JSS Mahavidyapeetha

JSS SCIENCE AND TECHNOLOGY UNIVERSITY
MYSURU-570006

FINAL YEAR B.E PROJECT REPORT
2020-2021

Autonomous docking and multi-robot coordination of 3 wheel drive omnidirectional mobile robots

Submitted by

Name	USN	e-mail	Phone-no
Abhinandan K	01JST17EC002	krishnanabhinandan@gmail.com	8746917235
Pannaga Sudarshan	01JST17EC062	pannagasudarshan@gmail.com	9449191976
Anoop CC	01JST17EC015	anoopc9899@gmail.com	9742087575
Rahul Kumar A M	01JST17EC015	amrahulkumar07@gmail.com	8217273122

Submitted in partial fulfilment of the requirement of academic event in BE

Under the Guidance of

Dr. Sudarshan Patil Kulkarni
Professor
Dept of Electronics & Communication

SRI JAYACHAMARAJENDRA COLLEGE OF ENGINEERING
MYSURU-570006

JSS Mahavidyapeetha

JSS SCIENCE AND TECHNOLOGY UNIVERSITY MYSURU-570006

CERTIFICATE

Certified that the project work entitled "**Autonomous docking and multi-robot coordination of 3 wheel drive omnidirectional mobile robots**" carried out by **Abhinandan K, Pannaga Sudarshan, Anoop C C, Rahul Kumar A M**, bonafide students of Sri Jayachamarajendra College of Engineering, Mysuru in partial fulfillment for the award of Bachelor of Engineering in ELECTRONICS & COMMNUNICATION of the JSS Science And Technology University, Mysuru during the year 2020-21. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the final report.The project report has been approved as it satisfies the requirements in respect of Project work prescribed for the degree.

Dr. M N Jayaram
Professor and Head,
Dept of E & C
SJCE Mysuru

Dr. Sudarshan Patil Kulkarni
Professor,
Dept of E & C
SJCE Mysuru

Dr. S B Kivade
Principal,SJCE Mysuru

External Viva

Name of Examiners	Signature with Date
1.	....
2.	....
3.	....

Contents

1 Introduction **1**
1.1 Introduction . . . 1
1.2 Motivation . . . 3
1.3 Problem Statement . . . 4
1.4 Objectives . . . 4

2 Literature Survey **6**
2.1 Previous Research . . . 6
2.2 Summary of Literature Review . . . 16

3 Requirements **21**
3.1 Hardware Requirements . . . 21
3.2 Software requirements . . . 22

4 System Architecture and Methodology **23**
4.1 Mobile Robot architecture . . . 23
4.1.1 Platform . . . 23
4.1.2 Electronics . . . 24
4.1.3 Controller . . . 24
4.2 System Block Diagram . . . 24
4.3 Methodology . . . 25
4.3.1 Navigation subsytem . . . 25
4.3.2 ROS Actions . . . 26
4.3.3 Path Planning subsystem . . . 26
4.3.4 A* algorithm . . . 27

5 Implementation and Testing **28**
5.1 Control algorithm for waypoints following . . . 28
5.2 Path planning for a single mobile robot . . . 30
5.3 Path planning for a multi-mobile robot system . . . 30
5.4 ROS communication . . . 31
5.5 Results and Discussions . . . 33
5.5.1 Single-Waypoint Following . . . 34

5.5.2 Multi-Waypoints Following 36

6 Conclusion and Future scope **46**
6.1 Conclusion 46
6.2 Future scope 46

Bibliography **47**

List of Figures

4.1 Top view of the robot 23
4.2 Mobile Robot System 24
4.3 Multi Waypoint following 25

5.1 Go to Pose algorithm control architecture 29
5.2 Block Diagram 29
5.3 ROS communication 31
5.4 ROS Action snippets 33
5.5 Straight line trajectory 34
5.6 Pose error Vs Time 34
5.7 θ_{robot} Vs Time 35
5.8 Wheel velocity profiles 36
5.9 Square trajectory 37
5.10 Pose error response 37
5.11 Theta response 38
5.12 Angular wheel velocity profiles during square trajectory 38
5.13 Hexagram trajectory 39
5.14 Pose error response 39
5.15 Theta response 40
5.16 Angular wheel velocity profiles during hexagram trajectory 40
5.17 Circular trajectory 41
5.18 Pose error response 42
5.19 Theta response 42
5.20 Angular wheel velocity profile during circular trajectory 43
5.21 Single robot path planning 44
5.22 Multi robot path planning 1 45
5.23 Multi robot path planning 2 45

List of Tables

5.1 Performance analysis 43

Chapter 1

Introduction

1.1 Introduction

The robotics industry is growing at a tremendous rate, and the diversity in which it is setting its foot is spread across a variety of intra-domain fields such as factory automation, dangerous environment detection, office automation, hospital, entertainment, space exploration, farm automation, military, education and security system and so on [b12].

The reason why there is such a trend in the robotics society is because of the fact that the entire world is moving towards automation of almost every task that a human being once performed. These tasks are not only restricted to a repetitive set of instructions such as the manual production lines, conveyor belts in the earlier years of technology, probably in the 19th century, but also to a very different area where we witness the fact that machines learn, in the 21st century. This is all possible due to the development of the technology and the way we tackle the problems that we face as human beings. Therefore, there has been an increase in demand for automation of machines in cross-domain industries, and the development of autonomous systems and robots that can be deployed in the society, for betterment of our lives. However, the extent to which the automation is done is still a challenge. Therefore, the problems related to autonomous robot systems have shown to be very attractive for researchers of diverse areas because of their complexity and challenging issues.

All mobile robots can be distinguished from other robots in general, by their ability to function autonomously to a certain level, their ability to make decisions and follow the same with an intelligence level that is adequate to respond and react when thrown into a situation at it. Every mobile robot must include input data, a technique to process the data, and a methodology to take actions or decisions in order to respond to a changing environment. These mobile robots are replacing humans in many fields due to their abilities. Their navigation requires the knowledge of planning algorithms and information theory deployed in the society for the betterment of our lives. As there are many concepts that are being turned into reality because of the tremendous advancements of eclectic nature in technological aspects.

Omnidirectional vehicles are widely used in a large number of terrain based applications, allowing movements in every possible direction, that is latitudinal and longitudinal, where the extra mobility is an important advantage in many ways. The robot is able to move from point A to point B with independent linear and angular velocities. Subsequently, this contributes to minimizing the time to react, the number of maneuvers it has to make and also the run time of the job or the task is significantly curtailed, thereby making the robot much quicker and efficient.

One of the versatile techniques for navigating the robot on a flat surface is to utilize omni directional wheels. With its capability of simultaneous translation and rotation movements, it attracts many researchers to build the mobile robot based on an omni-directional wheel platform. As a result, plenty of omni-directional wheel based platforms are under research which includes three, four, and six wheels. However, a minimum of three wheels is necessary to follow a given planar trajectory or path. Since the need for development of holonomic systems is very much essential for a large number of applications, it basically eases out the maneuverability process, as the total number of degrees of freedom (DoF) is equal to the number of controlled degrees of freedom. In order to provide an excellent maneuverability to a mobile robot in narrow or tight spaces, omni-directional wheeled solutions are optimum. On the other hand, a non-holonomic design of the system would drag down the system's performance, ability and efficiency when it comes to the motion and maneuverability of the robot in the same confined workspace.

The goal is to develop a robust algorithm and apply it to an omni-directional mobile robot platform, which has the ability to solve problems that may potentially arise in cases where there is a requirement of taking up an almost ideal trajectory for its traversal from point A to point B whilst maintaining it's pose as per the user's requirement. This project focuses on a very interesting and popular domain in robotics called the "Swarm Robotics". To grant intelligence for a multi-robot system, or in other words, known as the swarm of many individual robots, "Swarm intelligence" must be utilized. The term "Swarm Intelligence" refers to sophisticated collective behavior that can emerge from the combination of many simple individuals, each operating autonomously [21]. According to Cao et al. [22], swarm intelligence is "a property of systems of non-intelligent robots exhibiting collectively intelligent behavior". The application of swarm intelligence principles to collective robotics can be termed "Swarm-Robotics" [21] .

The project falls under the umbrella of autonomous docking and maneuvering of a multi-robot system which is specific to a warehouse implementation scenario. The warehouse as we know contains large heavy operated machinery that man has built for easement in the operation, such as huge indoor cranes, lifts, stationary arm robots, forklifts, etc, in order to transfer goods or shift the heavy cargo from one location to another within the warehouse. The automation in such an environment is very much necessary since the companies are putting human lives at stake. Also, there has been a large number of autonomous systems that are developed over the years to enhance the quality of such tasks, ease out human efforts, and probably even eliminate them.

Therefore, we propose to deploy a multi-robot system within the warehouse, to fulfill the maneuvering of goods from point A to point B within the warehouse.

The navigation of such a multi-robot system or also known as the swarm of robots depends on various algorithms used in the process. In this project, we aim to propose a solution or rather a proof of concept for "Autonomous docking and Coordination of swarm robots" that uses universal omni directional wheels for maneuvering around the warehouse. The multi-robot system has a huge versatility, that is, it can provide the heterogeneity of structures and functions required to undertake different missions in unknown environmental conditions. We propose an autonomous navigation of a system of 'Three Wheeled Omni Drive Mobile Robots' in the pre-defined workspace. This system of mobile robots can significantly curtail the amount of effort that was put into moving consignments across the warehouse and such self-assembly of indvidual mobile robots can enhance the efficiency of a group of autonomous cooperating robots in several different contexts as well [20].

1.2 Motivation

In the universe, a lot of things are unknown, plethora of questions unresolved, unanswered, extremely confusing, etc. The way nature conveys us solutions to a humongous amount of problems is extremely astonishing and interesting when inspected. The concept of a multi-robot system, in other words "Swarm robotics" and its related concept of, the swarm intelligence can be taken from a biological perspective.

Such meticulous tasks are basically day to day activities for many biological creatures on this planet. This is inspired by understanding of the decentralized mechanisms that underlie the organization of natural swarms found in nature such as bees, ants, fish, wolves, birds, and not to mention, human beings. Social insects provide one of the best known examples of biological self organized behavior. By means of local and limited communication, they are able to accomplish impressive behavioral feats: maintaining the health of the colony, caring for their young, responding to invasion and so on [23].

Nowadays, robots are capable of much more high graded tasks compared to what it was a decade ago. They are deployed in factory settings, surgery operating rooms, airport cargo warehouses, warehouses for different industries, institutions,etc. Often in such settings mobile robots require a capability to operate in a confined space[6]. By rigorous analysis of research papers and real world applications, we come to know that a vast majority of modern day mobile robots are implemented or constructed using wheels. Considering a range of options available for number of wheels, their type and configuration within a mobile robot platform base, the project aims to select the most suitable one.

In order to provide a good maneuverability of a mobile robot in a narrow space, omnidirectional wheeled solutions are considered. Since the need for development of holonomic systems is very much essential in swarm robotics' technology, it basically eases out the maneuverability process, since the number of controlled degrees of freedom (DoF) is equal to the total number of degrees of freedom [3]. On the other hand, a non-holonomic design of the system, would drag down the system's performance, ability and efficiency when it comes to the motion and maneuverability of the robot in the same confined workspace, or in our case the warehouse.

The need of the hour is to develop a team of autonomous robots or a multi-robot with intelligence, which have the ability to solve problems that may potentially arise in cases where there is a requirement of transporting large objects within a confined space. This is an important issue since such swarm robot teams would have the potential to solve many use cases in our day to day activities, be it in transportation or industrial scenarios, mainly implementing in warehouses.

Therefore, to provide the best possible solution, given the environment setting of the robot's workspace, the concept of multi-robot system seemed to solve it much more efficiently when compared to the amount of resources utilized by a single traditional robot. Also, when it comes to maneuverability around the workspace environment, developing a holonomic system seems to be optimum in such scenarios where the robot has to corner tight spaces in its environment. Considering all the above facts and research, choosing this project seemed to give us a real push towards making this project come alive.

1.3 Problem Statement

The day to day activities that are carried out in a warehouse environment are dangerous, timely organized, and the confinements are sometimes large that their maneuvering across the warehouse must be done carefully in order to protect them, and prevent any sort of transportational damage within the warehouse. These tasks are extremely done well when things are automated. When things are automated, it greatly reduces the manual workload laid on the workers, heavy machine operators, etc. Therefore, this project aims to deploy an autonomous mobile-robot system within the warehouse work environment, which is a subset of holonomic systems, and therefore uses universal omnidirectional wheels for maneuvering in tight spaces usually found in warehouse environments.The demand for efficient utilization of available resources for the multi-robot system, and how the productivity of homogeneous robots scales with group size,is addressed here and the decisions that the multi-robot model make are based on only local sensing and the communication established as a result, will lead naturally to swarms that will scale a to very large number of such individual robots. The novel autonomous docking principle will make a huge impact when the goods or the payloads exceed the limits or carrying capacity of an individual mobile-robot. Therefore, this project aims at developing a proof of concept for "Swarm Robotics", along with its applications, in a scenario which is very much crucial in industry level, but at a smaller scale.

1.4 Objectives

The Objectives of this project can be divided into three distinct parts namely Soft Alignment, Interlocking and Multi robot coordinated movement. These Objectives are put forth based on the assumption that the mobile robots have achieved stage 1 maneuver [24] are in close vicinity based on the objectives covered in [24]. Each of the above mentioned objectives are explicated as follows.

1. **Soft Alignment :** Eliminate the errors caused form the stage 1 maneuver Align the Mobile robots either laterally or longitudinally based on the given input.

2. **Interlocking :** Perform interlocking or docking of two mobile robots either longitudinally or laterally such that the robots can move together.

3. **Multi Robot Coordination :** Once interlocking between the individual robots is achieved, the individual interlocked robots, together act as a single bigger robot (a rigid body) with a larger load carrying capacity, and perform further coordinated maneuvers as instructed by the user.

These are the currently defined objectives for the project. Due to various constraints or systemic updates there were invariable shifts or deviation to the project's objectives by fractional amounts. However, the overall defined objectives which remain quintessential for the project have been implemented successfully.

Chapter 2

Literature Survey

2.1 Previous Research

Research papers, articles and journals were studied and the literature survey was conducted to draw the conclusions.

[1] A Low-Cost Laser Distance Sensor
In this paper, it briefly explains that many indoor robotics systems use laser rangefinders as their primary sensor for mapping, localization, and obstacle avoidance. The cost and power of such systems is a major roadblock to the deployment of low-cost, efficient consumer robot platforms for use. This paper describes a compact, planar laser distance sensor (LDS) that has capabilities comparable to current laser scanners: 3cm accuracy out of 6cm, 10 Hz acquisition, and 1 degree resolution over a full 360 degree scan. The paper also discusses the various systems that involve in "Single-Point Distance Module", which relies on an innovative laser point sensor module that works on triangulation principle.It states out clearly that it is slightly larger than the current IR distance sensors such as Sharp IR devices, but has much better accuracy and speed. There were many challenges in transitioning from proof-of-concept sensors to a consumer product. The advantages were the high rigidity of the laser-to-imager interface and the rapid subpixel localization of the laser dot, all using standard low-cost optics and electronics.

[2] Autonomous Docking Based on Infrared System for Electric Vehicle Charging in Urban Areas
An implementation of docking in real life scale for electric vehicle charging is discussed here. According to the authors, there are still some problems to be solved related to energy storage, electric charging and autonomy. This paper presents an autonomous docking system for electric vehicles recharging based on an embarked infrared camera performing infrared beacons detection installed in the infrastructure. The onboard autonomous docking algorithm implemented in this paper consists of an infrared camera for the localization of the vehicle in the reference frame of the charging station.

The docking station is equipped with eight infrared LEDs, their positions being precisely known in the station referential. The paper states that the high number of LEDs was chosen to allow the detection of several patterns, in case one or several lights were obstructed or failing. The experiments conducted in this paper, showed that six LEDs were enough in practice to accurately determine the vehicle position with regard to its docking station. The work done as discussed, is a control architecture for autonomous docking systems, based on an embedded perception system in an autonomous electric vehicle and a recharging station for urban parking areas, is presented. The information from the sensors had been processed and filtered and then sent to the control stage for automatic docking of the vehicle. The proposed work relies solely on the information from the camera on board the vehicle.

[3] Comparative Analysis of Mobile Robot Wheels Design

The paper analyses the design sector for wheeled mobile robot platforms.It clearly highlights that each platform is designed for a set of specific tasks and thus is supposed to work in previously known general condition of its environment. The work done, as per the authors, is that they had compared different types of mobile robot wheels, including conventional wheels, universal omnidirectional wheels, Mecanum wheels, caster wheels, and steering standard wheels, and analyzed the best scenario of design application. The paper explores a question of a wheeled mobile robot base design from a locomotion point of view tackling the choice of wheels and their configuration. The detailed information about the omni wheel as per the paper was that the basic idea of an omni wheel is a combination of a main active wheel and passive freely rotating rollers.

The active wheel and the rollers have their own rotation axes and in the case of universal wheels, axes of passive rollers are orthogonal to a main wheel axis. While an active wheel is rotating in clockwise or counterclockwise direction with respect to its rotational axis, combining active rotation of several active wheels with passively rotating rollers allows supporting locomotion almost in any direction. The paper compares certain types of wheels to facilitate the question of selecting the most suitable wheel type and configuration for constructing a mobile platform. However, omni wheels have complex manufacture design and high sensitivity to locomotion surface conditions. Therefore, the first step of designing and selecting the wheels is awareness of further robots workspace and application area.

[4] Design Principles for Robot Inclusive Spaces : A Case Study with Roomba*

Research focuses on service robots that deals with applications related to healthcare, logistics, residential, search and rescue since they are gaining significant momentum in recent years .The paper proposes a new philosophy of robot inclusive spaces, a cross disciplinary approach that brings together roboticists, architects and designers to solve numerous unsettled research problems in robotics community through design of inclusive interior spaces for robots where the latter live and operate. In conclusion, it states that given the current limitations of service robots in performing reliable autonomous works in dynamic human environments, it suggests here "design for robot" approach, to complement the conventional "designing robot" approach. The design principles presented here were illustrative rather than exhaustive and have been developed based on a case study and framed within the service robotics literature.

[5] Design, Modeling and Control of an Omni-Directional Mobile Robot

The structures of mobile robots with omni-directional wheels were explored here. The paper speaks out loud that one of the main issues of a mobile robot is to move in tight areas, to avoid obstacles, finding its way to the next location. These capabilities mainly depend on the wheels design. It highlights that an omni-directional drive mechanism is very attractive because it guarantees a very good mobility in such cases. This paper provides some information about the mechanical design of an omni-directional robot, as well as about its control. The development of an omni-directional vehicle was pursued to further prove the effectiveness and maneuverability of this type of architecture. Such a vehicle with four Mecanum wheels provides omni-directional movement without needing a conventional steering system. Two wheel designs were used for experimental tests in order to improve the wheel traction force.

[6] Dynamic Model With Slip for Wheeled Omnidirectional Robots

A dynamic model is presented for omnidirectional wheeled mobile robots, including wheel/motion surface slip. Dynamic simulation examples were presented to demonstrate omnidirectional motion with slip. After developing an improved friction model, compared to their initial model, the simulation results agreed well with experimentally-measured trajectory data with slip.The paper clearly highlights that not only high robot velocity and acceleration governed the resulting slipping motion, but also the rigid material existing in the discontinuities between omnidirectional wheel rollers played an equally important role in determining omnidirectional mobile robot dynamic slip motion, even at low rates and accelerations.

This paper has presented a dynamic model for omnidirectional wheeled mobile robots and vehicles, considering slipping between the wheels and motion surface. It derived the dynamics model, experimentally measured the friction coefficients, and validated the friction model by experimentally measuring the maximum force causing slip at various robot orientations. Simulation examples were presented to demonstrate slipping motion; the initial friction model results did not agree with experimental trajectory data.

Therefore, an improved friction model was developed, considering the rigid material in the discontinuities between omnidirectional wheel rollers. With this improved friction model, the simulation agreed well with the experimental data. Two motion surfaces, paper and carpet, were used in simulation and experiments, with different friction properties. A pure translational motion was commanded in simulation and experiment; simulations show that slipping for translational motions is not as severe, due to robot symmetry. With zero commanded rotational motion, the robot experienced undesirable slip in rotational motion. This paper is pertinent to any omnidirectional mobile robot design with or without discontinuity between rollers. Since the objective was to model and understand the sliding dynamics problem, this paper does not focus on real-time control.

[7] Dynamical Models for Omni-directional Robots with 3 and 4 Wheels

A discussion on 3 and 4 wheeled omni-directional robots was done here. The paper emphasises that omni-directional robots are becoming more and more common in recent robotic applications. It says that frequent applications include but are not limited to robotic competitions and service robotics. The paper's main goal was to find a precise dynamical model in order to predict the robot behavior. Models were found for two real world omni-directional robot configurations and their parameters were estimated using a prototype that can have 3 or 4 wheels. Simulations and experimental runs were presented in order to validate the presented work. Friction coefficients are most likely dependent on robot and wheels construction and also on the weight of the robot. The model was derived assuming no wheel slip as in most standard robotic applications. Observing estimated model parameters, the four wheel robot was found to have higher friction coefficients in certain directions.

[8] Experimental Analysis of Mecanum wheel and Omni wheel

Delineation of Omni and Mecanum wheels was done in this paper. It includes a few considerable points such as omni directional robotic platforms have vast advantages over a conventional design in terms of mobility in congested environments. They are capable of easily performing tasks in environments congested with static and dynamic obstacles and narrow aisles.

These environments are commonly found in factory workshops, offices, warehouses, hospitals and elderly care facilities. The paper discusses the qualitative view of the system's mobility performance. The forward and reverse motion seemed to be acceptable but did not utilize any function of the mecanum wheels and Omni wheels. Likewise with rotational motion, the system performed as would be expected of a standard differential drive platform.

The translational motion in x- axis, however, was not acceptable as the platform would tend to wander in the y-direction when attempting to traverse sideways. This paper talks about the strength of these wheels which have enhanced maneuverability of the mobile robot that needs extreme maneuverability in a congested environment. The paper also speaks about the design and development of an Omnidirectional platform using mechatronics system and Omni directional wheel to implement intelligent behavior and maneuvers, with the help of a microcontroller interface.

[9] Front and Back Movement Analysis of a Triangle-Structured Three-Wheeled Omnidirectional Mobile Robot by Varying the Angles between Two Selected Wheels

This paper focuses primarily on front and back movement, to analyse the square and triangle structured omnidirectional robot movements. It presents a design of a unique device of Angle Variable Chassis (AVC) for linear movement analysis of a three-wheeled omnidirectional mobile robot (TWOMR), at various angles (Θ) between the wheels. Basic mobility algorithm was developed by varying the angles between the two selected omnidirectional wheels in TWOMR. The experiment was carried out by varying the angles (Θ= 30° , 45° , 60° , 90° , and 120°) between the two selected omni wheels and analysing the movement of TWOMR in forward direction and reverse direction on a smooth cement surface.

A practical implementation of an omnidirectional mobile robot was the main focus of this work. A unique device of Angle Variable Chassis (AVC) was designed, and a three-wheeled omnidirectional mobile robot (TWOMR) was implemented and its kinematics model was described. There were some errors that originated from several sources such as the nonsteady condition in omnidirectional wheels during motion, the effect of friction in the working model, and the effect of omni wheel rollers on omni wheels. Irrespective of these errors, TWOMR showed effective movement at angles 120° and 60° . Hence, these two angles were found to be the optimum angles for the movement of TWOMR, as the deflections are minimum and negligible. The TWOMR was found to have greater speed at 60° compared to 120°. In this paper, only the linear movement analysis has been done.

[10] Laser Sensors for Displacement, Distance and Position

This paper discusses how laser sensors can be used to measure distances to objects and their related parameters (displacements, position, surface profiles and velocities).It lays out clear facts such as laser sensors are based on many different optical techniques, such as triangulation, time-of-flight, confocal and interferometric sensors. As laser sensor technology has improved, the size and cost of sensors have decreased, which has led to the widespread use of laser sensors in many areas. In addition to traditional manufacturing industry applications, laser sensors are increasingly used in robotics, surveillance, autonomous driving and biomedical areas. This paper outlines some of the recent efforts made towards laser sensors for displacement, distance and position.

[11] Motion Analysis of a Mobile Robot with Three Omni-Directional Wheels

This paper demonstrates the kinematics motion analysis of the three wheeled omni-directional mobile robot. The mobile robot is equipped with the three omni-directional wheels arranged at the vertices of an equilateral triangle and the wheel axles are aligned with the lines from the center of the equilateral triangle to each of the three wheels. Model of this omni-directional wheeled robot was developed and shows the different trajectory motion of the mobile robot by controlling the DC gear motor with different combinations.

A model was derived assuming no wheel slip as. By controlling motion of the three omni-directional wheels, immediate forward, backward, sideways and rotational motion, in other words, omni-directional motion was generated successfully. A mathematical kinematic model of the robot was done and velocity vector analysis on each omni-directional wheel was performed as well. A low-level program was created to simplify access to basic functionality of the robot.The paper also claims that it can be easily built up with high-level algorithms for localization, trajectory planning and tracking. However, the paper also discusses the fact that the kinematical and dynamical models are needed in this work and that can be used to study the limitations of the mechanical configuration and future enhancements of both mechanical configuration and controlling the device. This work will enable effective full comparison of three wheeled systems.

[12] Motion Planning of Multi-docking System for Intelligent Mobile Robots

The paper talks about development of a multi-docking system with a mobile robot and three docking stations. The docking structure was designed with one active degree of freedom and two passive degrees of freedom. The article focuses on the searching process using multiple sensors and a laser range finder for the mobile robot. The laser range finder searches the landmark of each docking station, and guides the mobile robot moving to the assigned docking station. In the experimental results, the power of the mobile robot is under the threshold value.

The paper seems to have developed a multi-docking system that has been integrated in the mobile robot. First, it discusses the design of a power detection module to measure the values of current and voltage, and transmit these measurement values to the main controller of the mobile robot, and a design of the power detection system for the docking station to measure the charging current. It also proposed a docking process to enhance the successful rate. When the power of the mobile robot is under the threshold value, the mobile robot uses a laser range finder to search the nearest and free docking station, and moves to the assigned docking station. The nearest docking station if it is found busy, the mobile robot can select the other station. In the docking process, the mobile robot communicates with the assigned docking station at any time, and knows the charging status of the recharging process.

[13] Object Transportation by Multiple Mobile robots controlled by Attractor dynamics: theory and implementation

With respect to object transportation or maneuvering by multi-robot system, this paper focuses on not just theory, but also implementation of the same. Here, dynamical systems theory is used as a theoretical language and tool to design a distributed control architecture for teams of mobile robots, that must transport a large object and simultaneously avoid collisions with (either static or dynamic) obstacles. The paper demonstrates in simulations and implementations in real robots that it is possible to simplify the architectures presented in previous work and to extend the approach to teams of n robots. The robots have no prior knowledge of the environment. The motion of each robot is controlled by a time series of asymptotically stable states. The attractor dynamics permits the integration of information from various sources in a graded manner. As a result, the robots show a strikingly smooth and stable team behaviour.

The complete dynamics architecture was implemented and evaluated on a team of three robots. In the implementation, the dynamics of heading direction and path velocity were integrated numerically using the forward Euler method. Sensory information is acquired once per computation cycle. The cycle (step) time was measured and was found to be approximately 50 ms for the leader and 60 ms for the helpers. In this paper, non-linear attractor dynamics was used as a tool to design a distributed control architecture that enables a team of n robots to transport a large object. The most striking feature of the robots is their smooth behaviour. This is a direct consequence of the attractor dynamics that permits information from various sources to affect in graded manner the overt behaviour. It was assumed that the robots have no prior knowledge of the environment.

The demonstrated robotic system has an obvious application. The global behavior was stable and trajectories were smooth. Very important, the ability to avoid collisions with either static or dynamic obstacles has been demonstrated.

[14] Scalable Multi-Robot Formations Using Local Sensing and Communication

In this paper, a solution to multi-robot formation is presented. All robots use only local sensing and local communication as the additional tool for the whole system performance. It assumes the solution to be local, robust to the size of a multi-robot team, stable in the case of occurrence of obstacles and able to support many formation shapes. Local inter-robot communication enables the group of robots to organize into formation without need of predetermined positions in the formation. The approach for scalable multi-robot formations using only local sensing and communication was presented. The key extension to the usual behavior-based approaches, according to the authors, was the application of so-called social roles representing positions in the formation and the use of local communication as the additional tool improving the whole system performance.

The author has described that local communication enables robots to provide information about actual formation occupation. The paper states that this enables the optimization and prevents from the formation shape violation. Proposed approach is robust for change in the size of the robot team. With the use of local communication, any new robot can join any existing organization keeping the formation shape clean. The redefinition of social roles and positions is not required. Organization of the robots grows after inter-robot formation occupation negotiation.

[15] Selection of Wheels in Robotics

This paper vividly describes and provides details about the wheels and the significance of selecting the optimum one for the respective robots which are required to compete in the challenge. The wheels were selected on the factors and methods based on the degrees of freedom, weight distribution, wheel geometry, field material and wheel material. The system defined in this paper, is cost effective and rugged. This system can be implemented for various types of Robots and provides the perfect fit in terms of mobility, locomotion and ideal stability. In this paper, a particular manual robot weighing 26 kilograms for the "Robocon Competition" was developed, which used four Omni wheels at an angle of 45 degrees to achieve desired direction of motion. With a calculated combination of the wheel alignment and the direction of rotation of the motor shaft, it achieved 8 directions of motion.

[16] Vision based Autonomous Docking and Recharging system for Mobile robot in Warehouse environment

This paper proposes a vision-based autonomous docking and recharging system that can operate in a warehouse environment. In addition, the docking mechanism applies two docking processes: searching zone process and approaching zone process. The experiment in the warehouse environment showed the docking success rate is around 97.33 percentage with high accuracy. Different start positions and approaching points may also influence the docking duration. The angle between the start point and charging station is negatively related to average duration time and the distance between approaching point and charging station is also negatively related to average duration. In this paper, a vision-based autonomous docking and recharging system for mobile robots working in a warehouse environment is presented. In conclusion, an AprilTag is used to develop the vision-based autonomous docking and recharging system for mobile robots working in a warehouse environment.This paper also states that through experiment, the overall success rate of this docking and charging system was convincing. The robot can dock successfully despite noisy images, obstacles in the path and can approach to the docking place in different conditions. However, as the light condition changes, the shielding of tags can interrupt the docking processes, which needs further research.

[17] Flexible Docking Mechanism Using Combination of Magnetic Force with error compensation capability

This paper presents a new docking mechanism with a localization error-compensation capability. The proposed mechanism uses the combination of mechanical structure and magnetic forces between the docking connectors. It is a structure to improve the allowance ranges of lateral and directional docking errors, in which the robot is able to dock into the docking station. Consequently, this mechanism reduces dependency of a robot control and allows easy docking with only mechanical configuration. In this paper, the superiority of the proposed mechanism is verified with experimental results. The paper claims that in experiments, the proposed mechanism showed a sufficiently high ability to compensate for the approaching errors of the mobile robot. Since robot technology progresses and the robot market grows, the robot will perform an increasingly wide range of important tasks, which require the ability to operate continuously. Furthermore, security and service robots in public institutions need to be able to recharge and return to their tasks autonomously without interruption. The proposed system is based on modularization with a simple and practical mechanism allowing easy adaptation to other mobile robots.

[18] An Extensive Review of Research in Swarm Robotics

The brilliant research conducted by the authors of this paper grabs the field our project falls in, which is “Swarm robotics”. According to the authors, swarm robotics is a new approach to the coordination of multi-robot systems which consist of large numbers of relatively simple robots which takes its inspiration from social insects. The most remarkable characteristic of swarm robots are the ability to work cooperatively to achieve a common goal. In this paper, classification of existing researches, problems and algorithms aroused in the study of swarm robotics are presented. The existing studies are classified into major areas and relevant sub-categories in the major areas.

In conclusion, most of the research conducted was based on the biological inspirations adopted from the behaviors of ants, bees and birds. Implicit communication seems to give more robustness in the communication architecture of swarm robotics. According to the authors, with respect to mapping and localization is concerned, work is still being carried out to fine tune the problems faced in this domain. In object transportation and manipulation, caging is preferred over the available methods as the constraints in the domain can be reduced and kept simple.

A lot of new heuristics and algorithms were introduced to solve the problems in this domain. In the learning domain, reinforcement learning (RL) was given much interest by the researchers. In the task allocation domain, heterogeneous and homogeneous systems were widely discussed.

[19] Autonomous Navigation System Applied to Collective Robotics with Ant-Inspired Communication

This paper clearly emphasizes the nature inspired deductions for our project. According to the paper, several approaches have already been tried in multirobot systems, but the bio-inspired ones are the most frequent. This paper proposes to augment an autonomous navigation system based on learning classifier systems for use in collective robotics, introducing an inter-robot communication mechanism inspired by ant stigmergy, with each robot acting independently and cooperatively. The navigation system has no innate basic behavior and all knowledge necessary to compose the decision making artifact is evolved as a function of the environmental feedback only, during navigation. Experiments are performed in simulation, with comparative results indicating that the presence of the pheromone trails is responsible for significant improvements in the capture rate and in the length of the route adopted by each robot.

In this work, a mechanism for collective robot navigation with indirect communication was proposed, inspired by pheromone trails in ant systems. Each robot was controlled by the autonomous navigation system (ANS), with an additional sensor for detecting the kind and level of pheromone present at the current location of the robot, and also an additional actuator to deposit pheromone following very simple rules. The robots have no initial knowledge and learning is accomplished by means of a classifier system composed of if-then rules with a particular configuration for the antecedent and consequent parts.

Three sets of experiments were performed. When comparing the ANS with and without the use of pheromone trails, all experiments guide to improvements caused by the presence of pheromone. Mainly in terms of target capturing, the gain was very significant. Related to trajectory optimization, the results indicate that the system is able to reduce the distances traveled between targets. Although it can be considered just as a side effect, the system capability of minimizing distances is promising, especially regarding that the ANS was not designed for this purpose.

[20] Cooperation through self-assembly in multi-robot systems

This paper illustrates the methods and results of two sets of experiments in which a group of mobile robots, called s-bots, are required to physically connect to each other - i.e., to self-assemble - to cope with environmental conditions that prevent them from carrying out their task individually. The experimental work illustrated in this paper summarises the research activities carried out by the authors with robots capable of physically connecting to each other -i.e., the s-bots.

The results of the first set of experiments proves that their work represents a sensible step forward with respect to the state of the art in the design of controllers for self-assembling robots, in particular when focused on (a) the number of robots involved in self-assembly, (b) the reliability of the system, (c) the speed with which the agents generate the assembled structure, and (d) the capability of the assembled structures to coordinate in order to transport a heavy object at high speed. In the second set of experiments, they started considering self-assembly within a framework in which the mechanisms for sensory-motor coordination are combined for decision making structures to allow the s-bots to decide when it is time to gather and pursue collective strategies. The aim of this work was to enhance the adaptiveness of a group of self-assembling robots by reducing to the minimum the priori assumptions concerning the nature of the control mechanisms that, by working on the agent's perceptual evidence, guide a multi-robot system in an intrinsically complex scenario. The results show that their methodology is promising: the evolved controllers are capable of displaying individual and collective obstacle avoidance, individual and collective photo axis, aggregation and self-assembly.

2.2 Summary of Literature Review

Sl no.	Reference	Work done	Drawbacks
1.	[1]	The paper describes a technique that relies on an innovative laser point sensor module that works on triangulation principle, called as	The accuracy of detection is 3cm out of 6cm, which is around 50, and uses expensive electronic devices such as cameras, which becomes an overkill for our
2.	[2]	The on board autonomous docking algorithm implemented in this paper consists of an infrared camera for the localization of the vehicle in the reference frame of the charging station. The paper states that the high number of LED's was chosen to allow the detection of several patterns, in case one or several lights were obstructed or failing.	The overall circuit complexity is higher, and it utilizes eight infrared LED's, to increase precision of the docking. The proposed system is for commercial electric vehicles and hence the work solely relies on the information from the camera on board the vehicle
3.	[3]	The paper describes various designs of wheeled mobile robot platforms, discusses briefly a comparative analysis done on mobile robot wheel design. The paper explores a question of a wheeled mobile robot base design from a locomotion point of view tackling the choice of wheels and their configuration	The paper however may seem very much promising of what we have chosen as the suitable wheel design, it speaks about the use of meccanum wheels for heavy acceleration and load carrying capacity, which is expensive and not required for our vision

4.	[4]	This paper proposes a new philosophy of robot inclusive spaces, a cross disciplinary approach that brings together experts from various backgrounds.	It talks more about the interior spaces that are somewhat different from the actual warehouse that we put forth in our project, and mainly focuses on the dynamic human interaction which is not so often in warehouse scenarios
5.	[5]	This paper provides a brief information about the mechanical design of an omnidirectional robot, as well as about its control.	One of the drawbacks is that only two wheel designs were used for experimental tests in order to improve the wheel traction force.
6.	[6]	In this paper, a dynamic model for omni wheel robots is presented considering the concept of sliping between the wheels and motion surface	The paper itself includes that the initial friction model results did not agree with experimental trajectory data. And more importantly, it is subjective to the material of the carpet, the material of the wheels, the quality of the wheels which were prime participants of the experiment.
7.	[7]	This paper discusses the dynamic models for omnidirectional robots with three and four wheels. The main goal was to find a precise dynamical model in order to predict the robot behaviour	The model was derived assuming no wheel slip, which however can be considered to only a certain extent. The model developed lacked mechanical suspension to even wheel pressure on the ground
8.	[8]	The paper discusses a qualitative view of the system's mobility performance when exposed to mecanum wheels and omni wheels.	The translational motion in x-axis was not acceptable as the platform developed would tend to wander in the y-direction when attempting to traverse sideways.

9.	[9]	The work done in this paper is an analysis of a triangle structured three wheeled omnidirectional robot by varying the angles between two selected wheels	The paper however focuses only on front and back movement of the robot. Only linear movement analysis has been done
10.	[10]	This paper discusses how laser sensors can be used to measure distance to objects and their related parameters such as displacements, position, surface profiles, velocities, etc.	The paper doesn't discuss the implementation of object detection by using a single beam of light ray, but instead uses a slightly higher level approach, which is not required in our case.
11.	[11]	This paper demonstrates the kinematics motion analysis of the three wheeled omnidirectional mobile robot.	However, the paper also discusses the fact that the kinematical and dynamical models are needed in this work and that can be used to study the limitations of the mechanical configuration and future enhancements of both mechanical configuration and controlling the device
12.	[12]	The paper describes the development of a multi-docking system that has been integrated on a mobile robot.	The implementation carried out here, is based on laser range finders, which are expensive, and is an overkill for our requirement, and is solely based on a threshold value.

13.	[13]	The paper demonstrates in simulations and implementations in real robots that it is possible to simplify the architectures presented in previous work and to extend the approach to teams of n robots.	It was assumed that the robots do not have any prior knowledge of the environment or the workspace, which is the opposite to what we wish to develop
14.	[14]	A solution of a multi-robot formation task is presented. The paper states that the local communication enables the optimization and prevents from the formation shape violation.	All the robots use only local sensing and local communication as the additional tool for the whole system performance, that is it doesn't make use of a ROS master as we have proposed.
15.	[15]	A brief explanation about the significance of selecting the optimum wheels in robotics is discussed. The wheels for the prototype developed here were selected based on the factors and methods based on the degrees of freedom, weight distribution, wheel geometry, field material and wheel material.	The paper is extremely specific to a manual robot weighing 26 kilograms for the "Robocon competition", and uses four omni wheels which is expensive, and not required for our scenario, and also doesn't propose a generic solution
16.	[16]	This paper proposes a vision-based autonomous docking and recharging system, that can operate in a warehouse environment. The paper claims put forth a novel vision based system.	The docking mechanism relies on an on board camera, which is power consuming, and frankly not required, the complexity is high, and the results are subjected to light conditions, and also the shielding tags can interrupt the docking process, which needs further research

17.	[17]	A flexible docking mechanism that uses a combination of magnetic force with error compensation capability is discussed here. The proposed system is based on a modularization mechanism that is adaptable to other mobile robots	The use of magnetic force is self error correcting methodology, it doesn't involve. However, the magnetic docking cannot be implemented in a warehouse scenario, since it may or may not interfere with the materials around the warehouse
18.	[18]	In this paper, classification of existing researches, problems and algorithms aroused in the study of swarm robotics are presented. The existing studies are classified into major areas and relevant subcategories in the major areas.	A lot of new heuristics and algorithms were introduced to solve the problems in this domain. In the learning domain, reinforcement learning (RL) was given much interest by the researchers.These algorithms does not fulfill our requirements.
19.	[19]	This paper proposes to augment an autonomous navigation system based on learning classifier systems for use in collective robotics, introducing an inter-robot communication mechanism inspired by ant stigmergy, with each robot acting independently and cooperatively.	The ability of the robot that was controlled by the autonomous navigation system with and without the pheromone, the system capability of minimizing distances between start and goal is not feasible.
20.	[20]	The experimental work illustrated in this paper summarises the research activities carried out by the authors with robots capable of physically connecting to each other.	The experimental work done here doesn't concern the requirements of our project mainly the warehouse, and does not give details of the robot itself, therefore a qualitative analysis couldn't be done.

Chapter 3

Requirements

3.1 Hardware Requirements

The main hardware requirements of the project are as follows:

1. **Ubuntu Machine running ROS Master**
    - Core i5 4th generation or above
    - 4GB RAM or above
    - 120GB HDD
    - Wireless connectivity
2. **Raspberry Pi 3B+ or 4B**
    - 2GB LPDDR4-3200 SDRAM or higher
    - 16GB memory Card
    - 1.2 Gigahertz Quad Core processor or higher
    - Dual Core GPU or higher
3. **STM32 32-bit MCU**
    - 32-bit processor
    - 72 MHz clock speed
    - 64 Kb of flash memory or higher
    - 10 kb SRAM or higher
4. **Power source**
    - Li-Ion 11.1V 2200mAh (2C)

4. **H-bridge Motor Drivers**
    - Maximum peak current of 1.2 A or above
    - Motor Voltage of 5V-36V
    - Maximum continuous current of 600 mA or above
5. **DC Geared encoder motors**
    - 12V rating
    - Quadrature encoder with 990 Pulse Per Revolution (PPR)
    - 110 No load RPM
    - 6.5 kgcm torque
7. **Omni Wheels**
    - 58 mm diameter - load capacity(3 Kg/Wheel)

3.2 Software requirements

The project relies on the following software to facilitate intercommunication of robots and to provide seamless interface between the user and the mobile robot system:

1. **Ubuntu Operating System**

   Ubuntu is a Linux distribution based on Debian and mostly composed of free and open-source software. Ubuntu is officially released in three editions: desktop, server, and core for Internet of things devices and robots.
2. **ROS Noetic Ninjemys**

   Robot Operating System is an open source robotics middleware. Although ROS is not an operating system, it provides services designed for a heterogeneous computer cluster such as hardware abstraction, low-level device control, implementation of commonly used functionality, message-passing between processes, and package management.
3. **Python**

   Python is an interpreted, high-level and general-purpose programming language. Python's design philosophy emphasizes code readability with its notable use of significant white-space.
4. **Arduino IDE**

   The open-source Arduino software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board, as well as a wide range of other MCUs using their respective drivers.

Chapter 4

System Architecture and Methodology

4.1 Mobile Robot architecture

4.1.1 Platform

The hardware design includes the entire robot that is constructed using acrylic sheet of 3mm thickness. The base of the robot houses three motors placed 120° apart from each other. [see Fig. 4.2].

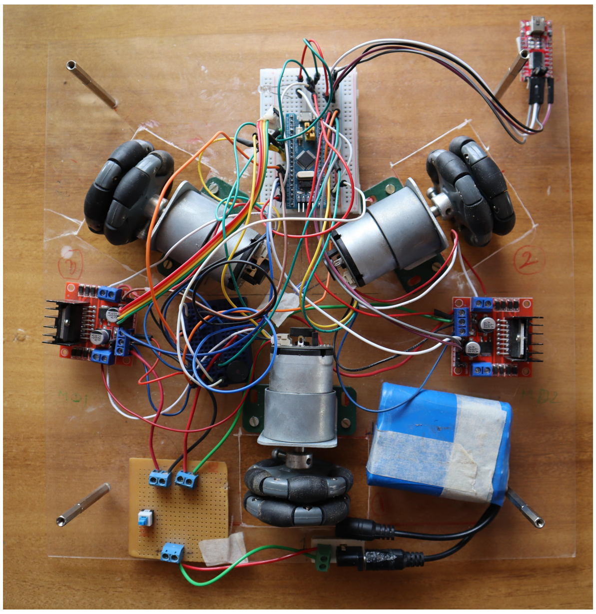

Figure 4.1: Top view of the robot

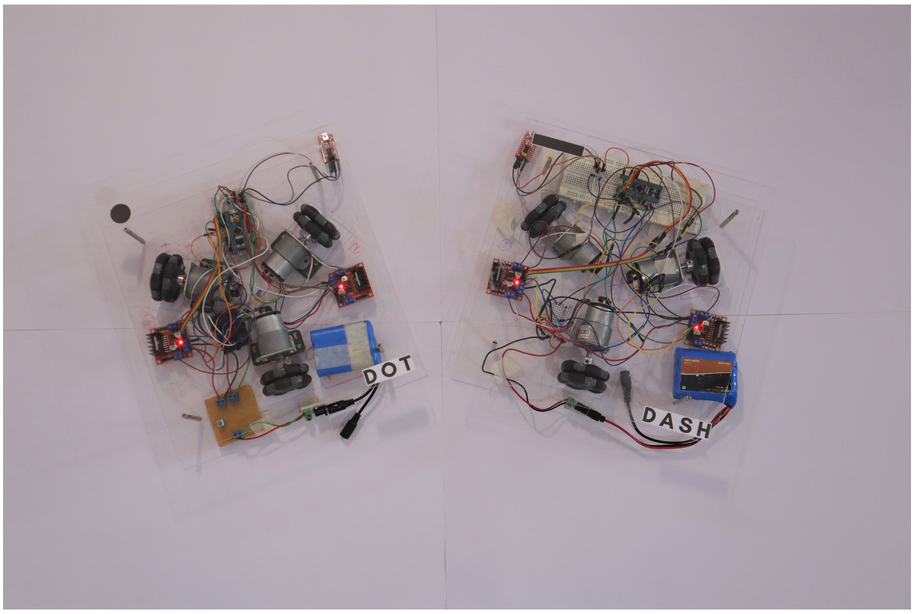

Figure 4.2: Mobile Robot System

4.1.2 Electronics

The entire robot is powered by a Li-ion 11.1V 2200mAH (2C) battery. The omni wheels are driven by Rhino GB37 12V 110 rpm 3.5kgcm DC geared encoder servo motor, which houses a quadrature magnetic encoder that has a PPR rating of 990, driven by two L298N motor drivers.

4.1.3 Controller

The brain of the robot is the STM32 (STM32F103C8T6) micro-controller also known as the 'blue pill', which houses Arm 32 Cortex-M3 CPU, with a clock speed of 72MHz, which is crucial for the computations of the kinematic equations that are further discussed. In order to indicate the start and end of a run, a buzzer has been incorporated to advertise the same.

In Fig 4.2, a system of two omni directional mobile robots are shown, which are identical to each other in every aspect. As explained in previous sections, in order to proove the concept of swarm robotics, a minimum of two robots is necessary.

4.2 System Block Diagram

The overall system consists of two major sub system, namely Navigation subsystem and the path planning subsystem as shown in Fig 5.2. The Navigation system resides on the mobile robot which uses waypoints following algorithm to maneuver the robot from initial pose to final pose as desired by the user. The waypoints following control calculates the required voltages that needs to be supplied to each wheel of the omni directional robot, this is further explained in detail in the following sections.

The path to be followed by the robot is given by the path planning subsystem. This subsystem consists of A* path following algorithm for both individual robot environment and multi robot environment. Multi-robot path planning algorithm uses a modified version of single robot path planning algorithm which takes into account the path followed by other robots while planning a path for the current robot. This subsystem is a centralised computation module which resides on a remote computer which has knowledge about all the robots in the space that needs to be taken care while planing a path.Further sections contains insight about both the algorithms.

The communication between the remote computer running the path planning subsystem and the Navigation subsystem running on the mobile robot is achieved using Actions server and Action client architecture provided by the Robot Operating system (ROS). ROS Actions provide a seamless integration of both the subsystems in the local area network.

4.3 Methodology

4.3.1 Navigation subsytem

The Waypoint following algorithm is divided into two types, namely, single waypoint following control sequence and multi-waypoints following control sequence. Single waypoint following uses Go to Pose algorithm which enables the mobile robot to reach a specified destination and orientation in the defined environment. The second type is multi-waypoints following, which uses a version of the same *Go to Pose* algorithm, where the destination point is updated to reach the next or a new destination, when the robot reaches the current destination, considering a margin of error as shown in Fig . Margin of error, S_{th}, is set to accommodate the inherent noise in the encoder readings and the wheel slippage.

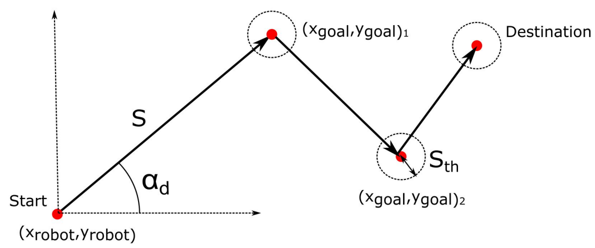

Figure 4.3: Multi Waypoint following

4.3.2 ROS Actions

ROS Actions have a client-to-server communication relationship with a specified protocol. The actions use ROS topics to send goal messages from a client to the server. Goals can be cancelled using the action client. After receiving a goal, the server processes it and can give information back to the client. This information includes the status of the server, the state of the current goal, feedback on that goal during operation, and finally a result message when the goal is complete.

- A **request message** is sent from an action client to an action server initiating a new goal.
- A **result message** is sent from an action server to an action client when a goal is done.
- **Feedback messages** are periodically sent from an action server to an action client with updates about a goal.

4.3.3 Path Planning subsystem

Path planning algorithm is one of the foremost and important research problems in the field of robotics. It deals with mapping the robot in the environment and finding the best path between the source and destination location. Because global environment information is not always available, path planning cannot always be done in advance. Path planning can be widely used in partially and unknown structured settings if an appropriate method is proposed.

A suitable trajectory is created as a series of operations to keep the robot moving from the starting point to the goal point through numerous intermediate states. Every decision in path planning algorithms is made based on the information available at the time and criteria, such as the shortest distance to the target point measured using euclidean distance computation. There may be several paths available from the same initial and final positions and, there may be times where there is no path available between the initial and final positions. In terms of optimization, the ideal path should cover the smallest distance, be free of obstacles and collisions, and take the least amount of time to reach the objective state or the destination or the goal coordinates. Since a mobile robot may have numerous motion constraints, such as the non-holonomic condition in under actuated systems, the chosen trajectory must be smooth and free of excessive twists. However, the proposed system is a holonomic system, and therefore, it eases out the maneuverability process, as the total number of degrees of freedom (DoF) is equal to the number of degrees of freedom that can be controlled [3].

For mobile robot path planning, there are variety of mature approaches for constructing an environment model. A grid decomposition map, quad split graph, visibility graph, and Voronoi diagram are some of the basic environment models that are now available. Following the creation of the environmental map, global path planning is carried out. Heuristic search methods and intelligent algorithms are the two basic forms of global path planning algorithms. The A* algorithm produced by the Dijkstra algorithm is the initial representation of the heuristic search technique or method. For state space, the A* algorithm is the most often used heuristic graph search technique. It is frequently used for robot path planning in addition to addressing problems based on state space.

4.3.4 A* algorithm

A* algorithm is a heuristic search algorithm to find the best possible path between the source and the destination. The most widely known form of best-first search is called A* search. It evaluates nodes by combining g(n), the cost to reach the node, and h(n), the cost to get from the node to the goal,

$$f(n) = g(n) + h(n) \tag{4.1}$$

Since $g(n)$ gives the path cost from the start node to node n, and $h(n)$ is the estimated cost of the cheapest path from n to the goal, we have,

$f(n)$ = estimated cost of the cheapest solution through n

Therefore, if we are trying to find the cheapest solution, it is reasonable to try first the node with the lowest value of $g(n) + h(n)$. It is evident that this strategy is more than just reasonable, provided that the heuristic function $h(n)$ satisfies certain conditions, A* search is both complete and optimal. The cost function used is the manhattan distance between the two coordinates of interest, and $g(n)$ is the manhattan distance between the current position of the robot which is obtained from the self localization method and the next position the robot considers as its destination. Also, $f(n)$ is the manhattan distance between the goal point and the next position. Summing up $h(n)$ and $g(n)$, gives the overall cost function. The obstacles can be defined by giving a cost of higher magnitude, usually infinity to those positions where obstacles are present. As a result the algorithm ignores these positions since the cost is very high. Overall, A* algorithm tries to minimise this cost function by choosing the least value of $f(n)$, thereby avoiding exploration of different nodes that might not lead to an optimal path to the destination.

Chapter 5

Implementation and Testing

5.1 Control algorithm for waypoints following

The overview of the control architecture used in the proposed *Go to Goal* algorithm is shown in Fig. 5.1 . At the very instant the mobile robot is powered on, it is fed the goal position (x_{goal}, y_{goal}), goal orientation θ_{goal}, and the desired velocity V_d of the mobile robot by the user in case of single waypoint following, or a list of goal positions and the final orientation in case of multi-waypoints following. The current pose of the mobile robot is obtained from the self localization as explained in [24]. The algorithm first calculates the euclidean distance S between the current position of the robot and the goal position, and now this distance error is compared with the preset threshold S_{th} or the error margin at every refresh rate, which is 50ms in our case.

$$\begin{aligned} S &= \sqrt{(x_{goal} - x_{robot})^2 + (y_{goal} - y_{robot})^2} \\ \alpha_d &= \tan^{-1} \frac{y_{goal} - y_{robot}}{x_{goal} - x_{robot}} \end{aligned} \tag{5.1}$$

Now, if this distance error thus calculated turns out to be less than the threshold value S_{th}, then as per the algorithm, the mobile robot comes to a complete halt or in other words seizes its motion in case of single waypoint following, or updates its destination point or the goal point, according to the list, in case of multi-way points following.

If the distance error above the threshold S_{th} still persists, the algorithm calculates the angle between the goal position (x_{goal}, y_{goal}) and the instantaneous position of the mobile robot (x_{robot}, y_{robot}) represented by α_d (5.1). As the robot is set into motion, the local axis begins to rotate. As a result, heading error starts to accumulate, in order to compensate for this heading error, the difference between α_d and the robot's orientation θ_{robot} is calculated which is represented by ψ. A PID controller is used to compensate for the orientation error, i.e, the difference between the goal orientation, θ_{goal} and the

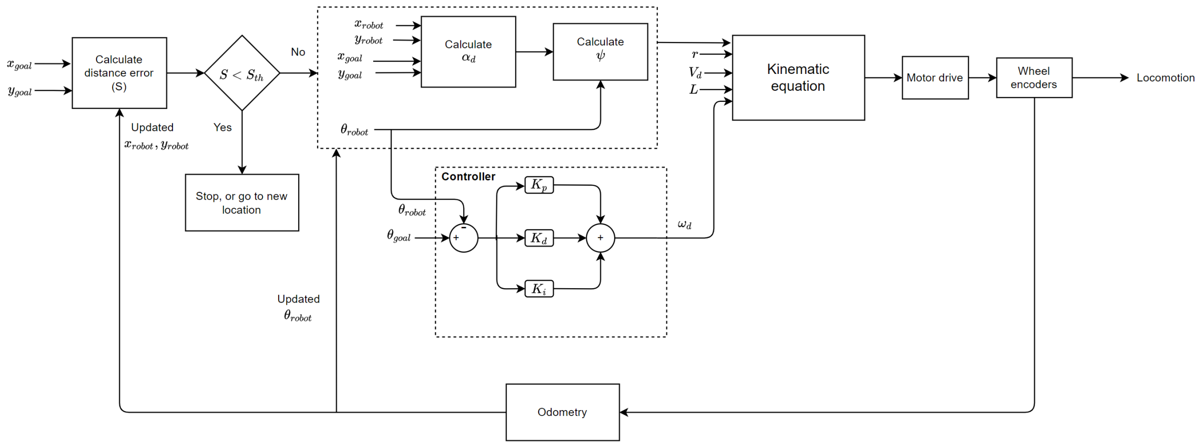

Figure 5.1: Go to Pose algorithm control architecture

robot's orientation, θ_{robot}. The output of the PID controller is represented by ω_d.

$$\begin{bmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \end{bmatrix} = \frac{1}{r} \begin{bmatrix} 1 & 0 & L \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} & L \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} & L \end{bmatrix} \begin{bmatrix} V_d \cos\psi \\ V_d \cos\psi \\ \omega_d \end{bmatrix} \tag{5.2}$$

The output from the controller ω_d, wheel radius r, desired velocity of the robot V_r, the distance from centre of mass of the robot to the center of the wheel L, and the heading error correction ψ, are used to obtain the angular wheel velocities $\dot{\phi}_1$, $\dot{\phi}_2$, and $\dot{\phi}_3$ using kinematic equation as given in (5.2). These angular wheel velocities are used to drive the motors by supplying corresponding voltages to the motors.

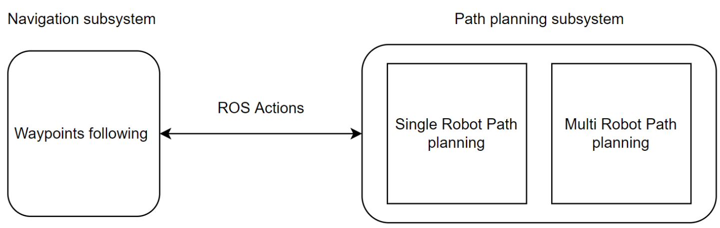

Figure 5.2: Block Diagram

5.2 Path planning for a single mobile robot

For implementation, it is necessary to map the environment into a 2D Cartesian coordinate system. The robot operates in this defined environment as guided by the path planning algorithm. The obstacles are mapped beforehand and fed into the algorithm. The path planning algorithm takes the environment mapped in a Cartesian coordinate space, the initial position, the final position and the list of obstacle coordinates if present any. The algorithm calculates the best possible path between the source and the destination and returns the path to the navigation module of the mobile robot to traverse the path.

Consider a grid of size p x q, therefore total number of nodes in the grid is p*q. Construct a graph G of size p*q. Each node has two attributes g and f, where g and f are the cost functions g(n) and f(n) respectively, as defined above. H(x) is the Heuristic function defined. Start and goal nodes are the initial and final destination coordinates.

Algorithm 1 SINGLE ROBOT PATH PLANNING

```
procedure PATHPLAN(start, goal, obstacle, list, graphG)
    for each node in G do
        n.f = infinity
        n.g = infinity
    create an empty list L
    while list not empty do
        current=node in the list with smallest f value
        remove current form list
        if current==goal then
            break
        for  each node, n that is adjacent to current do
            if n==obstacle then
                continue
            if n.g > ( current.g + cost of edge from n to current ) then
                n.g = current.g + cost of edge from n to current
                n.f = n.g+H(n)
                n.parent = current
                add n to list if not present already
        return list L
```

5.3 Path planning for a multi-mobile robot system

Multi-robot route planning can be classified as either centralised (assuming the existence of a central component that knows the state of the entire robot system) or decentralised (assuming the absence of a central component that knows the state of the entire robot system) (where no single component has the full picture, but cooperation must still be achieved). Depending on whether they are coupled or decoupled, centralised approaches can be further classified. Coupled techniques work in all of the robots' joint configuration space, allowing for completeness. Decoupled techniques, on the other hand, plan for each robot separately and resolve path conflicts as they happen, ensuring that collisions with other robots are avoided. Sequential programming, vehicle prioritising, and velocity tuning are all methods for decoupled planning. These solutions provide increased scalability at the expense of completeness and optimality.

We have used a simple but effective approach to plan plaths for multiple robots based on priority. Prioritized planning is a very efficient method because it allows robots to plan sequentially in space-time in order of priority, avoiding the combinatorial complications of linked systems. In this method, each robot calculates a minimum-cost path to its target that avoids any higher-priority robots' computed paths. Clearly, the priority order chosen will have an impact on the answer obtained. The path of the higher priority robot is considered as an obstacle in the defined space-time environment and the lower priority robot plans as to not avoid the path taken by the higher priority robot. Priority among the robots is chosen in such a way that the mobile robot near to the goal location is given higher priority and the robot farthest from the goal location is given the least priority. This reduces restrictions in the path for the lesser priority robots.

5.4 ROS communication

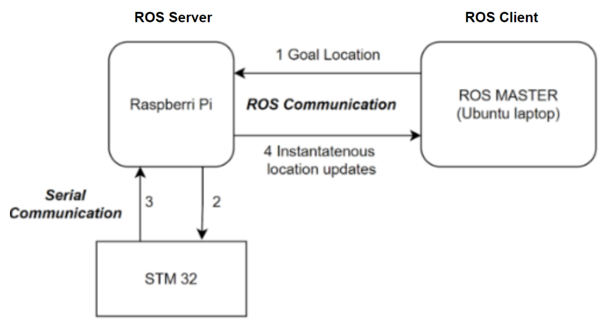

Figure 5.3: ROS communication

Fig. 5.3 depicts the communication architecture used in our system, it consists of a action server and action client. The action server runs on the mobile robot which accepts the goal location from the Action client running on the remote computer. The action server communicates with the underlying low level controller which runs the aforementioned way points following algorithm to enable the robot move from one point to another. The action client running on the remote server receives the path from

Algorithm 2 MULTI ROBOT PATH PLANNING

```
robot list = [ p:(startp, goalp), q:(startq, goalq)]
priority list → SORT ( euclidean distance ( robot list )
Multi-robot path planning( startp,startq, goalp, goalq, obstacles, graph G, priority list )
while list not empty do
    path = PATH PLAN ( start, goal, obstacles, graph G )
    obstacles = obstacles + path
    Append path to Path list
return path list


function PATH PLAN(start,goal,obstacle,list,graph G)
    for each node in G do
        n.f = infinity
        n.g = infinity
    create an empty list L
    start.g=0, start.f= H( start) add start to list
    while list not empty do
        current=node in the list with smallest f value
        remove current form list
        if current==goal then
            break
        for  each node, n that is adjacent to current do
            if n==obstacle then
                continue
            if n.g > ( current.g + cost of edge from n to current ) then
                n.g = current.g + cost of edge from n to current
                n.f = n.g+H(n)
                n.parent = current
                add n to list if not present already
        return list L
```

the aforementioned path planning subsystem and publishes this message to the action server. Once the robot gets the path and starts it maneuver, the ROS Server publishes the instantaneous location of the mobile robot which is in turn subscribed by the action client to send the data to the mobile application as implemented in [24]

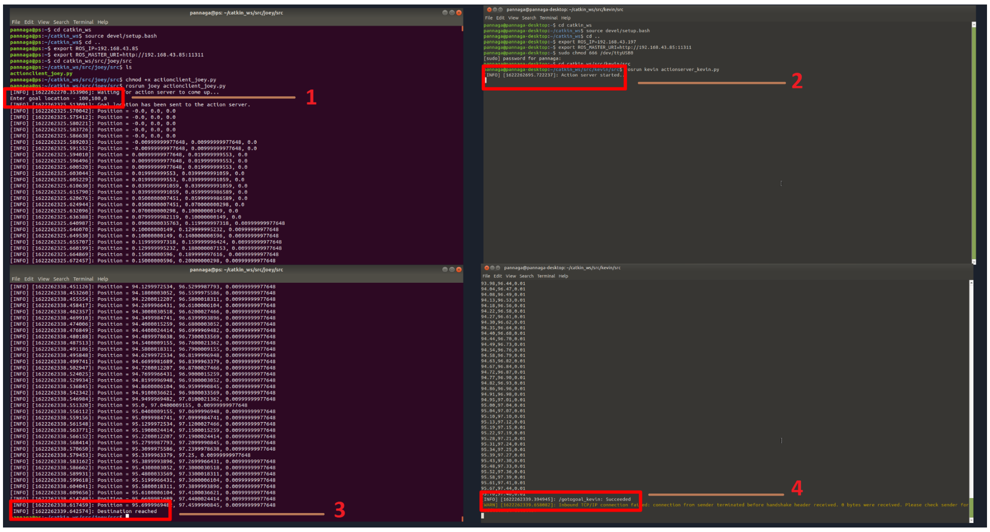

Figure 5.4: ROS Action snippets

The exchange of data between the robot and the remote machine is shown in Fig. 5.4. The top left window is the action client running on a remote Ubuntu machine which takes the goal location input from the user, and calculates the path for the robot using the path planning subsystem. Top right window represents the action server running on the robot, waiting for the goal location, once it receives the goal location, it send it to the waypoint following algorithm which maneuvers the robot to that goal location. Meanwhile, the action server also sends back the instantaneous location of the robot which is shown bottom right window of Fig 5.4. Once the robot reaches the goal location, the action server sends a result message back to the client, the current maneuver stops and the user is indicated that robot has reached the destination successfully as shown in bottom left window [Fig 5.4].

5.5 Results and Discussions

The above discussed navigation algorithm reinforced with odometry from [24] was implemented to a custom built Three Wheeled Omni-directional mobile robot as shown in Fig 4.2. The gain values of the PID controller, that is the K_p, K_d, and K_i values were tuned and set after conducting around 25-30 trials.

5.5.1 Single-Waypoint Following

The Fig 5.5 represents the traversal of the mobile robot from (0,0,0^c) to (1000,1000,0.7853^c). The blue dotted line indicates the shortest path possible for the robot, and the red line indicates the trajectory or the route taken by the robot. The arrow heads indicate the orientation of the robot at that instance in time. It is evident that pose of the robot started

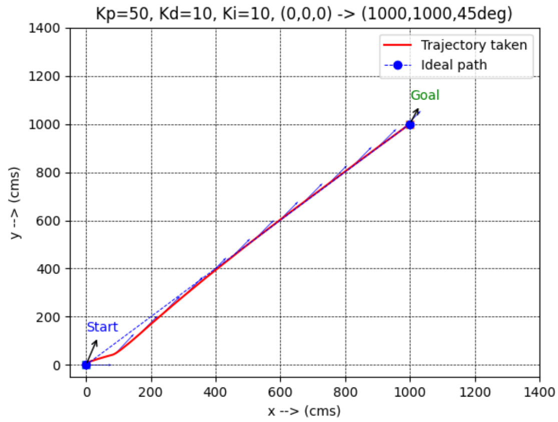

Figure 5.5: Straight line trajectory

at 0^c as was set in the program, and as time progressed, the pose is seen to correct itself to the desired pose, which is 0.7853^c or 45°, thereby eliminating the pose error that was present initially.

Figure 5.6: Pose error Vs Time

Fig 5.6 depicts the graph of 'Pose error response Vs Time'. The pose error is merely the difference between θ_{robot} and θ_{goal}, and this error is calculated every 50 milliseconds. It can be pronounced that the pose error is maximum initially, and gradually decreases over time, as the robot approaches the goal point. The oscillations die out eventually since they are governed by the K_p, K_d, and K_i values. We also note that the green horizontal line parallel to x axis, that is $y = 0$ is the ideal pose error.

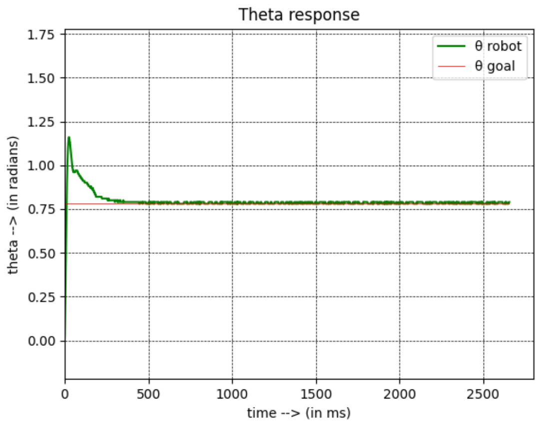

Figure 5.7: θ_{robot} Vs Time

The visual representation of the variation of θ_{robot} against time is shown in Fig 5.7. As per the algorithm, initially the pose is set to 0^c and the plotted θ_{robot} is updated every 50 milliseconds, and it is bold that it eventually catches up to meet θ_{goal} as desired, confirming the robustness of the algorithm thus developed.

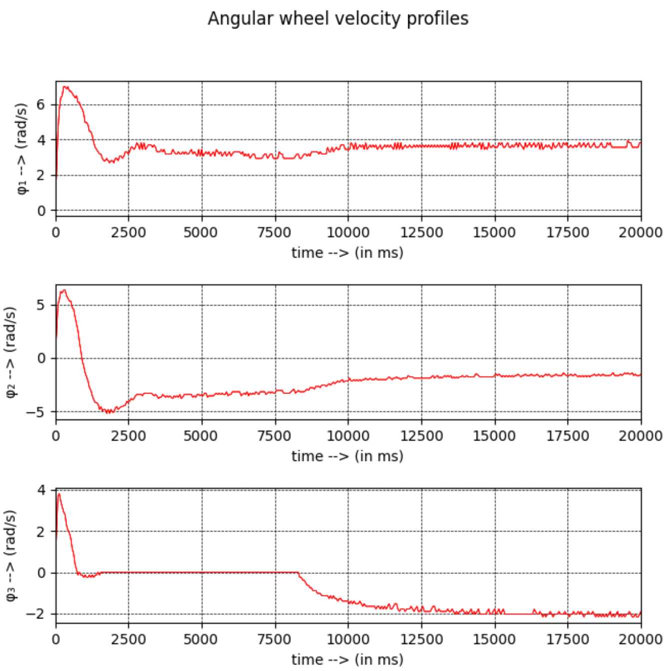

Figure 5.8: Wheel velocity profiles

Fig 5.8 depicts the individual wheel velocity profiles of the mobile robot. The y axis marks the velocity in rad/s and the x axis indicates the time in ms. The angular wheel velocities, i.e., ϕ_1, ϕ_2, and ϕ_3 are calculated after referring [24].

5.5.2 Multi-Waypoints Following

The multi point navigation, is a mere extension of the single waypoint navigation. To begin with, a Square, Hexagram, and a Circular trajectory was chosen for the experiment as shown in Fig. 5.9, Fig 5.13, Fig 5.17 respectively.

The following Fig 5.9, shows the mobile robot's traversal on a 'Square trajectory' of side 200 cms.

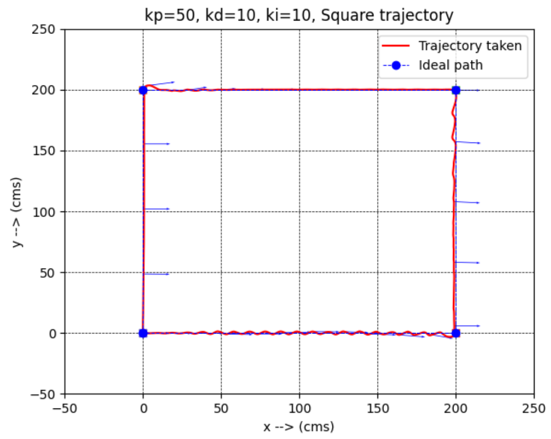

Figure 5.9: Square trajectory

The coordinates of the trajectory given by the user were $(0,0,0^c)$, $(0,200,0^c)$, $(200,200,0^c)$, $(200,0,0^c)$, and $(0,0,0^c)$. It is evident from the Fig 5.9 that the mobile robot has maintained its pose through-out its run. The trajectory taken is approximate to an ideal square path of side 200 cms. However, the feeble oscillations that is observed in the trajectory at corners exist because of sudden change in the coordinates of the destination of the mobile robot which is basically an orthogonal change of the path which the mobile robot has to travel and the fact that the mobile robot is trying to follow as close to an ideal path as possible, with the heading angle ψ which is PID controlled. The maximum deviation of the mobile robot in this case was found to be 3.52 cms, and the RMSE was 0.017 cms.

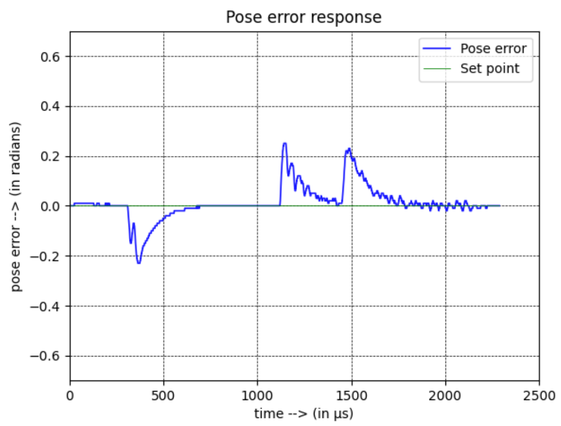

Figure 5.10: Pose error response

The pose error response curve of the square trajectory following is shown in Fig 5.10. The three extreme shoots in the curve exists since the mobile robot has to encounter three new corners, or in other words, three new coordinates. Also, the pose error is 0^c, since the initial and final first set of coordinates lie on the line $x = 0$. It is clear that the oscillations that exist die out eventually upon reaching the goal location or the goal coordinate.

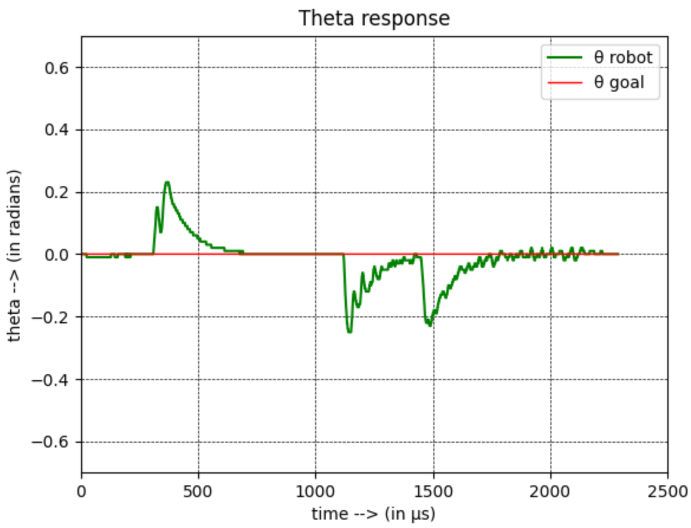

Figure 5.11: Theta response

The plot of theta response against time is shown in Fig 5.11. As explained above, the θ_{robot} is set to 0^c initially, and every θ_{goal} is set to 0^c, therefore, the mobile robot tries orient itself in a way where θ_{robot} and θ_{goal} are identical.

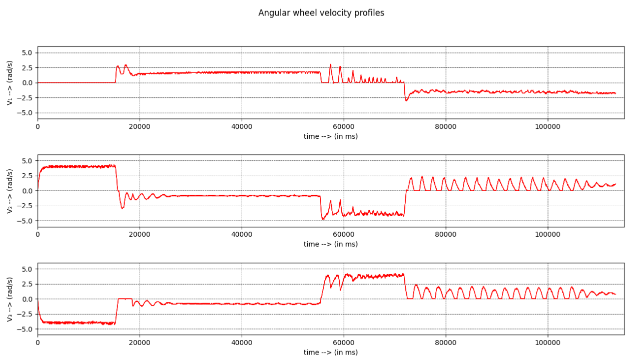

Figure 5.12: Angular wheel velocity profiles during square trajectory

The angular wheel velocity profiles are as depicted in Fig 5.17. These curves aid in visualizing the changes that were being done by the algorithm, during its traversal.

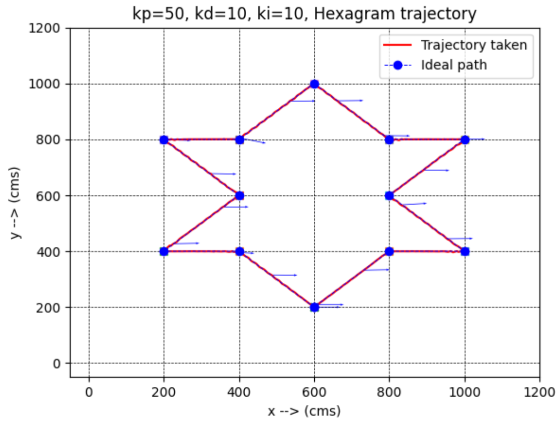

Figure 5.13: Hexagram trajectory

As depicted in Fig 5.13, the traversal coordinates were $(600, 200, 0^c)$, $(400, 4000^c)$, $(200, 4000^c)$, $(400, 6000^c)$, $(200, 8000^c)$, $(400, 8000^c)$, $(600, 10000^c)$, $(800, 8000^c)$, $(1000, 8000^c)$, $(800, 6000^c)$, $(1000, 4000^c)$, $(800, 4000^c)$, $(600, 2000^c)$. It is evident in Fig 5.13, that the mobile robot has travelled almost an ideal path during its run. The maximum deviation of the mobile robot away from the ideal path was found to be 3.6 cms, and the RMSE was 0.009 cms.

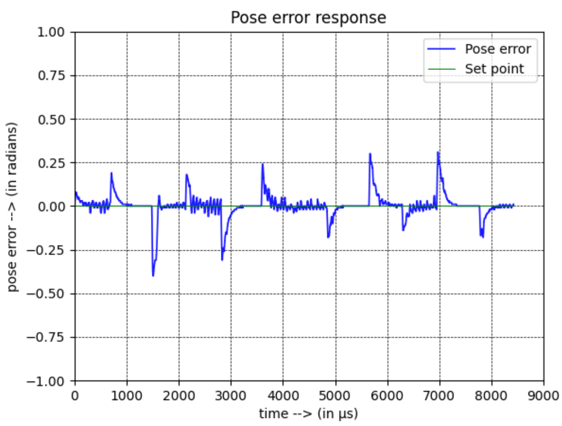

Figure 5.14: Pose error response

The above Fig 5.14 depicts the pose error plot. The pose error that is present initially is merely the difference between the instantaneous orientation of the mobile robot, i.e., θ_{robot}, and θ_{goal}. The fluctuations are a present because of the motion of the mobile robot. However, these fluctuations die out almost immediately when the mobile robot reaches its destination or the goal coordinate, or when the distance error is less that S_{th}.

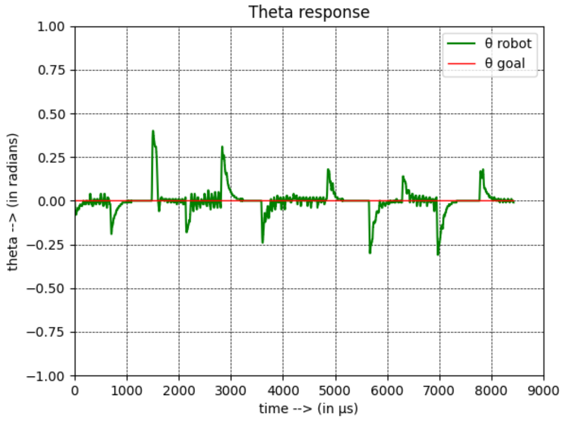

Figure 5.15: Theta response

The plot of θ_{robot} against time is shown in Fig 5.15. The θ_{goal} of the mobile robot is set to 0^c, at every location. Therefore, from the Fig 5.15, it is clear that θ_{robot} catches upto meet θ_{robot} eventually.

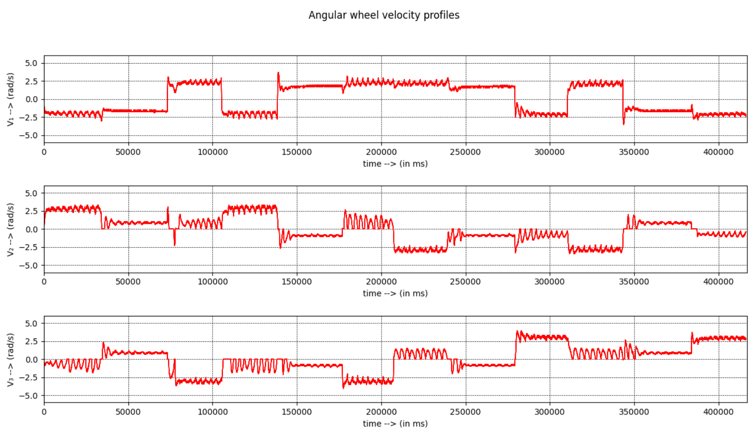

Figure 5.16: Angular wheel velocity profiles during hexagram trajectory

The angular wheel velocity profiles are as depicted in Fig 5.16. These plots indicate the angular wheel velocities of each wheel during its traversal on the ‘Hexagram’ trajectory.

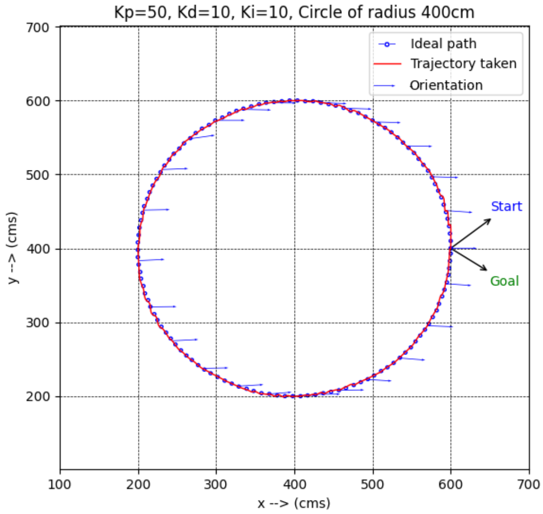

Figure 5.17: Circular trajectory

The circle,

$$x^2 + y^2 + 2gx + 2fy + c = 0 \tag{5.3}$$

with centre $(-g, -f)$ as $(400, 400)$ and radius $g^2 + f^2 - c$ as 200 cms was chosen as shown in Fig 5.17. The start and goal pose were $(600, 400, 0^c)$ as indicated, and the direction of traversal was counter-clockwise. In addition, we can see that the mobile robot has travelled well nigh in vicinity to the ideal path which is solely (5.3) plotted in x-y plane. The maximum deviation of the mobile robot here was found to be 2.66 cms, and the RMSE was 0.017 cms.

Figure 5.18: Pose error response

The pose error plot is as shown in Fig 5.18. The pose error plot clearly marks the difference in θ_{goal} and θ_{robot} at every 50 milliseconds. The oscillations exist since the circle is thought to be made of infinitesimally small line segments, or in this case the entire circle of radius 200 cms is divided uniformly into 126 line segments. Inspite of this approximation, the algorithm manages to keep the pose error as low as possible almost all the times during the robot's run.

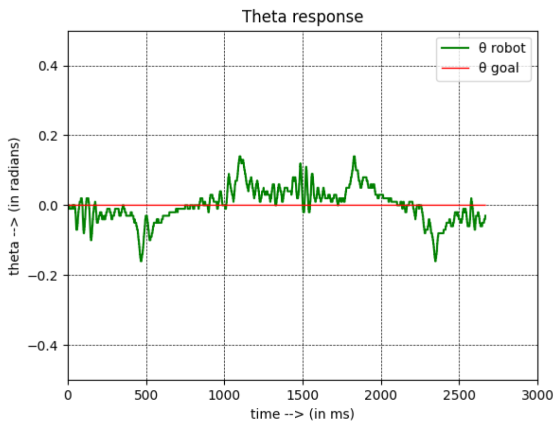

Figure 5.19: Theta response

The theta response against time is shown in Fig 5.19. The above explanation holds water for this case as well. It is also clear that the for circular trajectory, these response curves are affected primarily by how well the circle is defined, or in other words, the number of line segments that make up this desired circle of radius 200 cms, play a vital role in deciding the smoothness of both pose error plot and theta response curve.

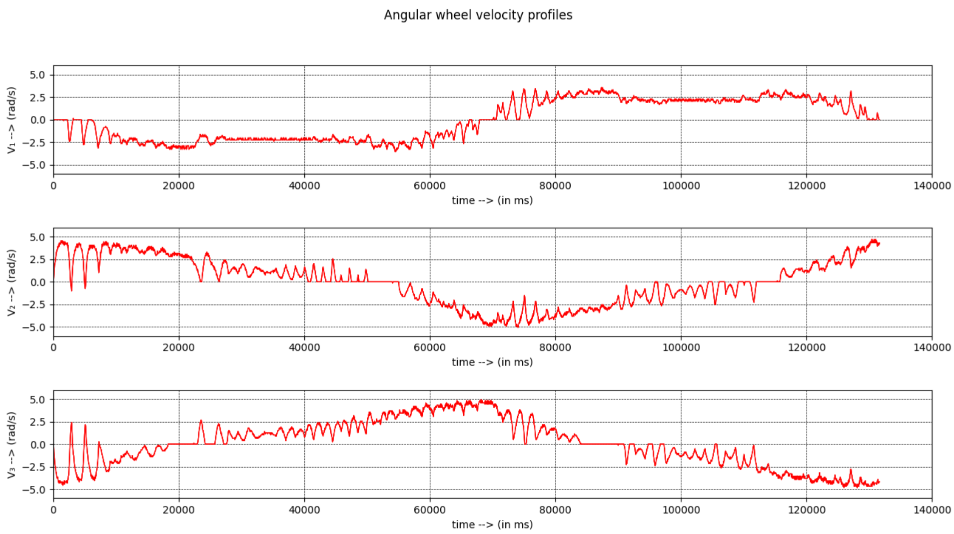

Figure 5.20: Angular wheel velocity profile during circular trajectory

The above Fig 5.20 indicates the plot of angular wheel velocities of each wheel against time. This information conveys that wheel 2 and wheel 3's angular velocities are nearly mirrored, indicating the mobile robot's direction of traversal as counter-clockwise. The angular wheel velocity plots as shown above gives a brief idea about the performance of the mobile robot during its course of traversal or during its run.

Table 5.1: Performance analysis

Trajectory type	**Maximum deviation** (cm)	**RMSE** (cm)
Straight line	31.2	0.237
Square	3.52	0.017
Hexagram	3.6	0.009
Circular	2.66	0.017

The results obtained from the experiments, shown in Table 5.1, gives an overview of the performance of the mobile robot during its run. The cross track error, which is the lateral deviation of the mobile robot with respect to the ideal or the best path possible between two successive waypoints is calculated and the root mean square of all the cross track errors is taken for calculating the required performance metric, for both single-waypoint following trajectory as shown in Fig 5.5, and in multi-waypoints following trajectory, as shown in Fig 5.13. According to table 5.1, in case of a 'single-waypoint' trajectory, the maximum deviation from the ideal path was found to be 31.2 cm and in case of 'multi-waypoints' following trajectory, the maximum deviation was found to be 3.52 cm, 3.6 cm, and 2.66 cm for Square, Hexagram, and Circular trajectory respectively. The root mean square of the cross track errors (RMSE), that have been accumulated over time gives us an approximation of how far away from the ideal path, the mobile robot has drifted laterally during its run. The RMSE was found to be 0.237 cm for 'single-waypoint' following, and 0.017 cm, 0.009 cm, 0.017 cm for Square, Hexagram, and Circular trajectory respectively, in case of 'multi-waypoint' following, all which comply with the set K_p, K_d, and K_i.

A $30\mathrm{units} \times 30\mathrm{units}$ layout was chosen with a grid size of $1\mathrm{unit} \times 1\mathrm{unit}$. Obstacles were statically entered to the path planning subsystem to simulate a warehouse environment.

Fig 5.21 depicts the path planned by the single robot path planning algorithm for a start location as $(21, 4)$ and the goal location as $(2, 44)$. The green circle indicates the robot start location in the grid and the red circle indicates the robot goal location.

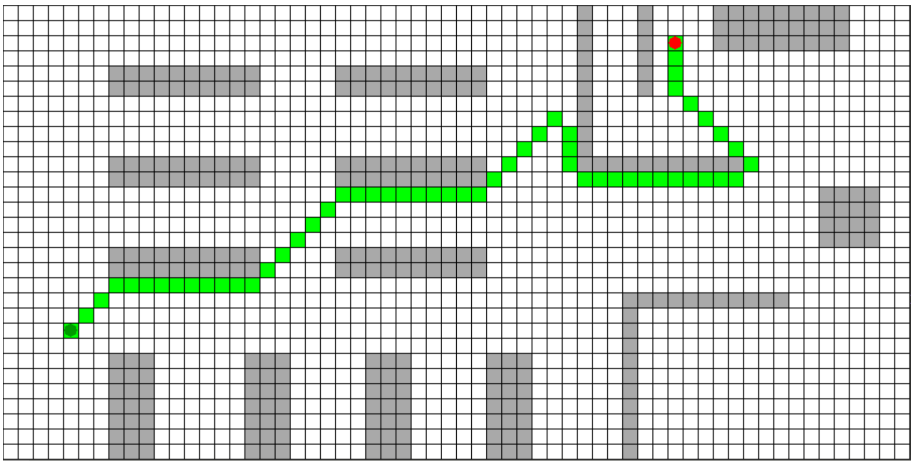

Figure 5.21: Single robot path planning

Fig 5.22 depicts the path planned by the multi robot path planning algorithm for a start location as $(21, 4)$ denoted by the green circle and $(26, 56)$ denoted by the blue circle. The common goal location for the two robots to converge was set as $(2, 44)$. The priority for the robot shown by blue circle was higher as it is located closer to the goal location compared to the robot shown by green circle.

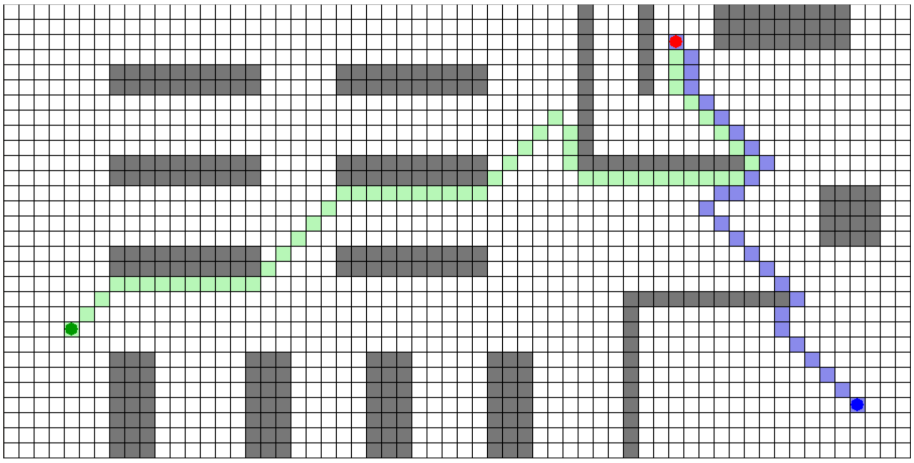

Figure 5.22: Multi robot path planning 1

Fig 5.23 depicts the path planned by the multi robot path planning algorithm for a start location as $(27, 20)$ denoted by the green circle and $(27, 4)$ denoted by the blue circle. The common goal location for the two robots to converge was set as $(11, 34)$. The priority for the robot shown by green circle was higher as it is located closer to the goal location compared to the robot shown by blue circle.

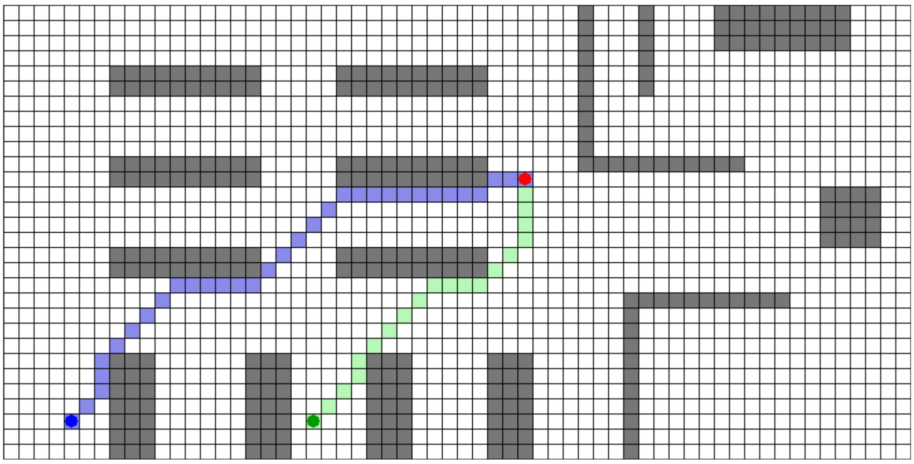

Figure 5.23: Multi robot path planning 2

Chapter 6

Conclusion and Future scope

6.1 Conclusion

This project demonstrates the self localization and path plannning of a system of three wheel omni drive mobile robots assuming no wheel slip. In addressing the problem of localization, ruling out the possibility of accumulation of inherent noise of the system, sensor noise, and aliasing is not possible in any situation. Although the algorithm developed here, takes care of eliminating the errors pertaining to the odometry aspect of the mobile robot, however, the additional accumulation of internal noise is inescapable, and will affect the mobile robot's localization in the long run.

This project provides a cost effective way of implementing the navigation algorithm for the mobile robot in the defined workspace. As explained in previous sections, for point to point navigation, the real time instantaneous coordinates of the robot is calculated by the feedback from the quadrature encoder motors, thereby eliminating the use of additional encoders dedicated to record the instantaneous coordinates of the mobile robot. It is observed that, for a given source and destination coordinates, the performance characteristics of the mobile robot is improved using multi-waypoints following, where multiple intermediate coordinates are given along the path of the source and destination coordinates.

6.2 Future scope

The Possible work that could be carried forward from this point onwards, is obstacle detection and avoidance in the defined work space. Also, an even more rigorous and robust alternative to the PID controller could be the application of 'Fuzzy Logic', which could provide even more accurate results during crucial tracking of way-points and when exposed to obstacles in proximity of the robot's path or trajectory. The implementation of path planning algorithm, i.e., the 'Prioritized A*' algorithm would be done in a layout simulating a warehouse environment. Further improvements in path planning will be researched and implemented with better and real time algorithms.

Bibliography

[1] Konolige, K., Augenbraun, J., Donaldson, N., Fiebig, C., Shah, P. (2008, May). *"A low-cost laser distance sensor"*. 2008 IEEE International Conference on Robotics and Automation. 2008 IEEE International Conference on Robotics and Automation (ICRA). https://doi.org/10.1109/robot.2008.4543666

[2] Pérez, J., Nashashibi, F., Lefaudeux, B., Resende, P., Pollard, E. (2013). *"Autonomous Docking Based on Infrared System for Electric Vehicle Charging in Urban Areas."* Sensors, 13(2), 2645–2663. https://doi.org/10.3390/s130202645

[3] K. Shabalina, A. Sagitov and E. Magid, *"Comparative Analysis of Mobile Robot Wheels Design,"* 2018 11th International Conference on Developments in eSystems Engineering (DeSE), Cambridge, United Kingdom, 2018, pp. 175-179, doi: 10.1109/DeSE.2018.00041

[4] M. R. Elara, N. Rojas and A. Chua, *"Design principles for robot inclusive spaces: A case study with Roomba,"* 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, 2014, pp. 5593-5599, doi: 10.1109/ICRA.2014.6907681

[5] Doroftei, I., Stirbu, B. (2010). Design, Modeling and Control of an Omni-Directional Mobile Robot. Solid State Phenomena, 166–167, 173–178. https://doi.org/10.4028/www.scientific.net/ssp.166-167.173.

[6] R. L. Williams, B. E. Carter, P. Gallina and G. Rosati, *"Dynamic model with slip for wheeled omnidirectional robots,"* in IEEE Transactions on Robotics and Automation, vol. 18, no. 3, pp. 285-293, June 2002, doi: 10.1109/TRA.2002.1019459.

[7] *"Dynamical Models for Omni-directional Robots with 3 and 4 Wheels"*; May 2008 Conference: 5th International Conference on Informatics in Control, Automation and Robotics.

[8] J. Bonneau, C. Herley, P. C. Van Oorschot and F. Stajano, *"The quest to replace passwords: A framework for comparative evaluation of web authentication schemes"*, 2012 IEEE Symposium on Security and Privacy (SP), May. 20–23, 2012.

[9] Sanket Soni , Trilok Mistry and Jayesh Hanath, *"Experimental Analysis of Mecanum wheel and Omni wheel"* , IJISET - International Journal of Innovative Science, Engineering Technology, Vol. 1 Issue 3, May 2014

[10] Mohanraj, A. P., Elango, A., Reddy, M. C. (2016). *Front and Back Movement Analysis of a Triangle-Structured Three-Wheeled Omnidirectional Mobile Robot by Varying the Angles between Two Selected Wheels*. The Scientific World Journal, 2016, 1–11. https://doi.org/10.1155/2016/7612945

[11] Ranjit Barua , Sajal Mandal and Dr. Samiran Mandal *" Motion Analysis of A Mobile Robot With Three Omni-Directional Wheels "*,IJISET - International Journal of Innovative Science, Engineering Technology, Vol. 2 Issue11, November 2015

[12] D. Kirovski and H. Malvar, *"Robust covert communication over a public audio channel using spread spectrum"*, Proc. 4th Int. Workshop Inf. Hiding, pp. 354-368, 2001

[13] 13. R. Soares, E. Bicho, T. Machado and W. Erlhagen, *"Object transportation by multiple mobile robots controlled by attractor dynamics: theory and implementation,"* 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, CA, 2007, pp. 937-944, doi: 10.1109/IROS.2007.4399019

[14] Su, K. L., Li, B. Y., Guo, J. H., Liao, Y. L. (2014, June).*" Motion Planning of Multi-docking System for Intelligent Mobile Robots"*, 2014 International Symposium on Computer, Consumer and Control. 2014 International Symposium on Computer, Consumer and Control (IS3C). https://doi.org/10.1109/is3c.2014.330

[15] Jigar J. Parmar, Chirag V. Savant, *"Selection of Wheels in Robotics"*, International Journal of Scientific Engineering Research, Volume 5, Issue 10, October-2014

[16] F. Guangrui and W. Geng,*"Vision-based autonomous docking and re-charging system for mobile robot in warehouse environment"*, 2017 2nd International Conference on Robotics and Automation Engineering (ICRAE), Shanghai, 2017, pp. 79-83, doi: 10.1109/ICRAE.2017.8291357.

[17] Se-gon Roh et al., *"Flexible docking mechanism using combination of magnetic force with error-compensation capability"*. 2008 IEEE International Conference on Automation Science and Engineering, Arlington, VA, 2008, pp. 697-702, doi: 10.1109/COASE.2008.4626554.

[18] Mohan, Y., Ponnambalam, S. G. (2009).*"An extensive review of research in swarm robotics"*, 2009 World Congress on Nature Biologically Inspired Computing (NaBIC). 2009 World Congress on Nature Biologically Inspired Computing (NaBIC). https://doi.org/10.1109/nabic.2009.5393617

[19] Cazangi, R. R., Von Zuben, F. J., Figueiredo, M. F.(2005).*"Autonomous navigation system applied to collective robotics with ant-inspired communication"*, Proceedings of the 2005 Conference on Genetic and Evolutionary Computation - GECCO '05. the 2005 conference. https://doi.org/10.1145/1068009.1068026

[20] Tuci, E., Groß, R., Trianni, V., Mondada, F., Bonani, M., Dorigo, M. (2006). *"Cooperation through self-assembly in multi-robot systems"*. ACM Transactions on Autonomous and Adaptive Systems, 1(2), 115–150. https://doi.org/10.1145/1186778.1186779

[21] Amanda J. C. Sharkey, *"The Application of Swarm Intelligence to Collective Robots"* in Advances in Applied Artificial Intelligence, John Fulcher, Idea Group Publishing, 2006, pp. 157 - 185.

[22] Uny Cao, Alex S. Fukunaga and Andrew B. Kahng, *"Cooperative Mobile Robotics: Antecedents and Directions"*, Autonomous Robots, vol. 4, 1997, pp. 1-23.

[23] Amanda J. C. Sharkey, *"Robots, Insects and Swarm Intelligence"*, Artificial Intelligence Review, vol. 26, 2006, pp. 255-268.

[24] Adithya, Preethi M H, Kandibedala Abhishek Kumar, Mabubhasha P, *"Localization and Control of 3 wheel drive omnidirectional mobile robots"*