Maze Solving using Autonomous Navigation

Submitted by

Isha Harish	**Nidhi V Ranjith**	**Diya Parekh**	**Tanmay Doshi**
200929098	200929100	2000929176	200929150

Under the guidance of

Dr ASHA C S

Associate Professor

Department of Mechatronics

MIT Manipal

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

BACHELOR OF TECHNOLOGY

IN

MECHATRONICS

DEPARTMENT OF MECHATRONICS

MANIPAL INSTITUTE OF TECHNOLOGY

(A Constituent of Manipal Academy of Higher Education)

MANIPAL - 576104, KARNATAKA, INDIA

June 2022

ABSTRACT

In recent years, the robot industry has made great progress, and has been widely used in automobile manufacturing, machining, transportation agriculture, medical treatment, food and other aspects. The autonomous navigation is a key technology in the development of robot intelligence.Here, in our project we have presented the Maze Solving behavior of robot in a Simulation based on ROS2. Computer Vision is the key focus with the integration of important robotics algorithms of In recent years, the robot industry has made great progress, and has been widely used in Motion Planning. The type of robot we will be using is Differential Drive Robot with a castor wheel.

TABLE OF CONTENTS

CHAPTER 1
Introduction

When it comes to solving a range of human problems, autonomous systems can be a huge help. This may have a variety of uses, including driverless cars, humanoid robots, assistive devices, home appliances, military equipment, and manipulator systems, to mention a few. Autonomous systems that assist people who need medical, mobility, household, physical, and mental assistance are known as assistive robots. Applications of this study area are growing in number, including autonomous wheelchair systems , autonomous walkers, intelligent canes ,lawn mowers , vacuum cleaners , and surveillance systems in facilities like assisted living . One of the most crucial things is data. These data are frequently gathered from the environment that the autonomous system operates in; examples of such data include the system's position and location coordinates in the environment, the static objects, the speed, velocity, or acceleration of the system, its peers, or any moving objects nearby, the direction that the vehicle is travelling in, the air pressure, and so on. The data is current since it is taken straight from the operational environment and can be accessed using either built-in or linked sensing equipment/devices. The navigation of an autonomous vehicle is the main topic of our project. We examine the studies conducted in the past and present employing imaging technologies, i.e., cameras . For tasks like object detection, obstacle avoidance, mapping, localisation, etc., the autonomous systems use sensor data. These two senses can complement one another, and are thus widely used for detection in autonomous systems.

CHAPTER 2
Related Work

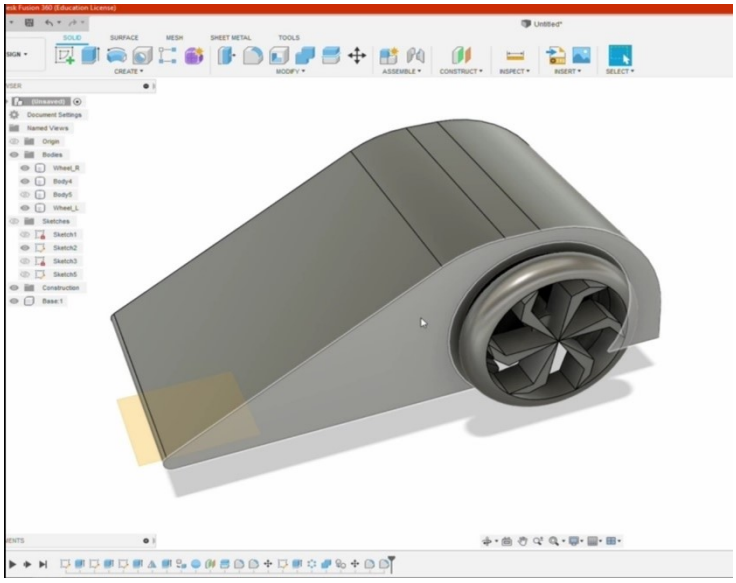
The Tremaux algorithm, which was later adopted in literature by other writers, is one of the most extensively used solutions in this field. In this method, a mobile robot operating in a virtual world would be able to discover the shortest path in the least amount of time. As a result, even though it was unable to manoeuvre it or tell another robot the answer, the robot was still able to show the quickest path. An crucial aspect of this work is that the robot was told where the exit was, which may not always be the case in our instance.

CHAPTER 3
Methodology

3.1 Custom Robot Creation

Here,in our project we have made use of fusion360 and blender software in order to design our robot(a simple working model of a two-wheeled car with a castor wheel). A differential drive robot was decided upon; one which maneuvers due to the difference in velocities in the two wheels fixed on either side of the body. The castor wheel attached towards the front portion of the robot body is free to move along the x-,y- and z- axes ,which aids in agility and maneuvering. No electronics or external hardware is used.

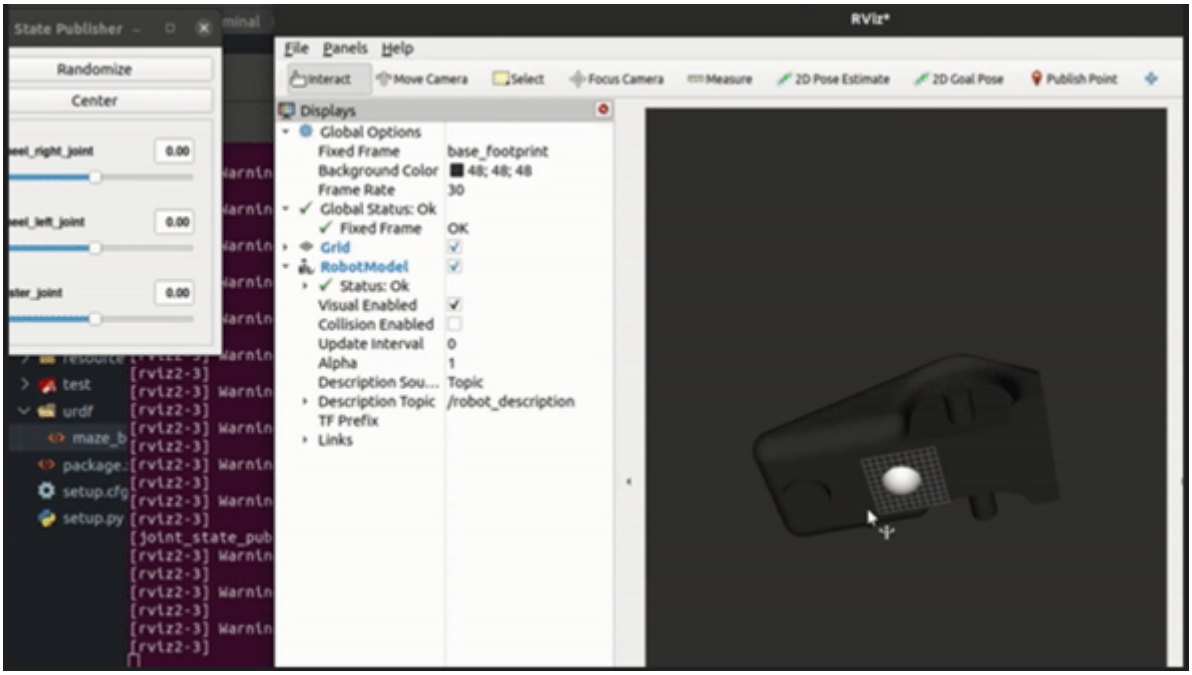
3d modelling: The model has been made from scratch on fusion360 in the required dimensions.

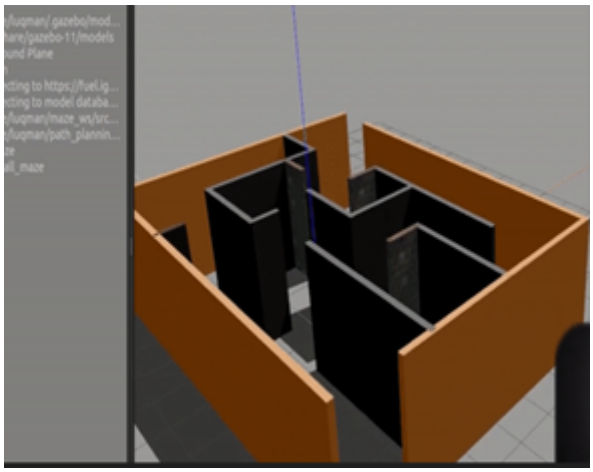


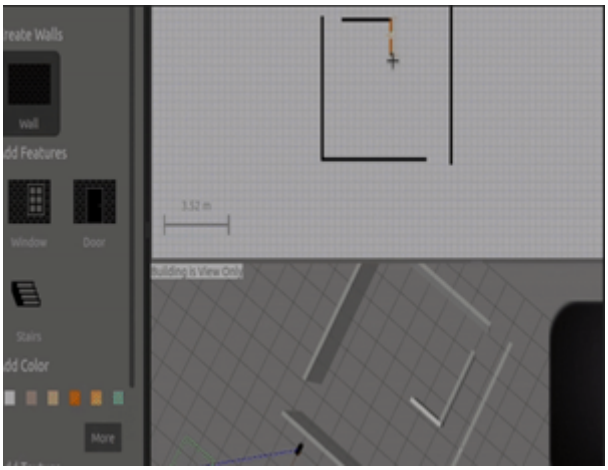
The file is then converted into a stl file and imported in ROS.

3.2 Gazebo and RViz Integrations

Rviz is further used for visualisation of the robot model .However due to the sheer size of the stl models ,as seen in the visualisation software, blender is used to compress the meshes. A joint axis test is conducted to ensure that the various components of the car move relative to one another, along the required axes, via joints. Gazebo is a robot simulation software with inbuilt libraries and tool-services.Here,we define the gravity,inertia and collision properties of the gazebo model is defined such that it is visible in our workspace and is properly constrained.Further the maze(rather,the world) is created on gazebo and saved as a launch file.

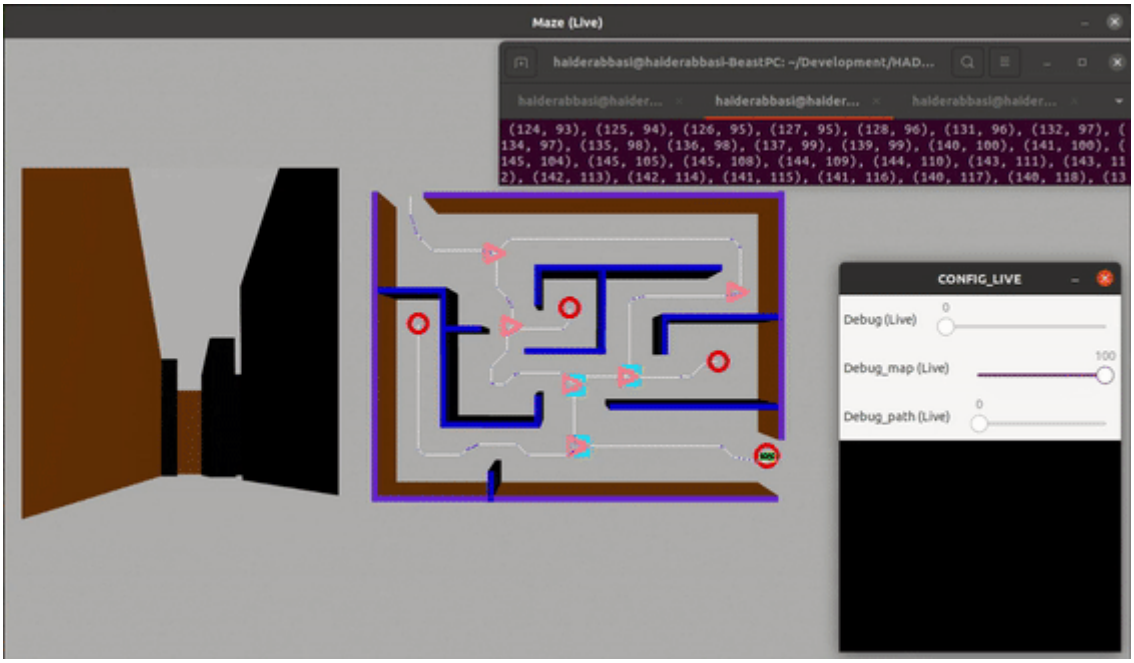






3.3 Mapping and Localization

In this project we make use of background subtraction for the localisation of the robot. The inputs received are an extracted frame from the video feed from camera and a frame to display the localized robot. The outputs using the bot_localization program are the co-ordinates (x,y) of the localized robot and the occupancy grid which is generated from the cropped maze.The purpose of this is to perform mapping to convert a top down maze view into a traversable graph using the bot_mapping python program. The input taken is the occupancy grid from the localization stage. The outputs are a graph generated from the given occupancy grid and an image that displays the pathways in the given maze input.By measuring angular velocity of the wheel, it can calculate the movement speed and direction of robot and conjecture the navigation track of the robot.

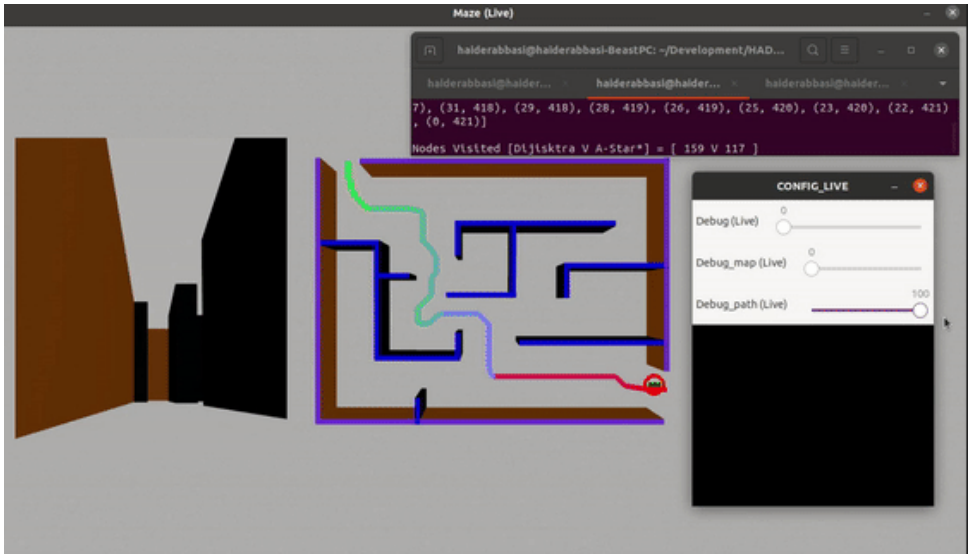


3.4 Path Planning

The autonomous vehicle needs three systems in order to follow the intended course. These systems include those for navigation, guiding, and control. Absolute positioning and relative positioning are the two primary position-estimation techniques typically used in navigation problems. Depending on the nature of environment, path planning can be divided into static and dynamic environment. Here ,we have assumed the environment to be static with only the robot being in motion.The maze is assumed to be static as well. The purpose of the bot_pathplanning program is to perform path planning from the source to the destination. The inputs taken are the graph extracted by the mapping module, the start and end points, the maze image and the methods to be used during path planning. The methods used are as follows:-

- DFS
- Min Heap
- Dijikstra
- Astar

As a result we receive the computed path from source to destination as a list of coordinates and the found path is highlighted in colour on the maze image.


Maze (Live)
haiderabbasi@haiderabbasi-BeastPC: ~/Development/HAD...
haiderabbasi@haider...
haiderabbasi@haider...
haiderabbasi@haider...
7), (31, 418), (29, 418), (28, 419), (26, 419), (25, 420), (23, 420), (22, 421)
, (0, 421)]
Nodes Visited [Dijsktra V A-Star*] = [159 V 117]
CONFIG_LIVE
Debug (Live)
Debug_map (Live)
Debug_path (Live)
100

CHAPTER 4
Contribution of Each Student

Nidhi V Ranjith

Contributions

- **Existing Solutions Review:** Worked on developing code for Localisation,path-planning and navigation of the robot through the maze. Also compiled the report together,after receiving inputs from peers.

Isha Harish

Contributions

- **Existing Solutions Review:** Worked on developing code for Localisation,path-planning and navigation of the robot through the maze.Also made the powerpoint presentation,after receiving inputs from peers.

Diya Parekh

Contributions

- **Existing Solutions Review:** Decided the type of robot to be used in the project and designed it using Fusion360 software.Exported the stl files and used it as meshes in the urdf.Also conducted literature reviews and studied various research papers regarding the same.

Tanmay Doshi

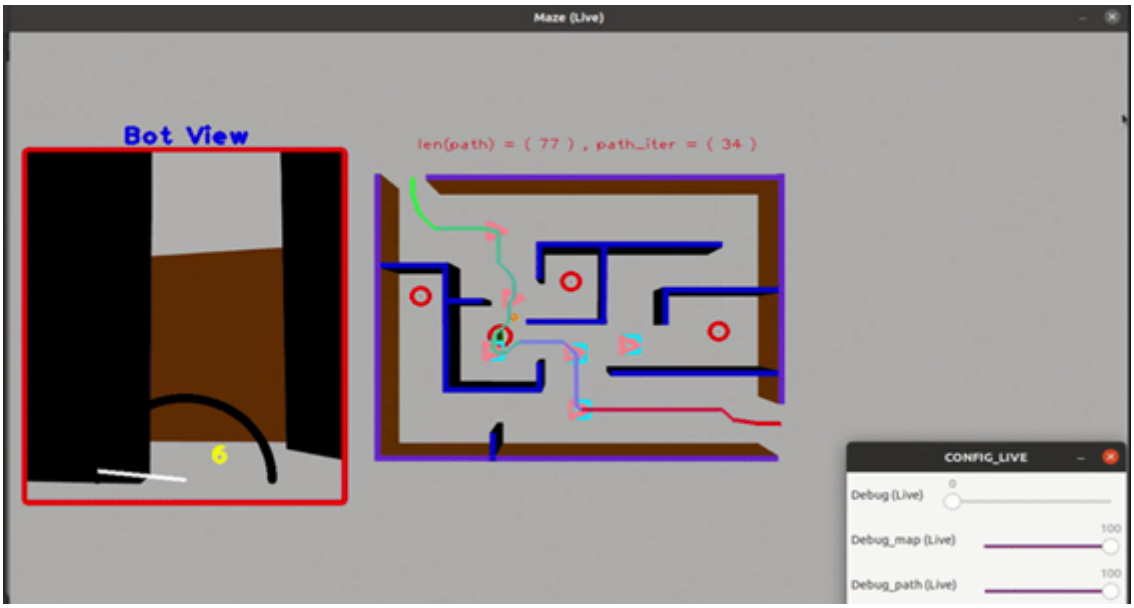
Contributions

- **Existing Solutions Review:** Worked on integrating RViz and Gazebo in the project.Designed the world and labyrinth through which the robot would have to navigate to the end goal

CHAPTER 5
Results and Discussion

5.1 Result

On running the maze_solver program, the node is initiated and the bot traverses along the maze finding the shortest path to its goal destination.



CHAPTER 6
Conclusion and Application

6.1 Conclusion

All specified targets were effectively finished completely. We planned the labyrinth into a navigable chart and afterward utilized DFS and Dijikstra to track down the best way between the source and the objective. The best way was featured utilizing a hued line which the robot follows.

6.2 Application

The Maze solver has different applications for the future, for example, in independent vehicles: assistive gadgets to help individuals with handicaps, radar directed journey control and so on. Expansion of sensors to distinguish hindrances, walls and so on make it ideal for errands, for example, entering mines and different regions which can't be accessed by people. It can likewise navigate environments which require evacuation and rescue operations. -Sensors stop collaboration - when we walk trespass a boundary/limit the robot closes down. -Speed checking control frameworks - as you stroll into a zone it dials back and the nearer you move the more slow it gets and at last stops.

Bibliography

[1] A comparative study of geometric path planning methods for a mobile robot: Potential

[2] A comparative study of A-star algorithms for search and rescue in perfect maze Source: IEEE

field and voronoi diagrams Source: IEEE

[3] A relative localisation system of a mobile robot Source: Springer Science and Business Media LLC

[4] J. Chen and D. M. Dawson, "Monocular Camera Visual Servo Control of Wheeled Mobile Robots", International Journal of Robotics Automation, vol. 26, no. 1, (2011), pp. 26-34.

[5] D. Brscic, T. Sasaki, M. L. Zhou, K. Yokoi, S. Zheng and H. Hashimoto, "Research and Development of an Autonomous Mobile Robot for Real World Environment". Proc. JSME Robotics and Mechatronics Conference, (2007) May 10-12, pp. 967-968, Akita, Japan 9