

Optimization of fuzzy rules using neural network to control mobile robot in non-structured environment

Ayman A. AbuBaker¹, Yazeed Yasin Ghadi²

¹Department of Electrical Engineering, Faculty of Engineering, Applied Science Private University, Amman, Jordan

²Department of Software Engineering and Computer Science Programs, Al Ain University, Abu Dhabi, United Arab Emirates

Article Info

Article history:

Received Oct 17, 2022

Revised Dec 20, 2022

Accepted Feb 3, 2023

Keywords:

Fuzzy logic

Hybrid neuro-fuzzy

Mobile robot

Neural networks

Unstructured environment

ABSTRACT

It is still a challenge for all authors to control an autonomous mobile robot in an unstructured environment. The purpose of this paper is to propose a new control method for mobile robots in unstructured environments using neuro fuzzy technique. The proposed algorithm reduces the processing time of the fuzzy logic controller (FLC) inference engine. The neural network (NN) will therefore select the optimum rule(s) directly from the inference engine. This means that all of the rules of the inference engine do not need to be processed. As a result, the inference engine process speed will decrease, and the fuzzy logic response will increase. An actual mobile robot with three distance sensors and one virtual orientation angle is used to test the proposed algorithm. Based on the results, the mobile robot is capable of avoiding all obstacles and reaching the target point accurately.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Ayman A. AbuBaker

Department of Electrical Engineering, Faculty of Engineering, Applied Science Private University

Amman, Jordan

Email: a_abubaker@asu.edu.jo

1. INTRODUCTION

Mobile robot is design as manipulator to sense the unknow topology of moving or stationary obstacle during its movements [1], [2]. Two main aspects need to be addressed in designing this type of robot: nonlinear and non-analytical controller and reliability of controller system. Fuzzy logic controller (FLC) is one of the controller systems that represent the human experience by set of linguistic rules. Although FLC is a robust controller in solving a tedious and difficult problem but it is very slow in response when used for large-scale problems [3], [4]. FLC is more efficient than all soft computing method since it can tolerantly noise and error of reading from the sensory system, also it can reflect the expertise of the human behavior [5], [6]. Two approaches are used to solve navigation problem in fuzzy logic, behavior based approach [7], [8] and classical fuzzy rule base approach [9]. The fuzzy inference engine is design based on using fuzzy logic rules. These rules represent the human experience as an if-then rules. These rules might be not correct and consistent, or do not possess complete domain knowledge, and/or could have a proportion of redundant rules. This will increase the processing time or produce a wrong decision from the fuzzy controller [10], [11].

Several successful reactive navigation approaches based on neural networks (NN) have been suggested in the literature [12], [13]. In spite of various suggested network topologies and learning methods, neural reactive navigators still perceive their knowledge and skills from demonstrating actions [14]. Therefore, they suffer from a very slow convergence, lack of generalization due to limited patterns to represent complicated environments, and finally information encapsulated within the network can not be interpreted into physical knowledge [15]. Consequently, the utilization of NN in reactive mobile robot

navigation is limited when compared to fuzzy logic. However, the role of NN has been found to be very useful and effective when integrated with fuzzy systems [16]. The birth of this integration between these two soft-computing paradigms is the neuro-fuzzy systems. Neuro-fuzzy systems provide an urgent synergy that can be found between the two paradigms, especially the capability to mimic human experts as in fuzzy logic, and learning from previous experience capability as in NN [17], [18].

The mobile robot is still a challenge to all authors. So, this paper will propose a new algorithm to control a mobile robot in an unstructured environment. The proposed algorithm is focused on increasing the response time for the inference engine. Therefore, NN are used to select and optimum fuzzy rule(s) from the inference engine. As a result, there is no need to process all the fuzzy rules in the inference engine and this will speed up the processing time and increase the mobile robot response.

This paper is organized as follows. A brief survey of previous work is presented in section 2. The materials and methods that are used in this proposed approach are presented in section 3. Section 4 presents the results, discussion and evaluation materials, while concluding remarks are given in section 5.

2. LITERATURE REVIEW

Different artificial intelligent techniques are implemented by many authors to control a mobile robot. A pattern recognition approach is one of the intelligent techniques implemented by Singh *et al.* [19] to control a mobile robot. The approach is designed based on collecting different data from ultrasonic sensors then develop a pattern mapping for the mobile robot based on these sensors. The mapping approach was designed using fuzzy kohonen clustering network (FKCN). As a result, the robot moves safely from the starting point to its target. Research by Yung and Ye [20], implement fuzzy logic in controlling a mobile robot. They used an alternative training approach and reactive fuzzy navigation to control the mobile robot. This method was 270 times faster than the traditional rule based fuzzy controller. A neuro-fuzzy controller also used in mobile robot [10]. The authors implement a new technology of hybrid neuro-fuzzy controller to control an autonomous mobile vehicle called Bearcat II. The controller has the ability to achieve target point, given a location, and reach in real time unexpected events like sudden appearance of obstacles. Others such as Carpenter *et al.* [21] implement a neuro-fuzzy controller in mobile robot. Due to the high response of the proposed algorithm, the mobile robot can avoid nonlinear dynamic problems. The controller is divide to three main stages: fuzzification, NN (instead of inference engine), and defuzzification.

3. METHOD

Mobile robot is required to explore several paths in a maze to reach its desired position safely. The mobile robot must therefore possess the necessary sensors in order to detect its path and the target point. In this paper, the mobile robot is equipped with three physical ultrasonic sensors and one virtual sensor as shown in Figure 1.

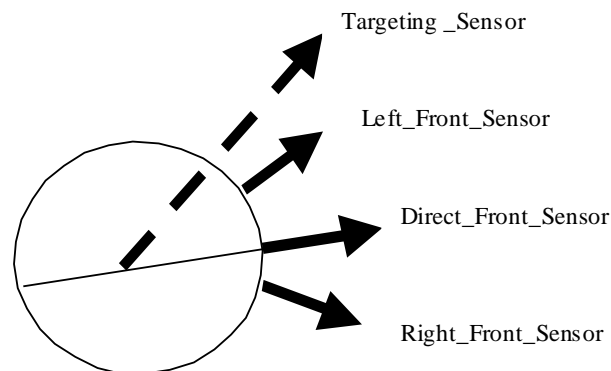


Figure 1. The four sensors of a mobile robot

Ultrasonic sensors which are used to detect the distance between the robot and the obstacles are set in three locations in the robot (right, left and center). The maximum distance of those sensors to detect any objects or obstacles is set to 6 m. A virtual sensor shows the robot's shortest path to the target point continuously.

In this paper, a neuro-fuzzy controller is designed to control a mobile robot in an unstructured environment. The proposed controller is divided into three main stages as shown in Figure 2. In the first stage, all crisp data from the sensors are converted to fuzzy data using a fuzzification engine. Then, neuro-fuzzy inference engine is worked to activate an optimum number of fuzzy rule(s). Lastly, the defuzzification engine will convert the fuzzy data into crisp data that can be used to control the mobile robot.

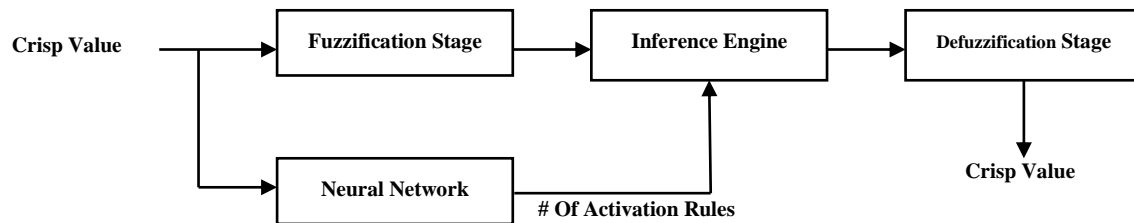


Figure 2. Proposed neuro-fuzzy controller

3.1. Fourty rules fuzzy navigator system

Xu and Tso [22] proposed a FLC40 to control an autonomous mobile robot. They analyzed and tested different cases for a mobile robot. When testing their algorithm on another complicated problem, the mobile robot failed to reach the target point. The main reason behind this failure was the limitation in number of membership function for each sensor (FAR and NEAR) as shown in Figure 3. To avoid these problems, a relaxation of the membership function was done by increasing number of fuzzy sets from two to five fuzzy sets. Therefore, the number of activation rules will be increased from 40 to 625 activation rules.

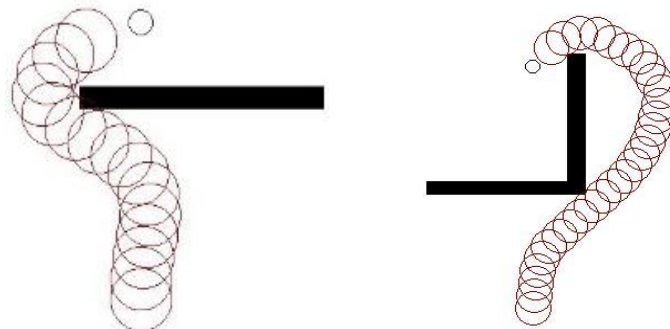


Figure 3. FLC40 simulated results

3.2. Improved fuzzy navigator system

FLC40 fails in avoiding collision with the edges of the obstacles as presented in the previous section. The reason behind this failure is the use of a limited number of fuzzy membership functions (NEAR and FAR). To overcome this problem, the number of fuzzy membership functions are increase from two to five membership functions for each input sensors. The linguistic labels for these fuzzy sets are entitled (VL, L, M, S, and VS) as shown in Figures 4(a)-(c). In this case, the fuzzy sets are shorter than before, so the controller is more accurate and performs better. In addition to the increase in the number of sets, the number of fuzzy rules increases as well, up to 625 activation rules ($5 \times 5 \times 5 \times 5 = 625$ activation rules). As an example, a sample is presented where the activation rules are: i) *IF dr is L and dc is L and dl is L and tr is LB THEN Sa is TLB*; ii) *IF dr is S and dc is M and dl is L and tr is RB THEN Sa is TZ*; and iii) *IF dr is L and dc is M and dl is S and tr is RS THEN Sa is TRS*.

A FLC62 can avoid collisions with obstacles as shown in Figure 5, but the processing of 625 activation rules will take a long time. As a result, the response time of the mobile robot will be decreased. Therefore, a NN is used in the inference engine to select the optimum number of activation rules for increasing mobile robot response, as described in the following section.

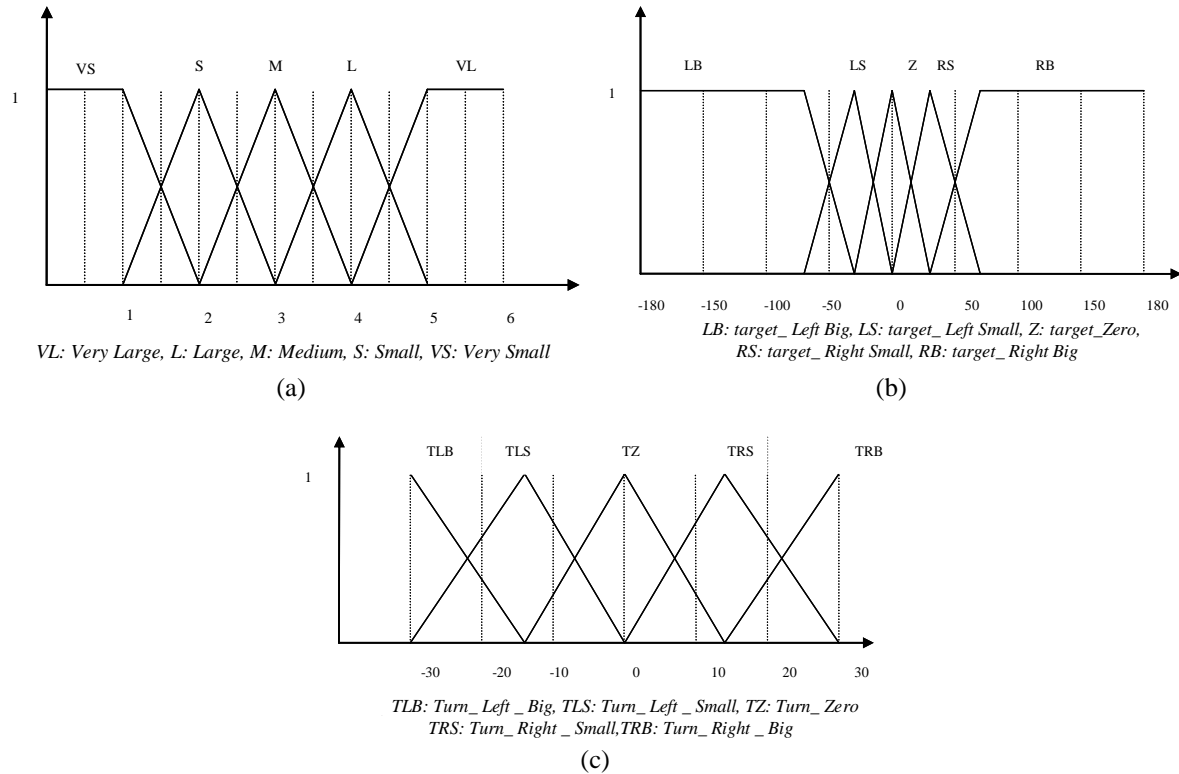


Figure 4. Fuzzy membership function (a) distance, (b) target orientation, and (c) turning angle

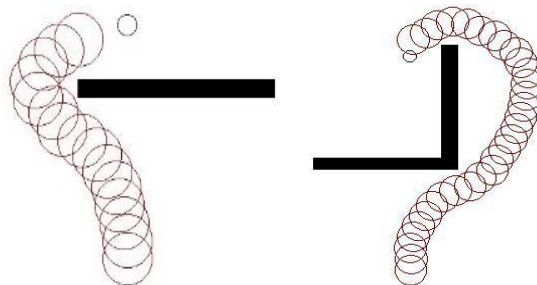


Figure 5. FLC625 simulation results

3.3. Optimized fuzzy rule base using neural network

Any FLC has a problem with its inference engine since it consists of many fuzzy rules. Processing these fuzzy rules will take a long time and decrease the response time for a FLC. To increase the response time of a fuzzy controller, an artificial NN is used to select the optimum number of activation rules in the inference engine. Therefore, the proposed controller will be a hybrid system of fuzzy logic and NN.

3.3.1. Creating input and output data

In supervised NN, both input and output data must be known. In our case, the inputs data are crisp data from three ultrasonic sensors and one virtual sensor. The detection range of the ultrasonic sensor is [0 to 6] m whereas the target orientation virtual sensor’s range [-180 to 180] degrees. The output is also a crisp value that presents the steering angle of the mobile robot which is in the range of [-65, 65] angle degree. To generate an input matrix, the distance is divided into seventeen points with step of 0.25 m between each point, whereas the target orientation’s step was 8.125 degree. As a result, the input matrix will of the size of four rows (four sensors) and 83,521 columns (17×17×17×17=83,521).

To find the output, the fuzzy rules in the inference engine are divided to 16 sections. Each section includes 40 rules. As a result, the total number of rules will be 640 activation rules. Where the 625 will actual activation rules and 15 do nothing rules that will be used null action in some cases. Therefore, the output nodes for the NN are set to 16 nodes which are the number of sections in the inference engine. For each

output node activation rule number that has a value [0.025 to 1] is set. The interval [0.025 to 1] comes from normalize the number of rules in the section based on the maximum number of the rules in the section which is 40. Therefore, the output matrix that have the desired activation rules for each input is set. As a result the size of the output matrix is 16×83,521 elements.

3.3.2. Training and learning the neural network

The feedforward NN were used in this paper, where the input and output data were arranged in a manner similar to Jackknife method [23], [24]. Approximately 70% of the data was used for training, and 30% for testing and validation. As we have four inputs and sixteen outputs, the NN topology was designed to satisfy these parameters. So, the input layer of the NN have four nodes (ultrasonic sensor left, ultrasonic sensor right, ultrasonic sensor center, virtual sensor) and sixteen output nodes (sections 1 to 16) in the output layer as shown in Figure 6.

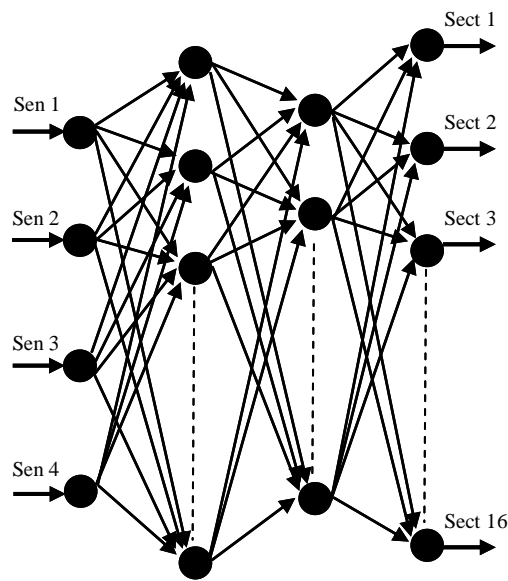
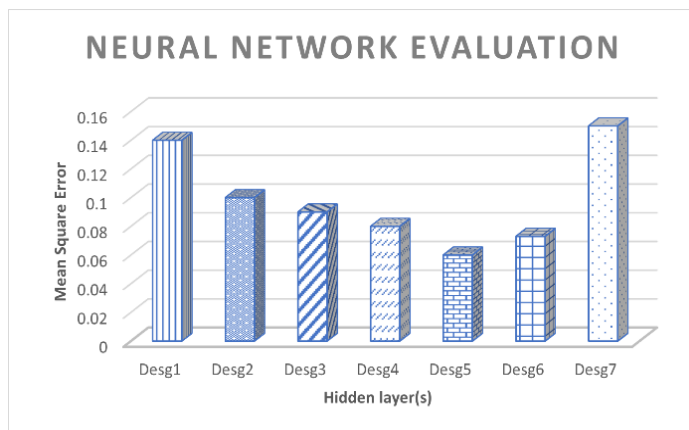


Figure 6. NN structure

Choosing the best number of hidden nodes or hidden layer depends on minimum mean square error (MSE) [25]. Therefore, many trials are carried out to choose the best performance of the NN as shown in Figure 7. As a result, the best topology was in using two hidden layers (9 nodes in the first layer and 5 in the second layer).



(Desg1: One hidden layer with three nodes, Desg2: One hidden layer with five nodes, Desg3: One hidden layer with seven nodes, Desg4: two hidden layers with five nodes in first layer and three nodes in second layer, Desg5: two hidden layers with nine nodes in first layer and five nodes in second layer, Desg6: three hidden layers with three nodes in first layer and three nodes in second layer and three layers in the third layer, Desg7: three hidden layers with five nodes in first layer and five nodes in second layer and three layers in the third layer)

Figure 7. NN Evaluation

4. RESULTS AND DISCUSSION

When using FLC40, the mobile robot is not capable of avoiding collision with edges of the obstacles due to the limitation in number of membership functions. Also, the inference engine in FLC625 needs a long processing time since it consists of large number of activation rules which affect on the mobile robot's response. Therefore, the NN is used to select the best number of activation rules in the inference engine. As a result, the processing time of the inference engine was reduced by 70% compared with traditional inference engine. The mobile robot's response was increased, and it can avoid collision with all the obstacles in different cases as shown in Figure 8. The three FLC (FLC40, FLC625, optimized FLC) were compared with each other in order to determine the total processing time per stage of each FLC. Based on Table 1, the optimized FLC had a 70% reduction in processing time compared to FLC 625.

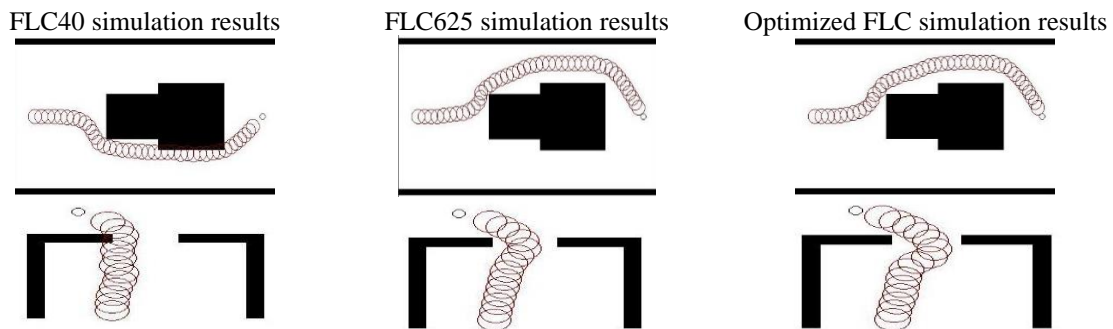


Figure 8. Comparison between three controllers (FLC40 cases, FLC625 cases, and optimized FLC cases)

Table 1. Performance evaluation of FLC40, FLC625, and optimized FLC

Variable	Performance	CPU time (μS)		
		Fuzzification time (μS)	Inference time (μS)	Total CPU time (μS)
FLC 40	Slightly or severely colliding with the obstacles	527	132	659
FLC 625	Avoid collision with the obstacles and smoothly reaches the target	1,038	1,736	2,757
Optimized FLC	No collision at all with the obstacles with high response time	840	349	1,189

5. CONCLUSION

The response time of FLC40 was fast, but unfortunately it is not capable of avoiding collision with the obstacles in different cases due to limitation in number of its fuzzy sets (FAR and NEAR). The FLC625 can successfully avoid collision with obstacles but the mobile robot's response time was high since the inference engine consists of large number activation rules. The proposed optimized FLC was designed to reduce the processing time of the inference engine. So, the NN is used to select the optimum number of activation rules in the inference engine. As a result, the processing time of the inference engine was reduced by 70% compared with traditional inference engine.

ACKNOWLEDGEMENTS

The author is grateful to the Applied Science Private University, Amman, Jordan, for the financial support granted to cover the publication fee of this research article.





REFERENCES

- [1] E. Aguirre and A. González, "Fuzzy behaviors for mobile robot navigation: design, coordination and fusion," *International Journal of Approximate Reasoning*, vol. 25, no. 3, pp. 255–289, Nov. 2000, doi: 10.1016/S0888-613X(00)00056-6.
- [2] K. Tanaka, *An introduction to fuzzy logic for practical applications*, 1st ed. New York: Springer, 1996.
- [3] T. J. Ross, *Fuzzy logic with engineering applications*. New York: McGraw-Hill, 1995.
- [4] L. A. Zadeh, "The roles of fuzzy logic and soft computing in the conception, design and deployment of intelligent systems," in *BT Technology Journal*, vol. 14, no. 4, Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 32–36.
- [5] A.-T. Nguyen and C.-T. Vu, "Mobile robot motion control using a combination of fuzzy logic method and kinematic model," in *Lecture Notes in Networks and Systems*, 2022, pp. 495–503, doi: 10.1007/978-981-19-3394-3_56.
- [6] S. Khesrani, A. Hassam, O. Boutalbi, and M. Boubezoula, "Motion planning and control of nonholonomic mobile robot using





- flatness and fuzzy logic concepts,” *International Journal of Dynamics and Control*, vol. 9, no. 4, pp. 1660–1671, Dec. 2021, doi: 10.1007/s40435-020-00754-4.
- [7] S. K. Harisha, R. K. P. M. Krishna, and S. C. Sharma, “Fuzzy logic reasoning to control mobile robot on pre-defined strip path,” *World Academy of Science, Engineering and Technology*, vol. 42, pp. 753–757, 2008.
- [8] S. K. Pradhan, D. R. Parhi, and A. K. Panda, “Fuzzy logic techniques for navigation of several mobile robots,” *Applied Soft Computing*, vol. 9, no. 1, pp. 290–304, Jan. 2009, doi: 10.1016/j.asoc.2008.04.008.
- [9] S. Parasuraman, B. Shirinzadeh, and V. Ganapathy, “Mobile robot navigation using alpha level fuzzy logic system: experimental investigations,” in *Evolutionary Computation*, London: InTech, 2009, pp. 557–563, doi: 10.5772/9624.
- [10] J. Cao, X. Liao, and E. L. Hall, “Reactive navigation for autonomous guided vehicle using neuro-fuzzy techniques,” in *Intelligent Robots and Computer Vision XVIII: Algorithms, Techniques, and Active Vision*, Aug. 1999, pp. 108–117, doi: 10.1117/12.360289.
- [11] C.-H. Chen, S.-Y. Jeng, and C.-J. Lin, “Mobile robot wall-following control using fuzzy logic controller with improved differential search and reinforcement learning,” *Mathematics*, vol. 8, no. 8, pp. 1–21, Jul. 2020, doi: 10.3390/math8081254.
- [12] R. Callan, *The essence of neural networks*. Prentice Hall Europe, 1999.
- [13] L. Fausett, *Fundamentals of neural networks*. Hoboken: Prentice Hall, 1994.
- [14] M. R. H. Al-Dahhan and M. M. Ali, “Path tracking control of a mobile robot using fuzzy logic,” in *2016 13th International Multi-Conference on Systems, Signals & Devices (SSD)*, Mar. 2016, pp. 82–88, doi: 10.1109/SSD.2016.7473656.
- [15] M. Abdelwahab, V. Parque, A. A. Abouelsoud, and A. M. R. Fath, “Navigation of omni-directional mobile robot in unstructured environments using fuzzy logic control,” in *2021 IEEE/SICE International Symposium on System Integration (SII)*, Jan. 2021, pp. 684–689, doi: 10.1109/IEEECONF49454.2021.9382654.
- [16] L. R. Medsker, *Hybrid intelligent systems*. Boston, MA: Springer US, 1995, doi: 10.1007/978-1-4615-2353-6.
- [17] N. H. Singh and K. Thongam, “Mobile robot navigation using fuzzy logic in static environments,” *Procedia Computer Science*, vol. 125, pp. 11–17, 2018, doi: 10.1016/j.procs.2017.12.004.
- [18] D. K. Mishra, A. Thomas, J. Kuruvilla, P. Kalyanasundaram, K. R. Prasad, and A. Haldorai, “Design of mobile robot navigation controller using neuro-fuzzy logic system,” *Computers and Electrical Engineering*, vol. 101, p. 108044, Jul. 2022, doi: 10.1016/j.compeleceng.2022.108044.
- [19] M. K. Singh, D. R. Parhi, S. Bhowmik, and S. K. Kashyap, “Intelligent controller for mobile robot: fuzzy logic approach,” in *The 12th International Conference of International Association for Computer Methods and Advances in Geomechanics (IACMAG)*, 2008, pp. 1755–1762.
- [20] N. H. C. Yung and C. Ye, “An intelligent mobile vehicle navigator based on fuzzy logic and reinforcement learning,” *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, vol. 29, no. 2, pp. 314–321, Apr. 1999, doi: 10.1109/3477.752807.
- [21] G. A. Carpenter, S. Grossberg, N. Markuzon, J. H. Reynolds, and D. B. Rosen, “Fuzzy ARTMAP: a neural network architecture for incremental supervised learning of analog multidimensional maps,” *IEEE Transactions on Neural Networks*, vol. 3, no. 5, pp. 698–713, 1992, doi: 10.1109/72.159059.
- [22] W. L. Xu and S. K. Tso, “Real-time self-reaction of a mobile robot in unstructured environments using fuzzy reasoning,” *Engineering Applications of Artificial Intelligence*, vol. 9, no. 5, pp. 475–485, Oct. 1996, doi: 10.1016/0952-1976(96)00047-4.
- [23] E. Vans, G. Vachkov, and A. Sharma, “Vision based autonomous path tracking of a mobile robot using fuzzy logic,” in *Asia-Pacific World Congress on Computer Science and Engineering*, Nov. 2014, pp. 1–8, doi: 10.1109/APWCCSE.2014.7053862.
- [24] A. A. AbuBaker, “An intelligent mobile robot controller using neuro-fuzzy techniques,” M.S. thesis, University of Jordan, Amman, Jordan, 2002.
- [25] S. Mondal and R. Ray and N. S. Reddy and S. Nandy, “Intelligent controller for nonholonomic wheeled mobile robot: A fuzzy path following combination,” *Mathematics and Computers in Simulation*, vol. 193, pp. 533–555, 2022, doi.org/10.1016/j.matcom.2021.10.028.

BIOGRAPHIES OF AUTHORS



Ayman A. AbuBaker     has obtained his Ph.D. in the area of in Electronic Imaging and Media Communications (EIMC) from School of Informatics, University of Bradford, UK. Currently, he is a professor in image processing and machine learning at the Faculty of Engineering and Technology, Applied Science Private University, Jordan. He has published more than 70 research articles/papers in journals, and conferences in his main research interest such as medical images processing including mammogram images (detecting microcalcifications and mass lesions), ultrasound and MRI. He has been actively working with several national professional communities and industries in the area of engineering education and accreditation such as ABET accreditation. Currently he is the dean of Faculty of Engineering and Technology at Applied Science Private University, Jordan. He can be contacted at email: a_abuabker@asu.edu.jo.



Yazeed Yasin Ghadi     received the Ph.D. degree from the Department of Biomedical Engineering, Central Queensland University, Australia. He was working as a Postdoctoral Research Fellow at CQU. He is currently an Assistant Professor with the Department of Computer Science and Software Engineering, Al Ain University, UAE. His research interests include multimedia contents and artificial intelligence, machine learning, computer vision, fuzzy logic, and renewable energy. He can be contacted at email: yazeed.ghadi@aau.ac.ae.