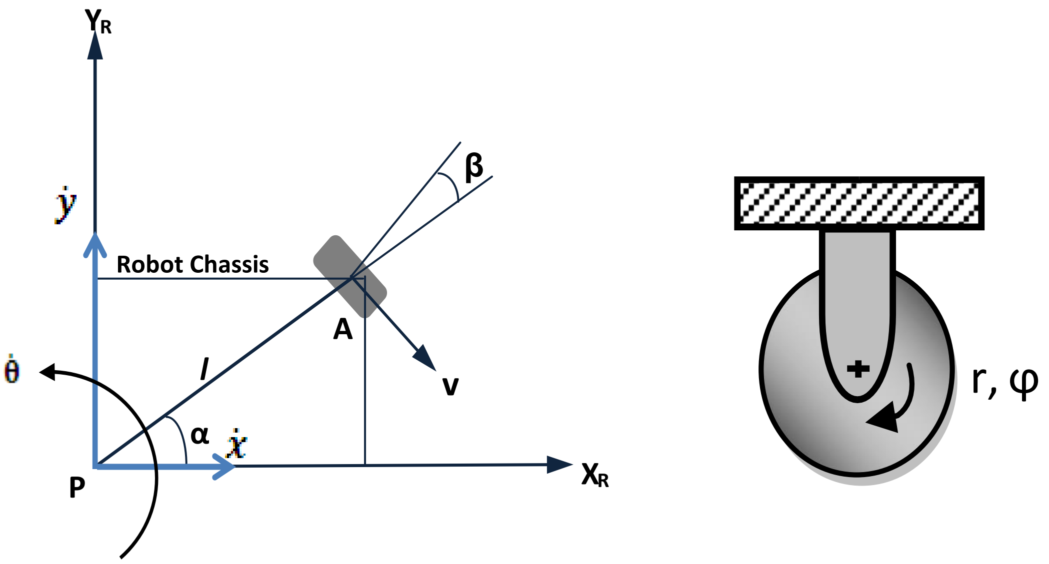

Fig 3.3: Geometric constraints of fixed standard wheel.

In this figure, the point 'A' represented by the center of the fixed wheel of mobile robot and this point is fixed with the reference frame. The position of 'A' is defined with the help of polar coordinates by distance PA=l and the angle α. The orientation of the plane of the wheel with respect to PA is representing by the constant angle β. The rotary motion angle of fixed wheels around its axle and it is denoted φ (t) and radius of wheel is 'r'. Therefore, location of fixed wheel defined using four parameters α, β, l, r and its movement by time changeable angles β (t). When components velocity of the contact point projected on the fixed wheel plane, we can consider two following constraints:

- Along the wheel plane

$$-sin\ \alpha+\beta\ \ cos\ \alpha+\beta\ \ lcos\beta\ \ R\ \ \theta\ \ \xi+r\varphi=0 \qquad (3.4)$$

- Orthogonal to the wheel plane

$$cos\ \alpha+\beta\ \ sin\ \alpha+\beta\ \ lsin\beta\ \ R\ \ \theta\ \ \xi=0 \qquad (3.5)$$

3.4.2. Steered standard wheel

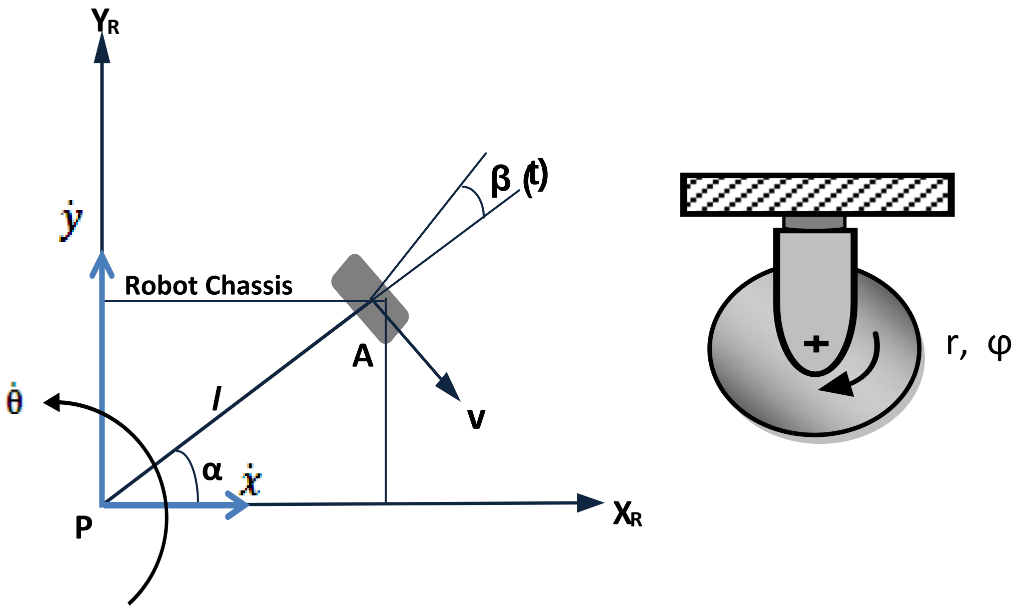

Fig.3.4: Geometric constraints of steered standard wheel.

A steered standard wheel is such that the movement of wheel with respect to the frame is a revolution around vertical axis goes through the center of the wheel as shown in Fig.3.5. The expression is same as for a fixed standard wheel, only difference is that now the angle β (t) is time varying. Therefore, position of wheel defined using three constant parameters l,α,r and its movement with respect to the frame by 2 time-varying angles β (t) and φ (t). We have the same expression form as above:

- Along the wheel plane

$$[-sin(\alpha+\beta) \quad cos(\alpha+\beta) \quad lcos\beta]R(\theta)\,(\xi)\ + r\varphi\ = 0 \tag{3.6}$$

- Orthogonal to the wheel plane

$$[cos(\alpha+\beta) \quad sin(\alpha+\beta) \quad lsin\beta]R(\theta)\,(\xi)\ = 0 \tag{3.7}$$

3.4.3. Castor wheel

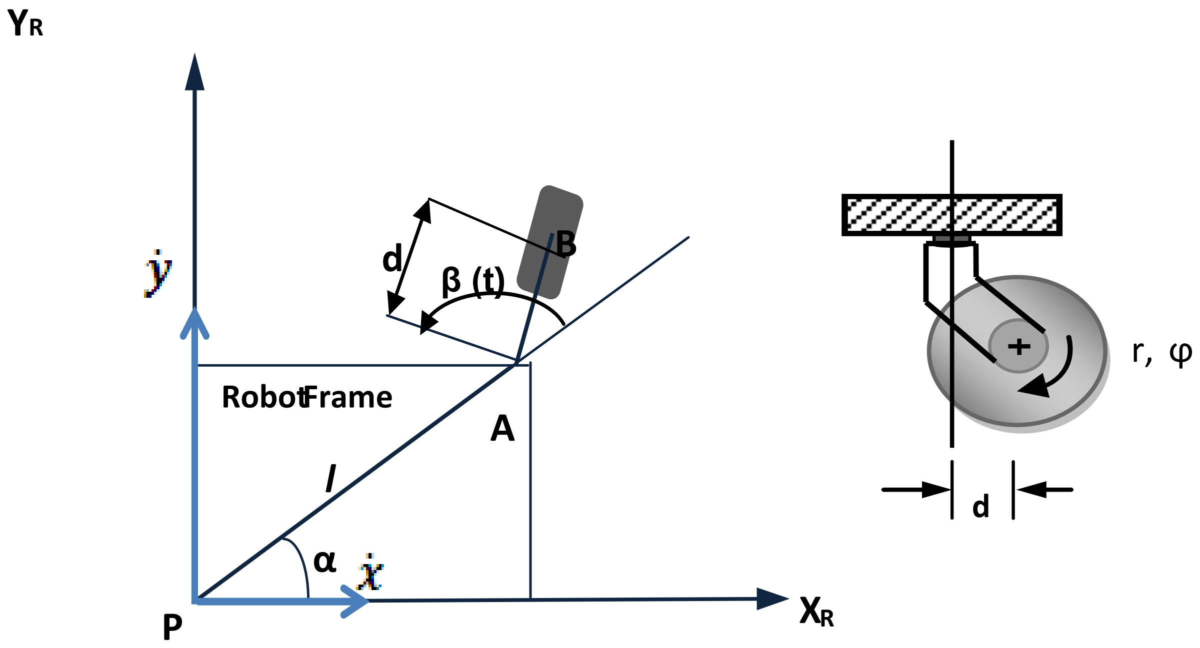

Fig.3.5: Geometric constraints of castor wheel.

In this type of wheel, the rotary motion of wheel surface is around vertical axis that does not go by through center of wheel (Fig.3.4). ‘B’ is the center of the wheel and is connected to the frame by a rigid bar AB of length ‘d’ which can be rotate around a fixed vertical axis at point ‘A’. The location of wheel defined using four parameters α, l, r, d and its movement using two changeable angles β (t) and φ (t).For this wheel constrains are in following form:

- Along the wheel plane

$$[-sin(\alpha+\beta) \quad cos(\alpha+\beta) \quad lcos\beta]R(\theta)\,(\xi)\ + r\varphi = 0 \tag{3.8}$$

- Orthogonal to the wheel plane

$$cos\ \alpha+\beta\ \ sin\ \alpha+\beta\ \ d+lsin\beta\ \ R\ \ \theta\ \ \xi+d\beta\ \ =0 \tag{3.9}$$

Chapter 4

METHODOLOGY

Algorithm

Simulation Results

4. METHODOLOGY

The most prominent part in robotic application is path planning. The methodology used for path planning depends on the type of application the robot is being used for, the kind of environment it is being used in etc.

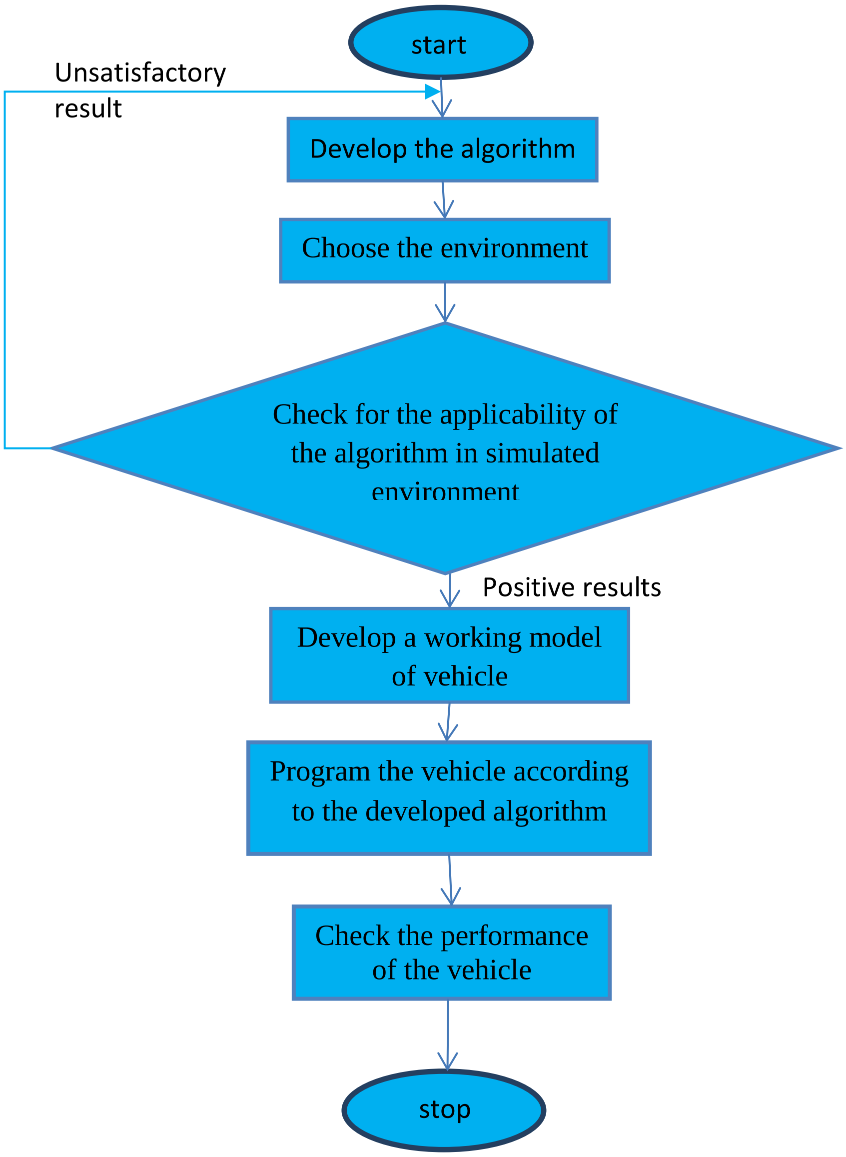

Fig. 4.1: Methodology used for path planning.

4.1. Algorithm

The path planning of an environment where the complete information of the environment is available is called global path planning, and the path planning where certain information of the environment is to be attained using sensors is called local path planning. The algorithm that is proposed in this paper is similar to the binary search algorithm used to find a number in data structures. It uses global path planner as known environment is used and the coordinate values and pixel values are available, and local path planner is used to ascertain that the path follows only the black coloured road provided. This can be implemented using Artificial Immune system algorithm.

4.1.1. Artificial Immune System (AIS) algorithm

Artificial Immune algorithm is developed basing on the principles of the Immune system of the Human beings. We know that in human beings when an antigen attacks the human body, anti-bodies are generated by the immune system present in the body to protect from the destructive action of the antigens. The produced anti-body must be counterproductive with respect to the action of the particular antigen attacking the human body.

In the Artificial Immune System based path planning algorithm the main consideration is to identify the conditions of the work space area or the environment. This information about the environment can be obtained by using the sensors connected to the robot. The main goal of Artificial Immune system based algorithm is to obtain the best suitable path from the source position to the goal position satisfying the predefined conditions.

The key problem to using Artificial Immune Network algorithm to a particular environment lies in defining the antigen and antibody of the system. Antibody is defined as the conditions of the mobile robot in the current environment, the information of the path, the distance to be travelled, the direction and orientation of the robot at the particular instant. The antibody is a set of conditions and the actions necessary to be taken in the occurrence of these conditions.

The conditions in this scenario are the presence of path or line on which the robot has to travel, the position of the destination, path identification when there are multiple paths at a branching node etc. and the action to be taken is the direction to be travelled and the orientation of robot at the given instant. Antigen can be defined as the condition of the environment at any particular instant along the travelling of the path and its structure is similar to that of the condition part of the antibody. Like there are a set of antibodies that fight the attacking antigens of particular kind, similarly the antibodies are the set of actions to be taken by the robot to achieve the task of reaching the destination following the most optimal path.

Here in this project the environment chosen to test this algorithm is the campus map of NIT Rourkela. Different academic departments and other prominent locations like auditorium etc. are chosen as the source and destination positions accordingly. The robot only has to move along the road paved on the campus as represented by black line in the simulation model and the scaled real time model used to test experimentally. Hence the conditions (or antigens) of concern can be defined as following:

1. The goal or destination position is known
2. The line direction can be known using IR sensors
3. The absence of path
4. The presence of multiple paths is known

The second condition can further be split into two subdivisions depending on the output of either of the sensors. The action to be taken can be defined by giving values to these conditions and the corresponding action is assigned to the particular value.

The goal position can be given values of 1, 2, 3 and 4 represented by 'a' depending on the location of the goal point on the axes of the environment. If the goal position is towards the positive end of the ordinate the value is 1. If it is closer to the positive end of abscissa the value is 2. And if it is closer to the negative end of the ordinate the value is 3. These values are with respect to the source being at negative end of abscissa.

The robot is ensured to move only along the paved path by giving values to sensors represented by 'b'. The left sensor is given value of 1 and right sensor value of 2 when it does not reflect light or is on a black line. So depending on the value obtained here the direction to be followed is determined. The presence of multiple paths can also be known from these values.

The actions to be taken are move forward, left, right or stop depending on the conditions.

Initialize the start position and goal position

Step 1: Check the value of b

Line follower:

Step 2:

If b=1, goto step 3

If b=2, goto step 4

If b=0, goto step 5

If b=3, goto step 6

Step 3: Take a slight left turn and goto step 1

Step 4: Take a slight right turn and goto step 1

Step 5: Keep moving forward and goto step 1

Choose path

Step 6: check the values of a

If a=1, goto step 3

If a=2, goto step 5

If a=3, goto step 4

If a=4, goal position is reached, halt and exit.

end.

4.2. Simulation Results

To validate the efficiency of the proposed algorithm, a known environment is considered as shown in fig.4.3. This environment is obtained from Google maps, which represents the roadmap of National Institute of Technology, Rourkela. This roadmap contains the locations of various departments and other important places in the campus. The problem statement of the current research work is to generate shortest paths between the given departmental locations.

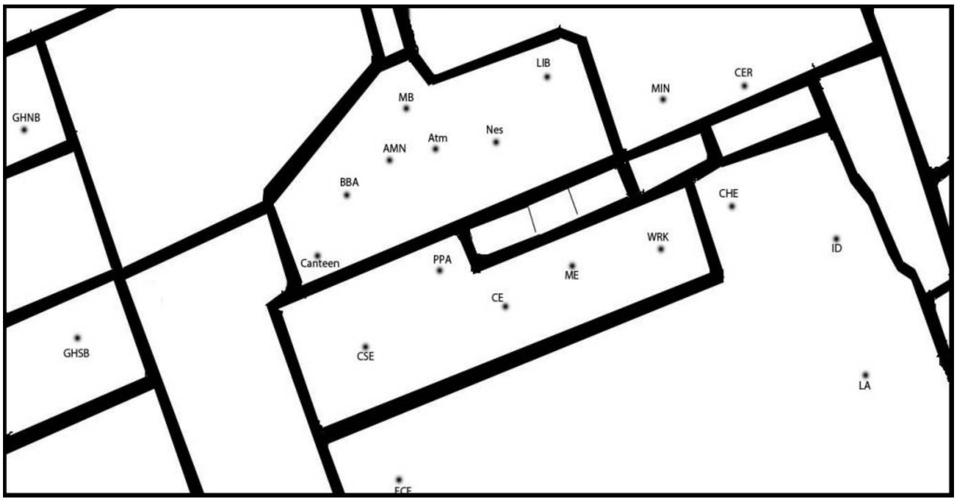

Fig.4.3: NIT Rourkela road map.

The roadmap is saved as an image file and then imported in MATLAB 2012b. For the various departments, corresponding coordinates are specified. First the map is studied and the location of the different departments are noted and verified. The path planning is done according to the process described in the flow chart as depicted in fig.4.4.

For simulation purpose the Unmanned Ground Vehicle (UGV) or the differential mobile robot considered here is represented as (robot x, robot y). Similarly all the departments and other prominent locations on the map are represented by the x and y coordinates.

The source and goal positions can be defined by

Case 1: entering the locations with the keyboard during program execution.

Case 2: choosing the desired positions on the roadmap by using mouse.

Once the source and the goal positions are attained, their respective position with one another are checked and the direction of the motion is determined. Here the whole map is checked and verified globally. If the source coordinates value is less than the goal position coordinate values, then the UGV moves in the forward direction towards left to right. Otherwise the UGV moves from right to left. Once the source and destination positions are given on the map, the UGV will move towards its goal position. If there are branching paths along the way, then the robot decides its direction motion according to the shortest path traversal criteria. The most important factor to be considered here is that always the shortest path is chosen for deciding the direction of motion of the robot towards the goal. When the vehicle is moving in the given path it has travel only along the road dedicated for its travel. It means that, the UGV always follows the black line indication on the road.

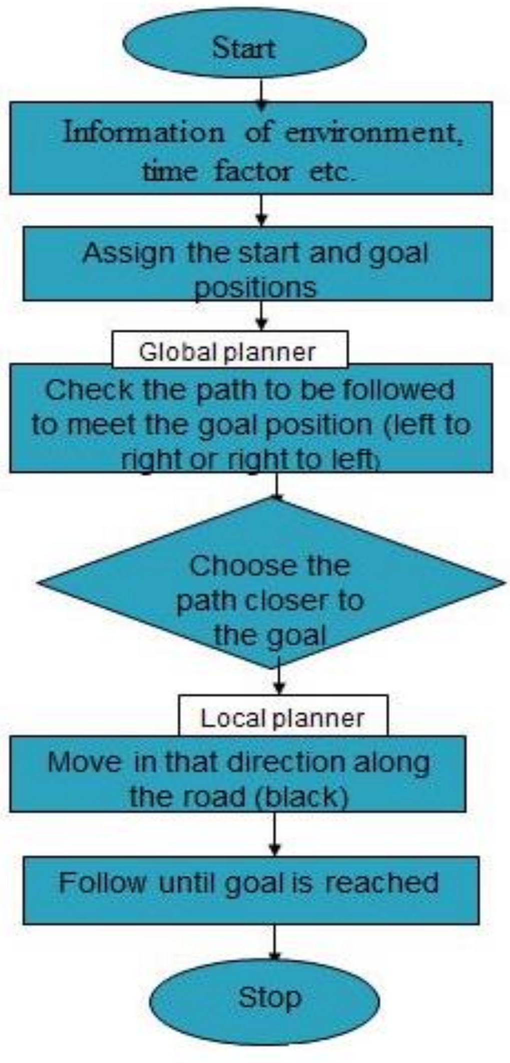

Fig.4.4: Flow chart of algorithm.

As described in the previous section, case 1 the source and goal position are entered with the mouse as shown in fig.4.5 and the simulation result containing the path for the corresponding positions is shown in fig.4.6. In this case it can be seen that goal coordinate value is higher than source position and hence the robot moves forward from left to right.

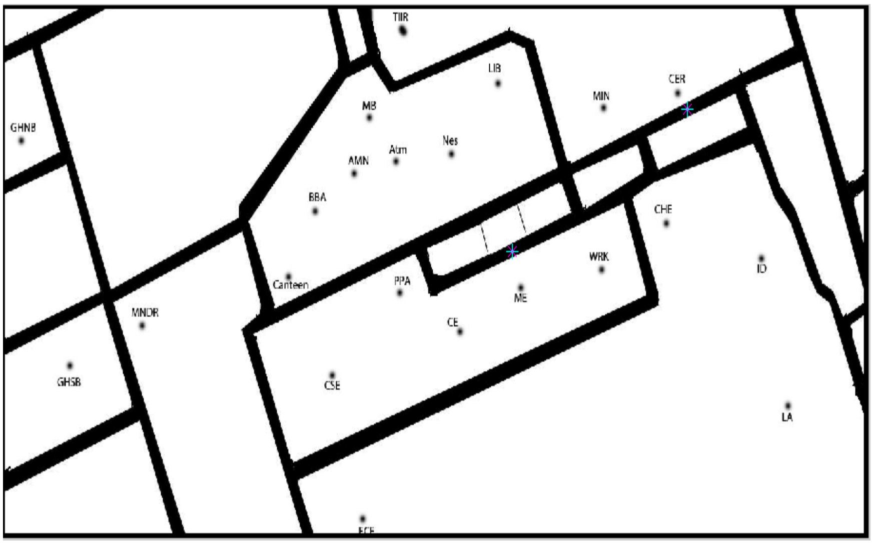

Fig.4.5: Source and goal positions entered with mouse.

Here the source is at mechanical (ME) department with coordinates x=805 and y=303. And the destination is ceramic (CER) department with coordinates x=1074 and y=127. Time taken for the simulation to take place is 4.95 seconds.

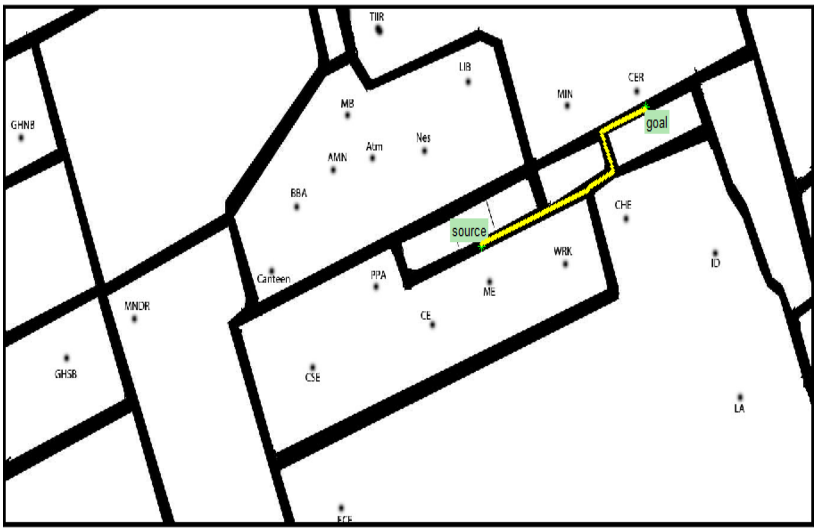

Fig.4.6: Path for the positions indicated in case 1.

Similarly as described in case 2 the source and goal positions can be entered manually using keyboard during execution as shown in fig.4.7

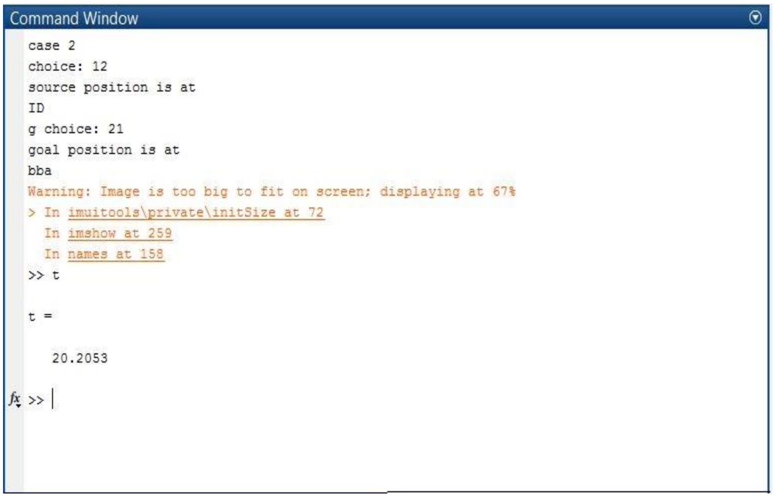

Fig.4.7: Command window showing source and goal positions specified during execution.

The path obtained for the specified locations in fig.4.7. is shown in fig.4.8.

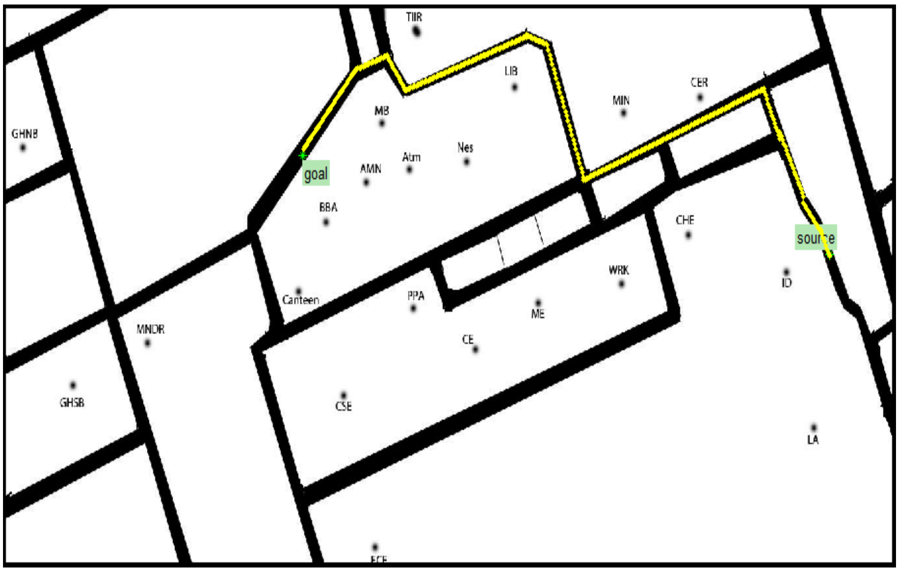

Fig.4.8: Path obtained from simulation for case 2.

For checking the output in case 2 the source position is at ID department with coordinates x=1264 and y=292and the goal position is at BBA with coordinates x=459 and y=171. Here the source position coordinates are higher than the goal position and hence the robot moves from right to left direction. The time taken to reach the destination is 20.053 seconds.

Chapter 5

EXPERIMENTAL SETUP

AVR Development Board 107010

The AVR Microcontrollers

Infrared Sensors

Wheel

Castor wheel

5. EXPERIMENTAL SETUP

The specifications of the mobile robot are given in the table 5.1. below:

Table 5.1.Specifications of robot

Microcontroller	Avr Atmega 16a
Flash Memory	16 KB (ATmeg16)
Operating Voltage	5V
SRAM	2 KB (ATmega 16)
Input Voltage (recommended)	5-20V
Input Voltage (limits)	5-40V (Driving Supply)
Motors	2 Direct Current, 300RPM DC Motor
Motors Driver	L293NE Up to 40V DC Motor Driver
Speed	Max: 300RPM, Min: 120RPM
Sensors	4 IR Range Sensor Distance measuring range: 2cm to 14cm
Communication	USB connection Serial Port
Size	Height: 7.5cm, Length: 27cm, Width:33cm,
Weight	Approx. 1.4kg
Payload	Approx. 400g
Power	Rechargeable Lithium Polymer 4 Cell, 12V, 2000mAh, 9v alkaline battery

5.1.AVR Development Board 107010

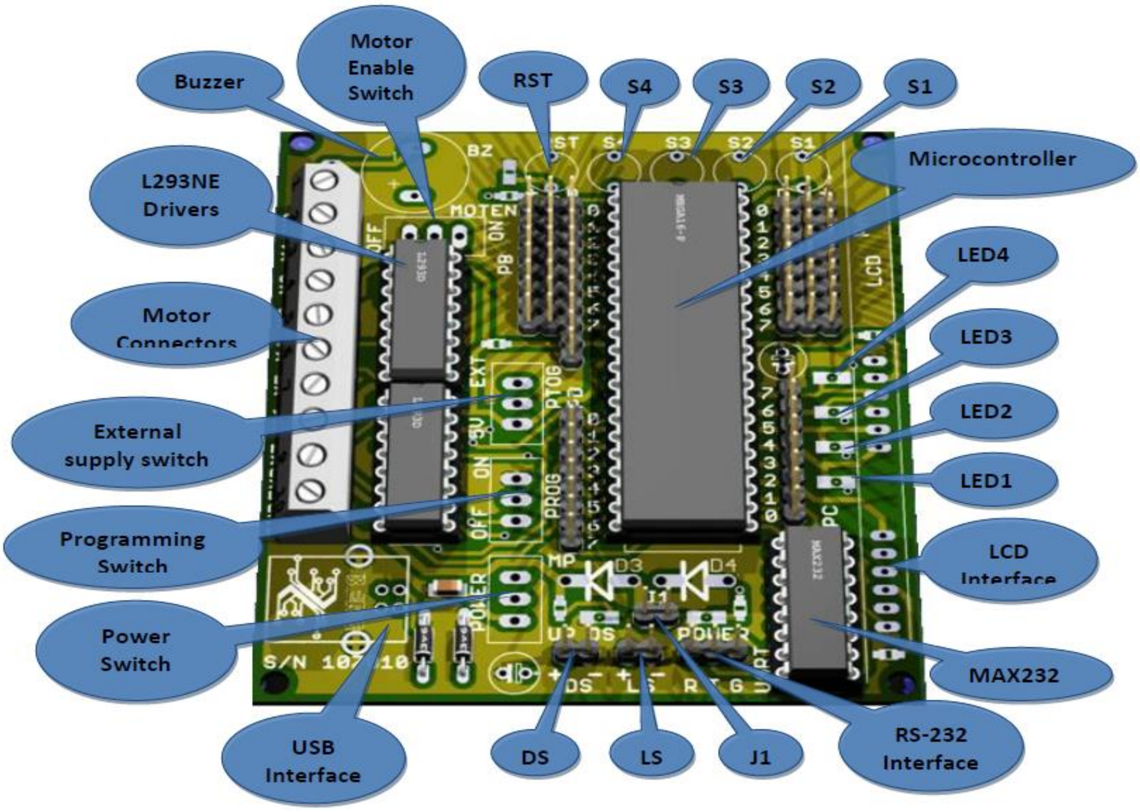

Fig.5.1: Mother Board used for programming the robot.

Some of the features of the AVR development Board 107010 shown in above fig 5.1 are:

- It includes Atmel’s ATmega16 Microcontroller with a flash memory of 16kb working at a speed of 16MIPS or ATmega8535 Microcontroller with a flash memory of 8kb working at a speed of 16MIPS.
- It can be used with almost all AVR 40 pin micro controllers.
- LCD interface is present on board itself (it can be used for most of the general purpose application).
- 2 Motor Drivers are available on the board, with which 4 DC motors or 2 Stepper motors can be connected.
- On-board servo interface is available
- On-board voltage regulator is present.

- It can be interfaced with PC through UART.
- On-board Buzzer is present.
- It has 16 MHz external crystal.
- All 32 I/O pins are exposed.
- 8 channel I/O pins are exposed for ADC and sensors with 5V/1A power supply.
- 8 channel I/O pins are exposed for servo and sensors with dual power supply.
- External input and reset can be controlled by using the five tact switches present on board.
- Four test surface LEDs are available for indicating status and debugging purpose.
- Two LEDs are used as supply indicators.
- Power can be supplied through either DC source (6V to 16V) or USB powered.
- USB programmer is present on board.
- Two power supply options are available (Dual or single).
- ISP pins are exposed for programming.

The major parts of the Avr Board are listed below:

- *Microcontroller:* A microcontroller is the heart of an embedded system. It is a standalone micro computer chip which stores the particular programs, executes them and takes action accordingly. The chip that is used here is Atmels manufactured AVR micro controller.
- *1117 voltage regulator:* It is a 5V voltage regulator IC with three terminals which is used to provide a constant 5V voltage supply to the micro controller and other peripherals (i.e. sensors etc.) connected to the main board.
- *MAX232:* It is an IC that takes care of voltage conversion necessary for the communication between the AVR Development board and PC's RS-232 (Serial/COM) port.
- *L293NE motor driver***:** It is nothing but a motor driver IC that receives input signal from the microcontroller and drives the DC and stepper motors by using different power source.

- *16 X 2/16 X 1 LCD interface*: The LCD interface can be interfaced with any 16x2 or 16x1 character LCD display in 4 bit mode. The LCD display can display any message, status or it can also be used for purpose of debugging. The ports used for LCD interfacing can also be used as a general purpose input output port. Following are the pin connections for interfacing an LCD to the board:
  - RS - PortC0
  - RW - PortC1
  - E - PortC2
  - D5 - PortC4
  - D6 - PortC5
  - D7 - PortC6
  - D8- PortC7
- *Switches:* For providing an external input to the board, the board has four tact switches along with a Reset switch. The switches are connected to the controller on board in the following manner:
  - S1 - PortA4
  - S2 - PortA5
  - S3 - PortA6
  - S4 - PortA7
- *RST (Reset switch):* The Reset switch as the name indicates is used to reset an executing program right to the starting point. Its application is similar to the reset switch of a PC.
- *MOTEN (Motor Enable Switch)***:** It is actually a toggle switch which toggles between the ON and OFF positions. Basing on the position of the switch (ON or OFF) mode the motor drivers are enabled or disabled accordingly.
- *POWER (Power on switch):* It is a simple toggle switch that toggles between MP (Main Power) or UP (USB Power) modes. It is used for providing power supply to the main board. We can use a battery power supply (through LS) to supply power to the board or we can use power drawn from USB source.

- *PTOG (Power Toggle switch):* It is a toggle switch which toggles between the 5V and EXT (external supply mode). The devices that are connected to PORTB can derive their power supply by using the internal power supply of 5V from the main board by throwing the switch in 5V mode or by using other external power sources that are connected in DS by throwing the switch in EXT mode. This EXT mode is usually used for high power consuming applications like driving servo motors etc.
- *PROG (Programming Switch)*: It is a toggle switch that can be used for programming of the microcontroller by using the USB programmer present on the board. When the controller is in normal operation mode the switch should be toggled towards the OFF position. When the microcontroller is to be programmed the switch should be toggled to ON position and the RESET switch should be pressed.
- *Power supply*
    - *LS (Logic Supply)*: It consists of two terminals one being the +ve terminal and the other being the –ve terminal. The power supply can be provided to the motherboard by connecting a battery or AC adaptor to these terminals. The peripherals present in the mother board receive the power supply from these terminals. The power supply from these terminals is regulated using a voltage regulator and this regulated power supply powers the external peripherals connected to the motherboard. The DC voltage provided by the source connected to this terminal must be in the range of 6 to 16 volt. When using the supply connected to LS pin the power switch must be toggled towards "MP" (Main power).
    - *DS (Driving Supply)*: It consists of two terminals one being the +ve terminal and the other being the –ve terminal. It is used for providing power supply to the high power applications. For operating DC motors the DC voltage provided by the source connected to this terminal must be in the range of 5 to 40 volt. For operating a servo motor the power source can be chosen according to the requirement of the particular motors being used (mostly servos operates at 4.5 to 6 volt). When PTOG switch is toggled towards EXT mode, the power derived from these pins directly goes to the supply pins of PORTB and to the driving supply of motor drivers.

- *J1 (Jumper 1)***:** we can use a single power supply for both DS and LS by using a simple jumper at these terminals. When we put a jumper here then only DC power supply must be provided otherwise both the motherboard and LS will be getting the power supply.

- *USB socket***:** This port is basically used for communicating with USB port of the PC. The necessary logic supply to the motherboard is provided by using this USB power supply by toggling the POWER switch towards UP (USB power) mode. When using the USB power some precautions to protect the board. We should avoid connecting any heavy loads to the board directly and we shouldn't use the J1.

- *LED's :* Following ports must be active high to turn on the status indicating LEDs:
  - RED LED1 – PORTC4
  - RED LED2 – PORTC5
  - RED LED3 – PORTC6
  - RED LED4 – PORTC7
  - GREEN LS - Logic Power ON indicator
  - GREEN DS - Driver Power ON indicator

- *BZ (Buzzer):* It gives an audible feedback from the controller when it is connected in an active high mode. The Buzzer is connected to the PORTC3 pin of the microcontroller. Hence when the PORT C3 pin is given active high or logic '1' the buzzer beeps. The buzzer also beeps when the positive and negative terminals of the power source come in contact.

- *ISP (In-System Programming) interface:* To download the programs in the microcontroller we can use any ISP programmer connected to the In-System Programming interface of the main board. It can also be used in SPI (Serial Peripheral Interface) communication. The pins connections for ISP are given below:
  - MO- Master Out Slave in PortB5
  - MI- Master in Slave out PortB6
  - SCK- Serial clock PortB7

- RST- Reset Reset
- GND- Ground Ground

- *RS – 232 interface*: This interface post consists of three terminals namely, R – Receiver, T - Transmitter, G – Ground. It can be used for PC controlled applications, debugging purpose, data communication with PC and for inter board data communication.
- *PA (Port A):* It is a general purpose I/O port. This port consists of eight pins that can be used as digital input, ADC in and digital output. These pins are in the form, DATA-VCC-GROUND (denoted as D + - respectively on the board). The Data pins are connected to the microcontroller. A 5V/1A power supply can be used to provide power to the VCC and Ground pins.
- *PB (Port B):* It is a general purpose I/O port. This port consists of eight pins that can be used as digital input and digital output. These pins are in the form, DATA-VCC-GROUND (denoted as D + - respectively on the board). A 5V/1A power supply can be used to provide power to the VCC and Ground pins. The supply to these pins can also be provided by using external supply connected in DS pin through EXT switch.
- *Motor driver connections:* The DC motors or stepper motors are run by using the motor drivers connected to the board which are controlled by the data from the microcontroller. The following are the connections of the motor drivers with the micro controller.
  - M0 - PortB0
  - M1 - PortB1
  - M2 - PortB2
  - M3 - PortB3
  - M4 - PortD4
  - M5 - PortD5
  - M6 - PortD6
  - M7 – PortD7

5.2.The AVR Microcontrollers

5.2.1. Description

Developed in 1996 by Atmel corporation AVR is a 8-bit RISC single chip microcontroller. The **AVR** has Modified Harvard architecture. Among microcontroller families AVR was the first to use on-chip flash memory for program storage. Most of the microcontrollers before AVR used One-Time Programmable ROM, EPROM, or EEPROM. Atmel's microcontroller operates on low power and has high performance. AVR microcontrollers handle demanding 8 and 16-bit applications. The lowest possible power consumption is possible because of the single cycle instruction set and RISC type of CPU, advanced innovative technology based on Pico Power®, and the architecture having a developed feature set. The tuned AVR instruction set helps in decreasing the size of the program and development time. Because of the well-defined I/O structure external components are not required and this helps in reducing the cost of development. Out of the four families of AVR microcontrollers released by Atmel Corporation we preferably use the microcontroller belonging to the mega AVR family especially “Atmeg16”.

5.3.Infrared Sensors

An infrared sensor is an electronic device that emits radiations and aids sensing a few aspects of the surrounding environment. An IR sensor can be used to measure the heat of an object and it can also detect the motion**.**

The typical wavelength of infrared waves is between 0.75 and 1000µm. The spectrum of infrared waves is split into three regions namely, the near infrared region which ranges from 0.75 to 3µm, the mid-infrared region which is in the range of 3 and 6µm and infrared radiations of wavelength greater than 6µm come under the far infrared region.

All the objects that are at a temperature higher than absolute zero (0 Kelvin) contains thermal energy and hence are sources of infrared radiation. Blackbody radiators, tungsten lamps and silicon carbide are mostly used as the Sources of infrared radiation. Typically infrared lasers and LEDs with specific infrared wavelengths are used as sources in Infrared sensors. For the transmission of infrared radiations a transmission medium is required, which can be comprised of vacuum, the atmosphere or an optical fiber.

The infrared sensors are more popular and used frequently because of their

- low power requirements
- simple circuitry
- portable features

Usually all the objects in the infrared spectrum radiate some sort of thermal radiations. The infrared radiations are usually invisible to the human eye but they can be detected by using an infrared sensor. In a simple IR sensor an IR LED (Light Emitting Diode) acts as an emitter and an IR photodiode that is sensitive to the infrared radiations of same wavelength as emitter.

The resistance of the photodiode changes proportionally with the magnitude of the incident IR light radiations and thus the output voltage changes. This voltage change can be used to detect presence and magnitude of the IR radiations.

5.3.1. Infrared sensors types

Infrared sensors can be broadly classified into two types namely:

- *Thermal Infrared Sensors*: This type of sensors uses heat or temperature difference due to the infrared energy. The photo sensitivity of the sensor does not depend on the wavelength being detected. Cooling is not required for Thermal detectors but they have slow response times and the detection low.

- *Quantum Infrared Sensors*: The photo sensitivity of the detector depends on wavelength. In order to obtain accurate measurements quantum detectors have to be cooled. They have faster response times and detection capabilities are high.

The infrared sensors can also be classified as

- *Active IR Sensor*: This type of sensors can both emit and measure the infrared radiations
- *Passive IR sensor*: This type of sensor can only measure infrared radiations.

5.3.2. IR sensor circuit diagram and working principle

An IR sensor module is most basic and popular in electronic devices. One of the most common applications of this module is to detect obstacles in real time environment. The components of an IR sensor circuit are as

- An operational amplifier-LM358 IC
- LED (Light Emitting Diode)
- Infrared LED (transmitter)
- Photodiode (receiver)
- 10 kilo ohm variable resistor
- Resistors of different ranges.

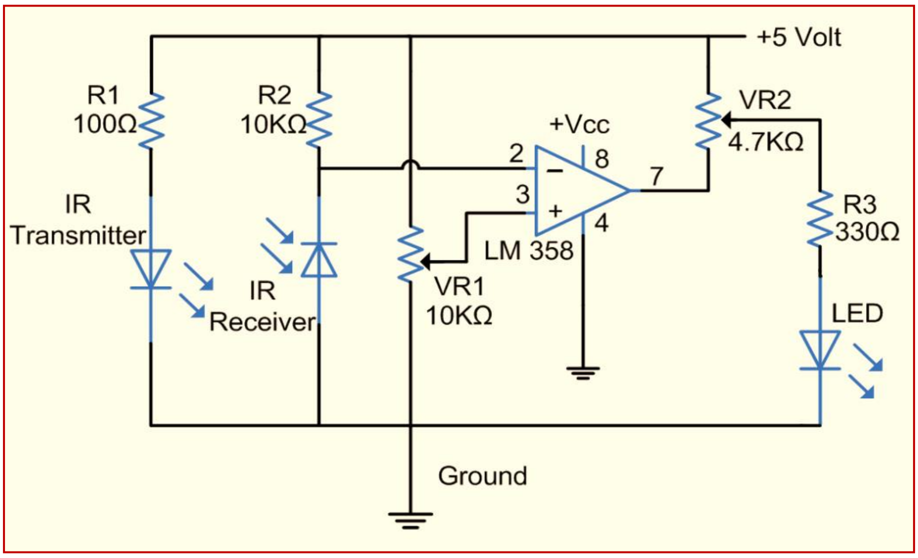

Fig.5.2: Circuit diagram of infrared sensors.

In the circuit shown in above figure an IR LED acts as transmitter that emits the IR rays continuously. This constitutes the transmitter section. The receiver module consists of a photodiode that receives the incident IR rays and the resistance of the photodiode changes according to the incidence of radiations. In order to analyze the variations in the values directly the output values are given to a comparator circuit, in this case an operational amplifier (op-amp) of LM 358 is used as comparator.

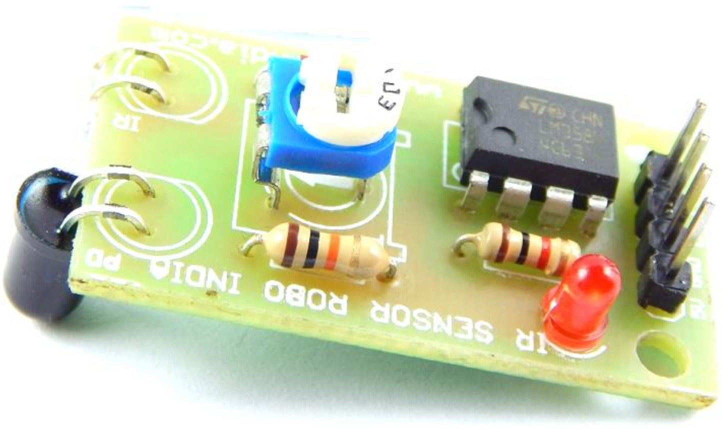

Fig.5.3: Infrared sensor module.

When the IR LED is not glowing or the photodiode is not in the range of incident radiations the inverting input value is greater than the non-inverting input value and hence the output of the comparator becomes low. Whereas when the IR rays are incident on photodiode the non-inverting input will be greater than the inverting input and hence the output of the comparator becomes high. When the output of the comparator is high the LED glows. Thus the LED acts as an indicator to the output status of the sensor module. Table 5.1.gives the specifications of an IR sensor module.

Table 5.2. Specifications of IR sensors

Detection range	Upto 14cm
Weight	10 Gram
Output Type	ANALOG and DIGITAL
Power supply voltage	5 volt
Average current consumption	33mA

5.3.3. The key applications of infrared sensors

- Night Vision Devices
- Infrared Tracking
- Gas detectors
- Water analysis
- Anesthesiology testing
- Rail safety

- Automatic Teller machine
- Gaming products

5.4.Wheel

A circular component that rotates on an axial bearing is known as wheel. Wheels, when used in conjunction with axles, they facilitate movement or transportation of heavy weights by supporting the load. They are also used to perform labor in machines. A ring-shaped covering that fits around a wheel's rim is surrounded by a ring shaped covering known as a tire (or tyre) that protects the wheel and ensures better vehicle performance. Synthetic rubber, natural rubber, fabric and wire, along with carbon black and other chemical compounds are the common materials used in modern pneumatic tires. The tyre mainly consists of a tread and a body. Traction is provided by the tread while the body provides the containment of compressed air. The specifications of the wheel shown in fig 5.4 is given in table 5.2

The main Performance characteristics of a wheel are:

Balance: When a wheel and tire rotate, they exert a centrifugal force on the axle that depends on the location of their center of mass and the orientation of their moment of inertia.

Rolling resistance: Rolling resistance is the resistance to rolling caused by deformation of the tire in contact with the road surface. As the tire rolls, tread enters the contact area and is deformed flat to conform to the roadway. The energy required to make the deformation depends on the inflation pressure, rotating speed, and numerous physical properties of the tire structure, such as spring force and stiffness.

Compatible motors that can be used with this wheel are:

10 RPM Center Shaft Metal Gear

100 RPM Center Shaft Metal Gear

10 RPM Johnson geared motor

100 RPM Johnson geared motor

1000 RPM Johnson geared motor

Round shape Stepper Motor (Low Torque)

Small Stepper Motor (Mid Torque)

Fig.5.4: Wheel used in robot.

Table.5.3. Specifications of the wheel

Weight	80 Gram
Wheel diameter	90mm
Wheel thickness	14mm
Hole diameter	6mm

5.5. Castor Wheel

Ball caster wheel is an Omni directional wheel. This wheel used as neutral wheel for the robot. It used in various applications such as shopping malls, office chairs and material handling equipment. High capability and heavy responsibility caster used in industrial applications such as platform tank, assembly lines. It used mostly in the smooth environment and flat surfaces. The specifications of castor wheel are listed in table 5.4.

Fig.5.4: Castor Wheel.

Table 5.4. Specification of castor wheel

Weight	45 gram
Base plate diameter	38.2mm
Caster wheel diameter	21.3mm
Wheel height	22.8mm
Mounting hole	Three
Angel between the hole	120 degree apart
Hole diameter	3.4mm

Chapter 6

EXPERIMENTAL ANALYSIS

Results and discussions

6. EXPERIMENTAL ANALYSIS

This chapter deals with the experimental analysis of mobile robot in the real time environment when it operates on the designed NIT Rourkela campus map. Top view of the robot showing the development board and the external connection to the board is given in fig. 6.1 and the front view showing the sensor connections, motors, battery etc is given in fig. 6.2.

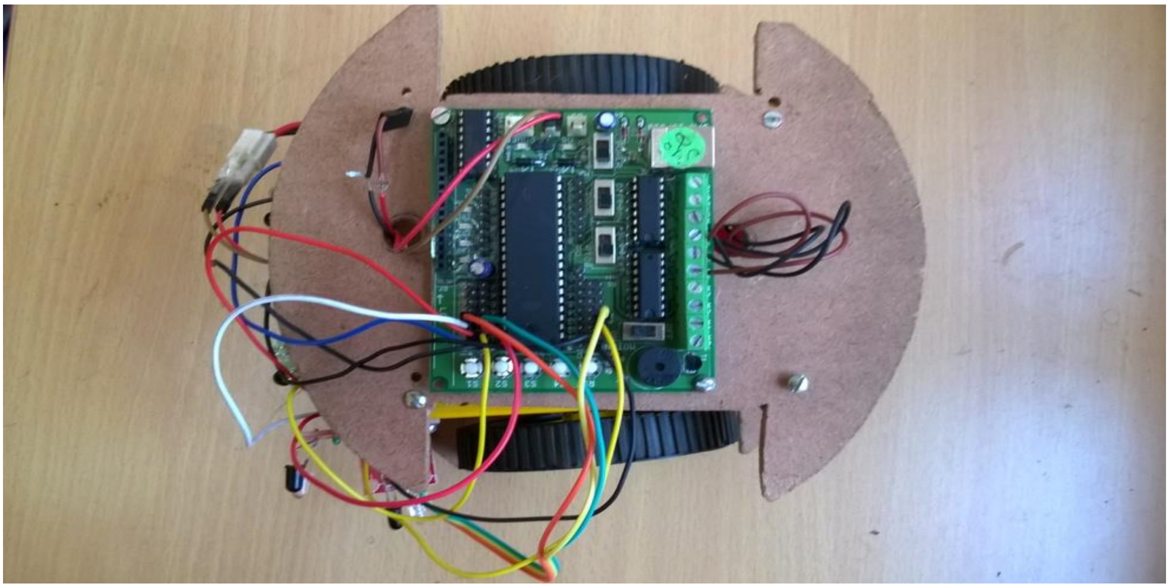

Fig.6.1: Top view of mobile robot.

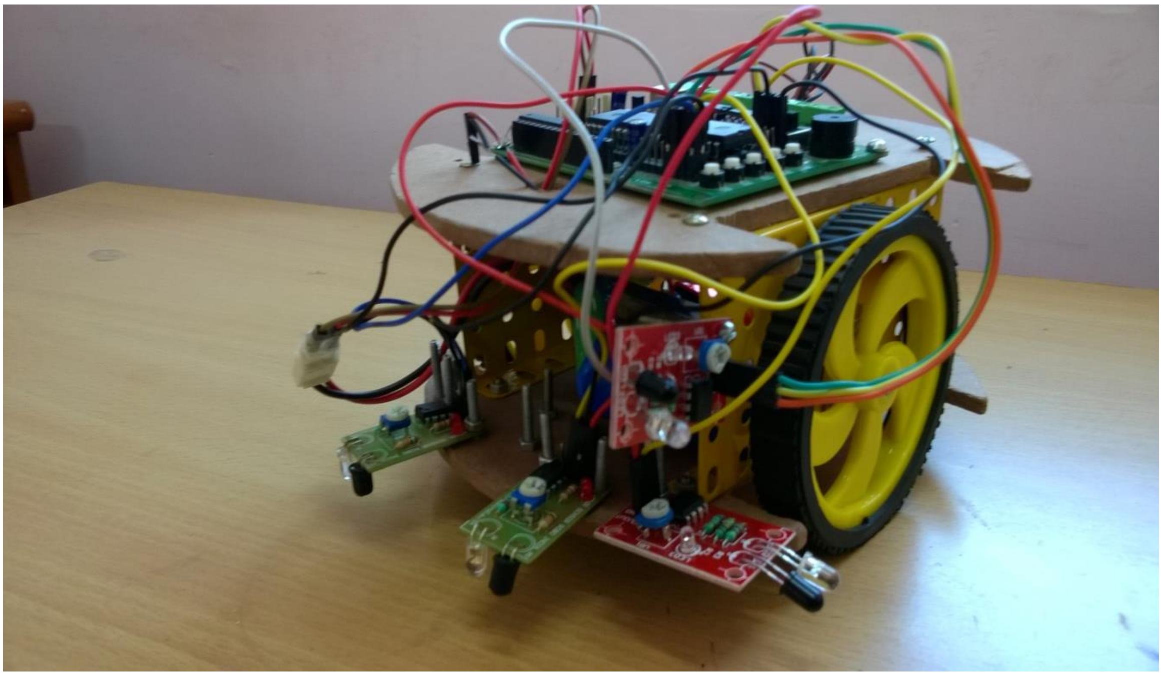

Fig.6.2: Front view of mobile robot.

The path planning of the mobile robot is done using the algorithm defined in chapter 4. Here 4 IR sensors are used as shown in fig 6.2. Two IR sensors are connected at the front of the robot. This pair of IR sensors is used to ensure that the robot moves along the given path only (Roads of NIT Rourkela campus map). Third sensor connected along the bottom corner is used to direct the robot and help it decide which direction it has to turn to when multiple paths are present. The fourth IR sensor used to detect when the destination point is reached.

6.1.Results And Discussions

The mobile robot is given different source and target locations and the behavior of robot is analyzed at different locations during the turns, at source and destinations positions etc. the time taken for travelling between specified locations is recorded and the results are compared to the simulation results

Case 1:

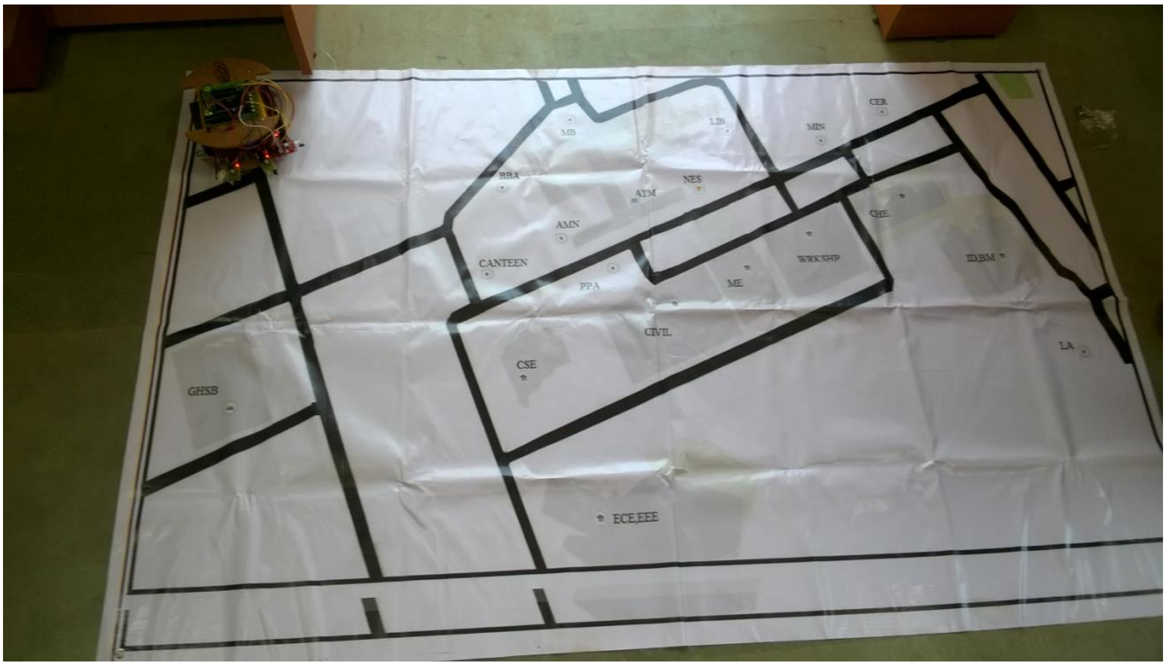

Fig. 6.3: Robot positioned at source (guest house north block).

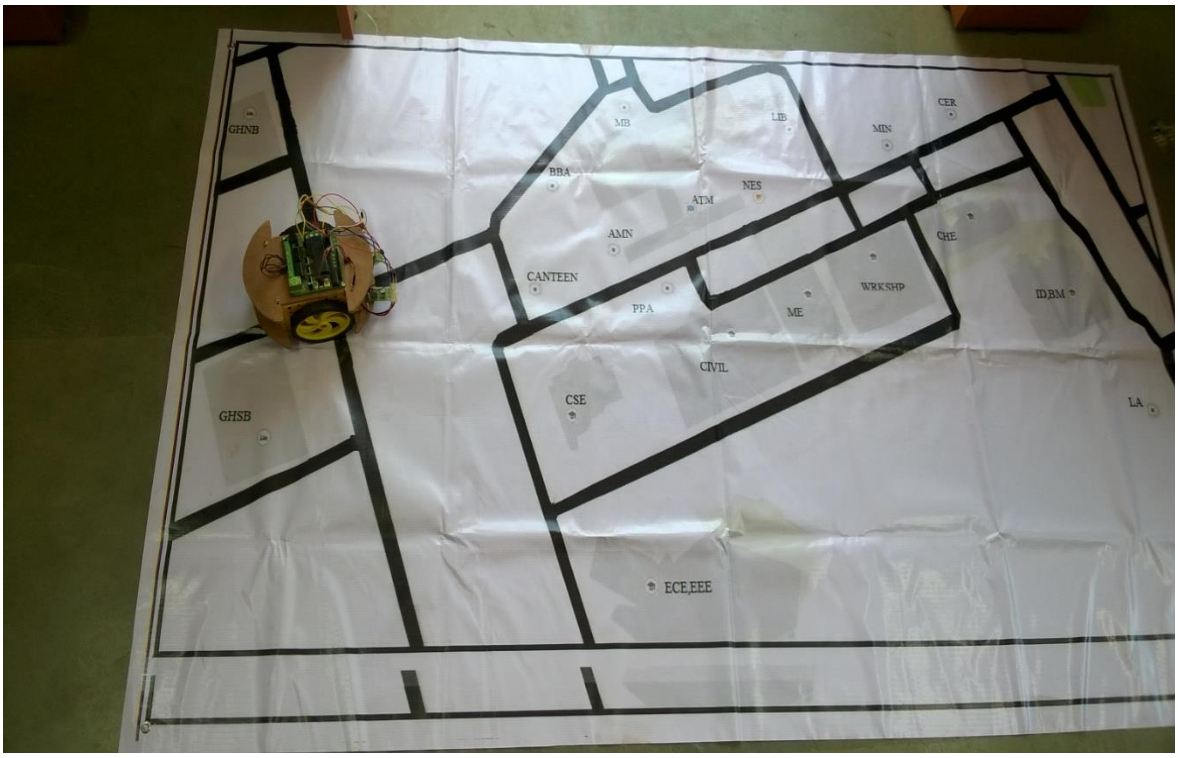

Fig. 6.4: Robot at branching point.

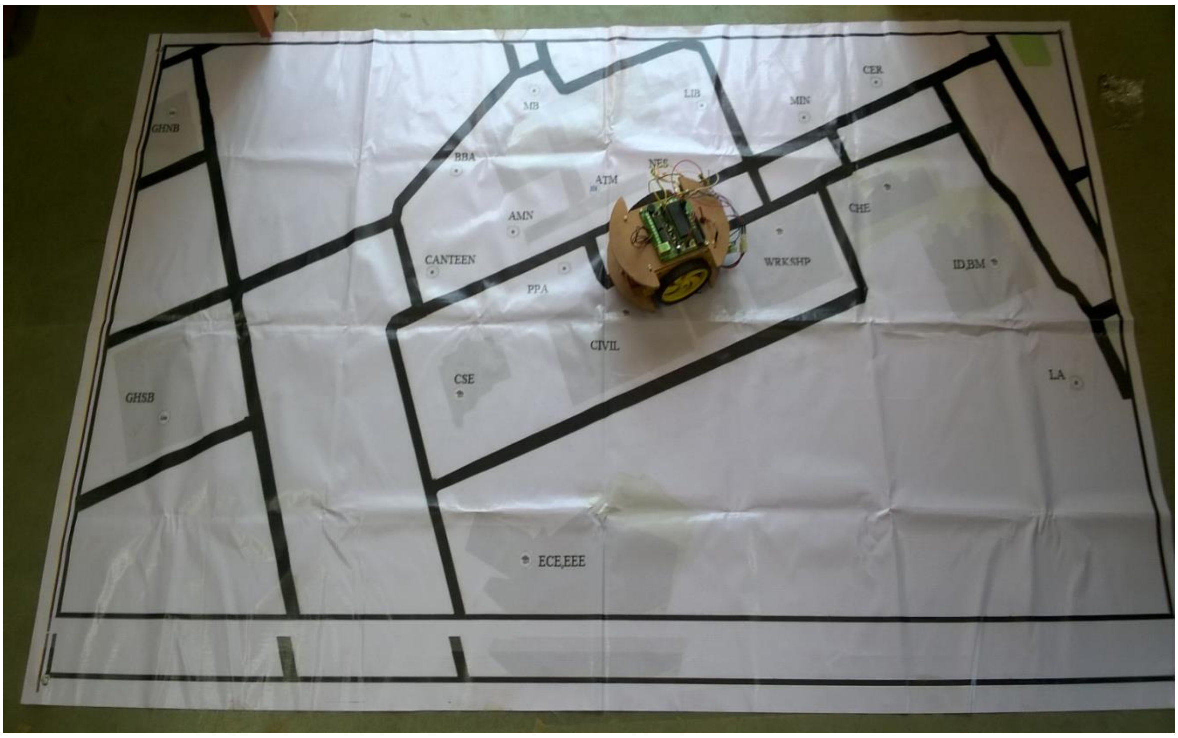

Fig.6.5: Robot at destination point (mechanical Dept.).

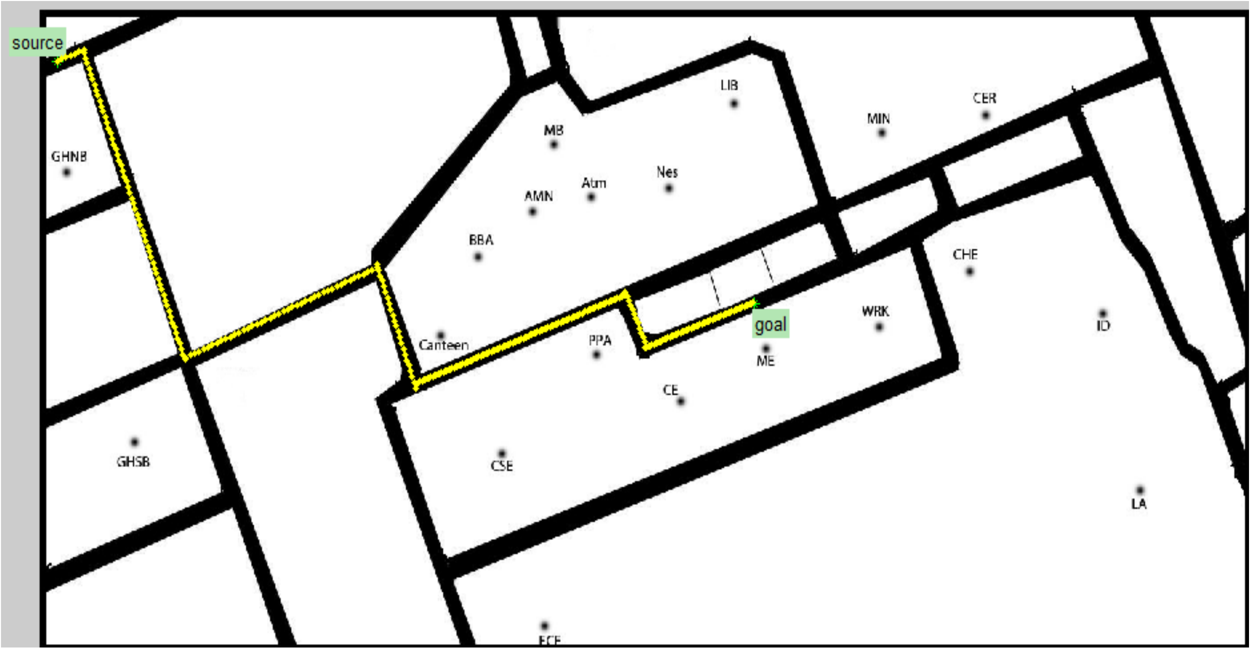

Fig.6.6: Simulation result for travelling between guest house and mechanical Dept.

Case 2

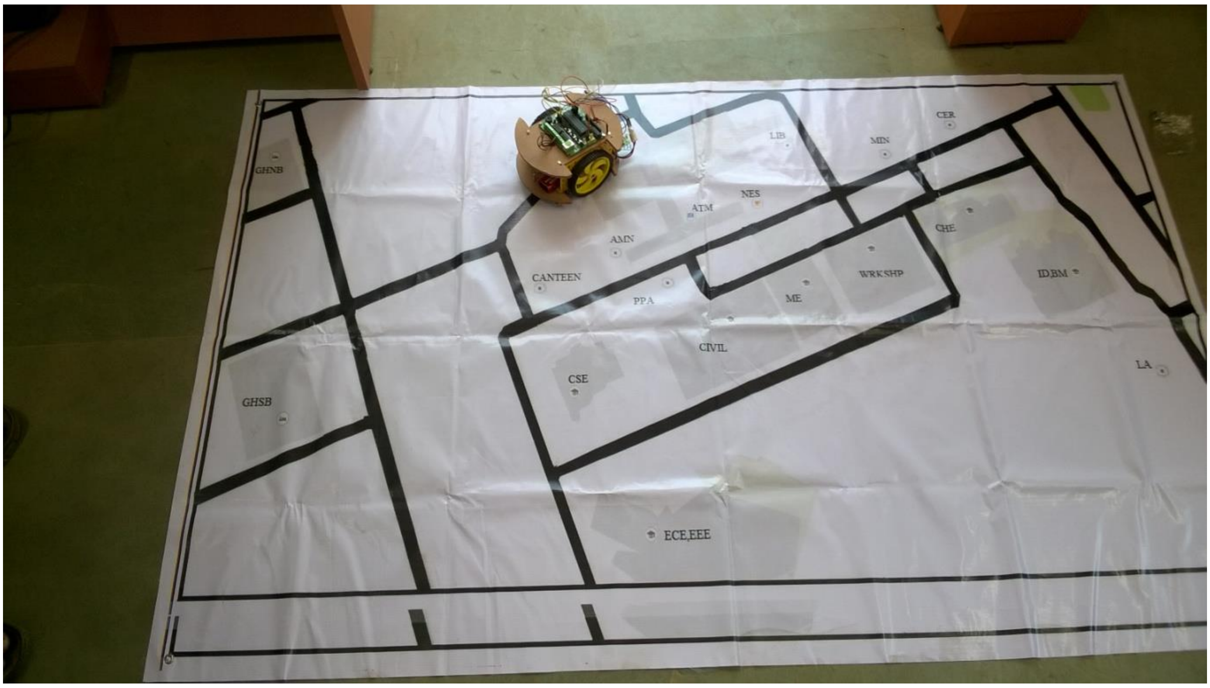

Fig. 6.7: Robot positioned at source (B.B. Auditorium).

Fig.6.8: Robot at destination point (ceramic Dept.).

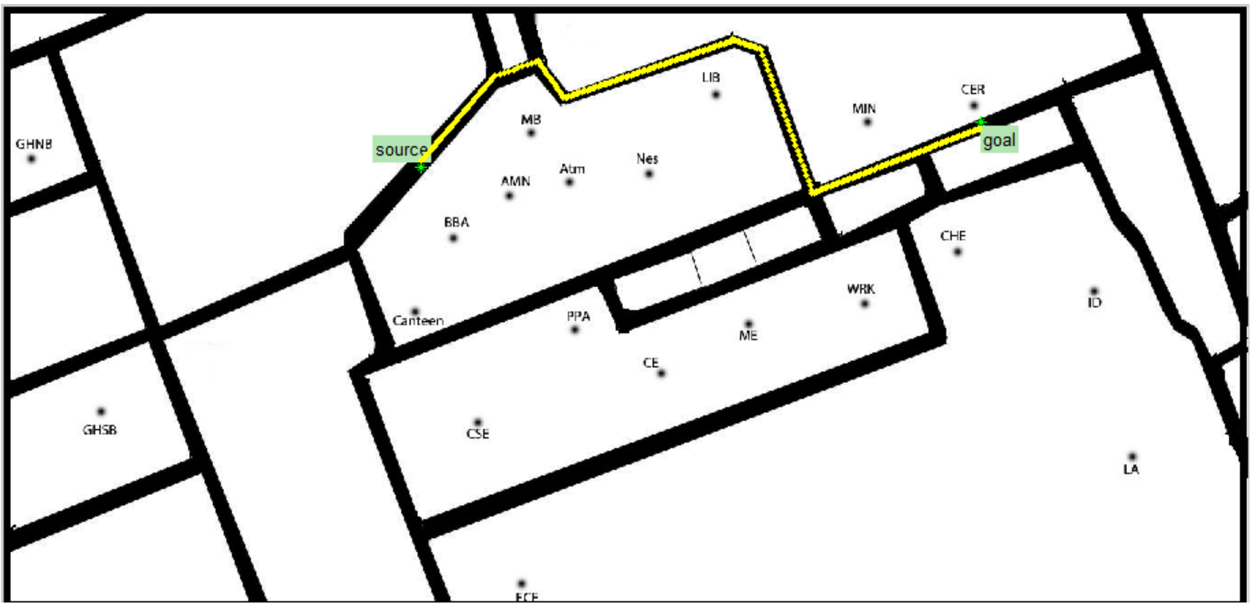

Fig. 6.9: Simulation result for travelling between B.B. Auditorium and ceramic Dept.

Case 3

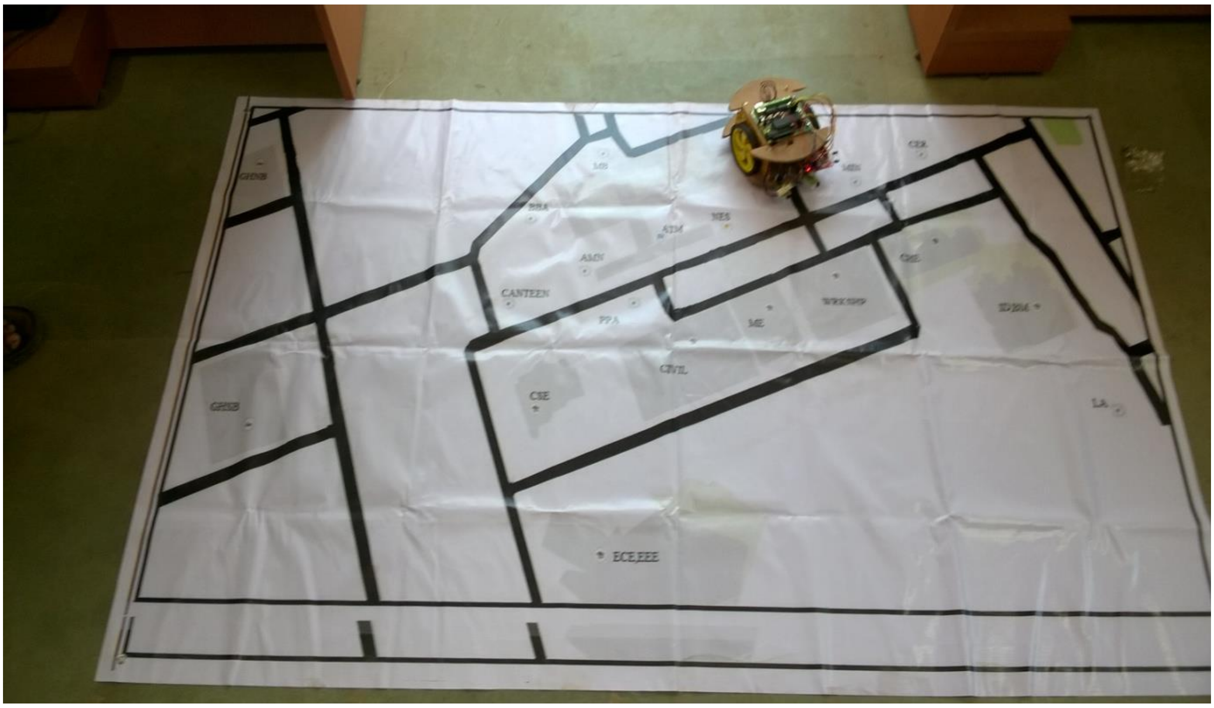

Fig.6.10: Robot positioned at source (library).

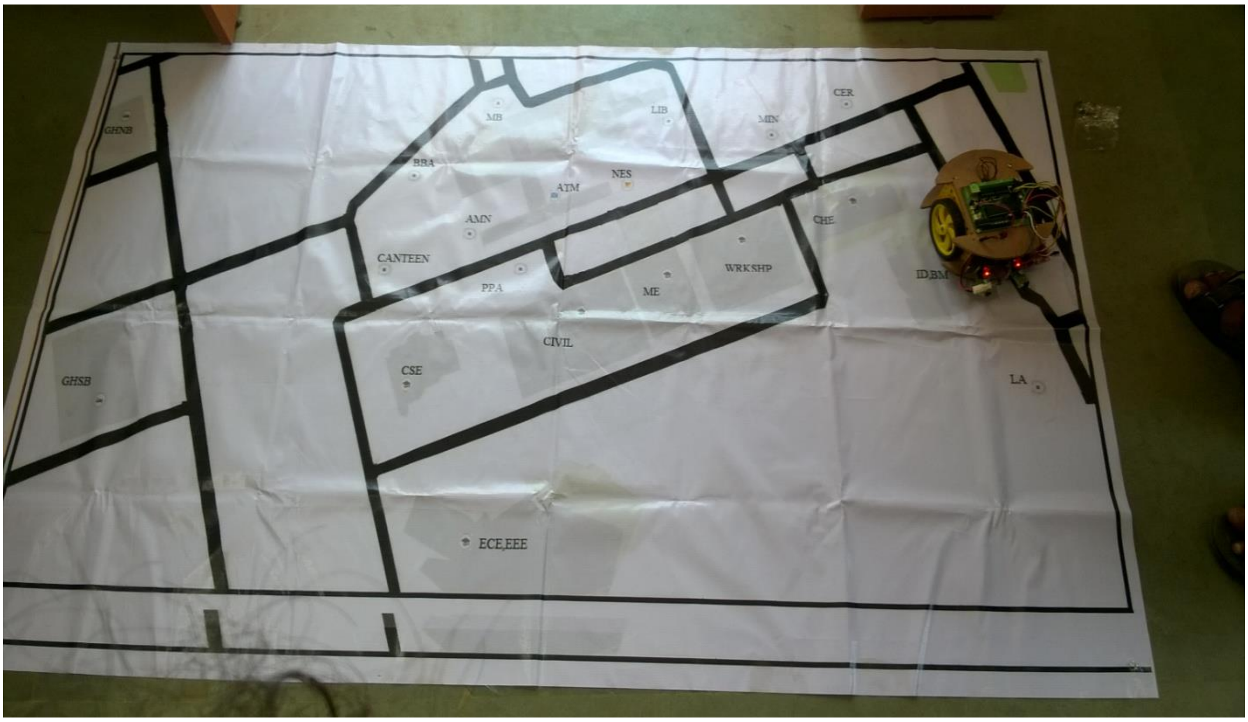

Fig.6.11: Robot at destination point (Industrial Design Dept.).

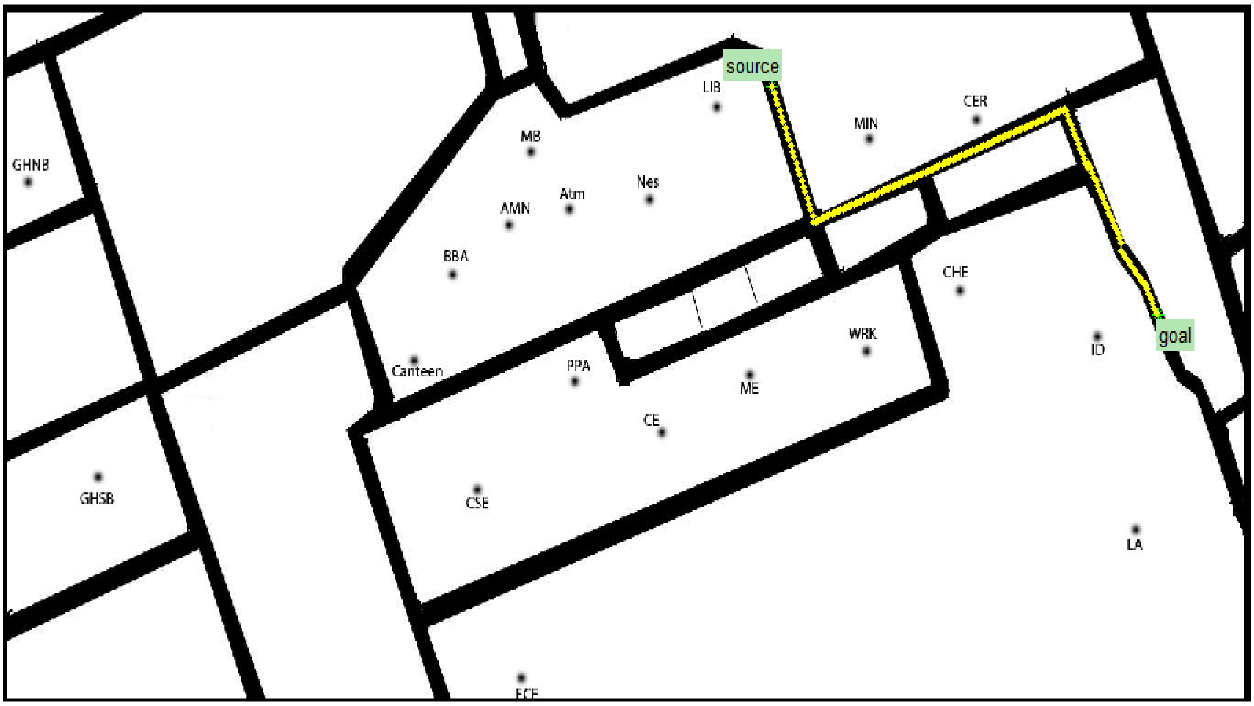

Fig.6.12: Simulation result for travelling between library and Industrial Design Dept.

Case 4

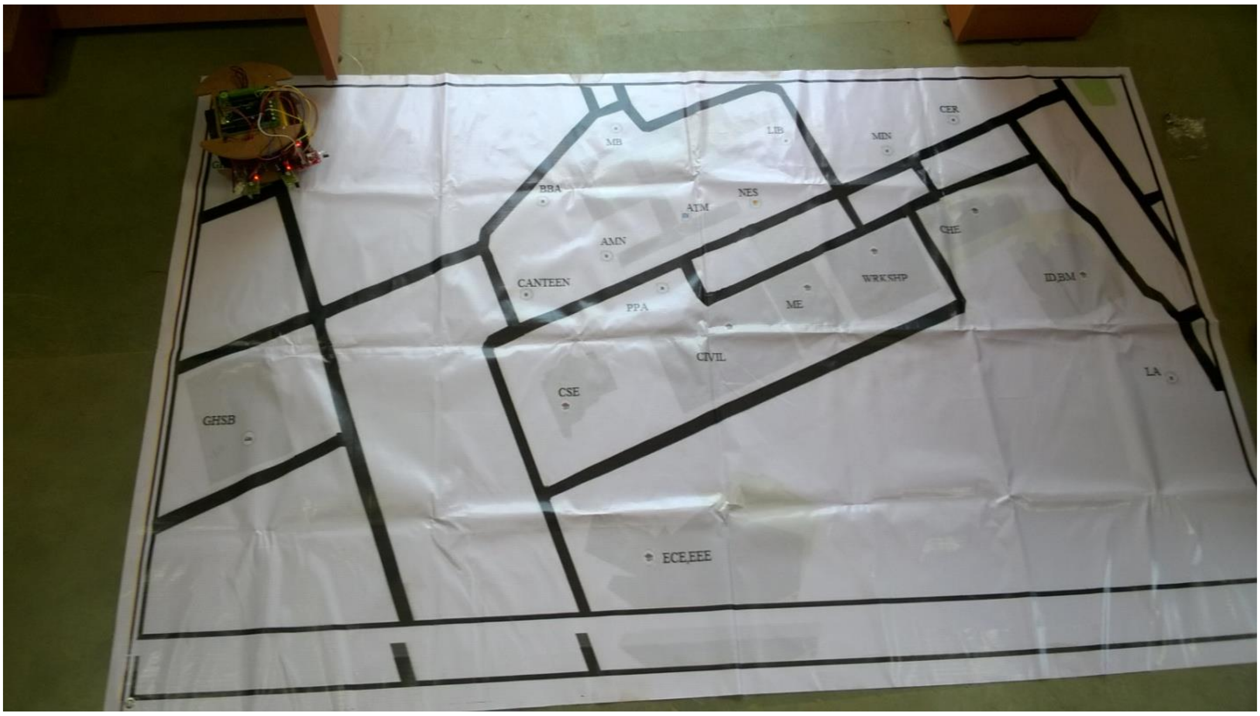

Fig.6.13: Robot positioned at source (guest house north block).

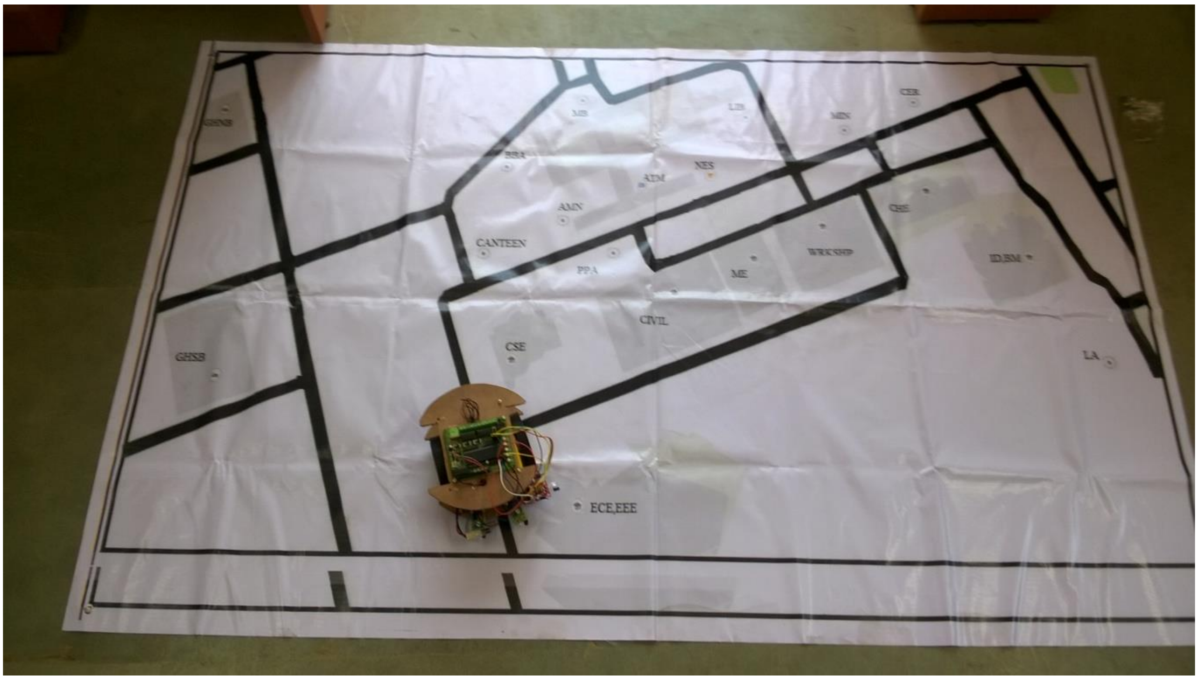

Fig.6.14: Robot at destination point (Electrical Dept.).

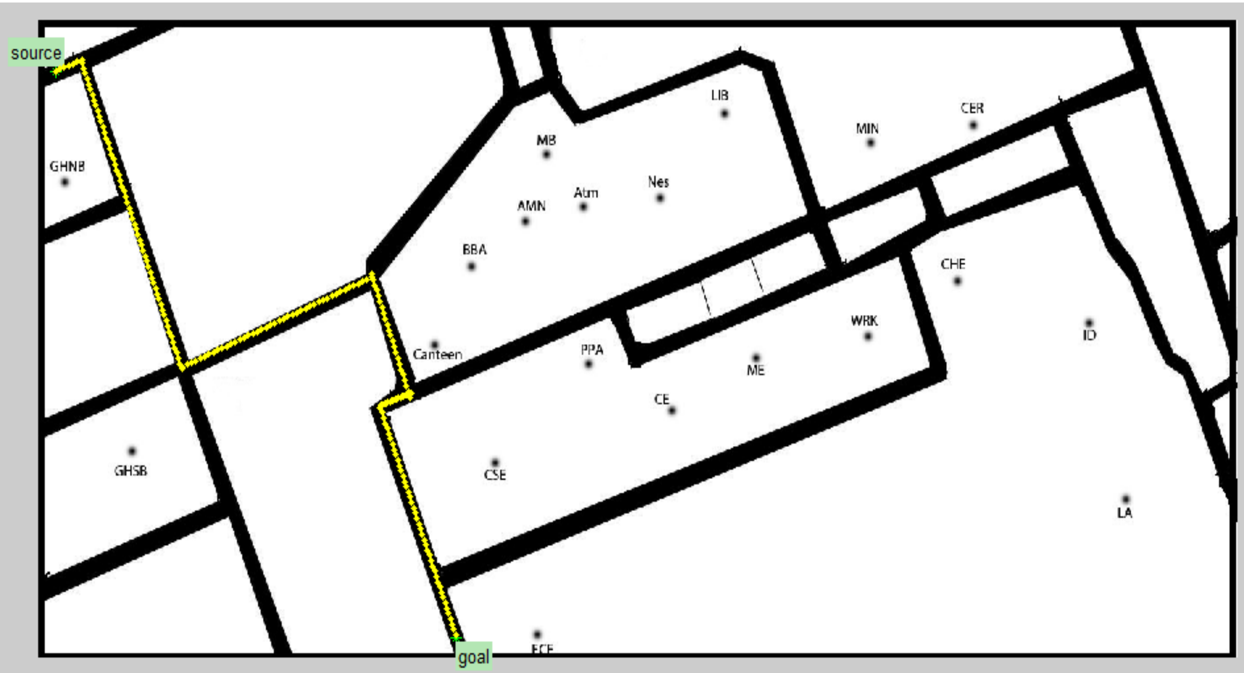

Fig.6.15: Simulation result for travelling between guest house and electrical Dept.

Table 6.1. Comparison between simulation and experimental results

case no.	**Source and goal positions**	**Time taken for simulation (in sec)**	**No. of iterations in simulation**	**Time taken in real time environment (in sec)**	**Distance between the positions in real time environment (in cm)**
1	Guest house north block to mechanical engineering Dept.	10.8249	361.4	10.88	125
2	B.B. Auditorium and ceramic Dept.	9.7851	326.1	9.83	107
3	Library to industrial design Dept.	7.2038	240.1	7.23	80
4	Guest house north block to electrical engineering Dept.	6.5102	216.3	6.52	70

Chapter 7

Conclusion

Future Scope

7.1.Conclusion

The simulation results for various choices of source and goal gives satisfactory results following the optimal path in each and every case. In the simulation process a time delay of 0.03 seconds is used to travel from one point to the adjacent point. Accordingly the vehicle is assumed to move at a speed of 33.33 units/second. We have the time taken for the robot to reach the destination in different scenarios. The robot moves at a speed of 11.5 cm/s with a delay of 0.5 seconds at 90 degree turn. Due to this delay and a few motion corrections in real time environment there seems to be a difference of 5% between the simulation and experimental values.

7.2.Future Scope

The robot that is developed in this project uses IR sensor for guidance. A better working model can be developed where in the global coordinate values can be obtained and the present location and data of the environment can be retrieved online and the action to be taken by the robot can be dictated accordingly. By further developments in the algorithm and model it can be used for automated driving cars etc. that can be very useful in day to day life.

5. REFERENCES

[1]. S.A.Stoeter and N.Papanikolopoulos, Kinematic motion model for jumping scout robots, IEEE transactions on robotics, vol. 22, no. 2, 2006, PP. 398-403.

[2]. K. Hosoda, T. Takuma, A. Nakamoto and S. Hayashi, Biped robot design powered by antagonistic pneumatic actuators for multi-modal locomotion, Robotics and autonomous systems, vol.56, no.1, 2008, PP. 46–53.

[3]. X. Wu and S. Ma, CPG-based control of serpentine locomotion of a snake-like robot, Mechatronics, vol. 20, no. 2, 2010, PP. 326–334.

[4]. Z.Y. Bayraktaroglu, Snake-like locomotion: Experimentations with a biologically inspired wheel-less snake robot, Mechanism and machine theory, vol. 44, no.3, 2009, PP.591–602.

[5]. J.L. Guzman, M. Berenguel, F. Rodriguez and S. Dormido, An interactive tool for mobile robot motion planning, Robotics and autonomous systems, vol. 56, no.5, 2008, PP.396–409.

[6]. J.C.Alexander and J.H.Maddocks, The kinematics and control of wheeled mobile robots, SRC TR 87-196, October 1987, PP. 1-15.

[7]. G.Campion, G.Bastin, and B.D.AndrCa-Novel, Structural properties and classification of kinematic and dynamic models of wheeled mobile robots, IEEE transactions on robotics and automation, vol.12, no.1, 1996, PP.47-62.

[8]. Lozano-Perez T, “Spatial planning: A configuration approach”. IEEE T. Comput. C, vol.32, no.3, 1983, PP.108-120.

[9]. R. A. Brooks, “Solving the find path problem by representing free space as generalized cones” A.I Memo No 674, Massachusetts Institute of Technology, 1982.

[10]. S. Kambhampati and L. S. Davis, “Multiresolution path planning for mobile robots,” in International Journal of Robotics Research, vol.5, no.10, 1986, PP.90-98.

[11]. M. Likhachev, D. Ferguson, G. Gordon, A. Stentz and S. Thrun “Anytime dynamic A*:An anytime, replanning algorithm”, Proceedings of the International Conference on Automated Planning and Scheduling, 2005, PP.262- 271.

[12]. O Hachour, “ Path planning of autonomous mobile robot”, International Journal of System Applied Engineering,Dev.,vol.4,no.2, 2008, PP. 178-190.

[13]. K.C. Fang and P.C. Lui, “Solving the find-path problem in mapped environments using modified A* search algorithm,” in IEEE Transactions on Systems, Man, and Cybernetics, vol.24, no.9, 1994, PP.1390-1396.

[14]. C. W. Warren, “Fast path planning method using modified A* method,” in. IEEE International Conference on Robotics and Automation, 1993, PP. 662-667.

[15]. O. Khatib, “Real time obstacle avoidance for manipulators and mobile robots,” in International Journal of Robotics Research, vol. 5, no.10, 1986, PP. 90-98.

[16]. L. Kavraki, P. Svestka, J.C. Latombe and M. Overmars, “Probabilistic roadmaps for path planning in high dimensional configuration spaces,” IEEE Transactions on Robotics and Automation, vol. 12, no. 4, 1996.

[17]. Q.R. Zhang and G.C. Gu, “Path planning based on improved binary particle swarm optimization Algorithm”, Proceedings of the IEEE International Conference on Robotics, Automation and Mechatronics, Chendu, China, 2008, PP. 462-466.

[18]. P. Raja and S. Pugazhenthi, “Optimal path planning of mobile robots: A review”, International Journal of Physical Sciences, vol.7, no.9, 2012, PP. 1314-1320.

[19]. M. Swati and B. Pankaj, “Maze Solving Algorithms For Micro Mouse”, IEEE International Conference on Signal Image Technology and Internet Based System, 2008.

[20]. R. R. Cazangi, and F. J. Von Zuben, “Immune Learning Classifier Networks: Evolving Nodes and Connections”, IEEE Congress on Evolutionary Computation, 2006.

[21]. D. L. Page, A. F. Koschan, M. A. Abidi and J. L. Overholt , “Ridge-Valley Path Planning for 3D Terrains”, Proceedings of the 2006 IEEE International Conference on Robotics and Automation, 2006.

[22]. B. B. V. L. Deepak, A. k. Jha and D. R Parhi, “Path Planning of an Autonomous Mobile Robot Using Artificial Immune System”, International Journal of Computer and Mathematical Sciences.

[23]. B. Deepak, D. R. Parhi, S. Kundu, “Innate immune based path planner of an autonomous mobile robot”, Procedia Engineering, vol. 38, PP. 2663-2671.