

01 Jan 2005

## An Embedded Real-Time Neuro-Fuzzy Controller for Mobile Robot Navigation

Nian Zhang

Daryl G. Beetner

*Missouri University of Science and Technology, daryl@mst.edu*

Donald C. Wunsch

*Missouri University of Science and Technology, dwunsch@mst.edu*

B. Hemmelman

*et. al. For a complete list of authors, see [https://scholarsmine.mst.edu/ele\\_comeng\\_facwork/865](https://scholarsmine.mst.edu/ele_comeng_facwork/865)*

Follow this and additional works at: [https://scholarsmine.mst.edu/ele\\_comeng\\_facwork](https://scholarsmine.mst.edu/ele_comeng_facwork)

 Part of the [Electrical and Computer Engineering Commons](#)

---

### Recommended Citation

N. Zhang et al., "An Embedded Real-Time Neuro-Fuzzy Controller for Mobile Robot Navigation," *Proceedings of the 14th IEEE International Conference on Fuzzy Systems (2005, Reno, NV)*, Institute of Electrical and Electronics Engineers (IEEE), Jan 2005.

The definitive version is available at <https://doi.org/10.1109/FUZZY.2005.1452413>

This Article - Conference proceedings is brought to you for free and open access by Scholars' Mine. It has been accepted for inclusion in Electrical and Computer Engineering Faculty Research & Creative Works by an authorized administrator of Scholars' Mine. This work is protected by U. S. Copyright Law. Unauthorized use including reproduction for redistribution requires the permission of the copyright holder. For more information, please contact [scholarsmine@mst.edu](mailto:scholarsmine@mst.edu).

# An Embedded Real-Time Neuro-Fuzzy Controller for Mobile Robot Navigation

<sup>1</sup>Nian Zhang, <sup>2</sup>Daryl Beetner, <sup>3</sup>Donald C. Wunsch II, <sup>4</sup>Brian Hemmelman and <sup>5</sup>Abul Hasan

<sup>1,4,5</sup>South Dakota School of Mines and Technology, Department of Electrical and Computer Engineering  
501 E. St. Joseph Street, Rapid City, SD 57701

<sup>2,3</sup>University of Missouri-Rolla, Department of Electrical and Computer Engineering  
1870 Miner Circle, Rolla, MO 65409

<sup>1</sup>[nian.zhang@sdsmt.edu](mailto:nian.zhang@sdsmt.edu), <sup>2</sup>[daryl@umr.edu](mailto:daryl@umr.edu), <sup>3</sup>[dwunsch@ece.umr.edu](mailto:dwunsch@ece.umr.edu), <sup>4</sup>[brian.hemmelman@sdsmt.edu](mailto:brian.hemmelman@sdsmt.edu),  
<sup>5</sup>[abul.hasan@sdsmt.edu](mailto:abul.hasan@sdsmt.edu)

**Abstract-** A reactive fuzzy logic based control strategy was developed for mobile robot navigation. To decrease the number of fuzzy rules and related processing, a RAM-based neural network was combined with the fuzzy logic strategy. The fuzzy rules are used to interpret sensor information. The neural network uses results from the fuzzy logic as well as environmental information to make navigation decisions. The feasibility of this neuro-fuzzy approach was demonstrated on a mobile robot using a simple, 8-bit microcontroller. Experiments show the approach works well, as the robot was able to successfully avoid objects while seeking a goal in real-time. The neuro-fuzzy approach is code-efficient, fast, and easy to relate to the physical world.

**Keywords:** Fuzzy systems, RAM-based neural network, robotic navigation, microprocessor system.

## I. INTRODUCTION

The challenge to autonomously navigate an unexplored or unknown environment is to find local obstacles, navigate around these obstacles, and reach a goal destination with very limited computational resources.

There are several difficulties with this problem. First, sensor data is often noisy and inaccurate. Determining the best actions based on sensor inputs is difficult, particularly in complex environments with many obstacles or where obstacles may be hidden or beyond sensor range.

As we know, fuzzy logic handles this imprecise and uncertain data to produce complex decision outcomes [1]. Fuzzy logic is also well suited for limited computational platforms as it is intrinsically modular and computationally simple [2].

The second difficulty is that the control strategy must be suitable for performance on a simple microprocessor system with modest capabilities in terms of both speed and memory.

A RAM-based neural network is a good choice for use with modest microcontroller systems [3]. RAM-based neural networks specialize in pattern classification and rely largely

on binary Boolean operations and memory reads and writes, rather than complex floating-point calculations.

A number of fuzzy logic approaches have been developed for mobile robot navigation. Chee et al. presented a two-layer fuzzy inference system that is capable of integrating the distance readings from different sensors and mapping the integrated results to the motion of the robot [4]. Fabrizi et al. proposed topology-based maps to represent the workspace of a mobile robot [5]. These maps capture the structure of the free space in the environment in terms of the basic topological notions of connectivity and adjacency. Topology-based maps can be automatically extracted from an occupancy grid built from sensor data defined on fuzzy values. Coradeschi et al. proposed a fuzzy computational theory of anchoring to incorporate cognitive processes into a physically embedded reasoning system [6]. Anchoring is the process of creating and maintaining the correspondence between symbols and percepts that refer to the same physical objects. Modeling this process using fuzzy set theoretic notions enables dealing with perceptual data that can be affected by uncertainty/impression and imprecise/vague linguistic descriptions of objects. Wasik et al. proposed a hierarchical behavior-based system that can perform several vision-based manipulation tasks by using different combinations of the same set of basic behaviors. Behaviors can run concurrently, and they are arbitrated through “if-then” rules [7]. Zhang et al. proposed a fuzzy logic approach to target the goal using a decision table and fuzzy rule base [8]. However, this approach was not applicable to obstacle avoidance. Yao et al. presented a RAM-based neural network method for obstacle avoidance [9]. It used a conventional RAM-based network to classify the local environment and choose appropriate steering actions. This method is good for a single task (i.e. obstacle avoidance). However, if the robot must perform multiple tasks (i.e. goal seeking and obstacle

avoidance), this method becomes incapable. All the above techniques were implemented only on computer simulation instead of a hardware implementation. None of these methods are well suited for application on a modest microcontroller-based robot system.

Some hardware implementations have been explored. Tunstel and Jamshidi described a fuzzy logic controller that provided a mobile robot equipped with a MC68HC11 microcontroller with the capability to exhibit a wall-following behavior [10]. Saffiotti et al. presented a mobile robot that can avoid obstacles on the way and seek for the goal. It was implemented on the mobile robot, named Flakey by SRI International [11]. Buschka et al. uses fuzzy logic to account for errors and imprecision in visual revognition, and for extreme uncertainty in the estimate of the robot's motion. It only requires an approximate model of the sensor system and a qualitative estimate of the robot's displacement, and it has a moderate computational cost. The method was demonstrated on a Sony AIBO legged robot in the RoboCup domain [12].

We present a novel neuro-fuzzy approach for mobile robot navigation that combines a fuzzy logic controller with a RAM-based neural network. The hardware implementation of the mobile robot embedded with a Phillips P89C51RC2 microcontroller was also described.

The rest of this paper is organized as follows. Section II introduces the mobile robot hardware design. Section III presents an approach for neuro-fuzzy mobile robot navigation. The design of the fuzzy system and neural network are described. Section IV presents the experimental results. In Section V, the performance of the neuro-fuzzy technique is discussed. The conclusions are also given.

## II. MOBILE ROBOT SYSTEM

Our experimental mobile robot was designed to find obstacles in its immediate environment, to navigate around those obstacles, and to detect and seek a goal destination.

The robot has two wheels used for driving and steering. A separate stepper motor is used to drive each back wheel to facilitate turns or forward and backward movement. The speed of the motors is set by the microcontroller. If there is no obstacle nearby, the robot will move quickly; otherwise, it will slow down.

The mobile robot is equipped with four infrared (IR) range sensors to detect obstacles. The sensors are evenly mounted in a semi-circle on the front of the robot, pointing in the left, left-front, right-front, and right directions. Distance is measured from 10 cm to 80 cm. In addition, four photoelectric sensors (i.e. goal-detectors) are mounted on the top of the robot to detect signals from a light source placed at the goal location. The sensors are numbered from one to four as shown in Fig. 1. The sensors are evenly spaced in a circle, pointing outward from the center. Each sensor has a typical detection angle of about  $90^\circ \pm 10^\circ$ .

## III. NEURO-FUZZY NAVIGATION SYSTEM DESIGN

While the robot explores an unknown environment, it is important for the robot to make a compromise between avoiding collisions with obstacles and moving toward the target. The robot detects obstacles and the goal location at the same time.

The neuro-fuzzy navigation system is shown in Fig. 1. The robot detects its environment using two groups of sensors. Four IR range sensors detect obstacles nearby and four photoelectric detectors find the orientation of the light source placed on the goal. Navigation is performed using two fuzzy logic controllers, one to process information from the left side of the robot and another to deal with the right. The fuzzy logic controllers consist of a rule base that fuzzifies range values and goal direction and produces defuzzified control actions. In addition, the controllers implement a reactive control strategy. For example, if the robot finds an obstacle in front of it and in the path to the goal, both fuzzy logic controllers will determine the best steering direction and speed to avoid collision while heading to the goal.

A RAM-based neural network chooses the reactive outputs from these fuzzy logic controllers based on its observation of the environment. The RAM-based neural network works as an action supervisor. It classifies the current environmental conditions based on sensor inputs and then chooses the best control response from the outputs of the two fuzzy logic controllers.

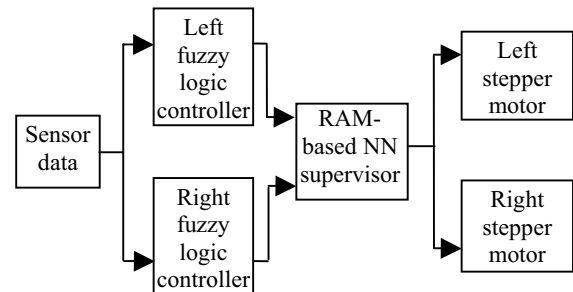


Fig. 1. Block diagram of the neuro-fuzzy navigation system.

The navigation algorithm uses a sense-act cycle where the robot senses nearby obstacles and the goal direction and makes navigation decisions. For each iteration, the fuzzy logic controllers and neural network evaluate data from the sensors and choose an appropriate steering direction and speed. Normally, the robot moves forward to the goal while surveying its environment until an obstacle is detected within a fuzzy range threshold. If an obstacle is detected, the steering direction and driving speed are modified by



## B. Neural Network Design

A RAM-based neural network is used to determine the direction of the obstacles, based on which the two fuzzy controllers would take action. The architecture of the neural network is shown in Fig. 5. It has two-layers. The input layer has four neurons and the output layer has six neurons. The inputs are *Left*, *Left-front*, *Right-front*, and *Right* obtained from the four IR range sensors. The six outputs are:

- Output 1: One or two obstacles in front of the robot (left-front side or right-front side or both);
- Output 2: One obstacle to the left-front of the robot, another to the right;
- Output 3: One obstacle to the right-front of the robot, another to the left;
- Output 4: One or two obstacles on the left side of the robot (left or left-front or both);
- Output 5: One or two obstacles on the right side of the robot (right or right-front or both);
- Output 6: More than two obstacles (obstacles on multiple sides).

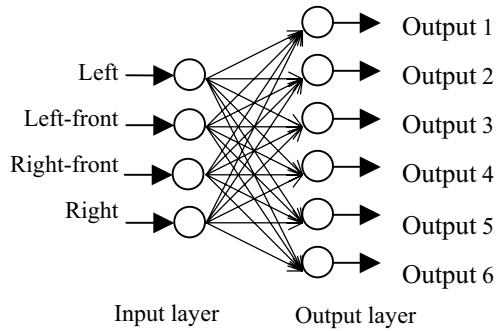


Fig. 5. The structure of the RAM-based neural network action supervisor.

The RAM-based network classifies the robot's immediate environment by recognizing patterns that were encountered during training. For RAM-based networks, the training does not require a direct measure of error, just an indication that the current training pattern is correct or incorrect. The training therefore proceeds very quickly. This is especially well suited to microcontroller-based real-time systems because it requires very modest computing capabilities.

The final command decision is based on the environment classification by the neural network and the outputs of the fuzzy logic controllers. The neural network classifier uses both fuzzy logic controllers and chooses appropriate actions to avoid obstacles and keep heading to the target. For case 1, the classifier selects the steering direction and velocity,  $(\theta, v)$  from either the left or right fuzzy logic controller; for case 2, 3 and 6, the classifier selects the larger  $\theta$  and  $v$  from the two fuzzy logic controllers; for case 4 the classifier selects the  $\theta$  and  $v$  from the left fuzzy logic controller; for case 5 the

classifier selects the  $\theta$  and  $v$  from the right fuzzy logic controller.

## IV. EXPERIMENTAL RESULTS

Three experiments were performed. Of particular concern was the case where a dead-end existed and the robot could be trapped, such as with a U-shaped obstacle pattern. The first experiment used a cluttered environment but with a clear path to the goal, as shown in Fig. 6. The goal is shown with a triangle. The robot's path is shown with circles. Obstacles are indicated with filled squares. The second experiment used multiple obstacles in a small U-shaped pattern, as shown in Fig. 7. The radius and the depth of this U-shape were no more than 30 cm. The third experiment used a combination of obstacles in a big U-shaped pattern, as shown in Fig. 8. Each environment was 2 m by 2 m in size. Navigation was attempted in these environments from a fixed starting point. In the figures shown, the robot started from the lower left corner and proceeded toward the goal located in the upper right corner.

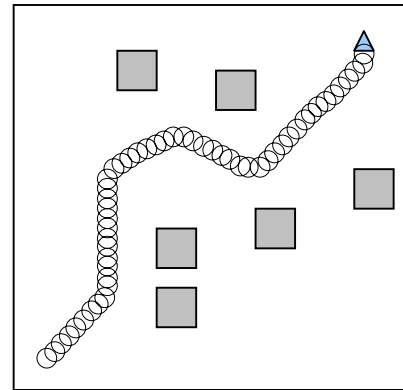


Fig. 6. The robot's path through a cluttered environment to its goal location.

In experiment 1, the robot was able to navigate its way through the obstacles to the goal with little difficulty. The environment shown in Fig. 7 was more challenging. The robot started by navigating into the U-shaped combination of objects; however, the neuro-fuzzy navigation algorithm was able to successfully navigate its way through these obstacles to the goal as long as the radius and the depth of the U-shape were not too large, i.e., less than 2 or 3 times the turning radius of the mobile robot. When the robot started navigating, it moved toward the goal at high speed because it saw no obstacles in its immediate environment. When it detected the U-shaped pattern of obstacles, it moved along the edge of the obstacles until it freed itself. Once free of the objects, the robot proceeded to the goal with high speed. In experiment

2, the robot successfully navigated past the “trap” to the goal.

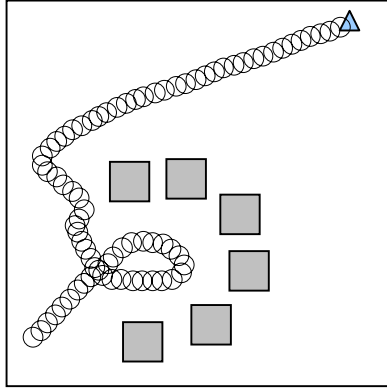


Fig. 7. The robot's path to its goal through an environment with obstacles in a small U-shaped configuration.

When we increased the radius of the U-shaped combination and moved the light source to the right behind the combination, the robot was trapped more easily. It turned around in the concave “trap” made by the U-shape and could not find an exit, as shown in Fig. 8.

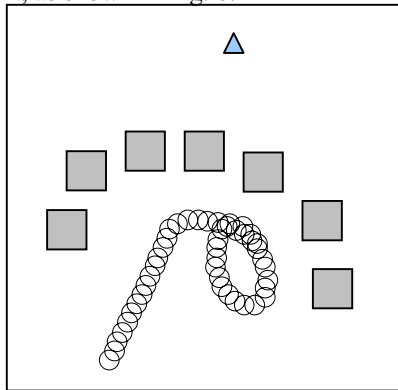


Fig. 8. The robot may get trapped in some large U-shaped obstacle configurations.

## V. DISCUSSION AND CONCLUSIONS

We proposed a neuro-fuzzy algorithm for reactive robot navigation. It performed reliably and in real-time on a simple 8-bit microcontroller system. This method can effectively deal with imprecise or noisy sensor information and coordinate conflicts among multiple reactive behaviors.

An efficient approach to reduce the number of fuzzy rules is presented. For our navigation system, the fuzzy controller is broken into two halves, which dramatically decreases the size of the overall rule base. A RAM-based neural network is used as a supervisor to these controllers (the two halves), to classify the environment and choose appropriate navigation decisions from the two controllers. The advantage of this combined fuzzy controller and RAM-based neural network

approach is that it requires very little computational power or memory while still maintaining the ability to handle imprecise or complex data. A possible drawback is that it may produce suboptimal solutions and cannot guarantee that the robot will reach its goal destination.

While this method of detecting the goal direction is simple, it is effective for demonstrating the quality of our navigation approach. Experimental results showed that the robot could avoid obstacles in a complex environment. The robot had a very modest microprocessor system (8 bits), a small amount of data memory (512 bytes), and only 16 KB of code memory, showing the algorithm could be implemented well on a comparatively modest computational platform. Compared with model-based navigation approaches, the neuro-fuzzy method reacts quickly and requires few computational resources.

Intelligent control strategies are needed to overcome the ambiguities in sensor data and reach appropriate navigation decisions.

## REFERENCES

- [1] John Yen, Reza Langari, *Fuzzy Logic: Intelligence, Control, and Information*, Prentice Hall, 1st edition, 1998.
- [2] Timothy J. Ross, *Fuzzy Logic with Engineering Applications*, John Wiley & Sons, 2nd edition, 2004.
- [3] J. Austin, *RAM-Based Neural Networks*, World Scientific, pp 8-9, 1998
- [4] B. Chee, S.Y.T. Lang and P.W.T. Tse, “Fuzzy Mobile Robot Navigation and Sensor Integration”, *Proceedings of the Fifth IEEE International Conference on Fuzzy Systems*, vol. 1, pp. 7 – 12, Sept. 8-11, 1996.
- [5] E. Fabrizi and A. Saffiotti, “Extracting Topology-Based Maps From Gridmaps,” *IEEE International Conference on Robotics and Automation*, vol. 3, pp. 2972-2978, 2000.
- [6] S. Coradeschi, D. Driankov, L. Karisson and A. Saffiotti, “Fuzzy Anchoring,” *The 10<sup>th</sup> IEEE International Conference on Fuzzy Systems*, vol. 1, pp. 111-114, 2001.
- [7] Z. Wasik and A. Saffiotti, “A Hierarchical Behavior-Based Approach to Manipulation Tasks,” *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 2, pp. 2780-2785, 2003.
- [8] N. Zhang and D. C. Wunsch II, “Fuzzy Logic in Collective Robotic Search”, *The 12<sup>th</sup> IEEE International Conference on Fuzzy Systems*, St. Louis, MO, vol. 2, pp. 1471 - 1475, May 25-28, 2003.

- [9] Q. Yao, D. G. Beetner, D. C. Wunsch II, and B. Osterloh, "A RAM-based Neural Network for Collision Avoidance in a Mobile Robot", *INNS-IEEE International Joint Conference on Neural Networks (IJCNN 2003)*, Portland, Oregon, pp. 3157-3160, July 20-24, 2003.
- [10] E. Tunstel and M. Jamshidi, "Embedded Fuzzy Logic-based Wall-following Behavior for Mobile Robot Navigation", *Proceedings of the First International Joint Conference of the North American Fuzzy Information Processing Society*, pp. 329 – 330, Dec. 18-21, 1994.
- [11] A. Saffioti, E. H. Ruspini and K. Konolige, "Blending Reactivity and Goal-Directedness In a Fuzzy Controller", *The 2<sup>nd</sup> IEEE International Conference on Fuzzy Systems*, vol. 1, pp. 134-139, 1993.
- [12] P. Buschka, A. Saffioti and Z. Wasik, "Fuzzy Landmark-Based Localization For a Legged Robot", *2000 IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2, pp. 1205-1210, 2000.