Journal of King Saud University – Computer and Information Sciences 34 (2022) 9060–9070

HOSTED BY

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

Robust mobile robot navigation in cluttered environments based on hybrid adaptive neuro-fuzzy inference and sensor fusion

Muhammad Husnain Haider [a,b], Zhonglai Wang [a,b,*], Abdullah Aman Khan [a,c], Hub Ali [d], Hao Zheng [a], Shaban Usman [a], Rajesh Kumar [a,b], M. Usman Maqbool Bhutta [e], Pengpeng Zhi [a,b,*]

[a] School of Mechanical and Electrical Engineering, University of Electronic Science and Technology of China, Chengdu, Sichuan 64000, China
[b] Yangtze Delta Region Institute, University of Electronic Science and Technology of China, Huzhou, Zheijang 313001, China
[c] Sichuan Artificial Intelligence Research Institute, Yibin, Sichuan 644000, China
[d] State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
[e] Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, Hong Kong 999077, Hong Kong Special Administrative Region, China

ARTICLE INFO

Article history:
Received 15 April 2022
Revised 17 August 2022
Accepted 26 August 2022
Available online 17 September 2022

Keyword:
ANFIS
GPS
Mobile robot
Obstacle avoidance
Autonomous navigation

ABSTRACT

Collision-free navigation of mobile robots is a challenging task, especially in unknown environments, and various studies have been carried out in this regard. However, the previous studies have shortcomings, such as low performance in cluttered and unknown environments, high computational costs, and multiple controller models for navigation. This paper proposes an adaptive neuro-fuzzy inference system (ANFIS) and global positioning system (GPS) for control and navigation to overcome these problems. The proposed method automates the navigation of a mobile robot while averting obstacles in unknown and densely cluttered environments. Furthermore, the mobile robots' global path planning and steering are controlled using GPS and heading sensor data fusion to achieve the target coordinates. A fuzzy inference system (FIS) is adopted to model obstacle avoidance where distance sensors data is converted into fuzzy linguistics. Moreover, a type-1 Takagi–Sugeno FIS is used to train a five-layered neural network for the local planning of the robot, and ANFIS parameters are tuned using a hybrid learning method. In addition, an algorithm is designed to generate a dataset for testing and training the ANFIS controller. All the testing and training are conducted in MATLAB, while simulations are carried out using CoppeliaSim. Comprehensive experiments are performed to validate the robustness of the proposed method. The results of the experiments show that the proposed approach outperforms various state-of-the-art neuro-fuzzy, CS-ANFIS, multi-ANFIS, and hybrid ANFIS navigation and obstacle avoidance methods in finding a near-optimal path in unknown environments.

1. Introduction

Autonomous navigation of a mobile robot is the ability to sense the environment, tackle obstacles, plan a trajectory from starting point to destination, and efficiently control the heading towards the target to achieve real-time navigation autonomously. Collision-free and autonomous navigation of mobile robots is a critical problem in various fields such as artificial intelligence (AI) and robotics (Thor and Manoonpong, 2022; Li et al., 2022; Xiao et al., 2022; Wu et al., 2021; Shahidinejad et al., 2020; Ma et al., 2018). Real-time navigation is relatively easy for humans and animals while avoiding all the obstacles in a dynamic environment. Though, it is still a monumental challenge for mobile robots. Numerous efforts have been devoted to achieving biomimetic-like behaviors in mobile robot navigation and obstacle avoidance. Previously, the main focus of building mobile robots was to navigate known environments such as warehouses, factory floors, indoor environments, etc. Such environments are comparatively more predictable and less challenging as compared to unknown and uncharted environments such as military operations, aerospace research, nuclear research, landmine detection, agriculture, rescue operations, medical aid during the COVID-19 pandemic, and others (Klancar et al., 2017; Baudoin and Habib, 2010; Troccaz, 2013; Cardona et al., 2020; Holland, 2004).

* Corresponding authors.
E-mail addresses: mhusnainhaider@outlook.com (M.H. Haider), wzhonglai@uestc.edu.cn (Z. Wang), zhipeng17@yeah.net (P. Zhi).

Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

https://doi.org/10.1016/j.jksuci.2022.08.031
1319-1578/

Fuzzy logic is well suited for the motion control of a robot as it is a protagonist to make inferences despite the existence of uncertainty in data (Naghsh et al., 2014; Ariffin et al., 2011). An intelligent fuzzy and feedback linearization controller effectively acquires the autonomous path planning of the mobile robot (Mondal et al., 2022). However, this method assumes that all obstacles have the same shape and size. The robot only follows a pre-defined trajectory, knowing the static obstacle positions. Another assumption is that virtual sensors precisely measure the obstacle's center, which is not practical in most cases. An intelligent autonomous parking system (Nakrani and Joshi, 2022) for a car-like mobile robot that efficiently parks the vehicle in dynamic environmental conditions is designed using two fuzzy controllers; one to handle the parking task and another to avoid obstacles while parking. But, the proposed obstacle avoidance controller needs an expert to define 81 different rules to avoid all possible situations. Additionally, a switch block is designed to switch between the modes. The adaptive fuzzy logic controller (Bakdi et al., 2017) for navigational control with two kinetic sensors and GA has the ability to find the target. Still, the map needs to be modeled offline before the GA path planning, the data fusion is not easy, and mapping the environment requires high memory and computational cost. Fuzzy membership and IR sensors-based obstacle avoidance (Aouf et al., 2019) can avoid obstacles with a safe margin, although this method demands partial or complete environmental information. A multi-layer decision-based fuzzy logic model (Kamil and Moghrabiah, 2021) performs well in comparatively simple and static environments for the autonomous navigation of mobile robots. The disadvantage of this method is the extremely high computational time in a cluttered and dynamic environment. Fuzzy neural network (Lv et al., 2021) for the unmanned ground vehicle based on multi-sensor information fusion smoothly avert obstacles coming in its way. However, it requires additional optimization of the neural network parameters to improve the design performance.

Adaptive neuro-fuzzy inference system (ANFIS) is the amalgamation of neural networks and fuzzy inference systems (FIS). ANFIS provides the opportunity for accurate knowledge representation with learning ability (Santoso et al., 2016). In reference (Elbatal et al., 2020), the authors proposed an autopilot using GA and ANFIS to increase the durability and robustness of unmanned air vehicles (UAV) under windy conditions. Still, multiple controllers are needed to tune each parameter, and authors must define 64 rules to satisfy the desired goal. The authors (Vu et al., 2018) applied ANFIS for the path planning of a robotic excavator arm to follow the desired trajectory under optimized shape conditions. Although, the path trajectory needs to be designed in the first place, and then ANFIS training can be carried out. In paper (Farahat et al., 2019), authors used a single ANFIS for mobile robot navigation with machine vision for global path planning and obstacle avoidance. Yet, all the input parameters have seven membership functions, leading to 2401 rules, making it computationally complex. A single ANFIS controller (Haider et al., 2022) for the autonomous path planning of the mobile robot shows success in the obstacle avoidance task by just using 16 rules; nonetheless, the global path is supplied to navigate in the environment. In reference (Karthikeyan et al., 2019), authors successfully presented an ANFIS-based model for distance calculation between the obstacles and robot with 104 rules and parameters tuning is attained by the backpropagation algorithm, which converges slower than hybrid propagation. These methods utilize multiple ANFIS-based controllers for each output parameter, which poses a hindrance to the training process. Moreover, these methods do not handle obstacle avoidance behavior, as a cluttered environment is challenging and computationally complex to map.

As discussed earlier, fuzzy logic has flaws when used alone for mobile robot navigation as it is incapable of learning. Even though several hybrid neuro-fuzzy and ANFIS techniques have been proposed, it is still challenging to perform autonomous mobile robot navigation utilizing only one controller and without expert-defined rules. It is a monumental task to move a robot autonomously from one point to another in a cluttered environment just using a few rules and a single ANFIS controller for local path planning without having information about obstacles' orientation and position. The methods described above are subject to numerous challenges, such as the requirement of partial or complete information about the environment and obstacles, adaptability to new environments, offline training for new environments, high computational demands for cluttered environments, data fusion problems, a large number of rules for simple tasks, and multiple controllers for obstacle avoidance and others. It motivates us to propose a robust hybrid method based on ANFIS and sensor fusion to overcome these issues for the local path planning of the mobile robot in this study. Global path planning is executed using GPS and heading sensor fusion to guide the robot towards the target when there is no obstacle in close proximity. The main contributions of this research work can be summarized as follows:

- This paper proposes a robust navigation approach. The proposed approach is more adaptable to unknown obstacle-prone clutter environments than previous methods. Moreover, path planning and obstacle avoidance are handled in a linguistic manner that is easily understandable by humans.
- Unlike the previous methods, a single ANFIS controller is proposed for local path planning with 27 rules contrary to multiple ANFIS and hundreds of rules. Additionally, the proposed method tackles all obstacles with a safe margin and reaches the target with a shorter path.
- Comprehensive experiments are performed, as well as several comparisons with state-of-the-art methods. The results indicate the superiority of the proposed approach.

The rest of the paper is organized as follows: Section 2 elaborates the proposed method and the preliminary knowledge. Moreover, Section 3 presents the details about the experiments and discussion on results followed by a conclusion in Section 4. Finally, Section 5 states the future work.

2. Proposed robust navigation controller

A general three-wheeled robot model having two front wheels and a passive omni wheel at the back is used for this research work. A GPS module is placed in the center of the robot while three sensors are placed around the right, front and left sides of the robot, respectively. GPS feeds the global navigation controller in every time interval δ_{gps}. Similarly, ultrasonic sensors feed the ANFIS module in every time interval δ_s, where δ_s is always smaller or equal to δ_{gps}. Fig. 1 represents the flow chart of the proposed algorithm for autonomous mobile robot navigation. .

2.1. Local navigation control

At first, the robot has to check if there are any obstacles less than the threshold or not, and then it decides the motion direction (as shown in Fig. 1). The robot receives near, mid, and far as obstacle distance information from the sensors during local path planning. It should be noticed that the sensors' information is imprecise and uncertain in nature. Additionally, conventional logical systems are not able to represent linguistic variables. Mathematical modeling of such variable variables is also challenging, but they are very commonly used in our daily life, and these are quite easy to use. Generally, human beings do not require precise

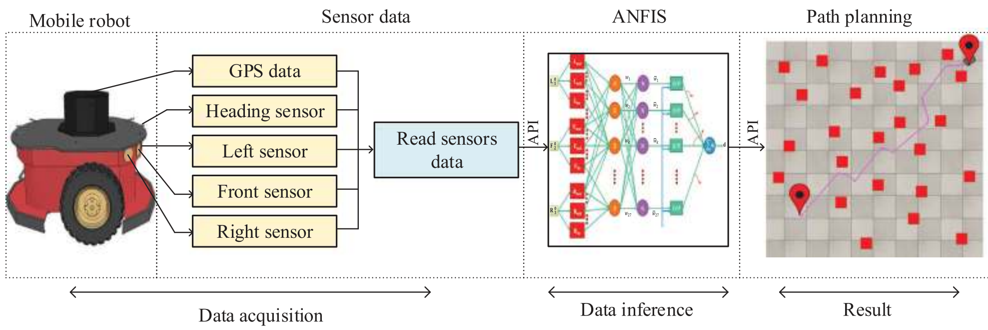

Fig. 1. A graphical abstract for the autonomous mobile robot navigation. The mobile robot performs the data acquisition task in the first step by collecting sensors data. An application programming interface (API) is established between CoppeliaSim and MATLAB to read the sensor measurements and govern the steering angle to successfully navigate the mobile robot while avoiding obstacles from a starting point to a target.

mathematical or logical numbers, yet their ability to control highly complex systems is remarkable. This is just because there is a concept of fuzzy logic in human perception.

The fuzzy logic system uses linguistic variables to model and deal with imperfect and imprecise data and represent it in a better and more logical way. Fuzzy logic gives a platform to represent uncertain and ambiguous data. Moreover, Fuzzy logic resembles the human linguistics rules to represent imprecise and inaccurate data. However, fuzzy logic lacks self-tuning and self-organizing. This creates the problem of the selection of membership function parameters. Self-tuning and self-learning require knowledge of the relationship between the input–output data. Experts formulate the rules and find the input–output relationship with the responding data.

On the other hand, neural networks are excellent performers for learning tasks. They can approximate a function without knowing the input–output relationship. Neural networks are used to develop many engineering systems. Neural networks are not good with modeling and logical reasoning of imprecise knowledge or data. ANFIS is the coproduct of fuzzy inference systems and neural networks. This allows for a favorable combination of circumstances to train the neural network using fuzzy logic. Hence, considering these attributes, ANFIS is chosen to perform the local path planning of the wheeled mobile robot. For local navigation, three ultrasonic sensors are mounted on the sides of the robot to measure the distance from obstacles on each side. Sensors are mounted on the right, middle, and left parts of the robot, covering the semicircular front to protect it from a collision. The sensors data is divided into three groups, right sensor data, left sensor data, and the front sensor data. The desired goal is to reach the destination without collision, with any obstacles coming in its way in unknown environments. The robot motion is divided into obstacle avoidance and goal-seeking motion planning. During the goal-seeking motion planning, the heading towards the destination is found using a goal-seeking algorithm, i.e., the global path planning and steering are adjusted accordingly. Otherwise, whenever an obstacle is within the close perimeter of the robot, then ANFIS takes over the steering control. Here, the threshold value for any distance sensor is adjusted to 20 cm.

2.2. Adaptive neuro-fuzzy inference system

This section is divided into three parts: (i) Fuzzy inference system, (ii) General schema of the ANFIS, and (iii) Hybrid training algorithm.

2.2.1. Fuzzy inference system

FIS is the critical component of ANFIS. The primary work of FIS is to make decisions using fuzzy logic data and rules. FIS decision rules use 'If-Then' statements along with 'AND-OR' connectors. FIS is composed of five functional blocks, namely, fuzzification, database, ruling, decision making, and defuzzification. A linguistic-based crisp input is converted into fuzzy variables, and FIS returns the crisp output after processing. There are two popular FIS methods for control applications, Mamdani FIS and Takagi–Sugeno FIS. Takagi–Sugeno FIS advances in terms of adjustable parameters more than Mamdani FIS. The ability to model exceedingly complicated systems and embed linear controllers is the beauty of Takagi–Sugeno FIS. The FIS for this study is constructed using the first order Takagi–Sugeno model as mentioned in multiple studies (Sugeno and Kang, 1988; Takagi and Sugeno, 1985).

FIS inputs are represented using linguistic terms, i.e., 'near', 'mid', 'far', and the universe of discourse is defined as 0, 20, 40, 60, 80, and 100 cm. The center values of corresponding linguistic terms 'near', 'mid', and 'far' are 12, 45, and 100 cm, respectively. As shown in Fig. 2, the bell-shaped membership functions are used to determine the degree of belongingness of fuzzy sets to their associated linguistic terms. FIS output is one of the a crisp value from −90, −45, 0, 45, 90, 180 deg. The If-then rules are used to select the right output in the ANFIS controller.

The proposed ANFIS architecture has five layers, three inputs, and a single output. Each input is subdivided into three membership functions, then 27 if-then rules are needed, as shown in Fig. 3. The details for the layers are provided as follows:

Input layer: The input layer is also known as the zero layer. This node receives the measured values from the distance sensors and then identifies the obstacles' locations and passes them to the first layer. Superscripts and subscripts represent the layer and sensor number, respectively.

$$\mathrm{L}_1^0 = \mathit{Output\ of\ left\ sensor}\ (cm) \tag{1}$$

$$\mathrm{F}_2^0 = \mathit{Output\ of\ front\ sensor}\ (cm) \tag{2}$$

$$\mathrm{R}_3^0 = \mathit{Output\ of\ right\ sensor}\ (cm) \tag{3}$$

Fuzzification layer: This layer receives the raw sensor inputs and converts them to fuzzy values. This is the first step of a FIS. The membership values of each node are determined by utilizing the membership belongingness of its relevant fuzzy set. The accuracy of the FIS depends on the number of membership functions, the higher number of membership functions results in higher accuracy, but it also increases the system's complexity. A complex system requires more memory and computational time. A trade-off can be found by keeping the desired objective in view. Gaussian-bell (g-bell) function membership is straightforward and popular to find the membership values (μ) in the fuzzification process. The g-bell function has been used to find the membership values. Every node in this layer is an adaptive node as given in Fig. 2.

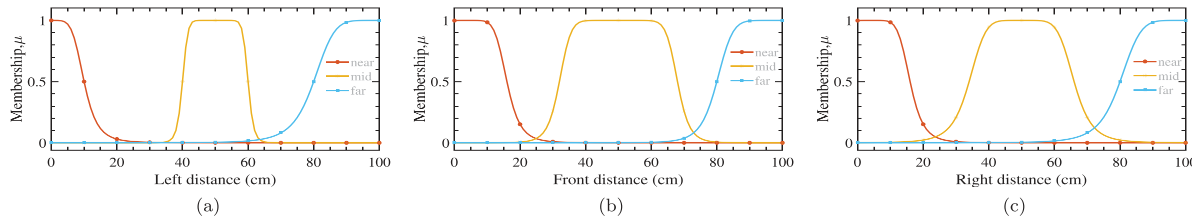

Fig. 2. Figures from (a)-(c) represent sonar sensor data before training. Obstacle distance readings are fuzzified into near, mid, and far linguistic variables for all the sensors. All the distance values are in cm, and μ stands for the membership value of each membership function of the corresponding set.

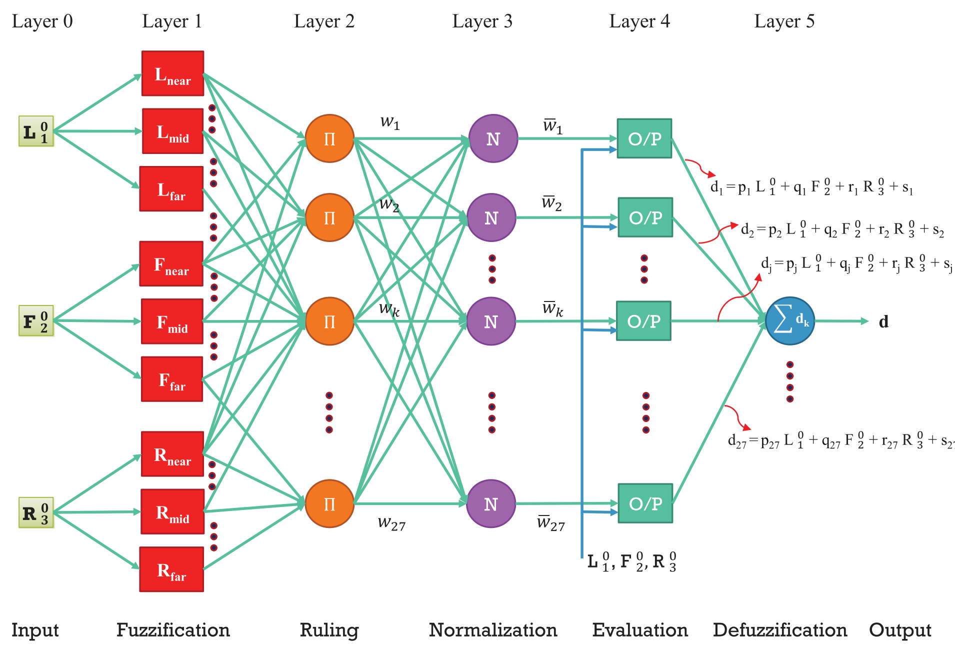

Fig. 3. The input data from the left, front and right obstacle detecting sensors are represented by L_1^0, F_2^0, and R_3^0 in the input layer; here, subscripts state sensor number and superscript state layer number. The final steering output is shown as d, while p_j, q_j, r_j, and s_j represent consequent parameters.Moreover, all the adaptive nodes are drawn in rectangular shape whereas all the fixed nodes are drawn in circles.

$$O_j^1 = \mu_{Lj}\left(L_1^0\right) = \frac{1}{1+\left(\frac{L_1^0-\alpha_j}{\gamma_j}\right)^{2\beta_j}} \quad \text{for } j = 1,\ 2,\ 3 \tag{4}$$

$$O_j^1 = \mu_{Fj-3}\left(F_2^0\right) = \frac{1}{1+\left(\frac{F_2^0-\alpha_j}{\gamma_j}\right)^{2\beta_j}} \quad \text{for } j = 4,\ 5,\ 6 \tag{5}$$

$$O_j^1 = \mu_{Rj-6}\left(R_3^0\right) = \frac{1}{1+\left(\frac{R_3^0-\alpha_j}{\gamma_j}\right)^{2\beta_j}} \quad \text{for } j = 7,\ 8,\ 9 \tag{6}$$

Premise parameters define the membership functions' shape and area. Here, antecedent parameters are donated by α_j, β_j, *and* γ_j for each neuron of the first layer, respectively. Based on the Takagi–Sugeno model for the ANFIS, the rules can be formulated as:

Rules j ($j \in \{1, 2, 3, \ldots, 27\}$) If left sensor data is L_j, front sensor data is F_j, and right sensor data is R_j then $d_j = p_j L_1^0 + q_j F_2^0 + r_j R_3^0 + s_j$ Where, L_1^0, F_2^0, R_3^0 are the fuzzy inputs, and d_j is the crisp output, $L_1^0 = \{L_{near}, L_{mid}, L_{far}\}$, $F_2^0 = \{F_{near}, F_{mid}, F_{far}\}$, $R_3^0 = \{R_{near}, R_{mid}, R_{far}\}$ are the fuzzy sets, and p_j, q_j, r_j, s_j, are the linear consequence parameters of the FIS.

Ruling layer: Each node in the second layer represents a defined If-Then rule under the Takagi–Sugeno model. These rules are the key for each node to evaluate the firing strength of the antecedents received from the previous layer. The output of each node generates the firing strength of the given set of fuzzy rules, which corresponds to a specific part of the rule applied. It is also important to note that the structure and area of the output function depend on the firing strength of every rule. Accordingly, the ruling layer outputs are the 'AND' product of the subsequent layer as provided in Eq. 7.

$$O_j^2 = w_j = \mu_{La}\left(L_1^0\right)\mu_{Fb}\left(F_2^0\right)\mu_{Rc}\left(R_3^0\right) \tag{7}$$

for $j \in \{1, 2, 3, \ldots, 27\}$ and $a, b, c \in \{1, 2, 3\}$. Every node in this layer is a fixed node.

Normalization layer: Each node is normalized according to the ratio of j^{th} rule strength and the sum of all the provided rules.

$$O_j^3 = \bar{w}_j = \frac{w_j}{\sum w_j} \tag{8}$$

All the nodes in this layer are fixed nodes. Each node in this layer receives the inputs from all the nodes of the ruling layer. The output of this layer is the weight ratio of the j^{th}node and the weight of all other nodes in this layer, as stated in Eq. 8.

Evaluation layer: In this layer, a relationship is established between input and output as stated in Eq. 9.

$$O_j^4 = \bar{w}_j d_j = \bar{w}_j\left(p_j L_1^0 + q_j F_2^0 + r_j R_3^0 + s_j\right) \tag{9}$$

The weighted average of the rules from the previous layer is multiplied with the equation having the consequent parameters.

Consequent parameters are vital in this layer as they govern the output. Additionally, all the nodes in this layer are adaptive in nature.

Defuzzification layer: Defuzzification is the final layer of the ANFIS. This layer sums up all previous layer outcomes to generate the final desired steering angle as given in Eq. 10.

$$O_j^5 = \bar{w}_j d_j = \frac{\sum w_j d_j}{\sum w_j} \tag{10}$$

This node corresponds to the adaptive behavior.

2.2.2. Hybrid training algorithm

ANFIS uses a hybrid algorithm to tune and update the premise and consequence parameters. Two methods used in conjunction are the gradient descent method and the least mean square method. The gradient descent method is applied to update premise parameters. Consequent parameters are tuned using the least square method. The consequent parameters are chosen by the least square method during each forward pass epoch. First, the consequent parameters are fixed using the backward error propagation method, and premise parameters are updated according to the gradient descent approach.

Algorithm 1: Dataset generation.

```
DistanceSamples  = 10000
ThresoldDistance  = 20
A[DistanceSamples, 4] =0
TD = ThresoldDistance
j = 1
  whilej ⩽DistanceSamplesdo
    A[j,1]= rand[100] %Left sensor data
    A[j,2]= rand[100] %Front sensor data
    A[j,3]= rand[100] %Right sensor data
    if(A[j,1] ⩽ TD) & (A[j,2] ⩽ TD) & (A[j,3] ⩽ TD)do
      A[j,4]=180
    elsif(A[j,1] ⩽ TD) & (A[j,2] ⩾ TD) & A[j,3] ⩽ TDthen
      A[j,4]=0
    elsif(A[j,1] ⩽ TD) & (A[j,2] ⩾ TD) & A[j,3] ⩾ TDthen
      A[j,4]=45
    elsif(A[j,1] ⩽ TD) & (A[j,2] ⩾ TD) & A[j,3] ⩾ TDthen
      A[j,4]=90
    elsif(A[j,1] ⩾ TD) & (A[j,2] ⩾ TD) & A[j,3] ⩽ TDthen
      A[j,4]=-45 %Anti Clockwise
    elsif(A[j,1] ⩾ TD) & (A[j,2] ⩽ TD) & A[j,2] ⩾ TDthen
      A[j,4]=90
    elsif(A[j,1] ⩾ TD) & (A[j,2] ⩽ TD) & A[j,3] ⩽ TDthen
      A[j,4]=-90 %Anti Clockwise
    end if
j  = j + 1
end while
```

2.3. Goal seeking planning

The robot gets its instantaneous position from the attached GPS module. Now, the robot direction is obtained using the two vector system and heading sensor. The working mechanism of the goal-seeking behavior is presented in Fig. 1. Moreover, Fig. 4 presents the geometrical features of the robot, GPS, and the heading sensor for global path planning. Assume x_1 and y_1 are the robot coordinates and $x_2,\ y_2$ are the destination coordinates. The values of x and y can be obtained using the subtraction of final coordinates and robot coordinates according to the following Eq. 11, Eq. 12, and Eq. 13.

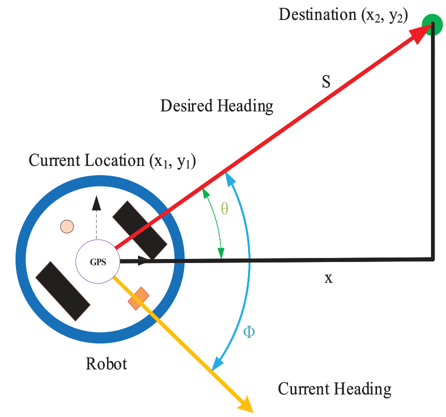

Fig. 4. Geometric representation of the destination, GPS, and the heading sensor with reference to the robot. Here, S represents the distance between the robot's current location and the destination, whereas x_1, y_1 and x_2, y_2 represents their corresponding coordinates, respectively.

$$S = \sqrt{x^2 + y^2} \tag{11}$$

$$S = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{12}$$

$$\theta = \tan^{-1}\left[(y_2 - y_1)/(x_2 - x_1)\right] \tag{13}$$

Moreover, a heading sensor needs to find the direction of the current direction of the robot. The target angle (ϕ) can be adjusted using the two points vector angle calculation. Finally, if the robot satisfies $S < S^{th}$ condition, it indicates the target position has been reached.

Fig. 4 shows the current location of the robot and the destination in the Cartesian coordinate system. The red line represents the desired path, while the yellow line represents the current heading path of the robot. Moreover, θ represents the angle in degrees the robot should deflect from the origin, and ϕ represents the angle between the current headed trajectory and the desired path.

2.4. Dataset generation

In order to train the proposed model, a dataset is generated that considers the sensor inputs and output steering angles. The dataset is comprised of 10,000 samples based on the sensor readings. In Algorithm 1, generated the dataset based on N data distances. All other operations such as conditional checks have a constant complexity of $\mathcal{O}(1)$. The main complexity depends on the number of distance samples *DistanceSamples*. Therefore the overall worst-case time complexity for Algorithm 1 is $\mathcal{O}(DistanceSamples)$ and the average runtime of Algorithm 1 is 0.0153752s.

We shed some light on the phenomena related to steering control. Seven cases drive the wheeled mobile robot's steering, as presented in Fig. 5, i.e., for the first two cases, there is an obstacle on either the right or the left side of the robot, and as a remedy, the robot selects a turning angle of $\pm 45^\circ$ towards left or right, away from the obstacle. Similarly, for the third and fourth cases shown in Fig. 5, a turning angle of $\pm 90^\circ$ is selected for a front-facing obstacle in addition to the side-facing obstacle. Moreover, the fifth case represents a corridor scenario, i.e., having obstructions on both sides, selecting a steering angle of 0°. In the sixth case, a steering angle of 180° is selected in order to avoid the front and side facing obstructions. A steering angle of 90° is selected to avoid the front-facing obstruction. Further details about the steering angles and the orientation of the obstacles are provided in Table 2 where the negative sign represents the anti-clockwise steering. Given

Table 1
The consequent parameters corresponding to each rule.

Rule	If-then Statement			Consequent Parameters			
	L_1	F_2	R_3	p_j	q_j	r_j	s_j
1	near	near	near	0.03807	-0.00715	0.03765	179.900
2	near	near	mid	0.00102	0.00807	0.00466	89.7085
3	near	near	far	0.01188	0.01030	0.00186	90.0044
4	near	mid	near	0.01363	0.01199	-0.02079	-0.56030
5	near	mid	mid	0.02153	0.00339	-0.00172	44.7967
6	near	mid	far	0.01498	0.00182	-0.00237	45.0038
7	near	far	near	0.01113	-0.00505	-0.00664	0.39617
8	near	far	mid	0.00896	0.00178	-0.00483	45.0320
9	near	far	far	0.01956	-0.00037	-0.00583	45.4269
10	mid	near	near	0.01188	0.00044	-0.04086	-90.5869
11	mid	near	mid	0.00133	0.02673	-0.00855	90.2378
12	mid	near	far	-0.00235	0.03727	-0.00547	90.3346
13	mid	mid	near	0.00120	-0.00265	-0.00439	-44.9368
14	mid	mid	mid	0.00265	0.00673	-0.00211	-0.39972
15	mid	mid	far	0.00282	0.00437	0.00078	-0.44664
16	mid	far	near	0.00256	-0.00301	-0.01107	-44.8034
17	mid	far	mid	0.00402	0.00447	-0.00343	-0.41950
18	mid	far	far	0.00176	0.00019	-0.00043	-0.07333
19	far	near	near	0.01817	0.00032	-0.05745	-91.2705
20	far	near	mid	0.00134	0.03862	-0.01989	90.6243
21	far	near	far	-0.00022	0.01739	-0.00318	90.2187
22	far	mid	near	0.00041	-0.00293	-0.01240	-44.8319
23	far	mid	mid	0.00157	0.00563	-0.00327	-0.26821
24	far	mid	far	-0.00100	0.00517	-0.00048	-0.14669
25	far	far	near	0.00142	0.00270	-0.01275	-45.2858
26	far	far	mid	0.00019	-0.00061	-0.00233	0.15740
27	far	far	far	0.00019	0.00034	-0.00017	-0.03242

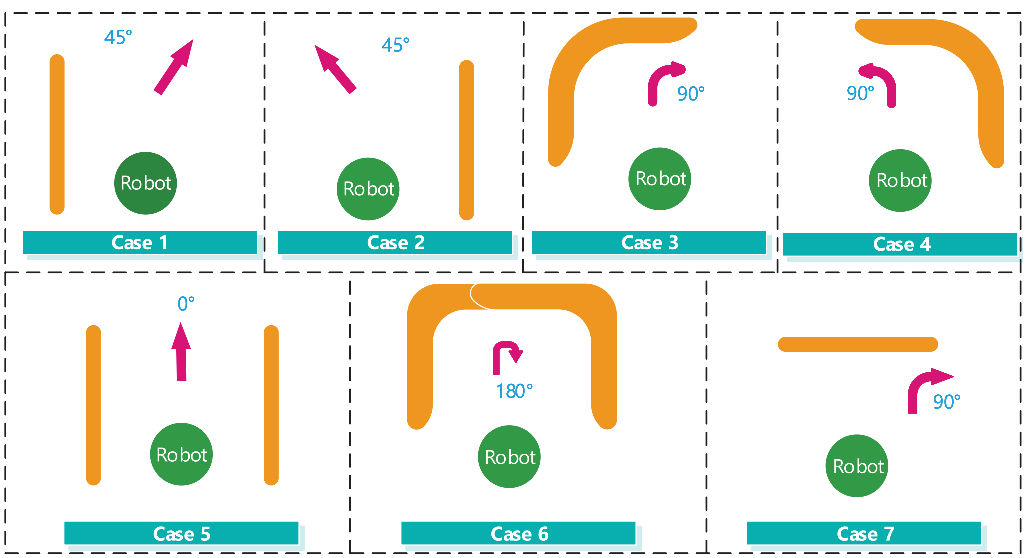

Fig. 5. Various cases for averting obstacles in a cluttered environment. The robot continuously collects surrounding data using sonar sensors and passes it to the ANFIS controller for a suitable steering angle if an obstacle exists. The arrow represents the steering angle in case of an obstruction.

Table 2
Dataset generation steering controls for ANFIS training.

Case	Obstacle Orientation	Steering Angle
1	Left	45
2	Right	−45 (anticlockwise)
3	Left Front Corner	90
4	Right Front Corner	−90 (anticlockwise)
5	Corridor	0
6	Left, Front and Right	180
7	Front	90

these scenarios, a dataset is generated using Algorithm 1. Algorithm 1 generates random variables ranging from 0 to 100 corresponding to the sample ultrasonic sensor readings. This data is generated for all the sensors to train and test the proposed ANFIS controller. This data is used as the input for the ANFIS. The output of the FIS is one of the pre-defined steering control as given in Table 2 that are governed by stated rules. Based on these rules, the test and training samples for the dataset are generated using Algorithm 1. The first column of the dataset represents the left, right, and front sensor data. At the same time, the last column represents the output steering angle. The proposed ANFIS is trained utilizing the generated dataset.

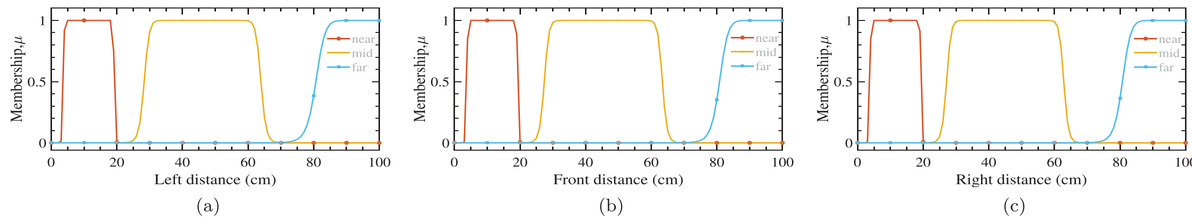

Fig. 6. Figures from (a)-(c) represent the post-training FIS membership functions for the designed ANFIS controller. All the FIS membership functions have changed shape under the influence of provided data for training. Notations are the same as provided in Fig. 2. It should be noted that there is a minor difference among all the FIS.

3. Experiments and results

This section provides details about the experiment, the related tools, and environment settings, followed by the simulation results and some discussion on the results.

3.1. Experiment details

This subsection provides details about the experiment settings and the dataset.

Simulation platform: CoppeliaSim (developed by Coppelia Robotics and previously known as V-REP) is a 3D robotics simulation platform with an integrated development environment that allows to model, edit, program a robot using sensors and perform simulations (Rohmer et al., 2013; Freese et al., 2010). It provides a plethora of features that can be easily integrated and combined with an extensive API and script functionality. MATLAB is integrated with CoppeliaSim to verify the proposed approach in this research work. All the testing and training are carried out in CoppeliaSim 4.2.0 and MATLAB 2021a.

Robot characteristics: A wheeled mobile robot 'Pioneer 3-DX', with multiple obstacles and a destination point, is included in the simulation scenes. Pioneer 3-DX robot has two front wheels with independent motor control and a passive omni wheel at the rear to balance the robot structure. The speed of the mobile robot is 0.2 m/s for all simulations.

Dataset: As mentioned earlier in subSection 2.4, the dataset is generated using Algorithm 1 for 10,000 random samples of left, right, and front sensor distances. Further, based on the threshold, a label (steering angle) is assigned to the input sensor data. The randomly generated data is synthetic in nature. However, it largely resembles the real-life robot sensor data.

Training and testing: For experiments, 25% of the dataset is utilized to test ANFIS, while 75% of the dataset is used to train the ANFIS. A slightly larger test set is used to study the proposed model's generalization capability and the performance response in real-world scenarios. To study in-depth and provide a clear picture to the readers, we generated two FIS using subtracting clustering and grid partitioning methods and tested for collision avoidance from obstacles, but they exhibited poor performance. Hence, a FIS is designed based on experience and literature studies (Jang, 1993). The final membership functions after training ANFIS under-designed FIS are shown in Fig. 6. ANFIS modifies the membership functions of FIS during the course of the training process. It also gradually reduces the error in the output steering angle, making it more accurate. Table 1 provides the post-training consequent parameters for all the 27 linguistic rules. Root mean square error (RMSE) validates the performance of the proposed FIS as given in Eq. 14. Fig. 7 shows the root mean square error (RMSE) of the sub clustering, grid partition, and the proposed ANFIS controller. The proposed FIS offers much accuracy over the subtracting clustering and grid partitioning methods. The testing data results verify that ANFIS has been trained accurately and efficiently.

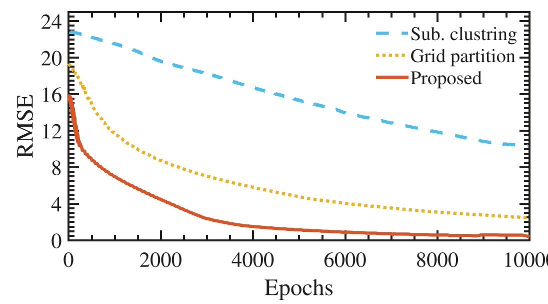

Fig. 7. RMSE plot of sub. clustering, grid partition, and the proposed FIS. It can be noticed that the proposed FIS has lower RMSE values, and the system converges in lesser epochs.

$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{(\hat{y}_i - y_i)^2}{n}} \tag{14}$$

Where $\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_n$ represent predicted data, $y_1, y_2, \cdots, y_n$ represent actual data, and n represent the number of iterations.

3.2. Simulation studies

The effectiveness of the proposed hybrid ANFIS is evaluated in this section using simulations. The robot first measures the distance between itself and obstacles during the simulation process under the obstacle avoidance behavior. If there are no obstacles nearby, the robot then switches to the target navigation behavior to determine the desired heading angle towards the destination. The mobile robot continuously measures the obstacle distances and desired angle, as illustrated in detail in earlier sections, and takes decisions accordingly. Fig. 8(a) and and Fig. 8-(b) shows autonomous mobile robot navigation and obstacle avoidance for two clutter environments. Simulation results for the dense clutter environment for two different destinations have been illustrated in Fig. 8(c) and Fig. 8-(d), respectively. In all scenarios, the robot is unfamiliar with the map. Like in Fig. 8-(a) after initialization, it checks to see if there are any obstacles in its immediate area. As there is no obstacle, it enters the target navigation behavior and begins moving toward the intended target. However, after moving a short distance, it encounters an obstacle, at which point it enters the obstacle avoidance behavior in order to avoid any collisions with a safe margin. It changes the target navigation behavior once more and calculates the destination angle after successfully avoiding the first obstacle. This process is repeated until the target is achieved. Simulation studies distinctly render that the robot can maneuver autonomously over different obstacles and achieve its target successfully. From Fig. 8, it is evident that the hybrid proposed motion planning of the autonomous robot based on ANFIS, GPS, and a heading sensor efficiently makes its way towards the

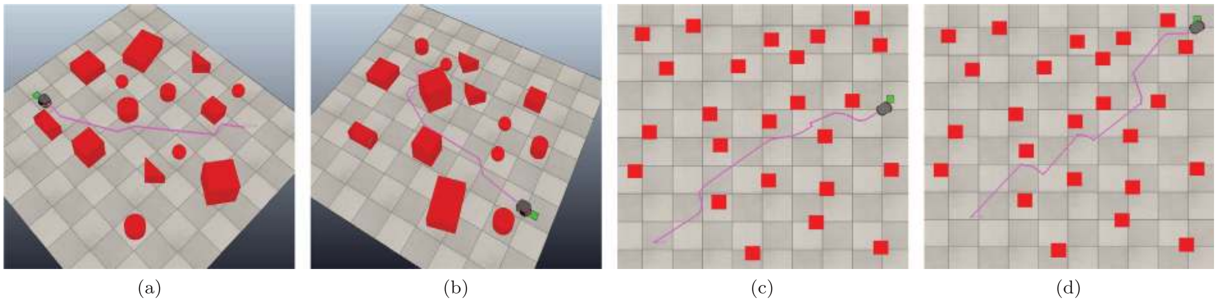

Fig. 8. Various scenes of autonomous mobile robot navigation and obstacle avoidance. (a) and (b) represents a 3D view of the obstacle avoidance and navigation of the robot. The line indicates the path followed from the start point to the target. Moreover, (c) and (d) represent a top view of autonomous obstacle avoidance and navigation in densely cluttered environments.

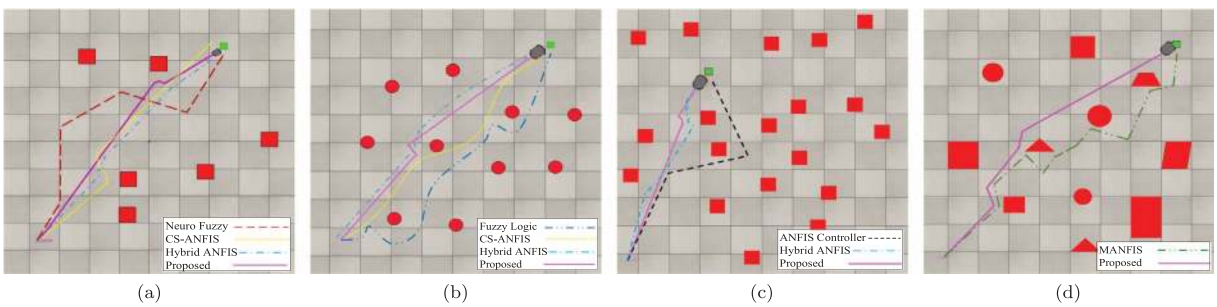

Fig. 9. This figure shows the paths followed by various methods. The sub figure (a) represents the path followed by Neuro-Fuzzy (Zhang et al., 2005), CS-ANFIS (Mohanty and Parhi, 2015), hybrid ANFIS (Gharajeh and Jond, 2020), and the proposed method in a cluttered environment. Similarly, (b) represents path planning and obstacle avoidance in a different cluttered environment among (Lakhmissi and Boumehraz, 2013; Mohanty and Parhi, 2015), hybrid ANFIS (Gharajeh and Jond, 2020), and the proposed method. The subfigure (c) represents obstacle avoidance and path planning in a densely cluttered environment by hybrid ANFIS (Gharajeh and Jond, 2020), ANFIS controller (Pothal and Parhi, 2015) and the proposed method. Furthermore, (d) presents a comparison of the path followed by MANFIS (Al-Mayyahi et al., 2014) and the proposed method in a densely cluttered environment with obstacles having different shapes and sizes.

destination without having any collision following a shorter path as stated in the proposed approach. The robot simulations are carried out in a CoppeliaSim with MATLAB controlling robot steering. It is worth noting that the performance of the proposed method does not rely on the obstacle shape so that it can work in most cases effectively. Simulation studies distinctly render that the robot can maneuver autonomously over different obstacles and achieve its target successfully.

A comparison of the proposed method and the other state-of-the-art methods is presented in Fig. 9. Fig. 9-(a) shows the path followed by Neuro-Fuzzy (Zhang et al., 2005), CS-ANFIS (Mohanty and Parhi, 2015), hybrid ANFIS (Gharajeh and Jond, 2020), and the proposed method. It can be observed that the proposed method is successfully capable of reaching the destination point while following a path that is more optimal as compared to Hybrid-ANFIS (Gharajeh and Jond, 2020) but much improved as compared to the path estimated by Neuro-Fuzzy (Zhang et al., 2005), and CS-ANFIS (Mohanty and Parhi, 2015). Moreover, Fig. 9-(b) presents a scenario with a dense obstacles environment. It can be seen that the proposed method follows the near-optimal path as compared to fuzzy logic (Lakhmissi and Boumehraz, 2013), CS-ANFIS (Mohanty and Parhi, 2015), and hybrid ANFIS (Gharajeh and Jond, 2020). Similarly, Fig. 9-(c) presents a path planning comparison with Hybrid ANFIS (Gharajeh and Jond, 2020) and ANFIS controller (Pothal and Parhi, 2015). It should be noticed that the environment is highly cluttered, and the proposed method follows the most optimal path to reach the final destination. Furthermore, an environment with obstacles having various sizes and shapes is presented in Fig. 9-(d). Regardless of the difference in sizes and shapes of the obstacles, the proposed method follows the near-optimal path as compared to MANFIS (Al-Mayyahi et al., 2014).

3.3. Statistical analysis

We conducted the Friedman rank-sum non-parametric statistical test with a 95% confidence level to ascertain the statistical significance of all competing methods with respect to the metric variables PP, Rules, CP, and time. The null hypothesis is defined as no statistical difference among all competing methods in this work. We then use the Nemenyi post hoc test to check the differences if the null hypothesis is rejected. The null hypothesis is rejected when the reported p-value of common significance is less than 0.05. The analysis is conducted on the average accuracy and average rank of all results as reported in Table 3. From Table 3, the p-value calculated for the competing methods in Fig. 9-(a) is 0.008. This p-value shows that our proposed method statistically differs from all competing methods. The Nemenyi post hoc test is further used to measure the true difference by constructing the critical difference diagram using the average ranks as given in Fig. 10-(a). The critical difference between the competing methods is 2.35.

For the competing methods in Fig. 9-(c), the reported p-value is 0.03, which rejects the null hypothesis. Specifically, our method is statistically different as compared to the competing methods. From Fig. 10-(b), the critical difference calculated from the Nemenyi post hoc test is 1.91. The p- values calculated for the competing

Table 3
Real time simulations and paramerters comparison. Additionally, PP represents Premise Parameters while CP denotes the number of Consequents Parameters. The bold values represent superior values and ↓denotes that the lesser is better.

Figure	Method	Controllers ↓	PP ↓	Rules ↓	CP ↓	Time (S) ↓
9(a)	Neuro Fuzzy	Two	32	45	765	13.2
	CS ANFIS	Single	36	81	405	11.35
	Hybrid ANFIS	Single	27	27	108	11.15
	Proposed	Single	**27**	**27**	**108**	**10.7**
9(b)	Fuzzy Logic	Two	26	56	12	14.65
	CS ANFIS	Single	36	81	405	12.8
	Hybrid ANFIS	Single	27	27	108	12.25
	Proposed	Single	**27**	**27**	**108**	**11.95**
9(c)	ANFIS Controller	Single	48	256	1280	10.85
	Hybrid ANFIS	Single	27	27	108	9.35
	Proposed	Single	**27**	**27**	**108**	**8.65**
9(d)	MANFIS	Four	125	137	137	16.9
	Proposed	Single	**27**	**27**	**108**	**13.5**

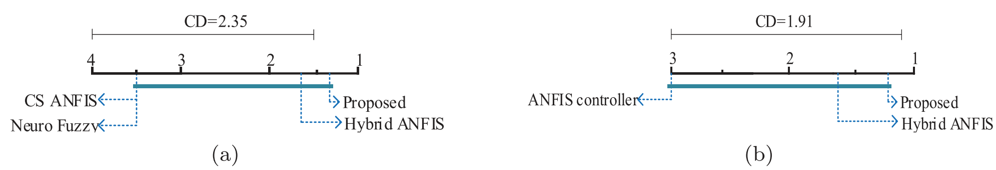

Fig. 10. (a) Represents Nemenyi test on competing methods Neuro-Fuzzy (Zhang et al., 2005), CS-ANFIS (Mohanty and Parhi, 2015), hybrid ANFIS (Gharajeh and Jond, 2020), and the proposed method in a cluttered environment as presented in Fig. 9-(a). Similarly, (b) Represents Nemenyi test on competing methods in another densely cluttered environment among hybrid ANFIS (Gharajeh and Jond, 2020), ANFIS controller (Pothal and Parhi, 2015) and the proposed method as provided in Fig. 9-(c).

methods in Figs. 9-(b) and 9-(d) are 0.11 and 0.13. Though the p-values for these two experiments do not show a statistically significant difference between the competing methods, our method has the fastest computational time and has a time performance gain of 0.3s and 3.4s as compared to the second-best time performance algorithms, hybrid ANFIS and MANFIS respectively.

3.4. Discussions

The proposed ANFIS only contains nine neurons for further processing. In contrast, others may need hundreds or more like the neural network presented in (Marichal et al., 2001 and Bozek et al., 2020) have 240 and 360 nodes in their first layers, respectively. Moreover, ANFIS is designed just with 27 rules, relatively fewer than other approaches. Generally, they need hundreds of rules, like FNN(Lv et al., 2021), which needs 160 rules (Rath et al., 2018) needs 256 rules, and invasive weed optimization (IWO)-based ANFIS (Parhi and Mohanty, 2016) needs 727 rules. A hybrid intelligent path planner (Mohanty and Parhi, 2015) based on ANFIS is designed to navigate a mobile robot. This method uses a single ANFIS controller to process inputs from the left, right, and front sensors, but it also requires the heading angle in addition to the left and right wheel velocities to calculate the correct steering angle. It indicates that the robot already knows the global path. In addition, this controller necessitates 727 rules for ANFIS's local path planning to be completed. It must also determine how to optimize 54 premise parameters and 5089 consequent parameters for training and testing, which makes it highly complex with high calculation time. A navigation controller (Subbash and Chong, 2019) for a mobile robot with a differential drive that is based on ANFIS actively plans local paths and avoids obstacles. However, this system requires two independent controls to manage the robot's left and right velocities. To gather the data set for the training and testing of the two ANFIS controllers, an additional fuzzy logic-based controller is required, doubling the effort and workload of the designer. In addition, this controller has difficulties adapting to new environments.

Fig. 9-(a) shows the comparison of the proposed approach with the neuro-fuzzy method (Zhang et al., 2005), CS-ANFIS (Mohanty and Parhi, 2015), hybrid ANFIS-GPS (Gharajeh and Jond, 2020), and the proposed approach. It can be seen that the proposed approach shows better performance as compared to both stated techniques. The quantitative results of the real-time simulations and other parameters of the other state-of-the-art methods are presented in Table 3. The proposed approach shows about 20% improvement in achieving the target in the same environments with minimum parameters employing a single ANFIS controller. Premise parameters and consequent parameters increase the system's complexity drastically. It can be observed that the neuro-fuzzy approach given in (Zhang et al., 2005) shows poor results as the path found is much longer than other techniques for mobile robot navigation.

CS-ANFIS (Mohanty and Parhi, 2015) reduced the consequent parameters from 765 to 405, but the path length remains long as the turning angles are less precise and it deviates to an extent from the destination. The hybrid ANFIS (Gharajeh and Jond, 2020) reduced the consequent parameters from 405 to 108. However, multiple turns to avoid the first obstacle add up to more travel distance and delays in reaching the destination. The proposed approach shows an improvement of 18.94% in achieving the target in the same environments with minimum parameters using a single ANFIS controller.

The authors (Lakhmissi and Boumehraz, 2013), have worked on a fuzzy logic method for obstacle avoidance and navigation of mobile robots. The simulation results of fuzzy logic (Lakhmissi and Boumehraz, 2013), CS-ANFIS (Mohanty and Parhi, 2015), hybrid ANFIS-GPS (Gharajeh and Jond, 2020), and proposed approach are presented in Fig. 9-(b) for comparison. The authors (Lakhmissi and Boumehraz, 2013) only used fuzzy logic. It is not efficient to find the goal and generates a long path as unnecessary detours are found while executing the target navigation and obstacle avoidance behavior. Similarly, CS-ANFIS (Mohanty and Parhi, 2015) method induces big curved turns resulting in a long path. Hybrid ANFIS-GPS (Gharajeh and Jond, 2020) can effectively navigate the target but obstacle avoidance behavior is not precise and multiple turns are seen while avoiding obstacles. The proposed method shows better results than the neuro-fuzzy (Zhang et al., 2005) method, CS-ANFIS (Mohanty and Parhi, 2015), and hybrid ANFIS-GPS (Gharajeh and Jond, 2020). The performance evaluation regarding path length is shorter than the neuro-fuzzy (Zhang et al., 2005) method, CS-ANFIS (Mohanty and Parhi, 2015), and hybrid ANFIS-GPS (Gharajeh and Jond, 2020). Fig. 9-(c) shows the navigation and obstacle avoidance comparison in a cluttered environment of the proposed approach, the work of (Pothal and Parhi, 2015; Gharajeh and Jond, 2020). The authors (Pothal and Parhi, 2015 and Gharajeh and Jond, 2020), have designed the method for robots to navigate through the environment using ANFIS. ANFIS controller (Pothal and Parhi, 2015) demonstrates the poor ability to follow the target navigation and turns away from the original course of direction losing its efficiency. Simulation results show that the proposed hybrid ANFIS-sensor fusion approach is more efficient as it provides a shorter path. Mayyahi (Al-Mayyahi et al., 2014) used four ANFIS controllers to navigate and avoid multiple obstacles. The comparison of the MANFIS (Al-Mayyahi et al., 2014) and the proposed approach is presented in Fig. 9-(d). The comparison shows that the path found by the proposed hybrid approach is near-optimal and shows around 20% more efficacy than other approaches in all scenarios.

4. Conclusion

This study proposes a novel hybrid navigation approach for autonomous path planning of mobile robots in cluttered environments. The approach illustrates the amalgamation of ANFIS for local motion control, whereas GPS and heading sensor for the global motion control. The proposed novel approach is more robust as it only requires 27 rules and 108 consequent parameters opposing hundreds of rules and thousands of consequent parameters compared to conventional neuro-fuzzy and ANFIS based approaches. Moreover, all the consequent parameters and rules are evaluated and clearly presented in this research work. The proposed approach is adaptive to new environments without additional training and has better repeatability to achieve the target. Additionally, the proposed obstacle avoidance method is independent of obstacle shape and size. The proposed approach maintains the physical and linguistic meaning of the parameters during the navigation and execution process, while conventional approaches lack this ability. We performed comprehensive experiments, and the results show 20% better efficacy of the proposed hybrid approach as compared to the state-of-the-art methods.

5. Future work

For our future work, we aim to enhance ANFIS and sensor fusion-based hybrid approach to solve the swarm autonomous navigation robotics problem along with hardware-based experiments.

Funding

This work is supported by Sichuan Science and Technology Program [2020JDJQ0036, 2020] and Natural Science Foundation of Sichuan Province [2022NSFSC1941, 2022].

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We wholeheartedly appreciate the assistance of Dr. Faez Iqbal Khan who moderated this paper and significantly improved it.

References

Al-Mayyahi, A., Wang, W., Birch, P., 2014. Adaptive neuro-fuzzy technique for autonomous ground vehicle navigation. Robotics 3, 349–370.
Aouf, A., Boussaid, L., Sakly, A., 2019. Same fuzzy logic controller for two-wheeled mobile robot navigation in strange environments. Journal of Robotics 2019.
Ariffin, A.A.B., Aziz, N.H.A., Othman, K.A., 2011. Implementation of gps for location tracking, in: 2011 IEEE control and system graduate research colloquium, IEEE. pp. 77–81.
Bakdi, A., Hentout, A., Boutami, H., Maoudj, A., Hachour, O., Bouzouia, B., 2017. Optimal path planning and execution for mobile robots using genetic algorithm and adaptive fuzzy-logic control. Robotics and Autonomous Systems 89, 95–109.
Baudoin, Y., Habib, M.K., 2010. Using robots in hazardous environments: Landmine detection, de-mining and other applications. Elsevier.
Bozek, P., Karavaev, Y.L., Ardentov, A.A., Yefremov, K.S., 2020. Neural network control of a wheeled mobile robot based on optimal trajectories. Int. J. Adv. Rob. Syst. 17. 1729881420916077.
Cardona, M., Cortez, F., Palacios, A., Cerros, K., 2020. Mobile robots application against covid-19 pandemic, in: 2020 IEEE ANDESCON, IEEE. pp. 1–5.
Elbatal, A., Elkhatib, M.M., Youssef, A.M., 2020. Intelligent autopilot design based on adaptive neuro-fuzzy technique and genetic algorithm. In: 2020 12th International Conference on Electrical Engineering (ICEENG). IEEE, pp. 377–382.
Farahat, H., Farid, S., Mahmoud, E.O., 2019. Adaptive neuro-fuzzy control of autonomous ground vehicle (agv) based on machine vision. Engineering Research Journal 163, 218–233.
Freese, M., Singh, S., Ozaki, F., Matsuhira, N., 2010. Virtual robot experimentation platform v-rep: A versatile 3d robot simulator. In: International Conference on Simulation, Modeling, and Programming for Autonomous Robots. Springer, pp. 51–62.
Gharajeh, M.S., Jond, H.B., 2020. Hybrid global positioning system-adaptive neuro-fuzzy inference system based autonomous mobile robot navigation. Robotics and Autonomous Systems 134, 103669.
Haider, M.H., Ali, H., Khan, A.A., Zheng, H., Bhutta, M.U.M., Usman, S., Zhi, P., Wang, Z., 2022. Autonomous mobile robot navigation using adaptive neuro fuzzy inference system. In: 2022 International Conference on Innovations and Development of Information Technologies and Robotics (IDITR). IEEE, pp. 93–99.
Holland, J.M., 2004. Designing autonomous mobile robots: Inside the mind of an intelligent machine. Elsevier.
Jang, J.S., 1993. Anfis: adaptive-network-based fuzzy inference system. IEEE transactions on systems, man, and cybernetics 23, 665–685.
Kamil, F., Moghrabiah, M.Y., 2021. Multilayer decision-based fuzzy logic model to navigate mobile robot in unknown dynamic environments. Fuzzy Information and Engineering, 1–23.
Karthikeyan, M., Sathiamoorthy, S., Vasudevan, M., 2019. Adaptive neuro fuzzy inference system based obstacle avoidance system for autonomous vehicle. In: International Conference on Innovative Data Communication Technologies and Application. Springer, pp. 118–126.
Klancar, G., Zdesar, A., Blazic, S., Skrjanc, I., 2017. Wheeled mobile robotics: from fundamentals towards autonomous systems. Butterworth-Heinemann.
Lakhmissi, C., Boumehraz, M., 2013. Fuzzy behavior based navigation approach for mobile robot in unknown environment. Journal of Electrical Engineering 13, 284–291.
Li, F.F., Du, Y., Jia, K.J., 2022. Path planning and smoothing of mobile robot based on improved artificial fish swarm algorithm. Scientific Reports 12, 1–16.
Lv, J., Qu, C., Du, S., Zhao, X., Yin, P., Zhao, N., Qu, S., 2021. Research on obstacle avoidance algorithm for unmanned ground vehicle based on multi-sensor information fusion. Mathematical Biosciences and Engineering 18, 1022–1039.
Ma, H., Ma, Y., Jiao, J., Bhutta, M.U.M., Bocus, M.J., Wang, L., Liu, M., Fan, R., 2018. Multiple lane detection algorithm based on optimised dense disparity map estimation. In: 2018 IEEE International Conference on Imaging Systems and Techniques (IST), pp. 1–5. https://doi.org/10.1109/IST.2018.8577122.
Marichal, G., Acosta, L., Moreno, L., Méndez, J.A., Rodrigo, J., Sigut, M., 2001. Obstacle avoidance for a mobile robot: A neuro-fuzzy approach. Fuzzy Sets Syst. 124, 171–179.
Mohanty, P.K., Parhi, D.R., 2015. A new hybrid intelligent path planner for mobile robot navigation based on adaptive neuro-fuzzy inference system. Australian Journal of Mechanical Engineering 13, 195–207.

Mohanty, P.K., Parhi, D.R., 2015. A new hybrid optimization algorithm for multiple mobile robots navigation based on the cs-anfis approach. Memetic Computing 7, 255–273.

Mondal, S., Ray, R., Reddy, S., Nandy, S., 2022. Intelligent controller for nonholonomic wheeled mobile robot: A fuzzy path following combination. Mathematics and Computers in Simulation 193, 533–555.

Naghsh, A., Sheikholeslam, F., Danesh, M., 2014. Design of an adaptive fuzzy estimator for force/position tracking in robot manipulators. Iranian Journal of Fuzzy Systems 11, 75–89.

Nakrani, N.M., Joshi, M.M., 2022. A human-like decision intelligence for obstacle avoidance in autonomous vehicle parking. Applied Intelligence 52, 3728–3747.

Parhi, D.R., Mohanty, P.K., 2016. Iwo-based adaptive neuro-fuzzy controller for mobile robot navigation in cluttered environments. The International Journal of Advanced Manufacturing Technology 83, 1607–1625.

Pothal, J.K., Parhi, D.R., 2015. Navigation of multiple mobile robots in a highly clutter terrains using adaptive neuro-fuzzy inference system. Robotics and Autonomous Systems 72, 48–58.

Rath, A.K., Parhi, D.R., Das, H.C., Muni, M.K., Kumar, P.B., 2018. Analysis and use of fuzzy intelligent technique for navigation of humanoid robot in obstacle prone zone. Defence technology 14, 677–682.

Rohmer, E., Singh, S.P., Freese, M., 2013. V-rep: A versatile and scalable robot simulation framework. In: in: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems IEEE, pp. 1321–1326.

Santoso, F., Garratt, M., Anavatti, S., 2016. Fuzzy system identification for the dynamics of the ar. drone quadcopter, in: Australasian Conference on Robotics and Automation, pp. 69–74.

Shahidinejad, A., Ghobaei-Arani, M., Esmaeili, L., 2020. An elastic controller using colored petri nets in cloud computing environment. Cluster Computing 23, 1045–1071.

Subbash, P., Chong, K.T., 2019. Adaptive network fuzzy inference system based navigation controller for mobile robot. Frontiers of Information Technology & Electronic Engineering 20, 141–151.

Sugeno, M., Kang, G., 1988. Structure identification of fuzzy model. Fuzzy sets and systems 28, 15–33.

Takagi, T., Sugeno, M., 1985. Fuzzy identification of systems and its applications to modeling and control. IEEE transactions on systems, man, and cybernetics, 116–132.

Thor, M., Manoonpong, P., 2022. Versatile modular neural locomotion control with fast learning. Nature Machine Intelligence, 1–11.

Troccaz, J., 2013. Medical robotics. John Wiley & Sons.

Vu, N.T.T., Tran, N.P., Nguyen, N.H., 2018. Adaptive neuro-fuzzy inference system based path planning for excavator arm. Journal of Robotics 2018.

Wu, R., Fan, J., Guo, L., Qiao, L., Bhutta, M.U.M., Hosking, B., Vityazev, S., Fan, R., 2021. Scale-adaptive pothole detection and tracking from 3-d road point clouds. In: 2021 IEEE International Conference on Imaging Systems and Techniques (IST), pp. 1–5. https://doi.org/10.1109/IST50367.2021.9651423.

Xiao, X., Liu, B., Warnell, G., Stone, P., 2022. Motion planning and control for mobile robot navigation using machine learning: a survey. Autonomous Robots, 1–29.

Zhang, N., Beetner, D., Wunsch, D.C., Hemmelman, B., Hasan, A., 2005. An embedded real-time neuro-fuzzy controller for mobile robot navigation, in: The 14th IEEE International Conference on Fuzzy Systems, 2005. FUZZ'05., IEEE. pp. 319–324.