Hindawi Publishing Corporation
Computational Intelligence and Neuroscience
Volume 2016, Article ID 9548482, 10 pages
http://dx.doi.org/10.1155/2016/9548482

Research Article

Fuzzy Logic Based Control for Autonomous Mobile Robot Navigation

Hajer Omrane, Mohamed Slim Masmoudi, and Mohamed Masmoudi

*Department of Electrical Engineering, METS Laboratory, National School of Engineers of Sfax, University of Sfax,
Street of Soukra Km. 4, BP 1173, 3038 Sfax, Tunisia*

Correspondence should be addressed to Hajer Omrane; hajeromrane2011@yahoo.com

Received 27 May 2016; Accepted 2 August 2016

Academic Editor: Dong W. Kim

This paper describes the design and the implementation of a trajectory tracking controller using fuzzy logic for mobile robot to navigate in indoor environments. Most of the previous works used two independent controllers for navigation and avoiding obstacles. The main contribution of the paper can be summarized in the fact that we use only one fuzzy controller for navigation and obstacle avoidance. The used mobile robot is equipped with DC motor, nine infrared range (IR) sensors to measure the distance to obstacles, and two optical encoders to provide the actual position and speeds. To evaluate the performances of the intelligent navigation algorithms, different trajectories are used and simulated using MATLAB software and SIMIAM navigation platform. Simulation results show the performances of the intelligent navigation algorithms in terms of simulation times and travelled path.

1. Introduction

Robotics is an important area of research that uses knowledge across several disciplines such as mechanics, electronics, and computer engineering, to move a mobile robot in defined environment with some degree of autonomy.

One of the major challenges of the autonomous navigation for mobile robots is the detection and obstacles avoidance during the robot navigation task. This problem can be solved by relating different methods or algorithms in order to attain best results.

Fuzzy logic is used in the design of possible solutions to perform local navigation, global navigation, path planning, steering control, and rate control of a mobile robot [1].

Many research literatures used soft computer algorithms to control mobile robots in academic field as well as in the engineering field.

Yu et al. used Taguchi method to design an optimized fuzzy logic controller for trajectory tracking of a wheeled mobile robot [2].

Xiong and Qu elaborated a method for intelligent vehicles path tracking with two fuzzy controllers combining with vehicle control direction [3].

Selekwa et al. detailed the design of a preference-based fuzzy behaviour system for navigating robotic vehicles employing the multivalued logic framework. The proposed technique permits the robot to effectively navigate through cluttered environments [4].

Jia et al. developed a method of fuzzy control of the smart tennis chair using pressure sensors and omnidirectional wheels. The wheelchair can be master of a manual way with joystick device [5].

For this paper, we developed a fuzzy controller for the mobile robot navigation with different desired positions. Then, a second intelligent controller based on fuzzy logic has been developed to achieve the navigation and obstacle avoidance tasks. The latter is provided by infrared range (IR) sensors and the discrete kinematic model.

Zero-order Sugeno model is used to develop different fuzzy logic controllers. The implication method is simply multiplication, and the aggregation operator just includes all of the singletons.

This paper is organized into seven sections. In Section 2 the mathematical model of the robot is presented.

In Section 3, the fuzzy logic controller is developed and discussed in detail. In Section 4 simulation results show the

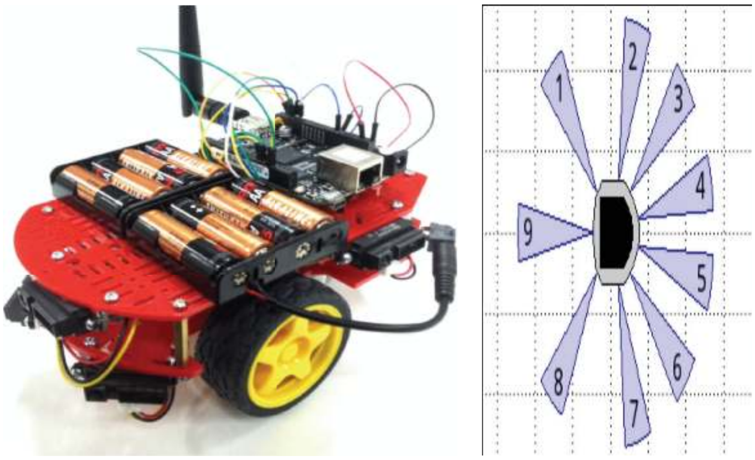


Figure 1: QuickBot robot and SIMIAM simulator model.

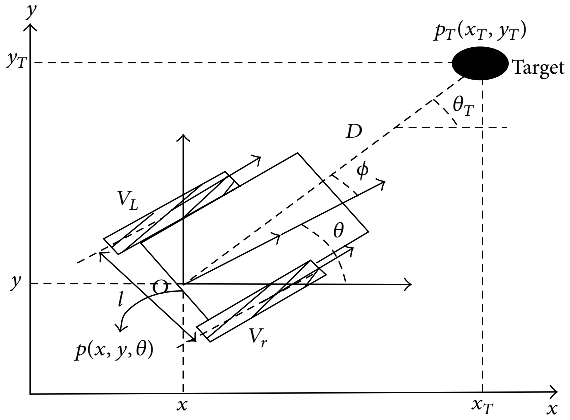


Figure 2: Kinematic model of the mobile robot.

efficiency of the proposed method. In Section 5, we describe the fuzzy logic controller to avoid obstacles. Section 6 presents the results of the simulation and experimental tests.

Finally, the conclusions based on the obtained results are given in Section 7.

2. The Kinematic Model of the Mobile Robot

Our mobile robot model is a unicycle robot type. It has two degrees of freedom and moved with two DC motors. The QuickBot is equipped with nine infrared range (IR) sensors, six are located in the front, and three are placed on its back. The robot has a two-wheel differential drive system moved by two DC motors equipped with optical encoder for each (Figure 1) [6].

The configuration of the mobile robot is characterized by the position (X, Y) and the orientation in a Cartesian coordinate.

Figure 2 shows the variables used in the kinematic model; it has several parameters that are as follows:

V_R: linear velocity of the right wheel.

V_L: linear velocity of the left wheel.

W: angular velocity of the mobile robot.

X: abscissa of the robot.

Y: intercept of the robot.

X, Y: the actual position coordinates.

θ: orientation of the robot.

L: the distance between the driving wheels.

The kinematic model [7] is given by these equations:

$$\begin{aligned} \frac{dx}{dt} &= \frac{V_L + V_R}{2}\cos\theta, \\ \frac{dy}{dt} &= \frac{V_L + V_R}{2}\sin\theta, \\ \frac{d\theta}{dt} &= \frac{V_L - V_R}{2}, \end{aligned} \tag{1}$$

where $(X,\ Y$ and $\theta)$ are the robot actual position and orientation angle in world reference frame. In simulation, we use the discrete form to build a model of the robot. These equations are used to simulate the robot in MATLAB software. Then, the discrete form of the kinematic model is given by the following equations:

$$\begin{aligned} X_{k+1} &= X_k + T\frac{V_{rk} + V_{lk}}{2}\cos\theta_k, \\ Y_{k+1} &= Y_k + T\frac{V_{rk} + V_{lk}}{2}\sin\theta_k, \\ \theta_{k+1} &= \theta_k + T\frac{V_{rk} + V_{lk}}{L}, \end{aligned} \tag{2}$$

where X_{k+1} and Y_{k+1} represent the position of the center axis of the mobile robot and T is the sampling time.

3. Design of the Fuzzy Logic Controller (FLC)

The developed fuzzy controller manages at the same time navigation and obstacle avoidance tasks. Many academic

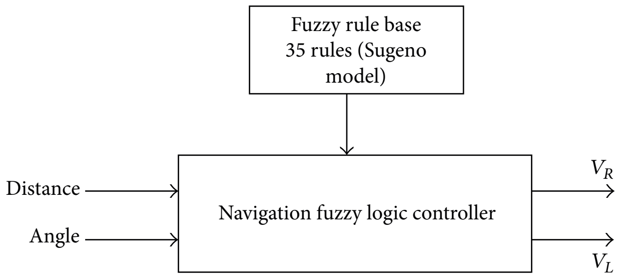


FIGURE 3: Methods of the conventional tracking fuzzy control system.

studies propose the fuzzy logic theory as a solution to control mobile robots [8–11]. The basic structure of the fuzzy controller is composed of three blocks: the fuzzification, inference, and defuzzification.

The first step to realize a fuzzy controller is fuzzification which transforms each real value's inputs and outputs into grades of membership for fuzzy control terms. The second part is fuzzy inference which combines the facts acquired from the fuzzification of the rule base and conducts the fuzzy reasoning process. There are some methods of fuzzy inference depending on the uses and the form of the membership function. When the input and the output variables and membership function are defined, the fuzzy rule is presented as the following form:

IF ⟨antecedents⟩ THEN ⟨conclusions⟩ rules.

The third part of fuzzy logic is defuzzification block. The objective of this part is to transform the subsets of the outputs which are calculated by the inference engine.

3.1. Fuzzy Logic Controller (FLC). The developed fuzzy logic controller (FLC) for navigation task used two inputs: the distance d and the angle orientation φ. The outputs of the controller are the speed of the right V_R and the left wheels V_L. The values of the two inputs are given by the following equations:

$$d = \sqrt{(X_T - X)^2 + (Y_T - Y)^2}, \quad \varphi = \theta_T - \theta, \tag{3}$$

with

$$\theta_T = \tan^{-1}\frac{(Y_T - Y)}{(X_T - X)}. \tag{4}$$

Figure 3 illustrates the block diagram of the robotic system using a fuzzy controller to reach the desired position. The final intelligent control loop is described in Figure 4.

3.2. Structure of the Membership Function. Triangular and trapezoidal membership functions are used in the fuzzification block.

The variable d is partitioned in the universe of discourse [0 500 mm] which is defined by five triangular membership functions: very small (VS), small (S), medium (M), big (B), and very big (VB) as shown in Figure 5.

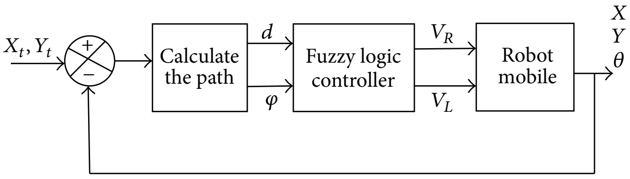


FIGURE 4: The block diagram of the robotic system.

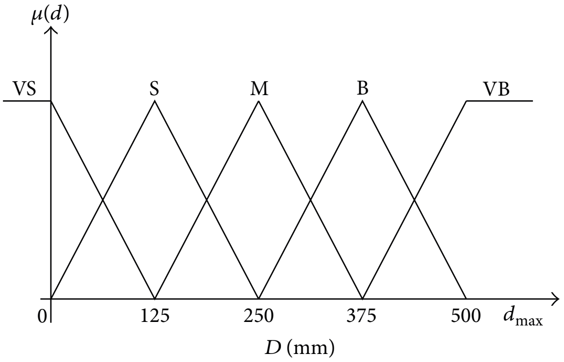


FIGURE 5: Input membership function for distance.

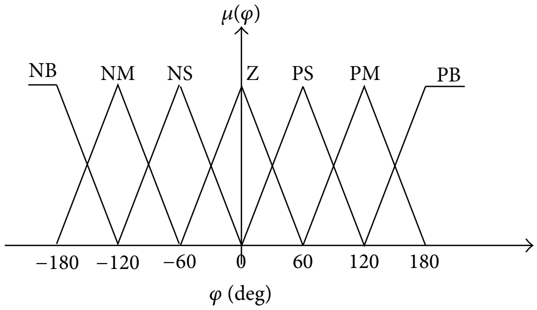


FIGURE 6: Input membership function for angle.

The angle should be partitioned in the universe of discourse [−180° 180°], and the setting is defined by seven membership functions: negative big (NB), negative medium (NM), negative small (NS), zero (Z), positive small (PS), positive medium (PM), and positive big (PB) as shown in Figure 6.

For speed five singleton memberships functions are determined: Z (zero), F (far), M (medium), B (big), and VB (very big) as shown in Figure 7.

3.3. The Structure of the Fuzzy Rule Bases. To fulfill the control objective, it is crucial to design a fuzzy logic control for the real velocities of the mobile robot which use fuzzy control in the inputs and outputs. After detailing membership functions, we define the fuzzy rule bases. The expert system is established based on 35 IF-THEN rules. The fuzzy rule bases are presented in Table 1.

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

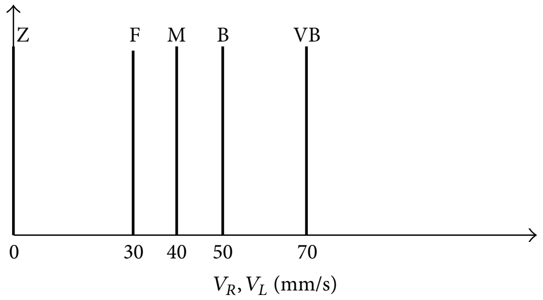


Figure 7: Output membership function for velocity.

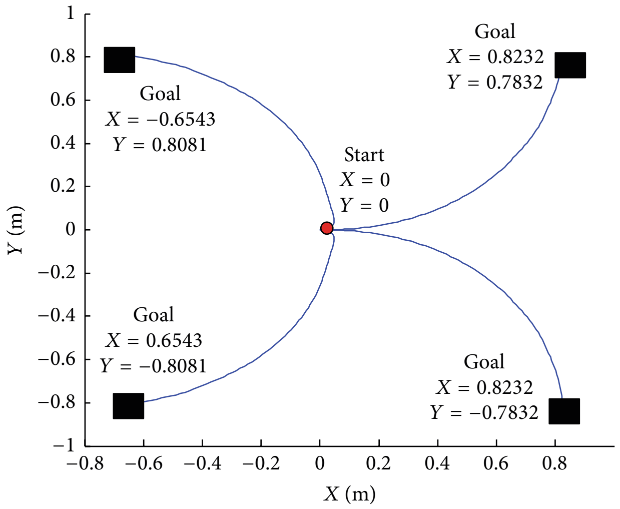


Figure 8: Navigation trajectories for different targets.

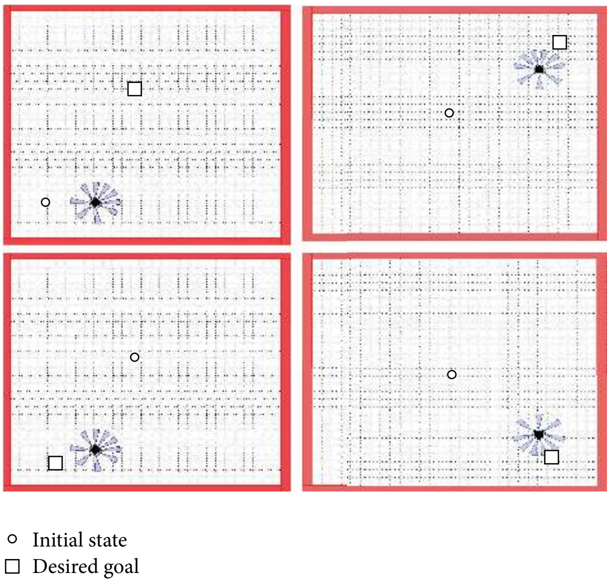


Figure 9: Simulation results for different target using SIMIAM platform.

4. Simulation Results

In this part, we will present the results of our simulation system using MATLAB and SIMIAM simulator. The robot starts at the beginning position (X_0, Y_0), moving based on its wheels velocity, to reach its target position (X_T, Y_T).

Several tests, for different configurations of the desired positions, have been carried out.

In these simulations, we have considered that the mobile robot always starts from the same point (0, 0) and moved reaching different target positions (Figure 6).

Figure 8 illustrates four trajectory positions.

The fuzzy controllers, previously described, have been tested with different situation. In Figure 8 we assume that the starting point is $X = 0$ and $Y = 0$ and the final point is $X_k = 0.8232$ and $Y_k = -0.7832$. It permits modifying the speed of the left and right wheel. This controller steers the robot to reach the target.

In Figure 10 we suppose that the starting point is $X = 0$ and $Y = 0$ and the final point is $X_k = 0.8232$ and $Y_k = 0.7832$. Figure 11 shows the initial position from $X = 0$ and $Y = 0$ to $X_k = 0.8232$ and $Y_k = 0.7832$.

Figure 9 explains the navigation of the mobile robot in various positions using the MATLAB software.

Figures 12 and 13 indicate the variation of angle orientation and distance at various simulation times without obstacles. At the starting simulation, the robot starts navigation by the distance of 1.22 mm. At 80 seconds, of simulation time, the robot attains the goal in the distance of 0.1 mm and the value of angle is 1.5 rad.

Figure 14 represents the right and the left velocity of the robot in simulation time. When the mobile robot has been oriented towards its target point, the FLC compute left and right robot wheel speeds to reach the goal.

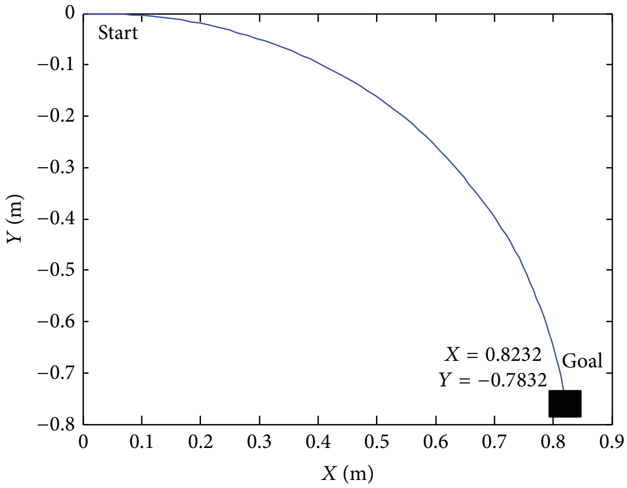


FIGURE 10: Navigation position with $X_k = 0.8232$ and $Y_k = -0.7832$.

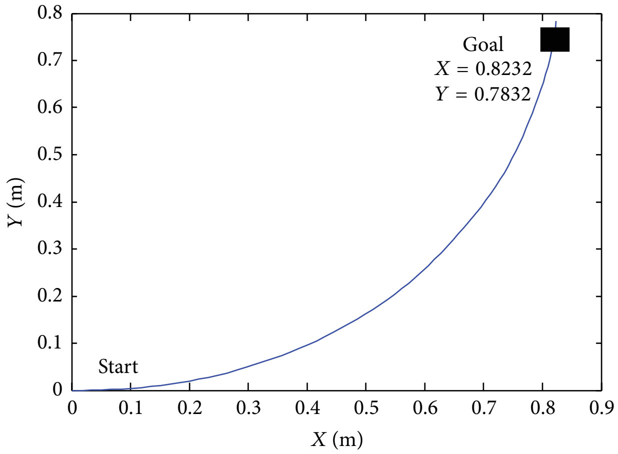


FIGURE 11: Navigation positions with $X_k = 0.8232$ and $Y_k = 0.7832$.

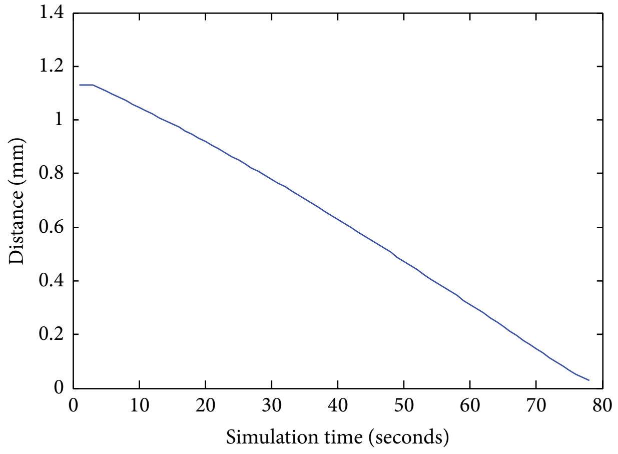


FIGURE 12: The distance D values during navigation.

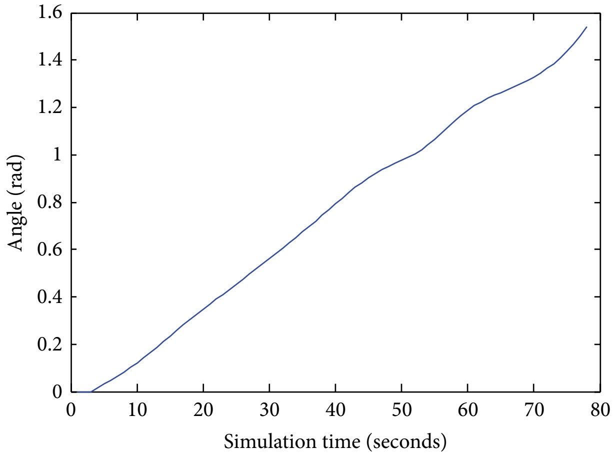


FIGURE 13: The angle φ values during navigation.

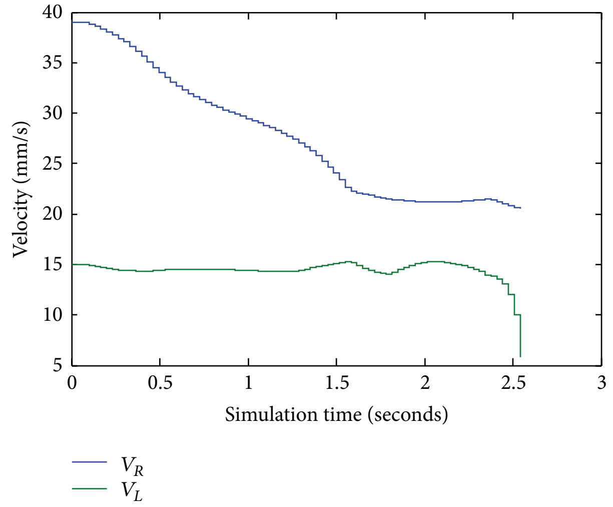


FIGURE 14: Left and right velocities values for $X_k = 0.8232$ and $Y_k = -0.7832$.

5. Obstacle Avoidance Controller (OAC)

In this part, we used a control system based on two fuzzy behaviours with a single coordination unit for the choice between the two controllers: the first controller for convergence towards the goal and the second controller for obstacle avoidance using the infrared sensors. Figure 15 illustrates the block diagram with obstacle avoidance.

The use of two fuzzy controllers consumes more resources and takes longer during navigating process of the mobile robot, so we optimize the configuration to use only one fuzzy controller for navigation and obstacle avoidance (FLCAO) (see Figure 22). Sugeno model is also used for its implementation performances.

Figure 16 shows the optimization of the configuration by using only one fuzzy controller for navigation and obstacles avoidance (FLCAO). SIMIAM robot is equipped with nine infrared sensors (Figure 17).

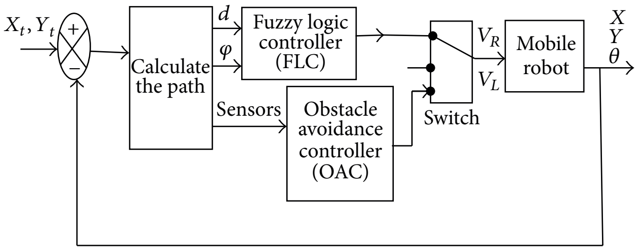


FIGURE 15: Block diagram with obstacle avoidance.

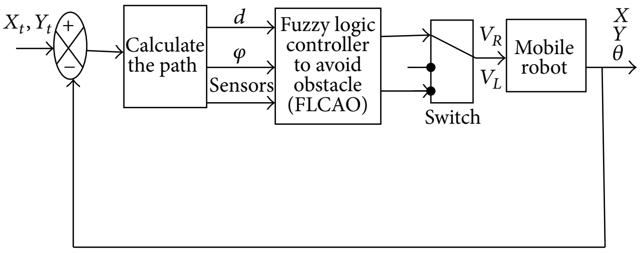


FIGURE 16: Block diagram with FLCAO.

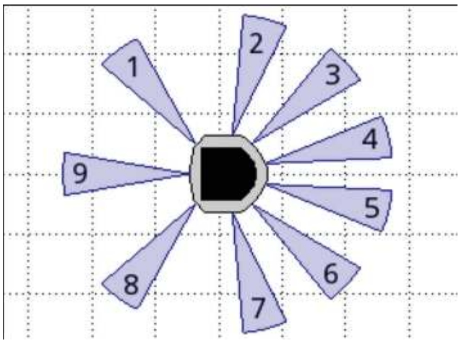


FIGURE 17: Infrared range (IR) sensor configuration.

The range sensors are effective in the range 0.02 m to 0.2 m. However, the sensors return raw values in the range of [18; 3960] instead of the measured distances [12] (Figure 17).

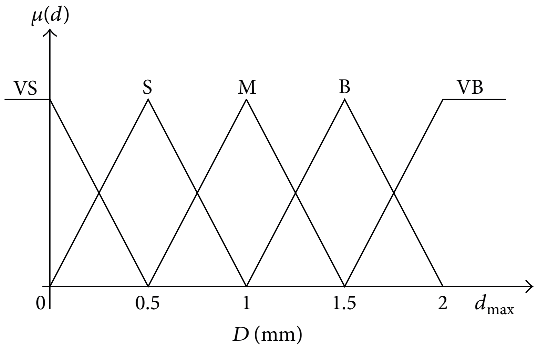


FIGURE 18: Input membership function for distance.

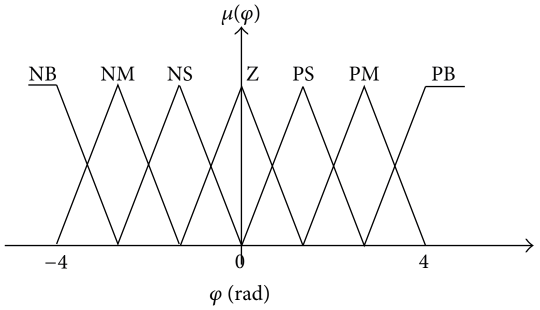


FIGURE 19: Input membership function for orientation angle.

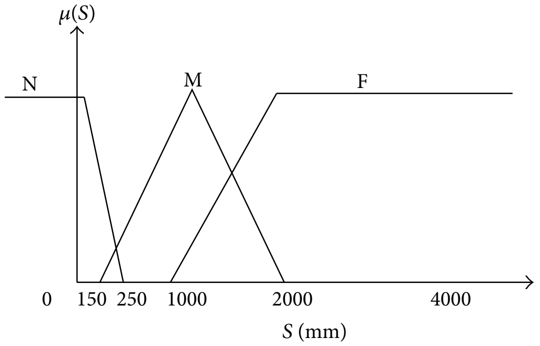


FIGURE 20: Input membership function for distance sensors.

5.1. Fuzzy Sets of the Input and Output. There are three inputs to the fuzzy logic system and two outputs. The inputs are basically the distance (Figure 18), the angle orientation (Figure 19), and values of eight distance sensors given by S1, S2, S3, S4, S5, S6, S7, and S8 (Figure 20). Two outputs are generated: left velocity (V_L) and right velocity (V_R).

The variable d is partitioned in the universe of discourse [0 2 mm] which is defined by five triangular membership functions: very small (VS), small (S), medium (M), big (B), and very big (VB) as shown in Figure 18.

The angle should be partitioned in the universe of discourse [−4 4 rad], and the setting is defined by seven membership functions: negative big (NB), negative medium (NM), negative small (NS), zero (Z), positive small (PS), positive medium (PM), and positive big (PB) as shown in Figure 19.

The sensor is divided into three membership functions: N (Near), M (Medium) and F (Far) (Figure 20).

Figure 21 shows the eight membership functions: Z (zero), F (far), M (medium), B (big) and VB (very big) for the left and the right velocity (V_R, V_L).

The controller was a Sugeno-type fuzzy model defined with set of 62 fuzzy control rules. Some fuzzy rule bases are shown in the following:

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)

If (Distance is VS) and (ANgle is NM) and (S1 is F) and (S2 is F) and (S3 is F) and (S4 is F) then (V_R is M) (V_L is 0)

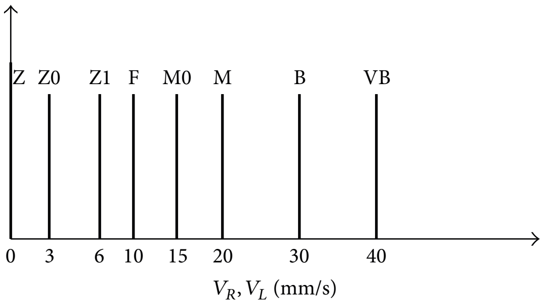


Figure 21: Input membership function for velocity.

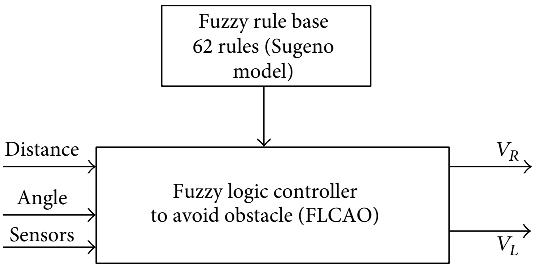

Fuzzy rule base 62 rules (Sugeno model)
Distance
Angle
Sensors
Fuzzy logic controller to avoid obstacle (FLCAO)
V_R
V_L

Figure 22: The FLCAO input/output configuration.

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 F) (V_L is 0)

⋮

If (S3 is M) and (S4 is M) then (V_R is M) (V_L is Z0)

6. Simulation Results with Obstacles Avoidance

In this section, the mobile robots have to navigate in environment with obstacles.

Figures 23 and 24 show that the proposed algorithm avoids the obstacle. We notice that if the infrared range sensors detect an obstacle, the robot is forced to make an adequate turn to avoid collision with the object.

Figures 25, 26, 27, 28, 29, 30, 31, and 32 show the distance sensor values. At the beginning of the simulation, the robot starts sensing the environment for possible obstacle detection.

Figure 33 shows the left and right wheels' velocities of the robot. Figure 34 describes the navigation and obstacle avoidance of mobile robot in various positions using MATLAB.

At 25 seconds of the simulation time, sensors S3 reached a value of 1000.

At 32 seconds, the robot detects another obstacle on its right side and moves forward. Then, the robot gets stuck in between two obstacles and another obstacle in front of it.

Sensors S4 attain a value of 500.

Besides, the robot detects another obstacle on its path and turns left at 35 s; sensors S5 reach a value of 3750. Finally, at 42 s, the robot picks up another obstacle on its right side when sensors S1 attain 2750.

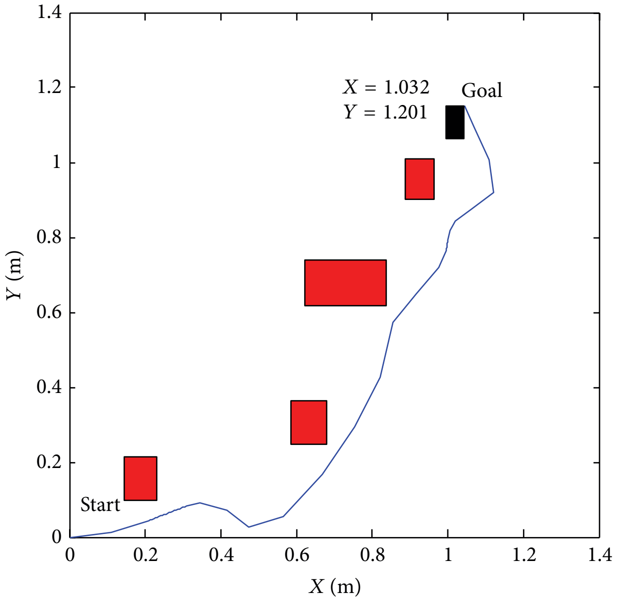


Figure 23: Reaching target with obstacles $X = 1.032$ and $y = 1.201$.

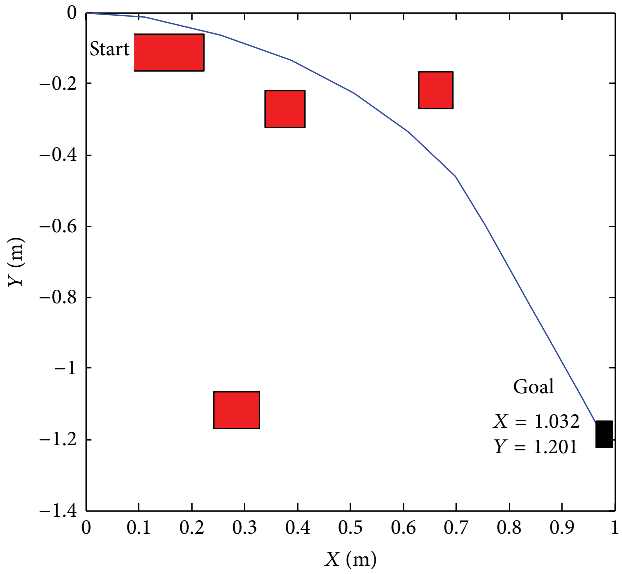


Figure 24: Reaching target with obstacles $X = 1.032$ and $y = -1.201$.

7. Conclusion

This paper presents the design and simulation of an intelligent mobile robot with encoders and infrared sensors detection computed with a navigation fuzzy control algorithm and obstacle avoidance system. First, we started with the development of the kinematic model of the robot system and then we designed the navigation fuzzy controller. The simulation results, using MATLAB and SIMIAM simulation platform,

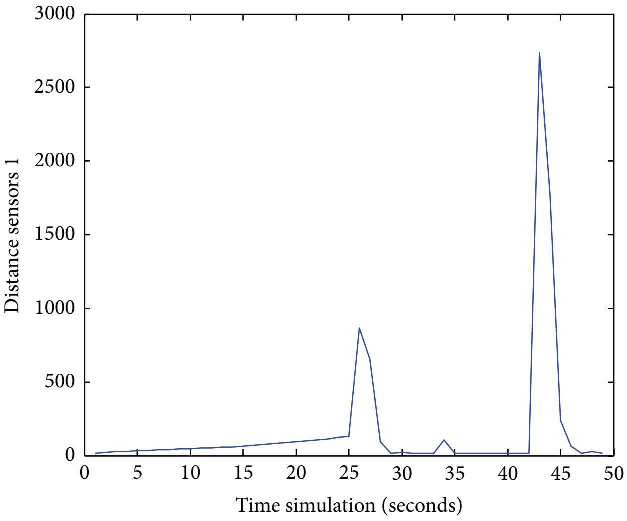


FIGURE 25: S1 values during obstacles avoidance with $X_k = 0.8232$ and $Y_k = -0.7832$.

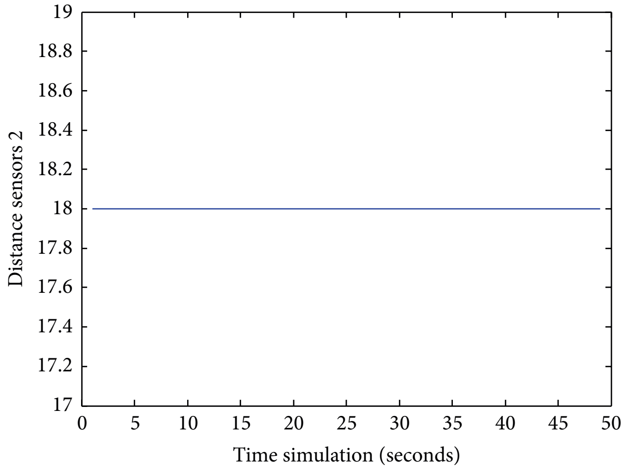


FIGURE 26: S2 values during obstacles avoidance.

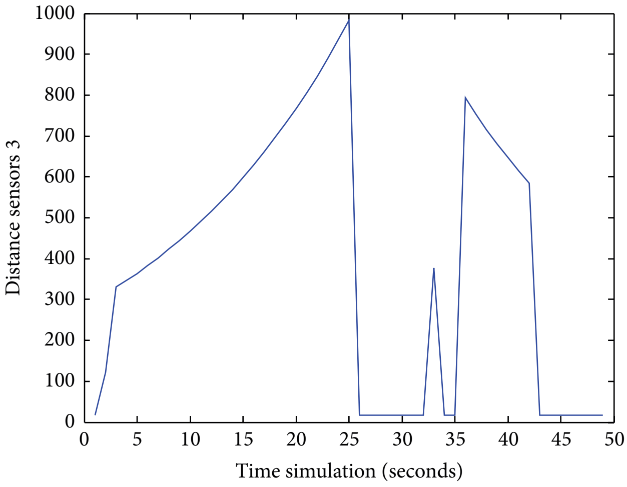


FIGURE 27: S3 values during obstacles avoidance.

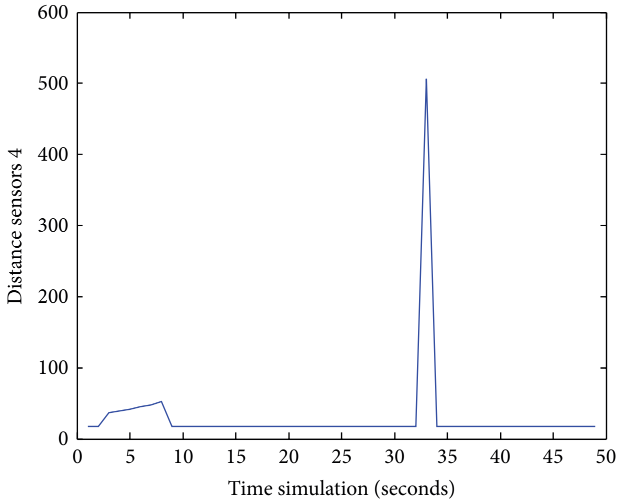


FIGURE 28: S4 values during obstacles avoidance.

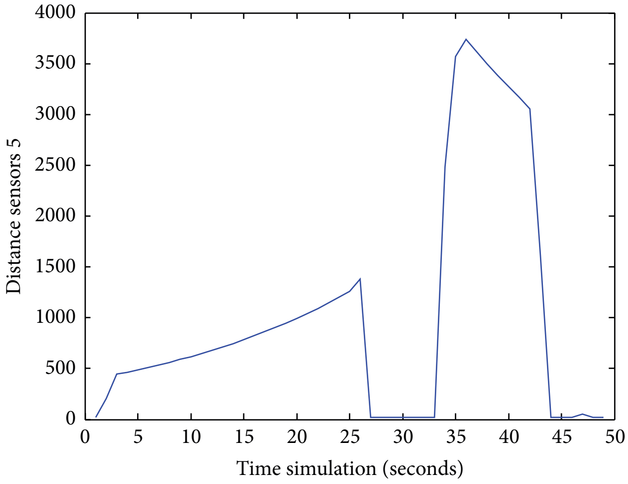


FIGURE 29: S5 values during obstacles avoidance.

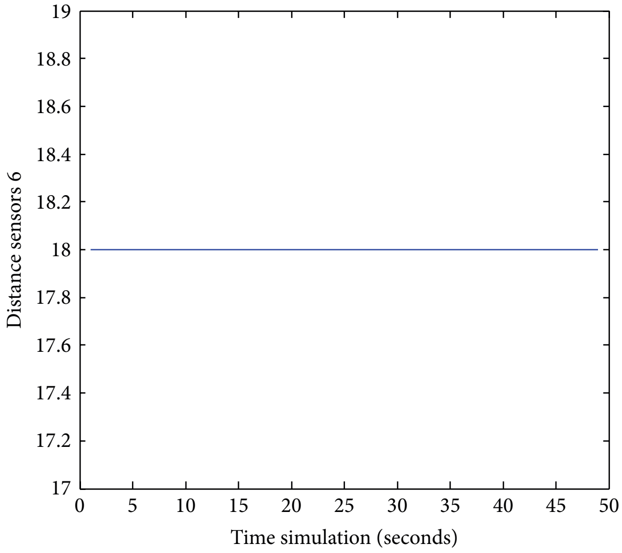


FIGURE 30: S6 values during obstacles avoidance.

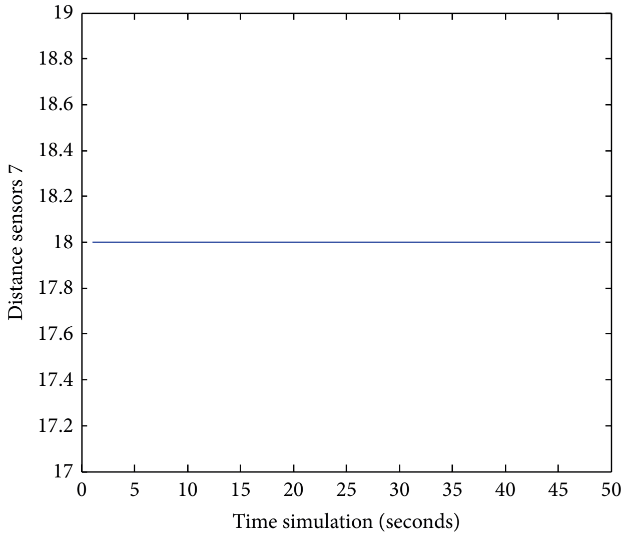


Figure 31: S7 values during obstacles avoidance.

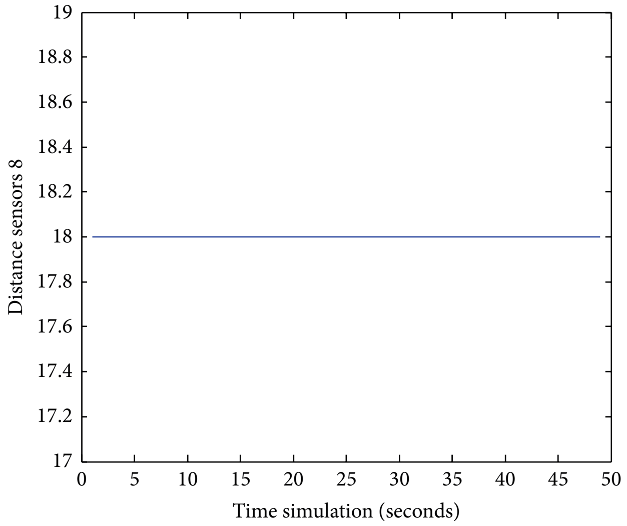


Figure 32: S8 values during obstacles avoidance.

have shown the effectiveness of the designed FLC giving good navigation performances.

To add the obstacle avoidance task we opted for a second fuzzy controller. This solution seems to be effective; but since the final objective is the implementation on a dedicated embedded platform we chose using a single fuzzy controller which performed both tasks.

Simulation results using SIMIAM platform have shown a clear performance improvement of the navigation system. Thus, the fuzzy implemented controllers have shown its adaptation to the dynamics of unicycle robots. In fact, QuickBot is capable of reaching the target position and avoiding obstacles based on an optimized intelligent controller.

The results of the simulation and experimental tests in a real environment showed the success of the FLC algorithm in controlling QuickBot.

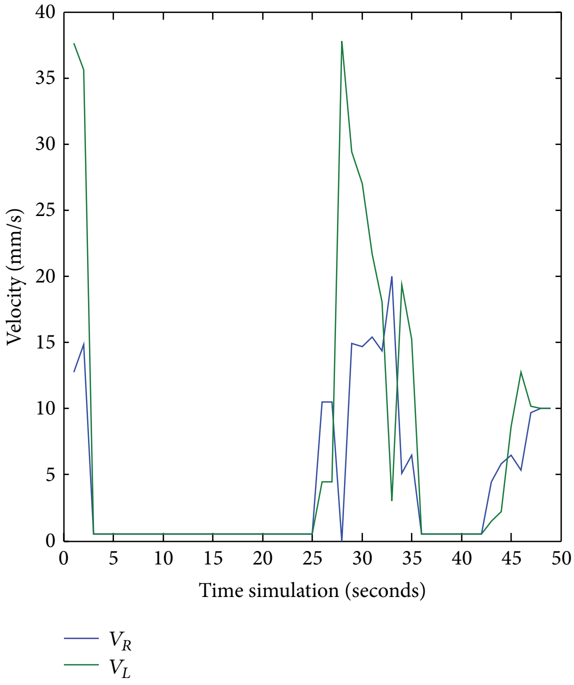


Figure 33: Left and right velocities for the robot during obstacle avoidance task. $X_k = 0.8232$ and $Y_k = -0.7832$.

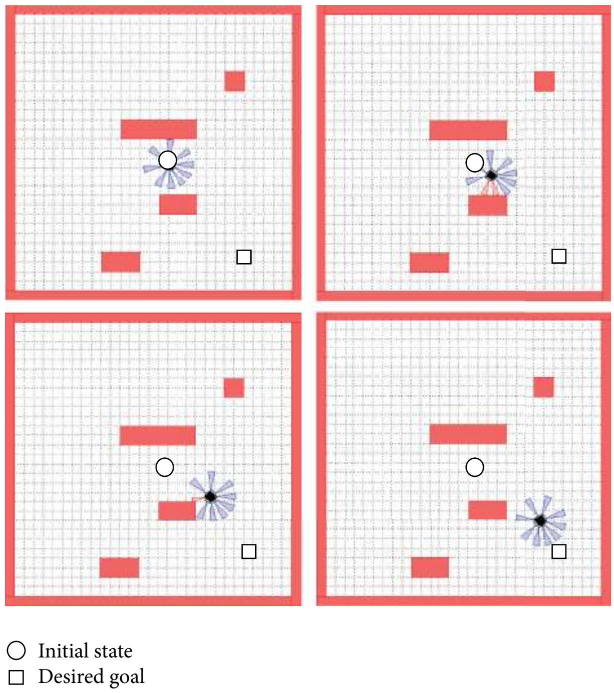


Figure 34: Sequence showing the MATLAB robot simulator SIMIAM implementing the navigation system with $X_k = 0.8232$ and $Y_k = -0.7832$.

The final fuzzy logic controller can be implemented on an intelligent wheelchair to help the elderly or disabled people to perform the navigation task.

Competing Interests

The authors declare that they have no competing interests.

References

[1] A. Prakash Moon and K. K. Jajulwar, "Design of adaptive fuzzy tracking controller for Autonomous navigation system," *International Journal of Recent Trend in Engineering and Research*, vol. 2, no. 2, pp. 268–275, 2016.

[2] G. R. Yu, K. Y. Bai, and M. C. Chen, "Applications of Taguchi method to fuzzy control for path tracking of a wheeled mobile robot," in *Proceedings of the 17th National Conference on Fuzzy Theory and Its Applications*, pp. 857–862, Kaohsiung, Taiwan, 2009.

[3] B. Xiong and S. R. Qu, "Intelligent vehicle's path tracking based on fuzzy control," *Journal of Transportation Systems Engineering and Information*, vol. 10, no. 2, pp. 70–75, 2010.

[4] M. F. Selekwa, D. D. Dunlap, D. Shi, and E. G. Collins Jr., "Robot navigation in very cluttered environments by preference-based fuzzy behaviors," *Robotics and Autonomous Systems*, vol. 56, no. 3, pp. 231–246, 2008.

[5] S. Jia, J. Yan, J. Fan, X. Li, and L. Gao, "Multimodal intelligent wheelchair control based on fuzzy algorithm," in *Proceedings of the IEEE International Conference on Information and Automation (ICIA '12)*, pp. 582–587, Shenyang, China, June 2012.

[6] A. V. Chavan and J. L. Minase, "Design of a Differential Drive Mobile Robot Platform for use in constrained enviroments," *International Journal of Innovations in Engineering Research and Technology [IJIERT]*, vol. 2, no. 6, 2015.

[7] M. Khaoula, N. Malek, and J. Mohamed, "Free navigation and obstacle avoidance based on fuzzy controller," in *Proceedings of the 2nd World Congress on Computer Applications and Information Systems*, N&N Global Technology, Hammamet, Tunisia, January 2015.

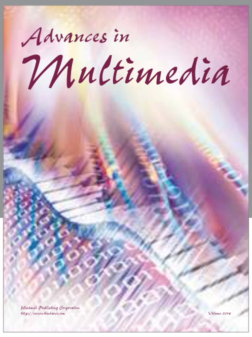
[8] A. Ranasinghe, N. Sornkarn, P. Dasgupta, K. Althoefer, J. Penders, and T. Nanayakkara, "Salient feature of haptic-based guidance of people in low visibility environments using hard reins," *IEEE Transactions on Cybernetics*, vol. 46, no. 2, pp. 568–579, 2015.

[9] M. Njah and M. Jallouli, "Wheelchair obstacle avoidance based on fuzzy controller and ultrasonic sensors," in *Proceedings of the International Conference on Computer Applications Technology (ICCAT '13)*, pp. 1–5, Sousse, Tunisia, January 2013.

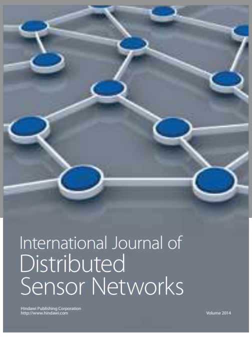
[10] T. T. Mac, C. Copot, R. De Keyser, T. D. Tran, and T. Vu, "MIMO fuzzy control for autonomous mobile robot," *Journal of Automation and Control Engineering*, vol. 4, no. 1, pp. 65–70, 2016.

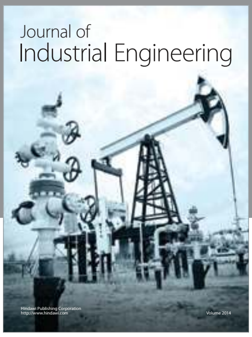
[11] S. Armah, S. Yi, and T. Abu-Lebdehet, "Implementation of autonomous navigation algorithms on two wheeled ground mobile robot," *American Journal of Engineering and Applied Sciences*, vol. 7, no. 1, pp. 149–164, 2014.

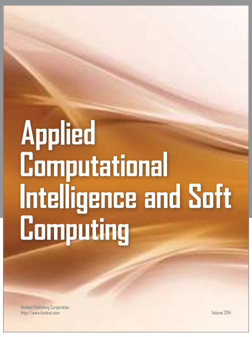
[12] *MATLAB Robot Simulator SIMIAM (Software)*, Georgia Institute of Technology, Georgia Tech, Atlanta, Ga, USA, 2013.

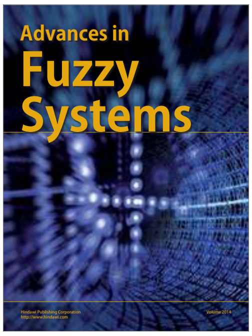

Advances in
Multimedia

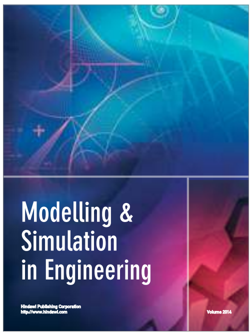
The Scientific World Journal

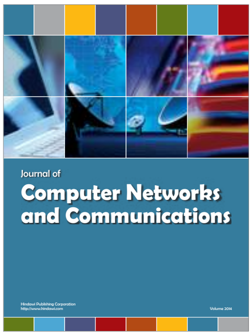

International Journal of
Distributed Sensor Networks

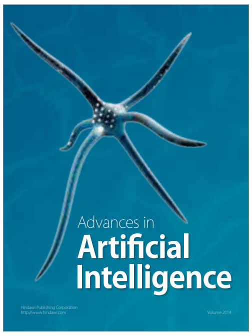

Journal of
Industrial Engineering

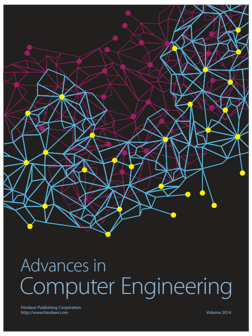

Applied Computational Intelligence and Soft Computing


Advances in
Fuzzy Systems

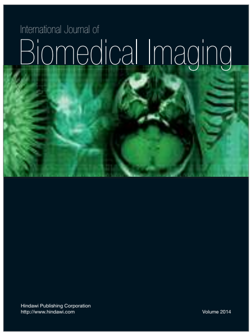

Modelling & Simulation in Engineering

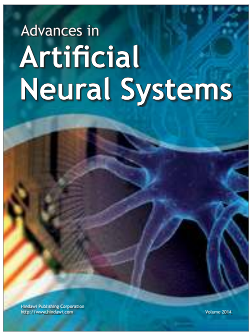

Journal of
Computer Networks and Communications

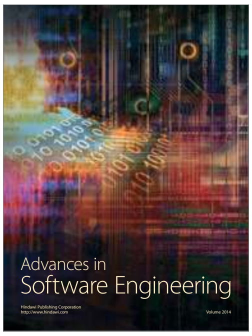

Advances in
Artificial Intelligence


Advances in
Computer Engineering

International Journal of
Computer Games Technology

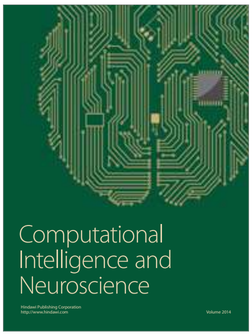

International Journal of
Biomedical Imaging


Advances in
Artificial Neural Systems


Advances in
Software Engineering

Journal of
Robotics

Advances in
Human-Computer Interaction


Computational Intelligence and Neuroscience

International Journal of
Reconfigurable Computing

Journal of
Electrical and Computer Engineering