



Journal of Applied Sciences

ISSN 1812-5654

science
alert

ANSI*net*
an open access publisher
<http://ansinet.com>

Fuzzy Based Obstacle Navigation and Effective Path Selection with Laser Range Finder

P. Prasanna, S. Sethuraman, N.S. Manigandan and K. Ramkumar

School of Electrical and Electronics Engineering, SASTRA University, 613 401, Tamil Nadu, India

Abstract: The efficient navigation of a mobile robot in any environment lies in effectively identifying the obstacles and the free space. Once they are identified the robot has to decide the free space so that it can reach the destination in an optimal path. There are several existing algorithms which has been adopted in the past to cater this issue. The algorithm's accountability not just rest with collision free navigation, but also in finding a solution with lesser time complexity. In this proposed method Laser Range Finder (LRF) readings are fuzzified using a linear fuzzy membership function. Defuzzification is done using a deciding distance which is applied as λ -cut. This provides a solution set with all possible collision free navigable paths locally (Perception phase). Finally, the path is chosen based on the environmental obstacle density (Decision phase). Experimental validation of the proposed algorithm was done in cluttered, sparse and closed environment with a Coroware's mobile robot and the performance was highly satisfactory.

Key words: Obstacle avoidance, non holonomic robot, navigation, path selection

INTRODUCTION

Mobile robot navigation avoiding collisions with obstacles in an unknown environment (Abiyev *et al.*, 2010; Shi *et al.*, 2010; Fernandez *et al.*, 2004) is a challenging problem to be addressed. The presence of an obstacle, its distance from the robot and its features can be detected using on board sensors like Laser Range Finder (LRF) (Qiu and Han, 2009), Ultrasound Sensors (US) (Borenstein and Koren, 1988), camera, Inertial Measurement Unit (IMU) and Global Positioning System (GPS). In sensor fusion techniques the reliability factor on the fused data increases with more number of sensors deployed for the purpose of information extraction. On the contrary as the number of sensors on board increases the following factors becomes an important issue. (1) The load on the mobile robot increases, (2) The power demand on the battery increases, (3) The quantum of data is very high thus occupying large memory space and (4) The navigation problem becomes computationally intensive hence system becomes slower. Having the above limitations in mind it may be concluded that information extraction and decision making from a single and the most reliable sensor can prove effective and efficient. The data from other sensors can be used when the reliability on the current sensor decreases, or there is a sensor failure. When decision has to be taken based on data from a single sensor it is better to go for some logical decision making tools, one such tool is fuzzy logic (Yang and Li, 2002).

Autonomous exploration is a fundamental problem to solve as an autonomous robot carries out tasks in real unknown environments (Leonard *et al.*, 1992; Lumelsky *et al.*, 1989; Chen and Huang, 1994; Edlinger and von Puttkamer, 1994; Senthilkumar and Bharadwaj, 2012). Sensor based exploration (Surmann *et al.*, 2003), motion planning, localization (Duckett and Nehmzow, 1998; Fox *et al.*, 1998), simultaneous mapping (Yamauchi *et al.*, 1998) and roadmap (Van den Berg and Overmars, 2005) are processes to be coordinated to achieve automation in navigation of mobile robots in unknown environments. Application of Kalman filter (Simon, 2010; Ramkumar and Manigandan, 2012) increases the computational complexity, though it provides a better result. In this study a method for effective obstacle navigation using LRF is proposed. The algorithm uses fuzzy based technique (Samsudin *et al.*, 2011) to identify every navigable path locally and to provide a knowledge string of the environment. Based on the solution set generated the robot perceives the nature of the environment and decides a path accordingly.

MATERIALS AND METHODS

System architecture: The algorithm has two phases perception and decision. The two phases are implemented as separate modules. The diagram in Fig. 1 depicts the schematic of the software architecture depicting flow of data in the process. The laser sensor and encoder hold

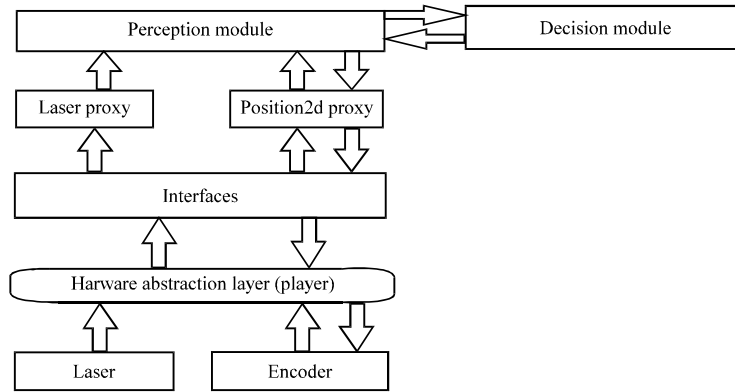


Fig. 1: Schematic of software architecture

communication with the perception module through the HAL (Hardware Abstraction Layer) provided by the player toolkit.

The Gazebo toolkit (PSGP, 2011) provides the functionality of proxies which abstracts the layer of communication between the device and the program. Corresponding device proxies provides appropriate methods to control the devices. The perception module senses the environment and updates the robot's current position from encoder by Laser proxy and Position2d proxy correspondingly. Perception module then interacts with the decision module to obtain the appropriate solution. As the solution is to control the motor which is done by Position2d proxy, the control flows from perception module to position2d proxy only.

Implementation: The algorithm implemented makes the robot move forward until it is confronted by an obstacle and then it decides a navigable path. The perception phase computes all the navigable paths in the environment. When the forward path is challenged, it passes the solution set to the decision phase so that an appropriate path is selected based on the obstacle density in the environment. The detailed implementation of the algorithm is as follows.

Perception phase: During this phase the robot keeps moving forward (Hsu and Hwang, 1998) sensing the environment. The algorithm is supplied with a parameter safe distance which keeps the sensor at a distance safe from the obstacles and prevents it from colliding from the obstacle. This should be greater than $2 \times l$, where l is the length of the robot. Another parameter is deciding distance; a suitable distance chosen that is greater than safe distance and closer to the range of the sensor with maximum credibility factor.

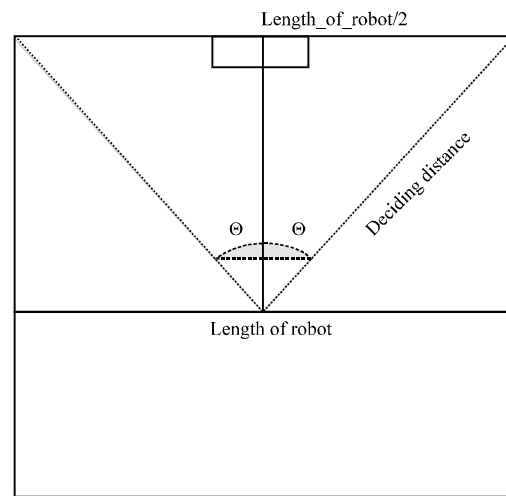


Fig. 2: Geometry to calculate THRES

Membership function: The membership function for the LRF readings is derived as $\mu(x) = \frac{x}{1+\max(x)}$, if $\text{safe_distance} \leq x \leq \text{max_range}$, where x LRF readings. This provides a fuzzy set with values (0,1) corresponding to the environment.

Defuzzification: The defuzzification by λ -cut is deployed to determine all the navigable paths for the robot. The λ value is chosen based on the deciding distance. The fuzzy value for deciding distance is the λ . Every fuzzy value λ -cut is assigned 1 otherwise 0. All the possible paths are determined from the defuzzified crisp set obtained. The minimum width (THRES) for navigation is obtained from the number of range readings lying above the deciding distance as follows.

With known angular resolution, dimension, safe distance, FOV, deciding distance, THRES is computed with Eq. 1 and from Fig. 2.

Deciding distance is user chosen value \geq Safe distance Safe:

$$\theta = \sin^{-1}(\text{length of robot}/(2 \times \text{deciding distance}))$$

$$\text{THRES} = 2 \times \theta / (\text{sensor angular resolution}) \quad (1)$$

The THRES consecutive number of range readings should be 1 in order to be considered as potential solution.

Decision phase: The following is the pseudo code in deciding the path by analyzing the nature of the environment.

```

If length of  $S_i > 0$ :
  If Counterfailed  $> 3$ :
    Environment = CLUTTERED
    Take the solution from the set S with largest count
  Else:
    Environment = SPARSE
    Take the solution from the set S with minimum turn
  End if
  Save the set  $S_i$  and the solution taken
Else:
  Environment = CLOSED
  Backtrack to the previous decision point
  Decide from  $S_{i-1}$  - {solution}
End if
    
```

Based on the value of Counterfailed, the robot learns the nature of the environment. The parameter Counterfailed associates to numbers free spaces that are not potential navigable paths. With this, the robot classifies the environment into one of the following three types and decides a path based on that.

Cluttered environment: When the Counterfailed > 3 which signifies that there are more than three free spaces but cannot accommodate the robot. The robot learns that it is in a cluttered environment (Fig. 3a). In such a case the robot finds a path having the widest navigable width from the set S which comprises of all the possible paths from that position of the robot. It tries to evade the cluttered obstacle in a way that provides a better navigable path.

Sparse environment: If the Counterfailed < 3 , the robot assumes that it is in a sparse environment (Fig. 3b). The robot from its set of solution, S chooses that path which has relatively minimum turn angle with respect to the current pose. This is better as the robot's orientation is not disturbed much when confronted with fewer obstacles.

Closed environment: The worst case lies in the fact that when the environment is closed (Fig. 3c). The



Fig. 3(a-c): (a) Mobile robot encountering a cluttered environment, (b) Mobile robot encountering a sparse environment and (c) Mobile robot encountering a closed environment

solution set is null. Whenever a solution set becomes null the robot retraces the path to reach previous decision point and chooses the path not chosen. Retracing the path in a static environment does not result in collision. When nth position is closed it goes to the (n-1)th position which is possible.

The algorithm saves the solution set at every decision point and laser scan to aid exploration and mapping.

ALGORITHM

The proposed algorithm is as follows. For each scan obtained from the LRF:

```

If path in front is navigable:
  Move forward
Else:
  S = { }
  Fuzzify the entire LRF reading using the function  $\mu(x) = x/(1+\text{maxrange})$ 
  Apply  $\lambda$ -cut to the fuzzy set
  Counterfailed = 0
  For each element in the fuzzy set:
    If element is 0 || end of the fuzzy set is reached:
      If count>THRES && count>0:
        Solution = (index,count)
        S = S U {Solution}
      Else:
        Counterfailed+ = 1
      Endif
    Count = 0
  Else:
    Increment count
  Endif
EndFor
EndIf

```

DECISION PHASE

```

If length of S>0:
  If Counterfailed>3:
    Environment = CLUTTERED
    Take the solution from the set S with largest count
  Else:
    Environment = SPARSE
    Take the solution from the set S with minimum turn
  End if
  Save the set S, and the solution taken
Else:
  Environment = CLOSED
  Backtrack to the previous decision point
  Decide from Sn-1-{solution}
Endif

```

The THRES in the algorithm is obtained from Eq. 1. S mentioned in the algorithm denotes the set of all possible solutions in the environment at a particular decision point. Counterfailed keeps count of set of continuous values

that are not potential solutions. The value of the Counter failed is directly proportional to the number of obstacles in the environment.

The algorithm above makes the robot move forward until confronted by obstacles. As the robot keeps sensing the environment, it finds every potential solution with respect to the deciding distance. Then chooses a path from the set of potential solutions. The chosen solution depends on the type of the environment the sensor is in.

The number of solutions and Counter failed is considered to categorize the environment. In a SPARSE environment, the solution having minimum turn is chosen. On the other hand in a CLUTTERED environment the sensor chooses the solution having maximum width. It goes to the previous decision taken point if the environment is CLOSED.

RESULTS AND DISCUSSION

The algorithm was tested in real time and graphs of fuzzy set and defuzzified crisp sets in different environments are provided below to explain the results briefly. The data is obtained from the LRF of the robot. Value of λ in λ -cut in all the examples quoted is assumed as 0.3.

In Fig. 4a, the robot is confronted by cluttered obstacles 3. This is known from the number of spaces that cannot accommodate the robot through it. The robot then decides to navigate through the path having larger width which is found from the count in the solution set. Figure 4b, shows the values post λ -cut. This clearly shows two navigable paths.

In Fig. 5a, the robot is in front of sparse obstacles. The set S, comprising of every possible solutions from that pose has lesser number of solutions and the Counterfailed also denotes the existence of sparse obstacles. The robot then decides to navigate through the path having minimum turn angle which is found from the solution set. The defuzzified crisp set's graphical representation in Fig. 5b, shows the values post λ -cut.

From Fig. 6a, it is clear that the robot has a closed environment in front of it which is learnt by the robot from the nature of the solution set which has zero navigable paths. The robot reverts back to previous decision point. The Fig. 6b, shows that the robot has not detected the threshold distance to navigate the obstacle, so it re-tracks.

Shi *et al.* (2010) uses coordinate transformation and Kalman filtering which are computationally very intensive and the environment considered is also partially known.

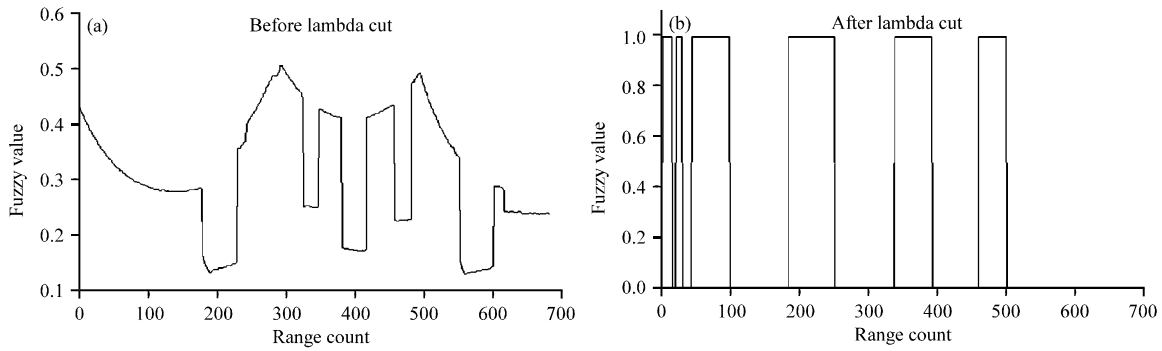


Fig. 4(a-b): (a) Fuzzy set-cluttered environment and (b) Defuzzified set-cluttered environment

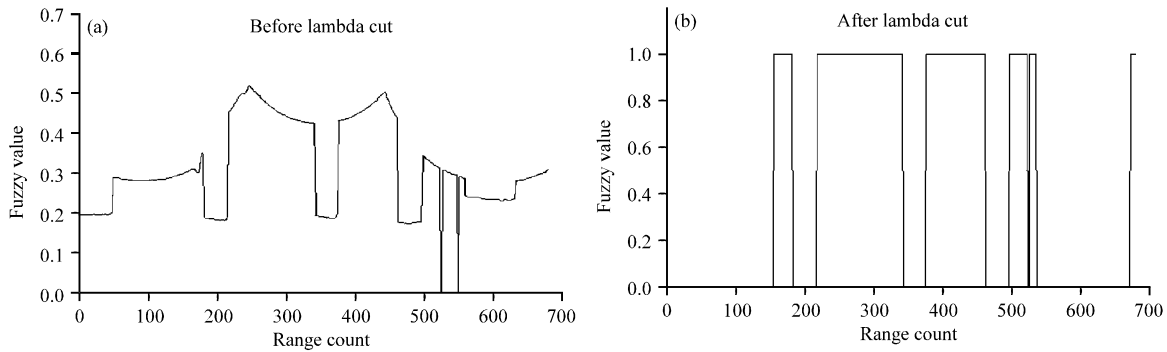


Fig. 5(a-b): (a) Fuzzy set-sparse obstacle environment and (b) Defuzzified set-sparse obstacle environment

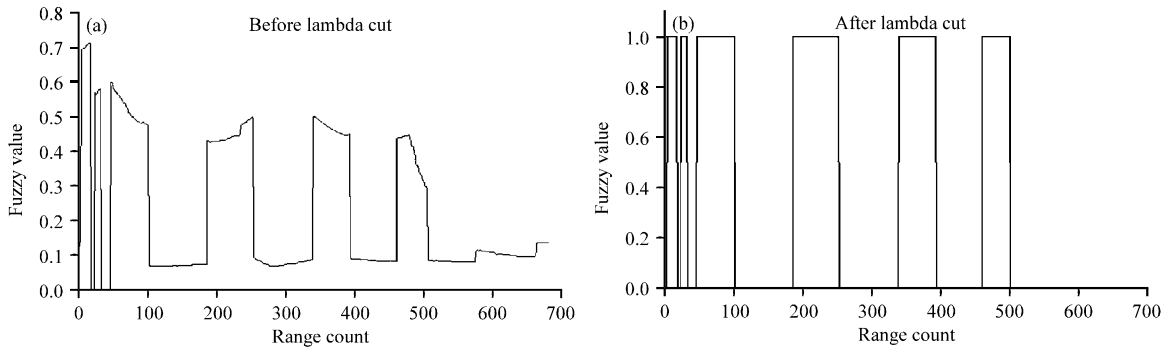


Fig. 6(a-b): (a) Fuzzy set-closed environment and (b) Defuzzified set-closed environment

The proposed algorithm in this study is fast, simple and direct to implement as it uses the raw LRF reading without any transformation in coordinates. This algorithm can be implemented in any unknown and static environment.

CONCLUSION

The obstacle avoidance algorithm proposed in this study is tailored to guide a robot in exploring an indoor environment so that every reachable area is visited

optimally. The perception module gives all navigable paths in the environment which is quite suitable for the graph-based exploration as a solution set. The robot selects the best solution from the set which provides optimal obstacle avoidance, minimum turning angle and shortest distance to reach a target. The solution set and the pose of the robot are saved after each decision thereby assisting the robot in backtracking and in choosing a different path to evade the obstacle and explore the environment. The environment classification

as cluttered, sparse and closed will be helpful in optimal path planning. When the robot backtracks to previous decision point, an alternate path is chosen from the solution set at this decision point exclusive of the path earlier taken. If a better path is desired then the solution set may be shrunk by increasing the λ -cut value. This process explores every solution obtained from the perception module and completes when all the solutions has been explored completely.

ACKNOWLEDGMENTS

The authors wish to express their sincere thanks to the CAIR-DRDO, New Delhi, India for their financial support (Project ID: ERIP-ER-0903808 dt 23.4.10). They also wish to acknowledge SASTRA University, Thanjavur for extending infrastructural support to carry out this work.

REFERENCES

- Abiyev, R., D. Ibrahim and B. Erin, 2010. Navigation of mobile robots in the presence of obstacles. *Adv. Eng. Software*, 41: 1179-1186.
- Borenstein, J. and Y. Koren, 1988. Obstacle avoidance with ultrasonic sensors. *IEEE Trans. Rob. Autom.*, 4: 213-218.
- Chen, Z. and C.M. Huang, 1994. Terrain exploration of a sensor-based robot moving among unknown obstacles of polygonal shape. *Robotica*, 12: 33-44.
- Duckett, T. and U. Nehmzow, 1998. Mobile robot self-localisation and measurement of performance in middle-scale environments. *Rob. Autom. Syst.*, 24: 57-69.
- Edlinger, T. and E. von Puttkamer, 1994. Exploration of an indoor-environment by an autonomous mobile robot. *Proceedings of the IEEE/RSJ/GI International Conference on Intelligent Robots and Systems*, Volume 2, September 12-16, 1994, Munich, Germany, pp: 1278-1284.
- Fernandez, J.L., R. Sanz, J.A. Benayas and A.R. Dieguez, 2004. Improving collision avoidance for mobile robots in partially known environments: The beam curvature method. *Rob. Autom. Syst.*, 46: 205-219.
- Fox, D., W. Burgard and S. Thrun, 1998. Active markov localization for mobile robots. *Rob. Autom. Syst.*, 25: 195-207.
- Hsu, J.Y.J. and L.S. Hwang, 1998. A graph-based exploration strategy of indoor environments by an autonomous mobile robot. *Proceedings of the IEEE International Conference on Robotics and Automation*, Volume 2, May 16-20, 1998, Leuven, Belgium, pp: 1262-1268.
- Leonard, J.J., H.F. Durrant-Whyte and I.J. Cox, 1992. Dynamic map building for an autonomous mobile robot. *Int. J. Rob. Res.*, 11: 286-298.
- Lumelsky, V., S. Mukhopadhyay and K. Sun, 1989. Sensor-Based terrain acquisition: Thesightseer strategy. *Proceedings of the 28th IEEE Conference on Decision and Control*, December 13-15, 1989, Tampa, FL., USA., pp: 1157-1161.
- PSGP, 2011. The player robot device interface. *Player Stage Gazebo Project*. <http://playerstage.sourceforge.net/doc/Player-3.0.2/player/index.html>
- Qiu, Q. and J. Han, 2009. An implementation of typical-obstacle detection and recognition with laser range finder. *Proceedings of the IEEE International Conference on Intelligent Computing and Intelligent Systems*, Volume 2, November 20-22, 2009, Shanghai, China, pp: 742-746.
- Ramkumar, K. and N.S. Manigandan, 2012. Stochastic filters for mobile robot SLAM problems: A review. *Sens. Transducers J.*, 138: 141-148.
- Samsudin, K., F.A. Ahmad and S. Mashohor, 2011. A highly interpretable fuzzy rule base using ordinal structure for obstacle avoidance of mobile robot. *Applied Soft Comput.*, 11: 1631-1637.
- Senthilkumar, K.S. and K.K. Bharadwaj, 2012. Multi-Robot exploration and terrain coverage in an unknown environment. *Rob. Autom. Syst.*, 60: 123-132.
- Shi, C., Y. Wang and J. Yang, 2010. A local obstacle avoidance method for mobile robots in partially known environment. *Rob. Autom. Syst.*, 58: 425-434.
- Simon, D., 2010. Kalman filtering with state constraints: A survey of linear and nonlinear algorithms. *IET Control Theory Appl.*, 4: 1303-1318.
- Surmann, H., A. Nuchter and J. Hertzberg, 2003. An autonomous mobile robot with a 3D laser range finder for 3D exploration and digitalization of indoor environments. *Rob. Autom. Syst.*, 45: 181-198.
- Van den Berg, J.P. and M.H. Overmars, 2005. Roadmap-Based motion planning in dynamic environments. *IEEE Trans. Robot.*, 21: 855-897.
- Yamauchi, B., A. Schultz and W. Adams, 1998. Mobile robot exploration and map-building with continuous localization. *Proceedings of the IEEE International Conference on Robotics and Automation*, Volume 4, May 16-20, 1998, Leuven, Belgium, pp: 3715-3720.
- Yang, S.X. and H. Li, 2002. An autonomous mobile robot with fuzzy obstacle avoidance behaviors and a visual landmark recognition system. *Proceedings of the 7th International Conference on Control, Automation, Robotics and Vision*, Volume 3, December 2-5, 2002, Singapore, pp: 1579-1584.