Navigation & obstacle avoidance on a mobile robot using fuzzy logic

Siddharth Shankar Jha
Roll No: 14EE30022
Department of Electrical Engineering
Indian Institute of Technology, Kharagpur 721302, India
Email: siddharthjha@outlook.com

Abstract

In this term paper, studies of one of the major challenges of the autonomous navigation for mobile robots, which is detection and avoidance of obstacles during the robot navigation task is performed. The problem revolves around ensuring the robot moves according to a planned path while avoiding obstacles and successfully reaching its target destination. It is a traditional mobile robot control system problem coupled with dynamic obstacle avoidance. In this paper, a fuzzy logic based navigation algorithm for two wheeled mobile robots is defined. Later, the algorithm is modified to include closed loop control and obstacle avoidance. All of these are tested on a MATLAB simulation.

Robot Kinematic Model

The mobile robot model is a unicycle robot type. It has two degrees of freedom in the X and Y direction and is moved with two DC motors each controlling a wheel. It is also called the simple differential drive model. nine infrared range (IR) sensors, 6 of them are in front of the robot and 3 of them are in the back. The robot has a two-wheel differential drive system actuated by 2 battery operated DC motors equipped with an optical encoder for both of them. The optical encoder is used for feedback about how much each wheel has moved.

This robot type is fairly common due to its ease of construction, ability to rotate at its own position and relatively simpler kinematics. Thus a design for such a robot would be applicable for a whole lot of mobile robots in the world.

The following figure shows the actual picture of the robot and a diagram showing the orientation and position of the 9 infrared sensors.

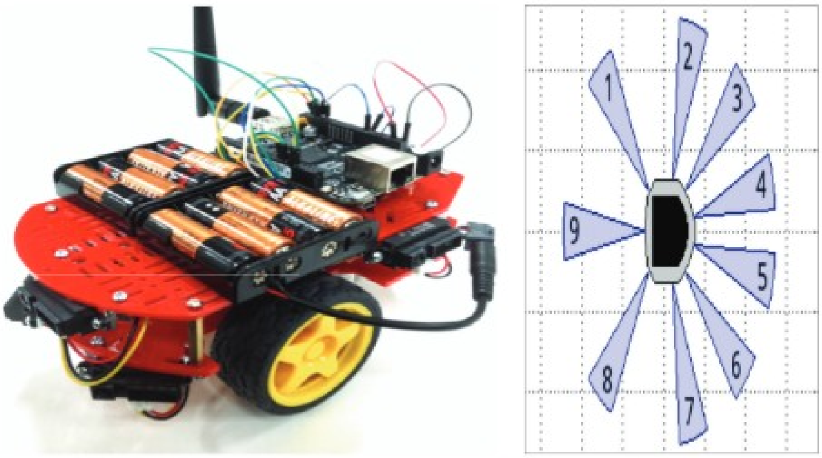

Figure 1: The robot and its sensor arrangement

The following figure shows the kinematic model of the robot. The wheel velocities are represented by V_L and V_R. The current orientation of the robot with respect to a certain fixed reference direction is given by θ. The position of the robot in a certain fixed coordinate system is given by (x,y). Thus the robot's state can be defined totally by the values (x,y, θ).

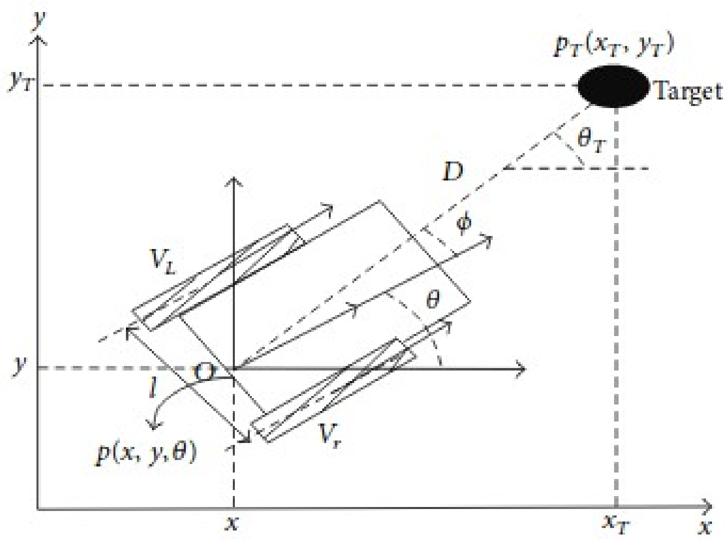

Figure 2: The robot kinematic model

The equations governing the robot's kinematics are given as:

$$\frac{dx}{dt} = \frac{V_L + V_R}{2}\cos(\theta)$$
$$\frac{dy}{dt} = \frac{V_L + V_R}{2}\sin(\theta) \qquad (1)$$
$$\frac{d\theta}{dt} = \frac{V_L - V_R}{2}$$

In a dead reckoning case, the update equations can be written as:

$$x_{k+1}=x_k+T\frac{V_{Lk}+V_{Rk}}{2}\cos(\theta)$$
$$y_{k+1}=y_k+T\frac{V_{Lk}+V_{Rk}}{2}\sin(\theta) \qquad (2)$$
$$\theta_{k+1}=\theta_k+T\frac{V_{Lk}-V_{Rk}}{2}$$

where T is the sampling time and the subscripts denote the number of the observation.

Design of the Fuzzy Logic Controller

A fuzzy logic controller design is proposed for both obstacle avoidance and navigation. Basically, the structure of the fuzzy controller is composed of the following 3 blocks: the fuzzification, inference, and defuzzification.

For the navigation task, the two inputs are given by
1. The Angle difference between the target and current orientation of the robot, as shown in Figure 2.
2. The distance of the robot from the target.
The membership function for distance is shown below:

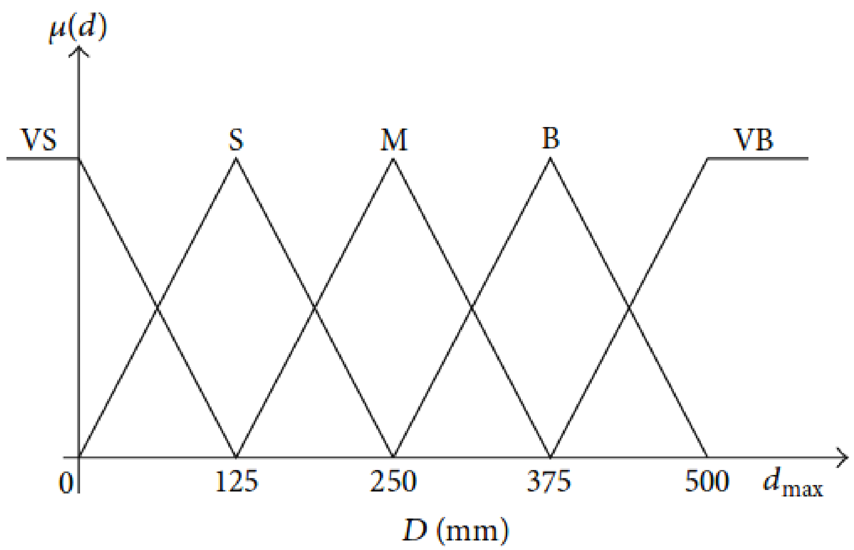

Figure 3: Membership function for distance from destination
VS: very small , S: small, M: medium, B: big, VB: very big

The membership function for angle is shown below:

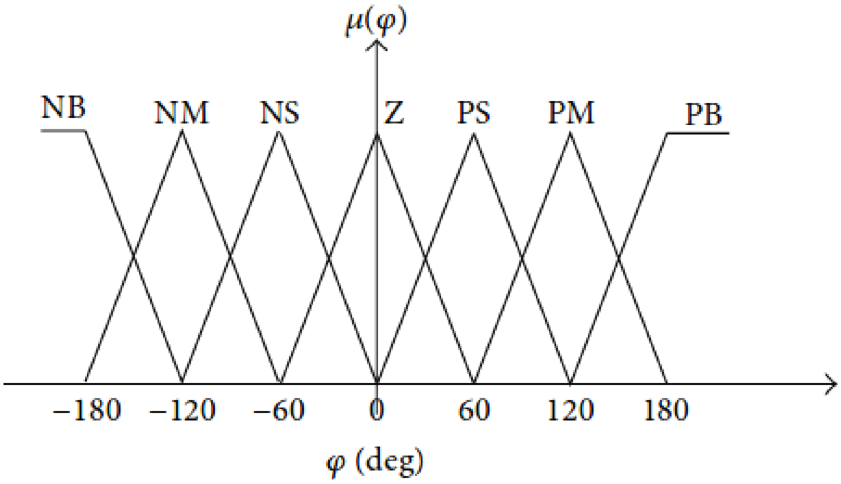

Figure 4: Membership function for error in angle

NB: negative big, NM: negative medium, NS: negative small, Z: zero, PS: positive small, PM: postive medium, PB: positive big

For the robot wheel velocities, a fuzzy control algorithm is proposed. The rule base is given as follows:

TABLE 1: The fuzzy rule bases for the controller.

		Angle (φ)													
		NB		NM		NS		Z		PS		PM		PB	
		V_R	V_L	V_R	V_L	V_R	V_L	V_R	V_L	V_R	V_L	V_R	V_L	V_R	V_L
Distance (D)	VS	B	Z	M	Z	F	Z	S	F	Z	F	Z	F	Z	M
	S	VB	Z	B	Z	M	Z	S	F	Z	M	Z	B	Z	VB
	M	VB	Z	VB	Z	B	Z	M	M	Z	B	Z	VB	Z	VB
	B	VB	Z	VB	Z	VB	Z	B	B	Z	VB	Z	VB	Z	VB
	VB	VB	Z	VB	Z	VB	Z	VB	VB	Z	VB	Z	VB	Z	VB

where VB: very big, B: big, M: medium, S: small, VS: very small, Z: zero.
For each combination of Angle and distance, both values of V_R and V_L are decided by this rule. For example If distance is medium and the angle difference is negative medium, then right wheel velocity is very big and left wheel velocity is zero.

For the values of VB, B, M, S, VS, Z following membership function is used.

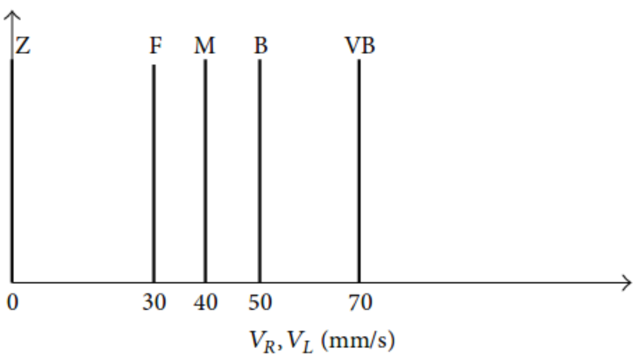

Figure 5: Velocity membership function

Obstacle Avoidance Controller (OAC)

After introduction of the Obstacle Avoidance Controller (OAC), we get a control system based on two kinds of fuzzy behaviours with a coordination unit for switching between these controllers. The first controller is for convergence towards the goal as described in the path planning algorithm in previous section and the second controller is for obstacle avoidance using the IR sensors. These two controllers can be combined into one Fuzzy Logic Controller to Avoid Obstacle (FLCAO). This is done because use of two fuzzy controllers consumes more resources. A Sugeno model is used for implementation.

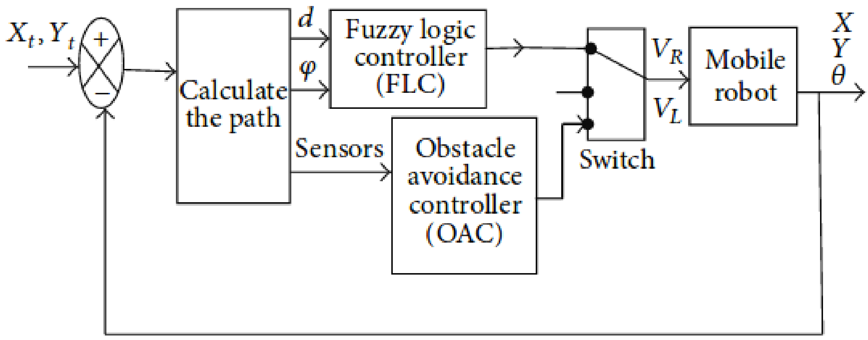

Figure 6: Controller with separate Fuzzy controller and Obstacle avoidance Controller

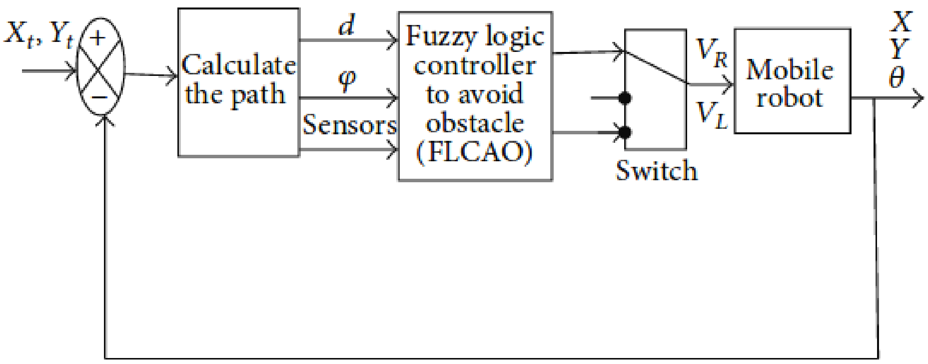

Figure 7: Combined navigation and obstacle avoidance controller (FLCAO).

There are 3 inputs (distance, angle orientation and the values of the 8 IR sensors) to the FLCAO and two outputs (VL and VR). For the FLCAO, the distance d has a universe of discourse between 0 to 2mm, for the angles it is -4 to 4 radians. Note that the distance is not from the destination but from the desired path, thus more like distance error. The following membership functions are defined for distance, angle and sensor measurements:

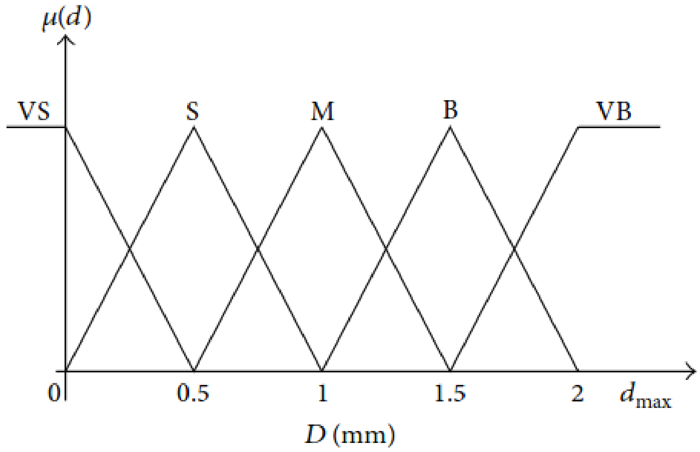

Figure 8: Membership function for distance error

VS: very small, S: small, M: medium, B: big, VB: very big

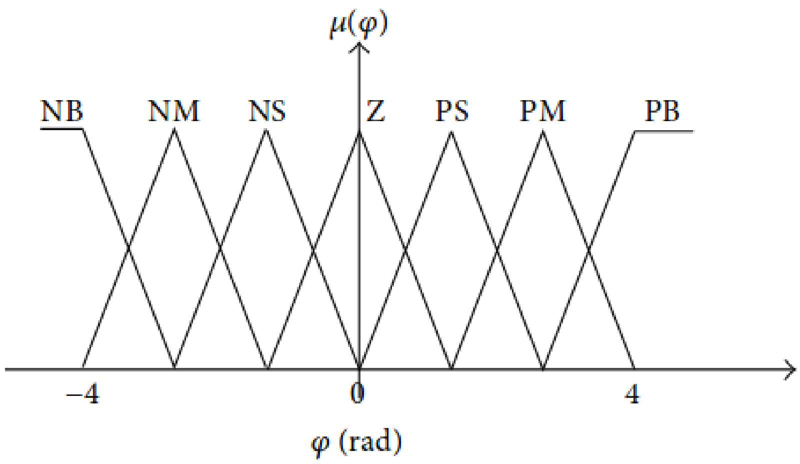

Figure 9: Membership function for angle error

NB: negative big, NM: negative medium, NS: negative small, Z: zero, PS: positive small, PM: postive medium, PB: positive big

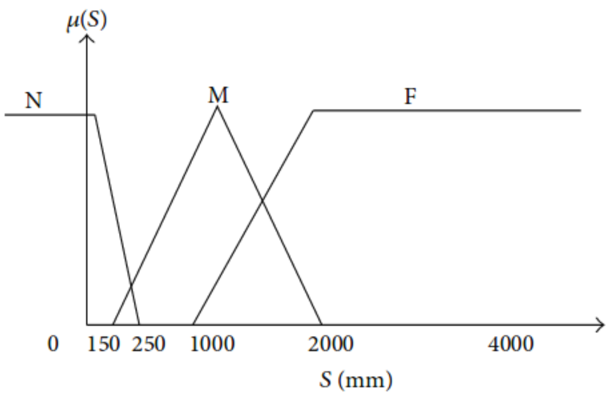

Figure 10: Membership function for sensor readings

N:Near, M: Medium, F: Far

Now a 62 valued rule base is defined using Sugeno type fuzzy model. Sugeno output membership functions are either linear or constant. The complete rule base is not given in the paper, but an example is as follows:

If (Distance is VS) and (Angle is NB) and (S1 is F) and (S2 is F) and (S3 is F) and (S4 is F) then (V_R is B) (V_L is 0)

The input membership function for velocity is given by:

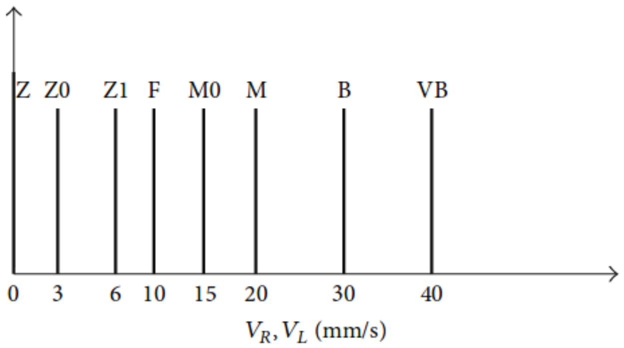

Figure 11: Velocity membership function

Results

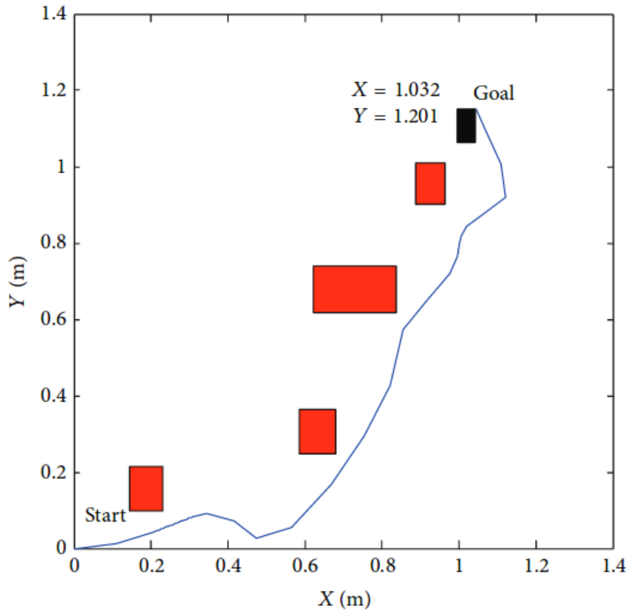

Figure 12: Obstacle avoidance simulation

The MATLAB simulation results show that the fuzzy controller is functional and is able to guide the robot to its destination while avoiding obstacles. If the path planning only result is visualized, without the OAC, it can still be seen that the path always converges to the destination.

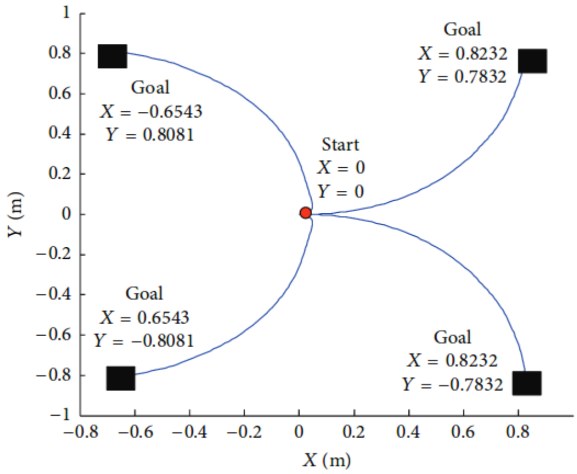

Figure 13: Navigation simulation

Summary

A navigation algorithm and a combined fuzzy logic controller and obstacle avoidance system for mobile robots is proposed in the paper. Simulation results show that the algorithm is convergent and the obstacle avoidance system is able to round the obstacles.

Future Work

This paper is entirely based upon simulation and thus doesn't take into account the error of real world sensors. Thus a real world demonstration of the algorithm working would be an achievement. Also it assumes that the distance of destination from the robot is accurately available which is not always the case in autonomous robotics, thus we may include the fuzziness in that variable as well for a separate research. Also, this paper uses low cost IR sensors that are generally used in hobby robotics, while for a further research scope, we may use state-of-the-art LASER based sensors for the same purpose, to make this algorithm effective for self driving cars as well. An analysis about the optimality and space-time complexity of the algorithm may be done too.

References

1. Hajer Omrane, Mohamed Slim Masmoudi, and Mohamed Masmoudi, “Fuzzy Logic Based Control for Autonomous Mobile Robot Navigation,” Computational Intelligence and Neuroscience, vol. 2016, Article ID 9548482, 10 pages, 2016. doi:10.1155/2016/9548482 https://www.hindawi.com/journals/cin/2016/9548482/

2. Mohammed Faisal, Ramdane Hedjar, Mansour Al Sulaiman, and Khalid Al-Mutib,Fuzzy logic navigation and obstacle avoidance by a mobile robot in an unknown dynamic environment,International Journal of Advanced Robotic Systems 10 (2013), no. 1, 37 http://journals.sagepub.com/doi/10.5772/54427

3. http://in.mathworks.com/help/fuzzy/what-is-sugeno-type-fuzzy-inference.html

4. K. Liu and F. L. Lewis, "Fuzzy logic-based navigation controller for an autonomous mobile robot," *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, San Antonio, TX, 1994, pp. 1782-1789 vol.2. doi: 10.1109/ICSMC.1994.400109 http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=400109&isnumber=9020