International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-2, July 2019

Obstacle Avoidance during Robot Navigation in Dynamic Environment using Fuzzy Controller

Shoaib Mohd Nasti, Zoltán Vámossy, Neerendra Kumar

Abstract: A Simulink model containing fuzzy logic controller for collision-free robot navigation in a dynamic environment is presented in this paper. Two controllers, pure pursuit and fuzzy logic controller, are considered to handle robot navigation with obstacle avoidance. Ignoring the obstacles, the pure pursuit controller computes the required linear and angular velocities to direct robot from start to goal location. However, if obstacles are present in the navigation path then the robot will get collided with obstacles in the path. As a result, the robot will not reach to the provided goal location. The fuzzy logic controller is used to avoid obstacles in the navigation path. The fuzzy logic controller takes obstacle distance, obstacle angle, target direction and the x coordinate of goal location as inputs. Consequently, the fuzzy logic controller outputs the required change in angular velocity for the robot. This change in angular velocity is applied to the angular velocity provided by the pure pursuit controller. The experimental work is performed using Turtlebot Gazebo simulator. The navigation including environment, obstacles and resultant paths are also manifested.

Index Terms: Fuzzy Logic Controller, Obstacle Avoidance, Robot Navigation, Simulink, Turtlebot Gazebo Simulator.

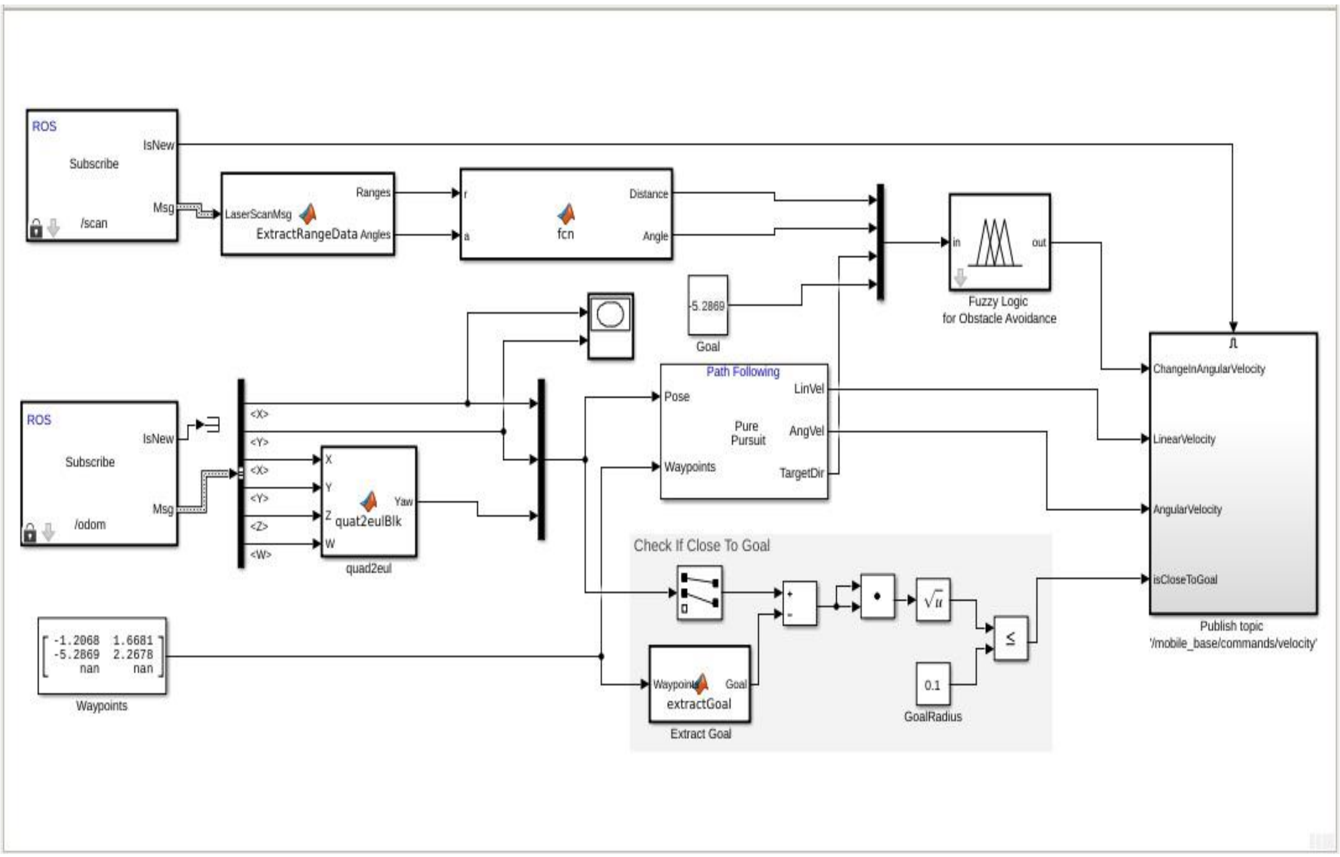
I. INTRODUCTION

The advantage of mobile robots is that they demand meagre human involvement in doing repetitive tasks. Navigation, which is an important characteristic of mobile robots, is the process of accurately determining one's location, and to plan and follow a path. The ability to navigate in a dynamic environment is significant in humans because to cross any street, we look at our right and left to analyze the movement of vehicles. If there is no movement of vehicles then we cross the street. Otherwise, we estimate the time vehicles will take to reach the location where we are standing and accordingly, decide whether to cross the street or not. In order to mimic humans, it is desirable for mobile robots in a dynamic environment to have similar behavior. For mobile robots, navigation in a dynamic environment is a challenging task because the location of obstacles is unknown before starting of the navigation. In this case, during the navigation, the robot needs to respond, dynamically, when any obstacle(s) is/are found by the robot sensors. Many of the researchers have contributed to collision-free robot navigation. Distances of various obstacles, in the navigation path of the robot, can be obtained by the distance-range sensors of the robot. Using the distance ranges of the obstacles, the robot can be prevented from aimless wandering and corner traps [1]. For collision-free robot navigation in a static environment, the Diligent-Bug algorithm to prevent both global and local loop trapping is developed in [2]. A Simulink model for robot navigation with obstacle avoidance in an unknown environment is presented in [3]. In [3], an algorithm based on robot position, laser scans and duration of scans is implemented to recognize the re-occurrences of obstacles during the navigation. The collision-free mobile robot navigation and searching in different environments with different shapes is proposed in [4]. For pick and drop of the shelves on customer's order, the collision-free mobile robot navigation from start to goal location within a warehouse is proposed in [5]. Using the model of [5], the robot moves towards the goal location if its neighbouring point, closer to goal location, is vacant. A less costly mobile robot platform with four wheels has the ability to move as a line follower robot in a 2D environment with navigation, mapping and obstacle avoidance [6].Recently, numerous researchers are attracted to the fuzzy controllers for obstacle avoidance in robot navigation. To avoid collisions in a static environment, fuzzy controllers for mobile robot navigation are given in [7]. A Mamdani-type fuzzy inference system, having inputs as robot rotation angle, robot orientation and distance between obstacle to the robot, is developed in [8]. However, the fuzzy inference system developed in [8] is applicable to static environments only. Fuzzy control system with inputs as orientation angle and position to control the linear and angular velocities of the robot is presented in [9]. Fuzzy logic based algorithm for path planning and probability to boost the performance of fuzzy logic is proposed in [10]. A Simulink model for robot navigation in an unknown environment with two controllers, pure pursuit for calculating direct path from start to goal location and fuzzy logic controller to avoid obstacles in robot navigation is presented in [11]. For obstacle avoidance in robot navigation, a Simulink model with the pure pursuit and fuzzy logic controllers is developed in [12]. In [12], the pure pursuit controller finds out a direct path from start to the next goal location and the fuzzy logic controller with Mamdani-type and Sugeno-type as two types of the fuzzy inference systems,

Revised Manuscript Received on 30 July 2019.
* Correspondence Author
Shoaib Mohd Nasti*, Department of Computer Science and Information Technology, Central University of Jammu, Jammu, India.
Zoltán Vámossy, John von Neumann Faculty of Informatics, Óbuda University, Budapest Hungary
Neerendra Kumar, Department of Computer Science and Information Technology, Central University of Jammu, Jammu, India.

avoids the obstacles in robot navigation However, the fuzzy controllers in [11][12] are biased to the right turns for obstacles appearing on center of laser scans.
Therefore, in this paper, a new fuzzy logic controller is presented to remove the right biasness for obstacles appearing on the centre of laser scan. Depending on the goal position, the proposed controller turns the robot towards left or right accordingly.

Figure 1: Simulink model



navigation path. The fuzzy logic controller takes four inputs, two of which, 'Distance' and 'Angle' are obtained

II. PROPOSED SIMULINK MODEL

Fig. 1 shows the proposed Simulink model. In the model, there are two subscribers for receiving sensor data from the robot. The first one receives messages sent to the topic of "/scan". In order to extract scan ranges and angles, the "/scan" message is then processed. The second subscriber in the model receives messages sent to the topic "/Odom". Robot's odometer data sent on the "/Odom" topic. The robot's (x,y) location is then extracted from the messages of "/Odom". The set of waypoints is the path which the robot follows. Two-dimensional coordinate positions for the robot path are taken as waypoints. Three points are considered as waypoints. Out of these three waypoints, the first one is the starting position of the robot. The second point of the waypoints lies inside or very close to the obstacles, so that, it can be tested whether the robot will avoid an obstacle or not. The third waypoint is the goal position. Pure pursuit and fuzzy controllers are used in the proposed model to move the robot from start to goal location. The pure pursuit controller takes two inputs: the robot's (x,y) location and set of specified waypoints. The pure pursuit controller computes linear and angular velocities to move the robot from current to next goal location without considering obstacles in the path. An additional controller is therefore required to avoid obstacles. The proposed fuzzy logic controller, which is another controller in the model, is used to avoid obstacles in the

from MATLAB function, and other two inputs, 'TargetDir' and 'Goal" is taken from the output of the pure pursuit controller and constant block, respectively. The fuzzy logic controller outputs the required change in angular velocity (ΔW) for obstacle avoidance. This change in angular velocity is applied to angular velocity provided by pure pursuit controller. As a result, both of the controllers, in combination, perform the robot navigation with obstacle avoidance

III. FUZZY CONTROLLER

The proposed fuzzy controller is given in Fig. 2, turns the robot towards its goal location while avoiding the obstacles. The fuzzy controller is having four inputs as:

i. Distance: This gives the distance from the robot to the closest obstacle,
ii. Angle: This gives the angle between robot to the closest obstacle,
iii. TargetDirection: This gives the target direction of the robot,
iv. Goal: This gives the x coordinate of the goal location.

There is only one output from the fuzzy controller which is Change in angular velocity (ΔW). ΔW is the required change in angular velocity for obstacle avoidance during robot navigation. The membership functions and rule base of the fuzzy controller are given ahead.

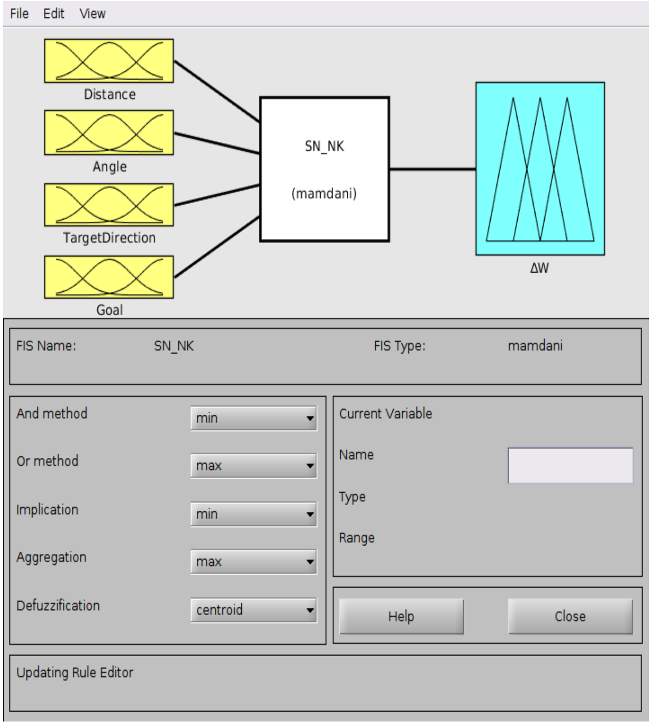


Figure 2: Fuzzy controller

A. Membership Functions

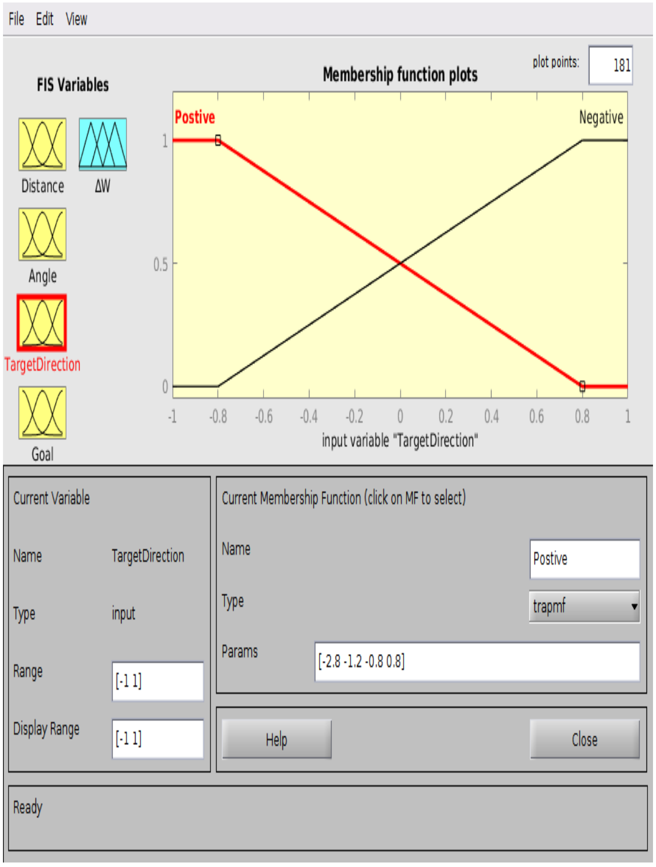


Figure 3: Membership function for target direction

Following [11], the membership functions for 'Distance', 'Angle' can be designed. However, for the two new input variables, 'TargetDirection' and 'Goal', the membership functions are defined as follows:

When an obstacle comes at the center of the robot scan, whether the robot has to move towards the right or left, is resolved by input variables 'TargetDirection' and 'Goal'. The membership function for the input variable 'TargetDirection' is divided into two categories 'Positive' and 'Negative'. Positive indicates that the robot is heading towards the right and negative indicate that the robot is heading towards left. The membership function plot for 'TargetDirection' is shown in Fig. 3. The membership function for the input variable 'Goal' is also divided into two categories 'Positive' and 'Negative'. Positive indicate that goal location is on the right side and negative indicates that goal location is on the left side. The membership function for 'Goal' is shown in Fig. 4. So, when robot moves and the goal is towards left then the robot is directed towards the left to avoid the obstacle in the path. Similarly, if the robot is heading towards left and goal is towards right then the robot is directed towards the right for the obstacle avoidance.

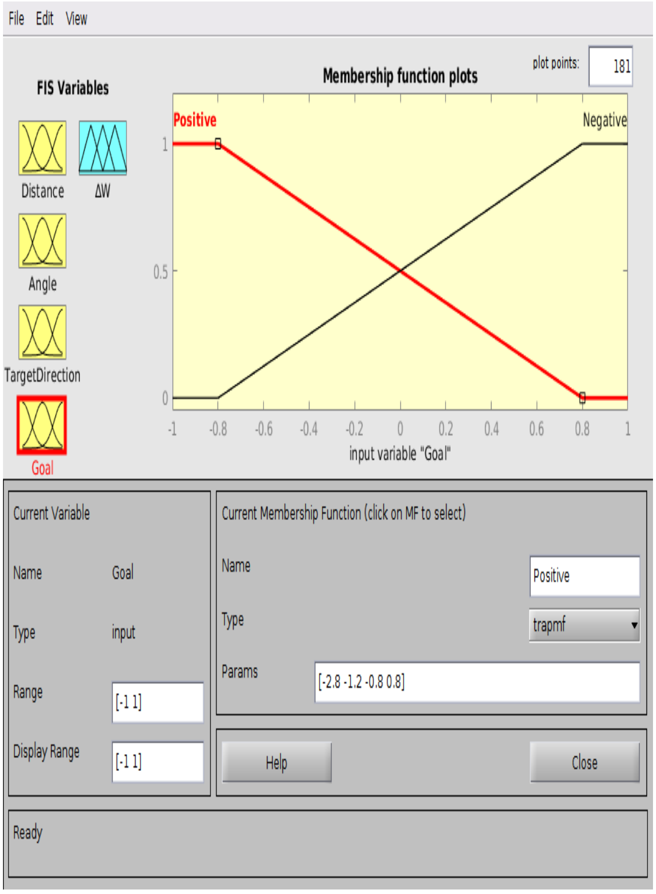


Figure 4: Membership Function for Goal

C. *Fuzzy Rule Base*

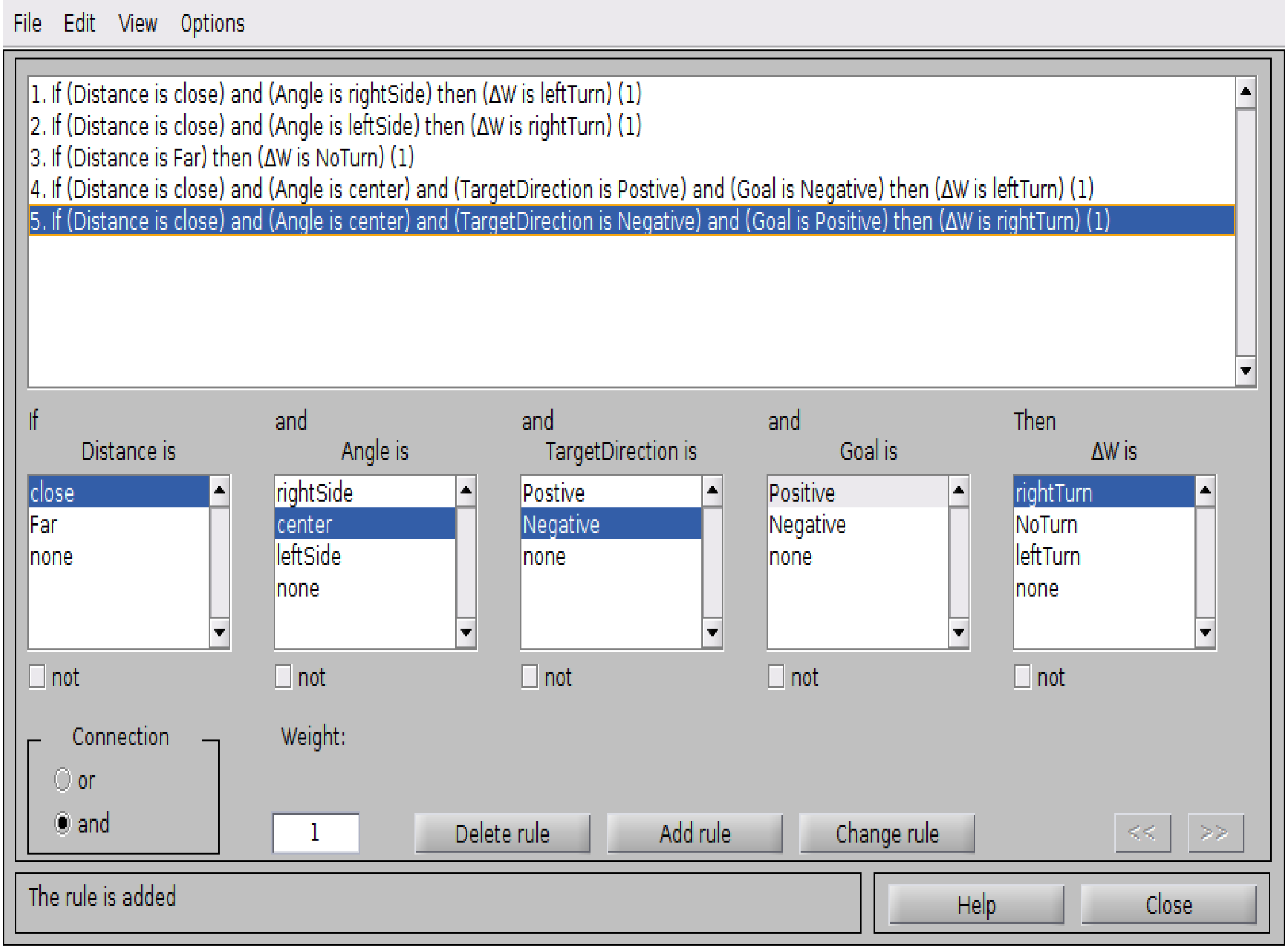


Figure 5: Rule base

As the fuzzy logic is used to mimic human reasoning. Thus, the fuzzy rules are given in Fig. 5 are taken much like how humans try to reach the goal while avoiding obstacles. The fuzzy rules (given in Fig. 5) can be summarized as follows:

1) If the obstacle is close and is right to the robot then turn the robot to the left side.

2) If the obstacle is close and is left to the robot then turn the robot to the right side.

3) If the obstacle is far from the robot then no change, in the direction of the robot, is required.

4) If the obstacle is close and is at the center of the robot scan and robot heading is right and the goal is at left then turn the robot to the left side.

5) If the obstacle is close and is at the center of the robot scan and robot heading is left and the goal is at right then turn the robot to the right side.

An instance of rule viewer is shown in Fig. 6

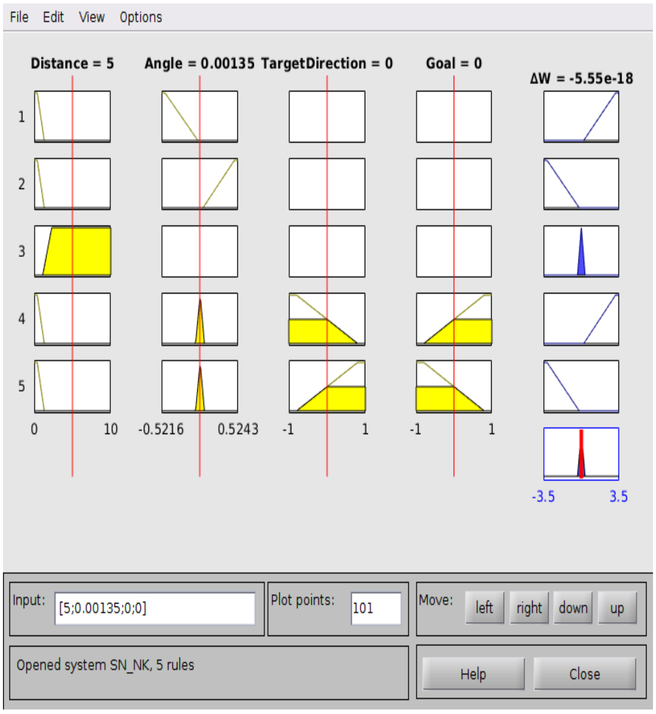


Figure 6: An instance of rule viewer

IV. EXPERIMENTAL SET-UP AND RESULTS

The experimental work is performed on a computer with dual-core 2.4 GHz + Intel i7 processor and 4 GB of RAM. The operating system used is Ubuntu 16.04 LTS. The version of the Robot Operating System (ROS) is Kinetic (1.12.2). Turtlebot-Gazebo simulator 7.0.0 is considered for the implementation of the model. MATLAB R2018a is used for programming purpose. Fig. 7 shows the Gazebo world with two spotlights indicating start and goal locations. Spotlight, where the robot is standing is the goal location obtained by the execution of the Simulink model given in Fig. 1.

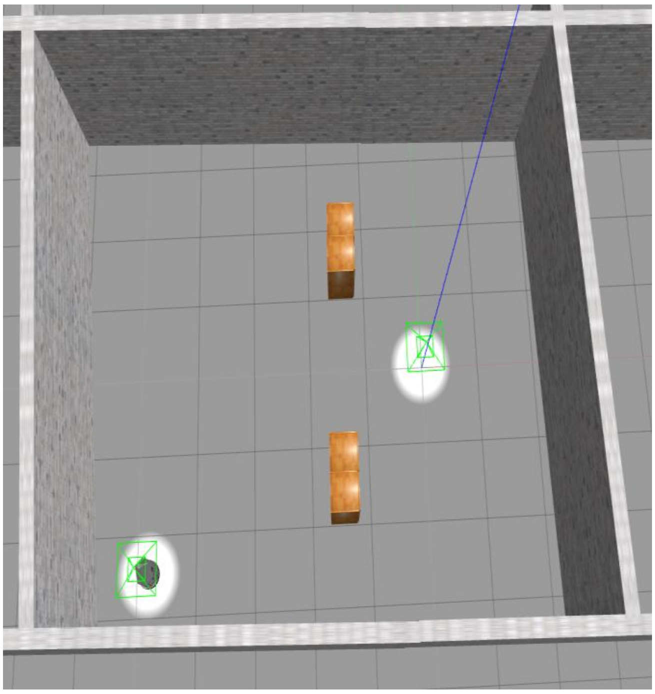


Figure 7: Navigation environment taken in Turtlebot-Gazebo Simulator

In Fig. 7, we have three waypoints which are provided to the Simulink model given in Fig. 1. The first and third waypoints correspond to start and goal locations, respectively. The second waypoint is very close to the obstacle. On execution of the Simulink model, the robot moves and obstacle appear at central of robot scan. Subsequently, as per defined fuzzy rules, the robot moves left to obstacles i.e. towards its goal position. Path of the robot is also being plotted and is shown in Fig. 8. Similarly, after the execution of the Simulink model, as robot moves and obstacle appear at the central of the robot scan. Thus, as per defined fuzzy rules, the robot moves left to obstacle i.e. towards his goal position. Path of the robot is also being plotted and is shown in Fig 9.

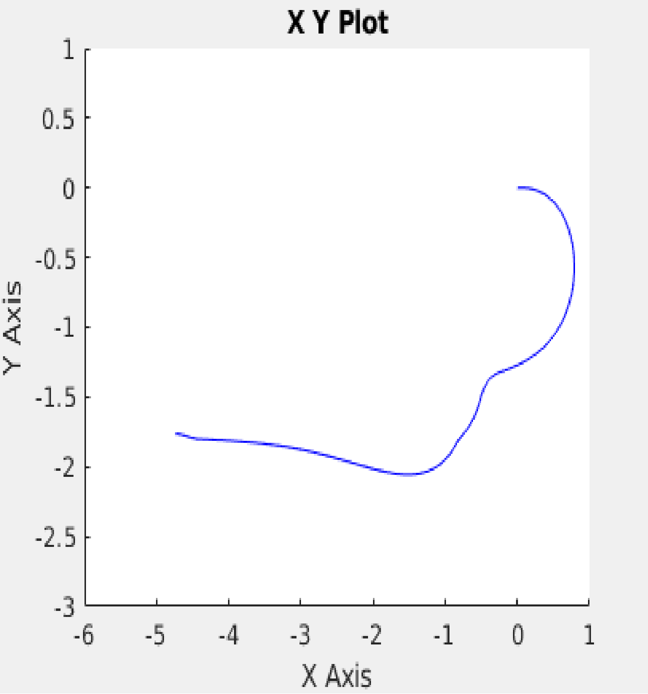


Figure 8: Robot's path for the goal on its left side

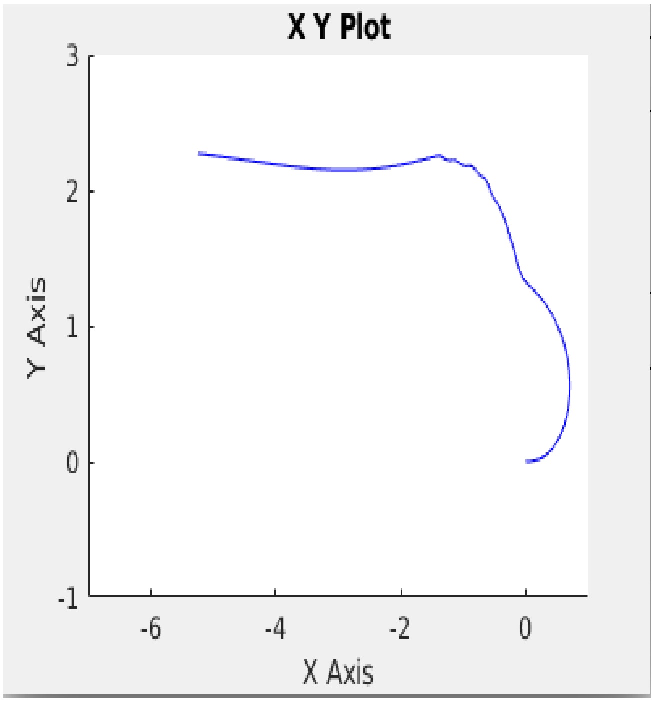


Figure 9: Robot's path for the goal on its right side

V. CONCLUSION

The proposed model is capable of obstacle avoidance regardless of the relative positions of the obstacles with respect to the robot. During navigation, when the closest obstacle is present at the left side in the robot's view area then the given model commands the robot to turn right for the obstacle avoidance. Similarly, for the obstacle at the right side in the robot's view area, the left turn command is computed and sent to the robot.

Retrieval Number: A1428058119/19©BEIESP
DOI: 10.35940/ijrte.A1428.078219
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Importantly, when the obstacle is at the center ray of the robot view area then the model overcomes the difficulty of left/right biased turning. In this situation, the fuzzy logic controller of the model uses future information to change the direction of the robot according to the goal location. Furthermore, before starting navigation from start to goal location, no information regarding the obstacles is required by the proposed model. The proposed model is, therefore, suitable for navigation in a dynamic and unknown environment where obstacles are not known beforehand or the obstacle positions change over time.

REFERENCES

1. [1] N. Kumar and Z. Vámossy, “Robot navigation in unknown environment with obstacle recognition using laser sensor,” *Int. J. Electr. Comput. Eng.*, vol. 9, no. 3, pp. 1773–1779, 2019.
2. [2] M. Abafogi, A. Durdu, and B. Akdemir, “A New Approach to Mobile Robot Navigation in Unknown Environments,” in *2018 10th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, 2018, pp. 1–5.
3. [3] N. Kumar and Z. Vámossy, “Obstacle recognition and avoidance during robot navigation in unknown environment,” *Int. J. Eng. Technol.*, vol. 7, no. 3, pp. 1400–1404, 2018.
4. [4] A. M. Rao, K. Ramji, and T. N. Kumar, “Intelligent Navigation of Mobile Robot Using Grey Wolf Colony Optimization,” *Mater. Today Proc.*, vol. 5, no. 9, Part 3, pp. 19116–19125, 2018.
5. [5] N. V. Kumar and C. S. Kumar, “Development of collision free path planning algorithm for warehouse mobile robot,” *Procedia Comput. Sci.*, vol. 133, pp. 456–463, 2018.
6. [6] S.-E. Oltean, “Mobile Robot Platform with Arduino Uno and Raspberry Pi for Autonomous Navigation,” *Procedia Manuf.*, vol. 32, pp. 572–577, 2019.
7. [7] A. Pandey and D. R. Parhi, “Optimum path planning of mobile robot in unknown static and dynamic environments using Fuzzy-Wind Driven Optimization algorithm,” *Def. Technol.*, vol. 13, no. 1, pp. 47–58, 2017.
8. [8] N. H. Singh and K. Thongam, “Mobile Robot Navigation Using Fuzzy Logic in Static Environments,” *Procedia Comput. Sci.*, vol. 125, pp. 11–17, 2018.
9. [9] R. H. Abiyev, I. S. Günsel, N. Akkaya, E. Aytac, A. Çağman, and S. Abizada, “Fuzzy control of omnidirectional robot,” *Procedia Comput. Sci.*, vol. 120, pp. 608–616, 2017.
10. [10] B. K. Patle, D. R. K. Parhi, A. Jagadeesh, and S. K. Kashyap, “Application of probability to enhance the performance of fuzzy based mobile robot navigation,” *Appl. Soft Comput.*, vol. 75, pp. 265–283, 2019.
11. [11] N. Kumar, M. Takács, and Z. Vámossy, “Robot navigation in unknown environment using fuzzy logic,” in *2017 IEEE 15th International Symposium on Applied Machine Intelligence and Informatics (SAMI)*, 2017, pp. 279–284.
12. [12] N. Kumar and Z. Vamossy, “Robot navigation with obstacle avoidance in unknown environment,” *Int. J. Eng. Technol.*, vol. 7, pp. 2410–2417, 2018.

AUTHORS PROFILE

Shoaib Mohd Nasti is currently pursuing M.tech in Computer Science & Technology from the Department of Computer Science and IT, Central University of Jammu, India. He has completed B.Tech in Computer Science & Engineering from Baba Ghulam Shah Badshah University Rajouri, India. He has also worked as a Software developer at CAPS TECHNOLOGIES, Alamgari Bazaar Srinagar, India. His research interests are Artificial Intelligence, Computer Vision and Robot Navigation.

Zoltán Vámossy is Associate Professor and Head at John von Neumann Faculty of Informatics of Óbuda University, Budapest, Hungary. In 2010, he has done his PhD in the area of Mobile Robot Navigation using Machine Vision based on PAL optics from Budapest University of Technology and Economics, Hungary. Presently, his research interests are Mobile Robot Navigation, Computer Vision and Image Processing. He has received various awards including the János Neumann Award (János Neumann Computer Science Association) in 2016. For Further information please visit the URL: http://uni-obuda.hu/en/node/701

Neerendra Kumar is working as an Assistant Professor at the Computer Science and IT department of Central University of Jammu, India. He has done his PhD studies from Óbuda University, Budapest, Hungary through Stipendium Hungaricum Scholarship. His teaching interests are in Programming languages, Algorithms, Artificial Intelligence and Robotics. His current research area is Robot Navigation and Path Planning. The details of his faculty profile is available at http://cujammu.ac.in//5084/5084_media/nareendra.pdf