robotics

MDPI

Article

An ANFIS-Based Strategy for Autonomous Robot Collision-Free Navigation in Dynamic Environments

Stavros Stavrinidis and Paraskevi Zacharia *

Department of Industrial Design and Production Engineering, University of West Attica, 12241 Egaleo, Greece; s.stavrinidis@uniwa.gr

* Correspondence: p.zacharia@uniwa.gr; Tel.: +30-210-5381055

Abstract: Autonomous navigation in dynamic environments is a significant challenge in robotics. The primary goals are to ensure smooth and safe movement. This study introduces a control strategy based on an Adaptive Neuro-Fuzzy Inference System (ANFIS). It enhances autonomous robot navigation in dynamic environments with a focus on collision-free path planning. The strategy uses a path-planning technique to develop a trajectory that allows the robot to navigate smoothly while avoiding both static and dynamic obstacles. The developed control system incorporates four ANFIS controllers: two are tasked with guiding the robot toward its end point, and the other two are activated for obstacle avoidance. The experimental setup conducted in CoppeliaSim involves a mobile robot equipped with ultrasonic sensors navigating in an environment with static and dynamic obstacles. Simulation experiments are conducted to demonstrate the model's capability in ensuring collision-free navigation, employing a path-planning algorithm to ascertain the shortest route to the target destination. The simulation results highlight the superiority of the ANFIS-based approach over conventional methods, particularly in terms of computational efficiency and navigational smoothness.

Keywords: mobile robot; navigation; sensors; path planning; static and dynamic obstacles; neuro-fuzzy controllers; CoppeliaSim

check for updates

Citation: Stavrinidis, S.; Zacharia, P. An ANFIS-Based Strategy for Autonomous Robot Collision-Free Navigation in Dynamic Environments. *Robotics* **2024**, *13*, 124. https://doi.org/10.3390/robotics13080124

Academic Editor: Giovanni Muscato

Received: 24 May 2024
Revised: 4 August 2024
Accepted: 21 August 2024
Published: 22 August 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The robotics research community focuses strongly on autonomous navigation in dynamic environments. This is especially true for the development of intelligent autonomous guided robots with significant industrial applications. The use of mobile robots in manufacturing, warehousing, and construction is increasing. Real-world scenarios pose challenges when robots and humans share the same environment. This necessitates effective human-robot interaction mechanisms. For example, in manufacturing, autonomous guided vehicles may encounter human obstructions and need to implement collision avoidance measures. Researchers have explored various methodologies incorporating different sensor types, including laser measurement systems [1], lidar sensors [2,3] and cameras [4].

Artificial intelligence (AI) techniques show significant potential in addressing complex challenges within dynamic and uncertain environments. Robots capable of demonstrating complex behaviors by interpreting their environment and adapting their actions through sensors and ANFIS control mechanisms are commonly referred to as behavior-based systems. The application of fuzzy logic and neural networks has shown effectiveness in the field of autonomous robot navigation.

Miao et al. [5] presented an obstacle avoidance fuzzy controller utilizing fuzzy control algorithms. This controller utilized operational data to establish an AGV adaptive neuro-fuzzy network system, capable of providing precise behavioral commands for dynamic obstacle avoidance in AGVs.

Haider et al. [6] introduced a solution that leverages an Adaptive Neuro-Fuzzy Inference System (ANFIS) in conjunction with a Global Positioning System (GPS) for control

and navigation, effectively addressing issues such as low performance in cluttered and unknown environments, high computational costs and multiple controller models for navigation. The proposed approach automates mobile robot navigation in densely cluttered environments, integrating GPS and heading sensor data fusion for global path planning and steering. A fuzzy inference system (FIS) is employed for obstacle avoidance, incorporating distance sensor data as fuzzy linguistics. Furthermore, a type-1 Takagi–Sugeno FIS is utilized to train a five-layered neural network for local robot planning, with ANFIS parameters fine-tuned through a hybrid learning method.

Farahat et al. [7] focuses on an extensive analysis and investigation into AI algorithm techniques for controlling the path planning of a four-wheel autonomous robot, enabling it to autonomously reach predefined objectives while intelligently circumventing static and dynamic obstacles. The research explores the viability of employing Fuzzy Logic, Neural Networks, and the Adaptive Neuro-Fuzzy Inference System (ANFIS) as control algorithms for this intricate navigation task.

Jung et al. [8] explore a vision guidance system for Automated Guided Vehicles (AGVs) that leverages an ANFIS. The vision guidance system is founded on a driving method capable of recognizing salient features such as driving lines and landmarks, providing a data-rich alternative to other induction sensors in AGVs. Due to the camera's sensitivity to lighting variations, a "dark-room environment" was developed to ensure consistent lighting conditions, minimizing brightness disturbances that affect camera performance. This controlled environment limits external light interference, stabilizing illumination to enhance the reliability of the vision system. Although this approach restricts the camera's viewing angle and presents control challenges at high speeds or sudden changes in direction, it proves effective with ANFIS-enhanced steering control, offering an improvement over traditional Proportional–Integral–Derivative (PID) methods. This specialized setup acknowledges the challenges of camera-only systems and contrasts with broader autonomous driving technologies that integrate multiple sensors to adapt to diverse environmental conditions.

Khelchandra et al. [9] introduced a method for path planning considering mobile robot navigation within environments with static obstacles, which combines neuro-genetic-fuzzy techniques. This approach involves training an artificial neural network to select the most suitable collision-free path from a range of available paths. Additionally, a genetic algorithm is employed to enhance the efficiency of the fuzzy logic system by identifying optimal positions within the collision-free zones.

Faisal et al. [10] presented a technique for fuzzy logic-based navigation and obstacle avoidance in dynamic environments. This method integrates two distinct fuzzy logic controllers: a tracking fuzzy logic controller, which steers the robot toward its target, and an obstacle avoidance fuzzy logic controller. The principal aim of this research is to demonstrate the utility of mobile robots in material handling tasks within warehouse environments. It introduces a notable departure from the traditional colored line tracking approach, replacing it with wireless communication methods.

A thorough review of recent literature reveals various control strategies employed in robotic systems for navigation in cluttered environments. Several studies have focused on the application of ANFIS for optimal heading adjustments, real-time decision-making, and improved navigation efficiency in dynamic environments. For instance, ref. [11] discusses a multiple ANFIS architecture for mobile robot navigation, demonstrating its effectiveness in navigating environments with both static and dynamic obstacles. Other studies, such as [12], propose ANFIS-based path-planning techniques using ultrasonic sensors to enhance collision-free navigation. Similarly, ref. [13] explores the use of ANFIS in real-world dynamic environments, highlighting the hybrid system's ability to combine fuzzy logic and neural network advantages. Moreover, ref. [14] presents an ANFIS controller for efficient navigation in densely cluttered environments, while [15] introduces a hybrid system combining ANFIS with ant colony optimization for enhanced navigation. Additionally, ref. [6] uses ANFIS and GPS for robust navigation and obstacle avoidance in unknown environments.

Furthermore, research on fuzzy logic has demonstrated its utility in handling uncertainty and imprecise data, contributing to the development of robust fuzzy logic controllers integrated into the ANFIS framework. For example, ref. [16] employs a fuzzy logic controller for sensor-actuator control in mobile robots, showcasing improved navigation in the presence of static and moving obstacles. In [17], fuzzy logic is applied to optimize the trajectory of autonomous robots, providing efficient and adaptive path planning. Study [18] explores sensor fusion techniques using fuzzy logic to enhance the accuracy of environmental perception. Additionally, the authors in [19] investigate the application of fuzzy logic in adaptive control systems, which enables robots to adjust to varying environmental conditions dynamically. Research [20] highlights the integration of fuzzy logic in hybrid systems for effective obstacle avoidance. The work in [21] presents a fuzzy-based approach for real-time decision-making in autonomous navigation. The work in [22] applies fuzzy logic to improve the precision of motion control systems in robots. Furthermore, ref. [23] examines the role of fuzzy logic in enhancing the reliability of sensor data interpretation. The work in [24] focuses on developing fuzzy logic algorithms for dynamic obstacle detection and avoidance. In [25], fuzzy logic is utilized to improve the robustness of control systems under uncertain conditions. The research in [26] integrates fuzzy logic with neural networks to optimize navigation strategies. The work in [27] explores the application of fuzzy logic in multi-robot coordination and path planning. Lastly, the study presented in [28] demonstrates the effectiveness of fuzzy logic in improving the adaptability and responsiveness of autonomous navigation systems. These studies cover various aspects such as trajectory optimization, sensor fusion, and adaptive control, all of which form the foundation for the design of our ANFIS-based navigation strategy.

Various artificial intelligence techniques including reinforcement learning, neural networks, fuzzy logic and genetic algorithms enhance the reactive navigation capabilities of mobile robots. Particularly, fuzzy logic is notable for its capacity to articulate linguistic terms and make reliable decisions amidst uncertainty and imprecise data, proving invaluable in control systems. Fuzzy control systems, which are rule-based and draw on domain knowledge or expert insights, use fuzzy if-then rules to simplify decision-making without needing detailed computations. These systems are widely recognized for their ability to handle a broad range of tasks effectively. Fuzzy logic excels in dynamic or unknown environments by mimicking human reasoning and decision-making processes, thus facilitating robust and quick-responsive navigation. This approach not only adapts to the unpredictable nature of real-world settings but also does not require precise environmental models for effective navigation [29].

A notable limitation of fuzzy controllers is the absence of a systematic design methodology, frequently resulting in labor-intensive tuning of membership function parameters. The integration of neural network learning techniques with fuzzy logic has enabled the development of neuro-fuzzy controllers, significantly accelerating the design process and enhancing performance. This hybrid approach has become a prominent area of research within the field of robot navigation in uncertain environments [13].

Recent research has advanced the field with studies such as [30], which explores virtual navigation of a Pioneer P3-DX mobile robot using ANFIS and ANFIS-PSO algorithms, demonstrating PSO's optimization for better trajectory tracking and obstacle avoidance. Additionally, ref. [31] proposes a Sliding Mode Control (SMC) with Nonlinear Disturbance Observer (NDO) and Neural State Observer (NSO) to improve trajectory tracking under disturbances, showing superior performance in ROS/Gazebo simulations. Another study [32] employs an ANFIS-based sensor fusion approach to combine LiDAR and RTK-GNSS/INS data, significantly enhancing positioning accuracy and reliability for outdoor mobile robots.

Significant progress has been made in applying fuzzy control and ANFIS to mobile robotics, enhancing adaptability in dynamic environments. Despite these advances, traditional systems still struggle with rapid adaptability to new or evolving conditions and exhibit computational inefficiencies in complex scenarios. This underscores the necessity for further research aimed at developing simpler, more efficient control models. Simplifying

these systems would not only expedite response times but also decrease computational demands, thereby advancing the capabilities of autonomous navigation technologies.

In our present work, we employ a combination of four Adaptive Neuro-Fuzzy Inference Systems (ANFIS) to facilitate mobile robot navigation towards a desired destination while ensuring collision avoidance capabilities for both stationary and mobile obstacles. This approach focuses on the application of a path-planning algorithm to establish an efficient route from the start point to the end point, ensuring a smooth path within a static environment. During this initial phase, two ANFIS controllers are dedicated to tracking and guiding the robot towards the end point. However, if the robot encounters a mobile obstacle during its trajectory, another pair of ANFIS controllers is activated, specifically designed for obstacle avoidance maneuvers. Once the obstacle avoidance action is executed, the robot smoothly resumes its original planned path.

In our experimental evaluations, we conduct a comparative analysis between the ANFIS controllers and fuzzy logic controllers. Significantly, our ANFIS controllers exhibit improved operational efficiency, employing fewer rules compared to the fuzzy logic controllers, emphasizing a simpler approach. The proposed control algorithm, with ANFIS at its core, is proved to be both simple yet effective, providing valuable insights into robotic navigation in dynamic environments within the existing literature.

The innovation of this work is summarized around four key points:

1. Development of a novel control strategy that employs the ANFIS model's capabilities to simplify and facilitate the decision-making process, thereby enhancing the system's computational efficiency.
2. Application of four Adaptive Neuro-Fuzzy Inference Systems (ANFIS) that are expertly optimized to minimize the rule set of fuzzy controllers, while maintaining system efficiency.
3. Enabling efficient navigation in environments cluttered with obstacles through a simplified rule structure.
4. Integration of a path-planning algorithm that adds a layer of sophistication, enhancing the determination of optimal trajectories and significantly improving the efficiency and effectiveness of autonomous robot navigation.

The paper is organized as follows: Section 2 provides a concise overview of the kinematics of Autonomous Guided Vehicles, essential for understanding the subsequent development of the control system. Section 3 introduces the novel path-planning strategy, crucial for autonomous navigation. Section 4 presents a comparative analysis of ANFIS and PID controllers, underscoring the advantages of ANFIS in dynamic environments. Section 5 presents the design of the ANFIS controllers, explaining their roles in tracking and avoidance within the robotic control system. In Section 6, the simulation experiments conducted in CoppeliaSim are presented. Finally, conclusions and directions for future advancements in robotic navigation are presented in Section 7.

2. Kinematics of a Two-Wheel Differential Drive Robot

This research focuses on designing a control system for safe and reliable navigation of a mobile robot in dynamic environments. This requires careful consideration of the robot's kinematic and geometric properties. The following section presents the kinematics of a two-wheel differential drive robot, like the Pioneer p_3dx [33] used in this study.

Figure 1 illustrates two critical frames of reference that describe the robot's movement: a stationary global reference frame (I), denoted by X_I and Y_I, and a moving local reference frame (R), denoted by X_R and Y_R, fixed to the robot. Establishing these frames is fundamental for developing a robust control strategy that can dynamically adapt to the environment. It is also noted that θ represents the robot's orientation, and G marks its position.

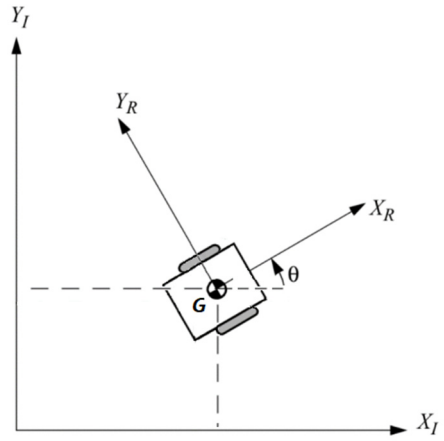

Figure 1. A visual depiction of the robot within both the global and local reference frames.

The inertial frame is considered as the fixed reference frame, typically aligned with global coordinates. The robot frame is attached to the mobile robot, with its origin at the center of the robot and axes aligned with the robot's forward, lateral, and vertical directions. The sensor frame is specific to each sensor, with its origin at the sensor's location on the robot and axes oriented according to the sensor's mounting position.

The kinematic model for the wheeled robot's motion builds upon three foundational assumptions: each wheel maintains perpendicularity to its respective plane; there is a singular contact point between the wheel and the plane; and the wheel rolls without slippage. Integrated into this model are two key constraints: firstly, the wheel's translational motion along the plane corresponds directly with its rotational velocity; secondly, no translational motion occurs perpendicularly to the wheel plane.

In accordance with [24], the differential drive configuration is characterized by two wheels placed on the same axis, each independently driven by separate motors. The motion of the robot is a cumulative outcome of the contributions from each of these wheels, and understanding this dynamic is essential for implementing precise control. The model is mathematically described as follows:

$$u = \frac{r}{2}\left(\omega_{right} + \omega_{left}\right) \tag{1}$$

$$\omega = \frac{r}{L}\left(\omega_{right} - \omega_{left}\right) \tag{2}$$

where u denotes the linear velocity of the robot, ω its angular velocity, r the radius of the wheels, ω_{right} and ω_{left} the angular velocities of the right and left wheels, respectively, and L the distance between the two wheels.

3. Strategy for Path Planning

3.1. Path Planning Concepts

Path planning is a crucial and essential issue in autonomous mobile robot navigation. Over the past two decades, extensive research, as presented in [24], has been dedicated to the domain of path planning for mobile robots in various environments. This procedure involves the identification of a collision-free path that connects two points while minimizing the associated route cost. In online path planning, the robot utilizes sensor data to gather information about its surroundings and construct an environmental map. Conversely, offline path planning entails the robot independently acquiring environmental information without dependence on sensors. This process plays a crucial role in enabling effective autonomous navigation.

3.2. The Bump-Surface Concept

In the present study, we implement the Bump-Surface concept [34] as a strategic approach to path planning. This method offers a solution to the path-planning challenges encountered by mobile robots operating within 2D static environments. The development of the Bump-Surface method involves the utilization of a control-point network, the density of which can be adjusted to fulfill specific path-planning precision requirements. In summary, increased grid density results in enhanced precision. Moreover, leveraging the flexibility of B-Spline surfaces enables us to achieve the desired level of accuracy, utilizing their features for both local and global control. Subsequently, a genetic algorithm (GA) is employed to explore this surface, aiming to find an optimal collision-free path that aligns with the objectives and constraints of motion planning.

4. Basic Concepts on the Adaptive Neuro-Fuzzy Inference System (ANFIS)

The Adaptive Neuro-Fuzzy Inference System [35] represents a versatile computational framework that integrates the merits of fuzzy logic and neural networks. This fusion equips ANFIS to handle intricate problems that might pose challenges for conventional mathematical methodologies. Essentially, ANFIS combines the human-like, intuitive reasoning of fuzzy logic with the data-driven capabilities of neural networks. It serves as a valuable tool for tasks involving modeling, prediction, and rule-based decision-making. ANFIS models encompass fuzzy logic membership functions that collaborate with adaptive neural network nodes. These nodes undergo learning and enhancement through training. The integration of fuzzy and neural components renders ANFIS adaptable and robust, making it a powerful tool for addressing real-world issues, ranging from system control to pattern recognition.

The ANFIS method features a structure that combines a fuzzy inference system with a neural network, utilizing input and output data pairs. This configuration acts as a self-adapting and flexible hybrid controller with embedded learning algorithms. Fundamentally, this method enables fuzzy logic to dynamically adjust the parameters of membership functions, aligning the fuzzy inference system with the input and output data of the ANFIS model. Adapting neural networks to manage fuzzy rules necessitates specific modifications in the conventional neural network structure, enabling the integration of these two vital components.

ANFIS combines the strengths of fuzzy logic and neural networks, making it highly effective for complex problems that challenge standard mathematical methods. While PID controllers are effective for environments with linear and nonlinear characteristics, ANFIS exhibits superior performance particularly in dynamically evolving environments without necessitating a complex mathematical model, allowing it to handle unpredictably changing conditions more effectively. This capability allows ANFIS to navigate through unpredictably changing conditions more effectively, as it fuses the intuitive, human-like reasoning of fuzzy logic with the learning power of neural networks, enhancing its adaptability and overall effectiveness.

ANFIS models use fuzzy logic membership functions and adaptive neural network nodes that learn and improve through training. This allows ANFIS to adjust automatically to unexpected changes, such as obstacles or varying terrain—something PID controllers struggle with. The system integrates fuzzy and neural elements to become both adaptable and robust, making it ideal for real-world applications like system control and pattern recognition. ANFIS features a unique setup that merges a fuzzy inference system with a neural network, using data pairs to guide adjustments. This setup forms a flexible, self-learning hybrid controller that can adjust fuzzy rules dynamically, a necessary feature for integrating fuzzy logic with neural networks.

5. Designing the ANFIS Controllers for the Automated Guided Vehicle

In this section, we conduct a comprehensive analysis of the proposed ANFIS controllers, which have been designed and implemented using the ANFIS Toolbox in MATLAB R2019b. Notably, we have devised four individual ANFIS controllers, each serving a distinct role: initially, two ANFIS controllers are designed to facilitate reaching the end point by following a predefined path, while the remaining two ANFIS controllers are created to manage obstacle avoidance tasks.

5.1. ANFIS Tracking Controllers

The primary goal of our robot is to navigate towards a specified end point within its environment, utilizing sensor feedback for real-time adjustments. The tracking controller's role is to align the robot's orientation with the desired point. It operates based on two inputs: the position error and the heading error. The position error is calculated using the Euclidean distance between the current position of the mobile robot and its target point along the planned trajectory. This error is measured in meters and normalized within the range of [0, 1]. The heading error, measured in degrees, represents the angular discrepancy between the robot's current heading and the direct line to the target. It is normalized between $[-1, 1]$.

To achieve the goal of reaching the end point, two ANFIS tracking controllers are introduced. One controller manages the right motor velocity, while the other is responsible for the left motor velocity. Both controllers have two inputs: the position error and the heading error, which collectively determine the motor velocity (right and left), as shown in Figure 2. Importantly, when the position error reaches zero, indicating that the robot has reached its target, the robot stops moving to prevent further unnecessary motion.

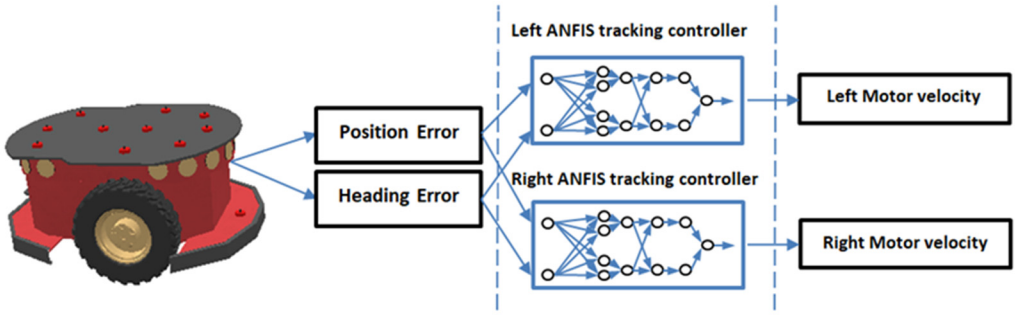

Figure 2. ANFIS Tracking Controllers: inputs—position and heading errors, outputs—motor velocities for path tracking.

5.1.1. Data Set for ANFIS Tracking Controllers

To create the dataset for training the ANFIS tracking controllers, we first developed a fuzzy logic tracking controller. Following that, we utilized this fuzzy logic tracking controller to generate the dataset required for training the ANFIS tracking controllers.

The fuzzy logic tracking controller has two inputs and two outputs. The first input is the position error, and the second is the heading error. The first output controls the velocity of the right motor, and the second output controls the velocity of the left motor.

The position error is measured in meters and normalized within the range of [0, 1], and the heading error is measured in degrees and normalized between $[-1, 1]$. The fuzzy sets expressing the position error use trapezoid membership functions to define linguistic terms, namely "small", "medium" and "large" (Figure 3). These terms are employed to represent position error values as follows.

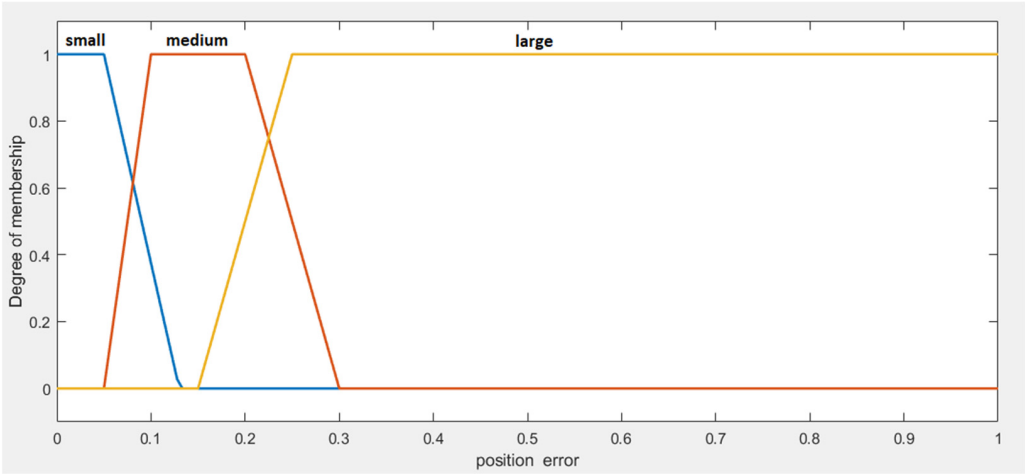

Figure 3. Fuzzy logic tracking controller position error fuzzy sets.

To analyze position errors more thoroughly within the range of 0 to 1, we have categorized them into three fuzzy sets: 'small', 'medium', and 'large' (Figure 3). Each fuzzy set represents a different degree of error.

- 'small' represents values from 0 to 0.13, indicating that the robot is very close to the target.
- 'medium' ranges from 0.05 to 0.3, suggesting a moderate distance from the target.
- 'large' encompasses values from 0.15 to 1, indicating that the robot is far from the target.

The same approach is applied to heading error, as shown in Figure 4: Five fuzzy sets, namely 'negative big', 'negative small', 'zero', 'positive small', and 'positive big', represent different ranges of heading deviations. This approach helps us better understand and manage the AGV's orientation during navigation tasks.

- 'negative big' represents values ranging from -1 to -0.3, indicating a significant deviation in the negative direction.
- 'negative small' spans values from -0.55 to -0.07, suggesting a moderate negative deviation.
- 'zero' encompasses values from -0.15 to 0.15, indicating no deviation from the desired heading.
- 'positive small' extends from 0.07 to 0.55, representing a moderate positive deviation.
- 'positive big' ranges from 0.3 to 1, indicating a significant deviation in the positive direction.

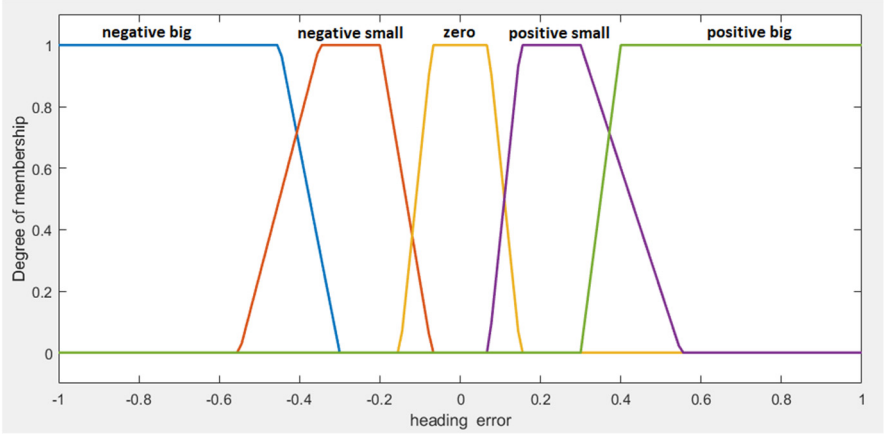

Figure 4. Fuzzy logic tracking controller heading error fuzzy sets.

In this study, the output velocity (m/s) of both the left and right motors plays a crucial role in the performance of our AGV navigation system. For better understanding of the velocity output, we have categorized this range into three fuzzy sets: "slow", "medium" and "fast" (Figure 5), as follows:

- 'slow': covering values from 0 to 0.4, representing slower motor speeds.
- 'medium': ranging from 0.27 to 0.6, indicating intermediate motor speeds.
- 'fast': extending from 0.47 to 1, denoting higher motor speeds.

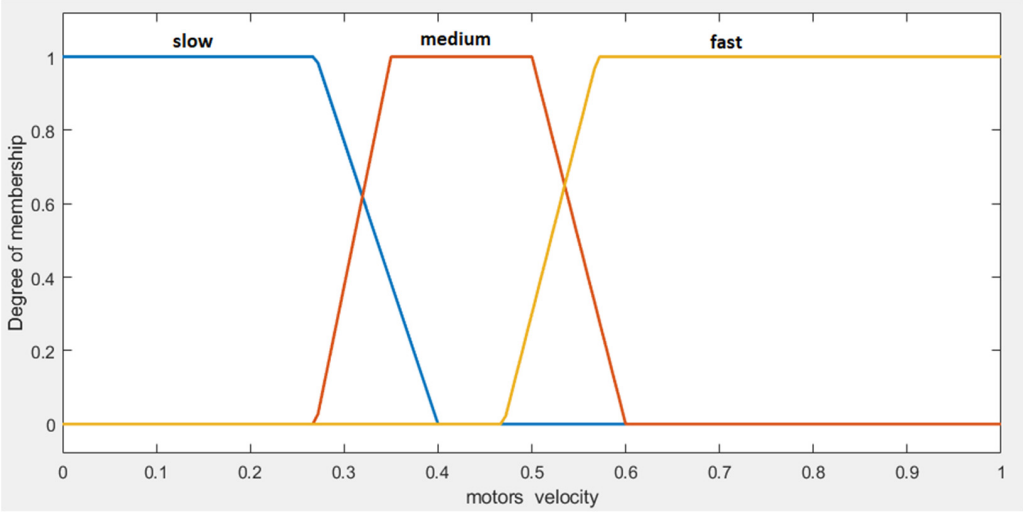

Figure 5. Fuzzy logic tracking controller output velocity fuzzy sets (right and left motors).

Based on our knowledge of the AGV navigation system, a set of 15 expert rules is defined guiding the AGV's actions across various scenarios, effectively establishing a fuzzy tracking controller. This set of rules is a crucial component of our navigation control strategy (Table 1).

Table 1. Rule set for fuzzy tracking controller.

Rule	Position Error	Heading Error	Right Motor Velocity	Left Motor Velocity
1	large	negative big	medium	fast
2	large	negative small	slow	medium
3	large	zero	fast	fast
4	large	positive small	medium	slow
5	large	positive big	fast	medium
6	medium	negative big	medium	fast
7	medium	negative small	slow	medium
8	medium	zero	fast	fast
9	medium	positive small	medium	slow
10	medium	positive big	fast	medium
11	small	negative big	slow	fast
12	small	negative small	slow	fast
13	small	zero	slow	slow
14	small	positive small	fast	slow
15	small	positive big	fast	slow

The training dataset has been generated in accordance with the expert logical rules described previously. These rules define the connections between position error, heading error, and motor velocities within the AGV navigation system. Following these rules ensures that the training data thoroughly encompass a wide spectrum of scenarios and behaviors, accurately capturing the AGV's responses to various error conditions. This dataset is

fundamental for training the AGV navigation system and optimizing its performance. A sample of training data for the ANFIS tracking controller is presented in Table 2.

Table 2. Sample of training data generated by the fuzzy tracking controller.

Position Error	Heading Error	Expected Right Motor Velocity	Expected Left Motor Velocity
0.9	0	0.7578	0.7578
0.8	0	0.7578	0.7578
0.7	0	0.7578	0.7578
0.6	0	0.7578	0.7578
0.5	0	0.7578	0.7578
0.4	0	0.7578	0.7578
0.3	0	0.7578	0.7578
0.2	0	0.7572	0.7572
0.1	0	0.6055	0.6055
0	0.9	0.7578	0.1674
0	0.8	0.7578	0.1674
0	0.7	0.7578	0.1674
0	0.6	0.7578	0.1674
0	0.5	0.7578	0.1674
0	0.4	0.7546	0.1713

5.1.2. Training ANFIS for Tracking Controllers (Left and Right)

In this work, a dataset consisting of 228 data points is generated using the fuzzy logic tracking controller to train the ANFIS tracking controllers (left and right). In the proposed ANFIS system setup, three membership functions have been used for each input for each controller. These functions are of the trapezoidal type, which contributes to capturing complex patterns effectively. For the output, linear membership functions are chosen, ensuring straightforward mapping of rules. The system is generated using a grid partition approach for efficient input organization and a hybrid optimization method is applied during training to improve performance. Finally, the system is trained over 200 iterations (epochs). The training error is calculated as the root mean square error (RMSE) between the predicted motor velocities (outputs of the ANFIS controller) and the actual motor velocities (target outputs). These settings enhance the precision and adaptability of our system for various robotics applications. Figure 6 presents the training data for the left and right motor ANFIS tracking controller while Figure 7 depicts the training error for the same controllers through 200 epochs.

Figure 8 displays the training and FIS output data for the left and right motor ANFIS tracking controller after completion of the ANFIS training process. The actual training data points (generated from the fuzzy logic controller) are depicted as blue circles and the ANFIS output predicted after training are depicted as red asterisks. The proposed ANFIS model performed well during training with a minimal error of 0.0458 for the left motor and 0.0462 for the right motor.

This model uses a combination of 35 components, including 27 linear and 24 nonlinear parameters. The ANFIS tracking controller for the right motor follows a similar pattern, employing 9 rules, which is a reduction of 6 rules compared to the previous fuzzy controller with 15 rules (refer to Section 5.1.1). This ANFIS model is versatile and can be useful for various robotics applications.

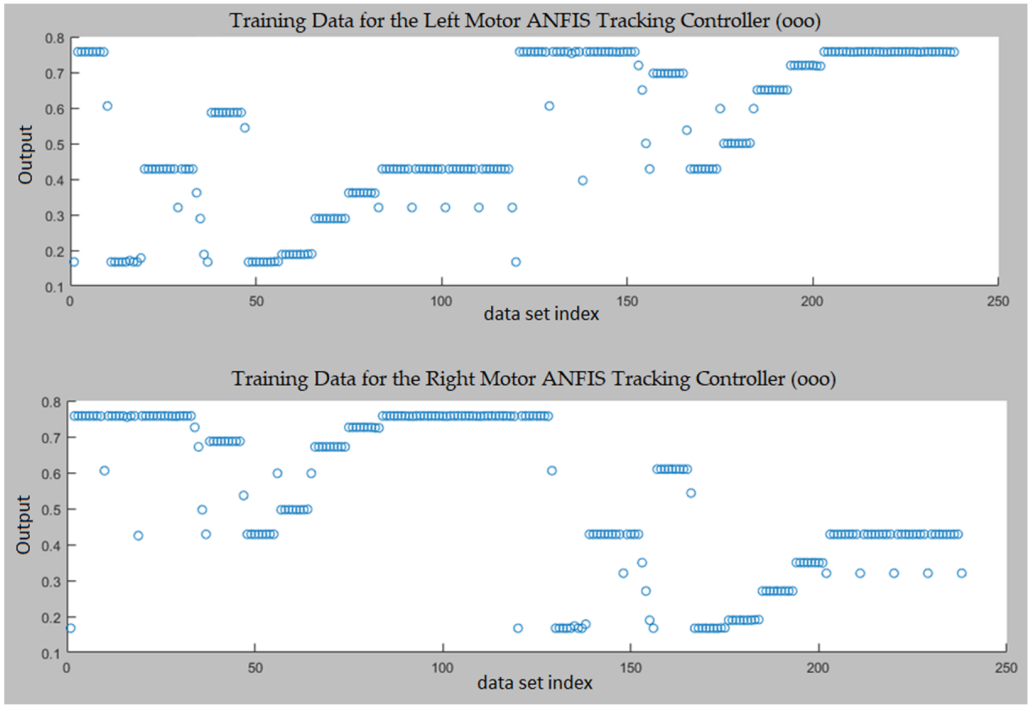

Figure 6. Training data for the left and right motor ANFIS tracking controller.

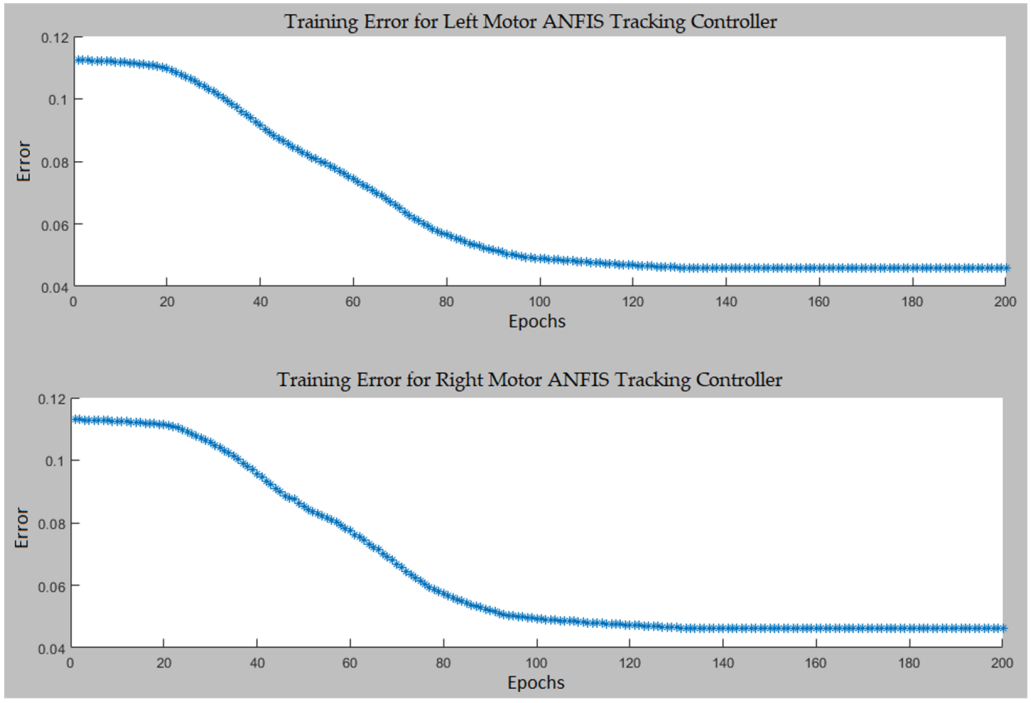

Figure 7. Training error for left and right motor ANFIS tracking controller.

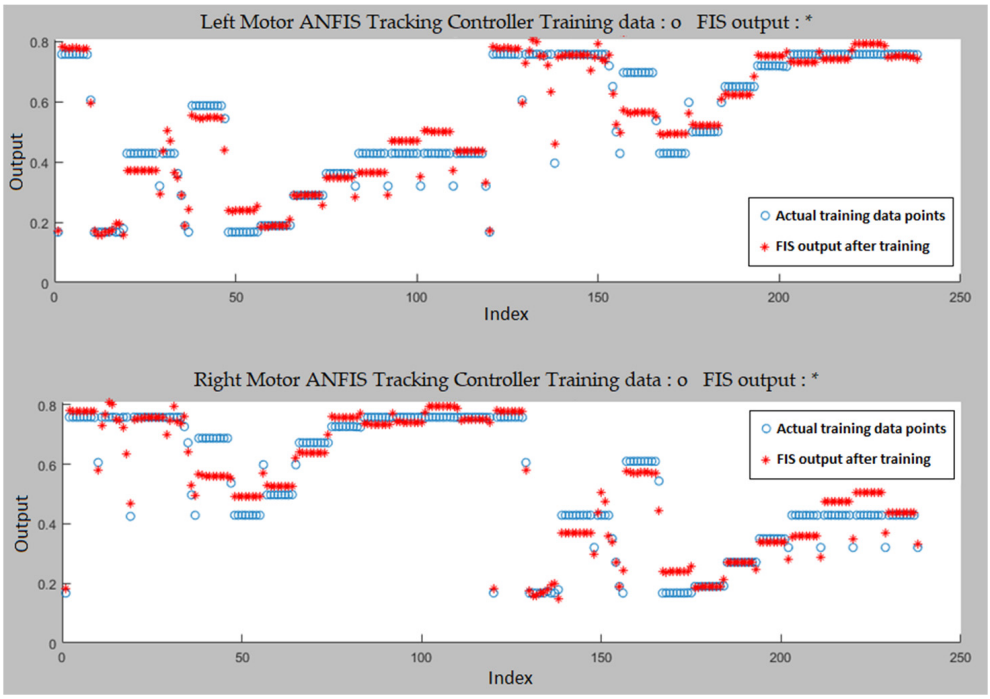

Figure 8. Data after the ANFIS is trained (left and right motor ANFIS tracking controller).

5.2. ANFIS Avoidance Controllers

The primary objective of the robot's obstacle avoidance controller is to navigate safely within its environment by avoiding collisions with obstacles. This controller is composed of two ANFIS avoidance controllers, one specifically governing the left motor and the other the right motor. Each controller takes two crucial inputs: the left sensor reading and the right sensor reading. These sensor values provide information about the proximity of obstacles on either side of the robot, with a range scaled from 0 to 1.

The output of each ANFIS avoidance controller corresponds to the velocity (m/s) of the respective motor (left and right) as shown in Figure 9. These controllers collaboratively work to dynamically adapt motor velocities, enabling the robot to navigate around obstacles and maintain a collision-free path. By processing sensor data, the ANFIS avoidance controllers are capable of making real-time decisions regarding motor speeds, ensuring effective obstacle avoidance in the robot's navigation process. This mechanism enhances the robot's safety and reliability.

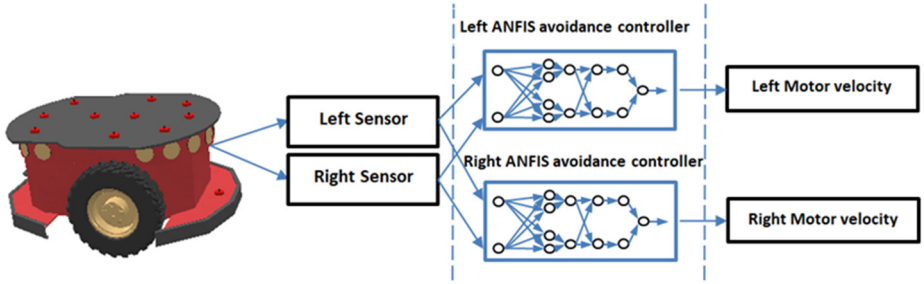

Figure 9. ANFIS Avoidance Controllers: inputs—sensor readings, outputs—motor velocities for obstacle navigation.

5.2.1. Data Set for ANFIS Avoidance Controllers

To create the dataset for training the ANFIS avoidance controllers, we first developed a fuzzy avoidance controller. Following that, we utilized this fuzzy avoidance controller to generate the dataset required for training the ANFIS avoidance controllers. The fuzzy avoidance controller has two inputs: the left sensor value and the right sensor value, as shown in Figure 9. The controller also features two outputs: the left motor velocity and the right motor velocity. For the two inputs (left and right sensors), trapezoidal membership functions are defined for the linguistic terms " very close", " close", " medium", "far", and "very far", as shown in Figure 10.

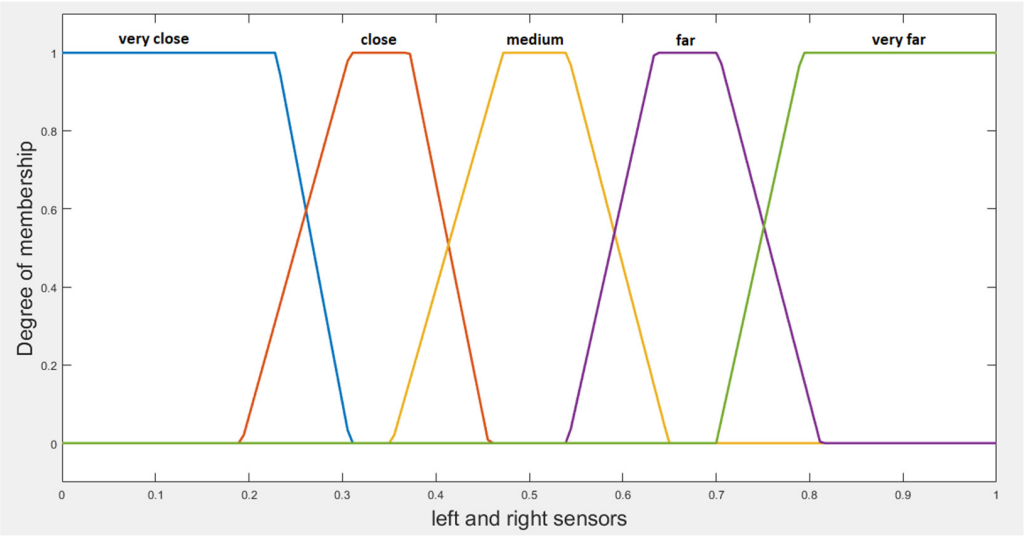

Figure 10. Fuzzy logic avoidance controller inputs (left and right sensor) fuzzy sets.

The membership functions for the outputs (left and right motor velocity) are shown in Figure 11 and are characterized by the linguistic terms 'slow' and 'fast'.

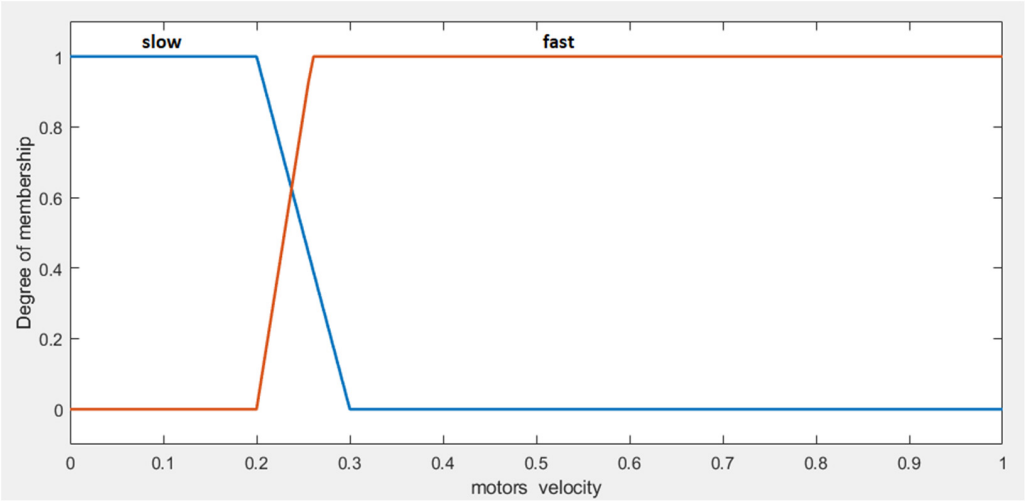

Figure 11. Fuzzy logic avoidance controller output velocity fuzzy sets (right and left motors).

Afterwards, a set of 25 expert rules shown in Table 3 is established that governs the behavior of the AGV in different situations.

Table 3. Rule set for fuzzy avoidance controller.

Rule	Left Sensor	Right Sensor	Right Motor Velocity	Left Motor Velocity
1	very close	very close	slow	slow
2	very close	close	slow	fast
3	very close	medium	slow	fast
4	very close	far	slow	fast
5	very close	very far	slow	fast
6	close	very close	fast	slow
7	close	close	slow	slow
8	close	medium	slow	fast
9	close	far	slow	fast
10	close	very far	slow	fast
11	medium	very close	fast	slow
12	medium	close	fast	slow
13	medium	medium	slow	slow
14	medium	far	slow	fast
15	medium	very far	slow	fast
16	far	very close	fast	slow
17	far	close	fast	slow
18	far	medium	fast	slow
19	far	far	fast	fast
20	far	very far	slow	fast
21	very far	very close	fast	slow
22	very far	close	fast	slow
23	very far	medium	fast	slow
24	very far	far	fast	slow
25	very far	very far	fast	fast

This process effectively forms a fuzzy avoidance controller providing the foundation for generating training data for our ANFIS avoidance controller. The training dataset follows the expert logical rules outlined in Table 3. These rules precisely define how the AGV's sensors relate to its motor velocities. By following these rules, a diverse dataset is generated encompassing a wide range of scenarios and behaviors. This systematic approach guarantees that the AGV is ready to navigate and adapt to real world situations ensuring efficient and reliable performance. A sample of training data for the ANFIS avoidance controllers is presented in Table 4.

Table 4. Sample of training data for the ANFIS avoidance controllers.

Left Sensor	Right Sensor	Expected Right Motor Velocity	Expected Left Motor Velocity
0	1	0.6186	0.1276
0.1	0.9	0.6186	0.1276
0.2	0.8	0.6186	0.1276
0.3	0.7	0.6175	0.1292
0.4	0.6	0.5377	0.1364
0.5	0.5	0.1276	0.1276
0.6	0.4	0.1363	0.5377
0.7	0.3	0.1292	0.6175
0.8	0.2	0.1276	0.6186
0.9	0.1	0.1276	0.6186
1	0	0.1276	0.6186

5.2.2. Training ANFIS for Avoidance Controllers (Left and Right)

In this study, 32 datasets are generated using the fuzzy logic avoidance controller to train the ANFIS avoidance controllers (left and right). In the proposed ANFIS model, four membership functions are established for each input (left and right sensor), characterized

by trapezoidal shapes. The output membership functions are linear. The Fuzzy Inference System is constructed employing grid partitioning, and the hybrid optimization method is applied. The model was trained over 50 epochs. The training error is calculated as the root mean square error (RMSE) between the predicted motor velocities (outputs of the ANFIS controller) and the actual motor velocities (target outputs).

Figure 12 displays the training data for the left and right motor ANFIS avoidance controller and Figure 13 depicts the training error for the left and right motor ANFIS avoidance controller.

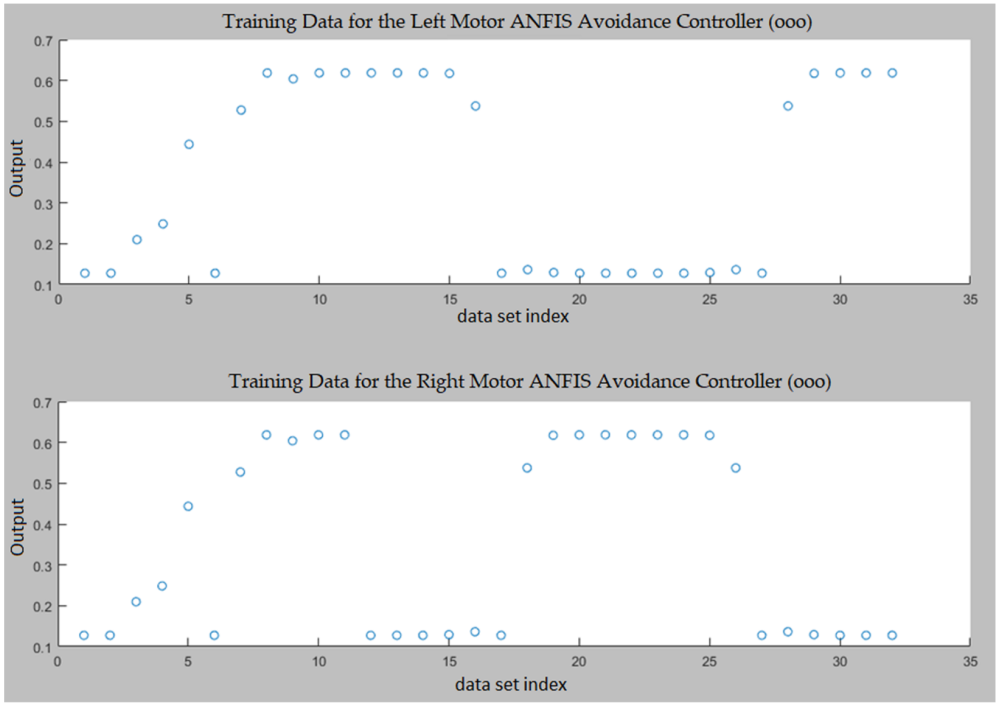

Figure 12. Training data for the left and right motor ANFIS avoidance controller.

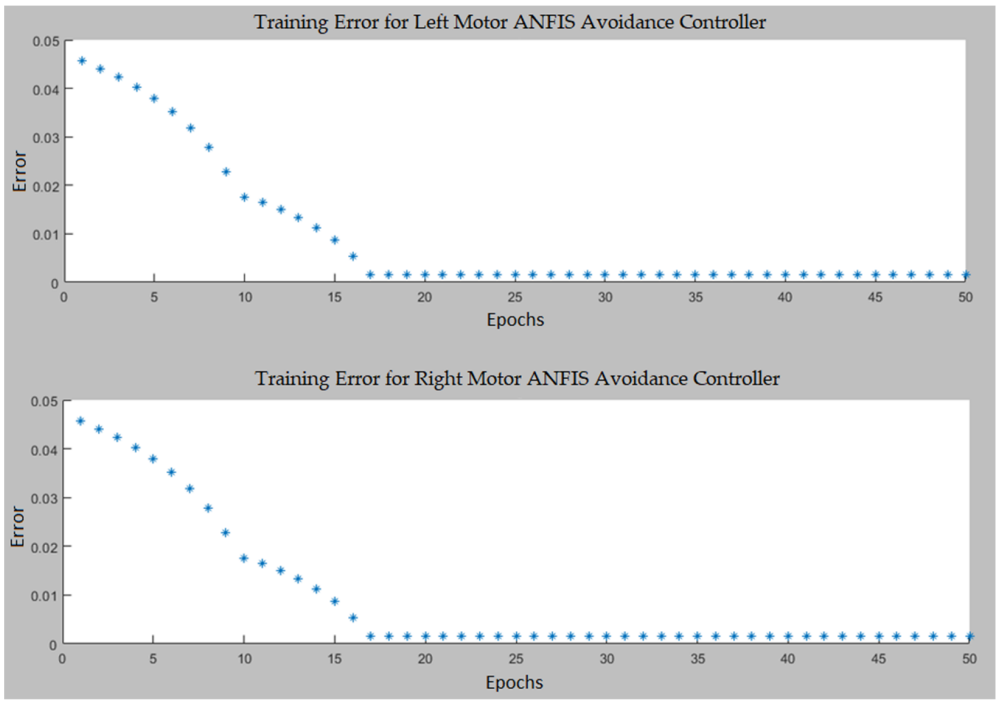

Figure 13. Training error for left and right motor velocity ANFIS avoidance controller.

Figure 14 displays the training and FIS output data for the left and right motor ANFIS avoidance controllers after the ANFIS is trained. The actual training data points are depicted as blue circles and the predicted output after ANFIS training are depicted as red asterisks. In this ANFIS model, a minimal training RMSE of 0.0017 for the left motor and 0.0017 for the right motor is achieved. The model utilizes 53 nodes with a combination of 48 linear and 32 nonlinear parameters, totaling 80 parameters. Notably, this model is built upon 16 fuzzy rules, demonstrating a notable decrease when compared to the fuzzy avoidance controller, which is composed of 25 rules (see Section 5.2.1).

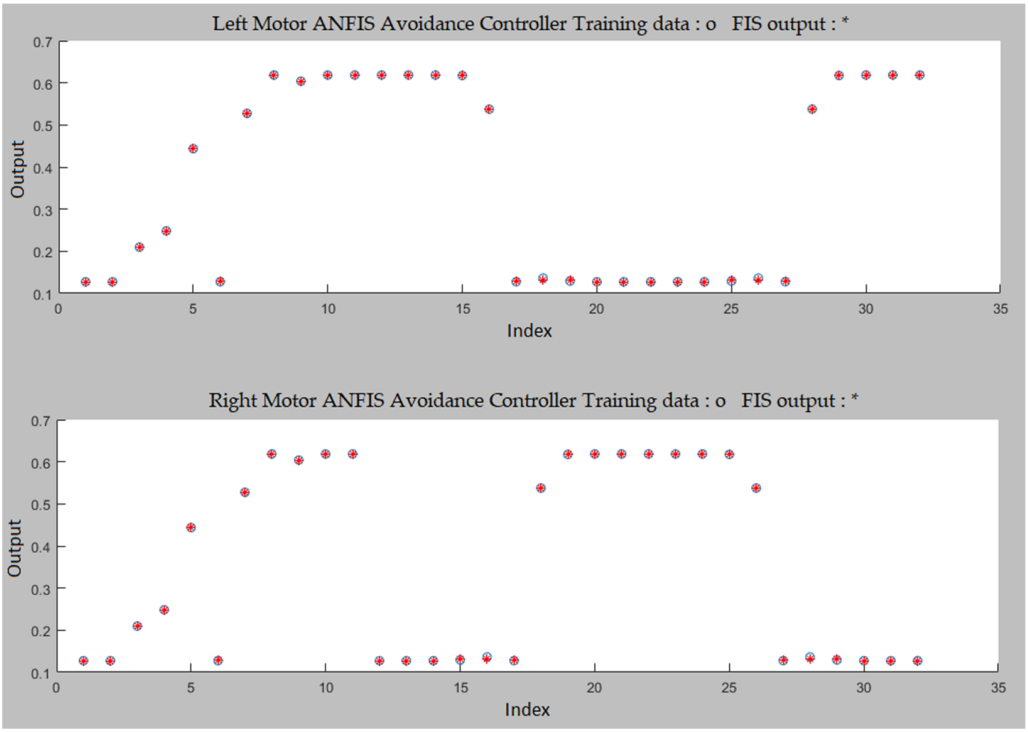

Figure 14. Data after the ANFIS is trained (left and right motor ANFIS avoidance controller).

6. Simulation Experiments

The experiments were conducted in a simulated environment configured using CoppeliaSim, a robot simulator that provided the necessary robustness and versatility for preliminary validation. The use of CoppeliaSim was crucial in overcoming the practical limitations of real-world experimentation, such as time and cost constraints, enhancing both the comprehensiveness and efficiency of the proposed control strategy.

6.1. Robot System and Sensor Arrangement

The mobile robot chosen for this research is the Pioneer p_3dx, a compact two-wheel differential drive robot, notable for its integration of 16 ultrasonic sensors.

The ultrasonic sensors are strategically positioned around the robot to ensure optimal coverage and obstacle detection. Specifically, the sensors are placed at equal angular intervals around the front half of the robot, covering an angle of 180 degrees as shown in Figure 15.

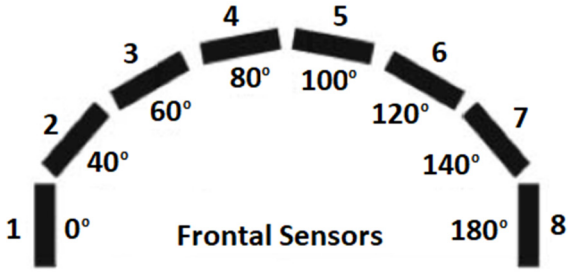

Figure 15. Frontal sensors of Pioneer p_3dx.

The maximum sensing range of the ultrasonic sensors used on the Pioneer p_3dx robot is set to 1 m. This setting standardizes the measurement scale, ensuring that all distance readings are directly comparable and appropriately scaled for input into our ANFIS controllers. In this study, an effort is made to simplify the complexity of the ANFIS obstacle avoidance controller by exclusively utilizing the six frontal sensors (sensors 2 to 7 as shown in Figure 15), organized into two sets, as shown in Figure 16.

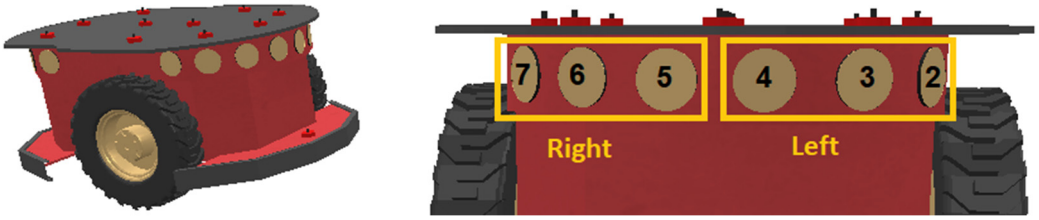

Figure 16. Left and Right Sensor.

The determination of the right and left sensor values is computed as follows:

Right Sensor Value = (Measurement from Sensor 5 + Measurement from Sensor 6 + Measurement from Sensor 7)/3

Left Sensor Value = (Measurement from Sensor 2 + Measurement from Sensor 3 + Measurement from Sensor 4)/3

The choice to calculate the left and right sensor values by averaging the readings from the three frontal sensors on each side (sensors 2, 3, 4 for the left, and sensors 5, 6, 7 for the right) is based on the following reasons: Firstly, averaging multiple sensor readings helps reduce the impact of noise and anomalies from individual sensors, resulting in more stable and reliable input values for the ANFIS obstacle avoidance controller. Secondly, by consolidating the readings into two values, the complexity of the obstacle avoidance controller is reduced, allowing for more efficient processing and decision-making within the control system. Simplifying the sensor data in this manner helps ensure that the system operates efficiently, which is particularly important in real-time applications. The effectiveness of this method is demonstrated in our experimental results presented in Section 6.2, where the robot successfully navigates the environment while avoiding obstacles.

6.2. Simulation Results

Simulations have been conducted in three environmental setups including different obstacle arrangements with varying numbers, sizes, and types (both static and dynamic) of obstacles. To test the robustness of our navigation method, 10 test cases were conducted for each environment, each featuring varied start and end points which led to different paths determined by the path-planning algorithm and obstacles that originated from different locations and moved in various directions. Due to space limitations, two representative test cases conducted from different environmental setups are presented, illustrating the robot's ability to successfully navigate to the end point, effectively handling different numbers, shapes, and sizes of both dynamic and static obstacles.

Firstly, a scene is established using CoppeliaSim, where both the start and end points are specified for the robot's navigation. Our primary objective is to guide the robot towards its designated end point while avoiding any obstacles, either static or dynamic. This scene includes five warehouse racks, four boxes in the lower left corner and a control room in the upper right corner, as shown in Figure 17.

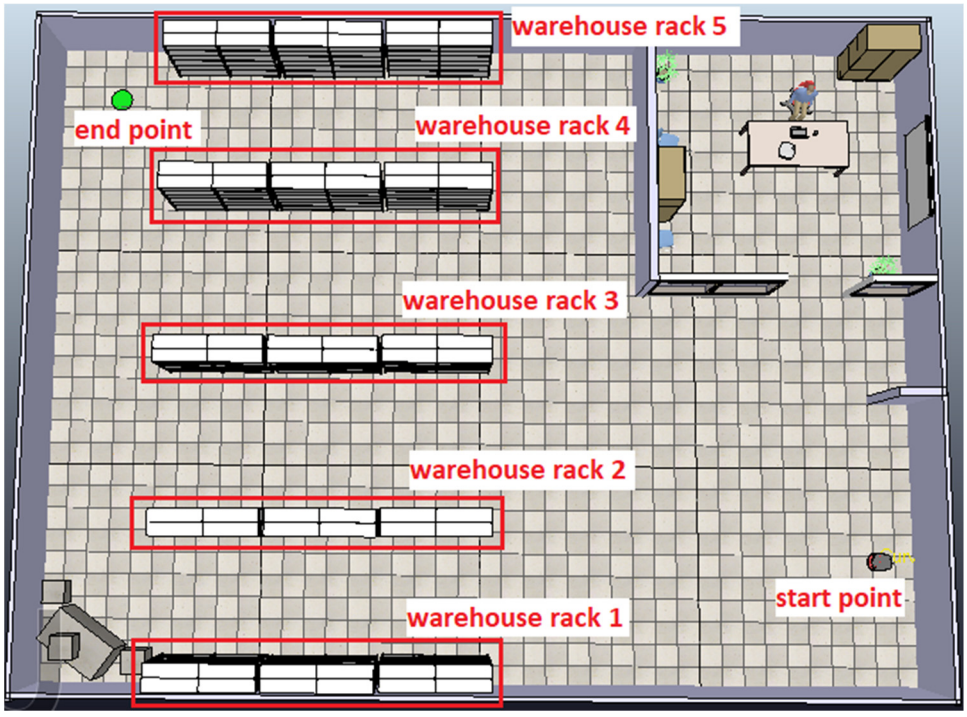

Figure 17. CoppeliaSim scene and static obstacles.

Based on this scene, the Bump-Surface concept is employed to determine the shortest and optimal path for the robot to traverse from its start position to the designated end point. The resulting optimal path, computed using a path-planning algorithm, is depicted in green color in Figure 18. This path serves as a visual representation of the route the robot is intended to follow within the simulated environment.

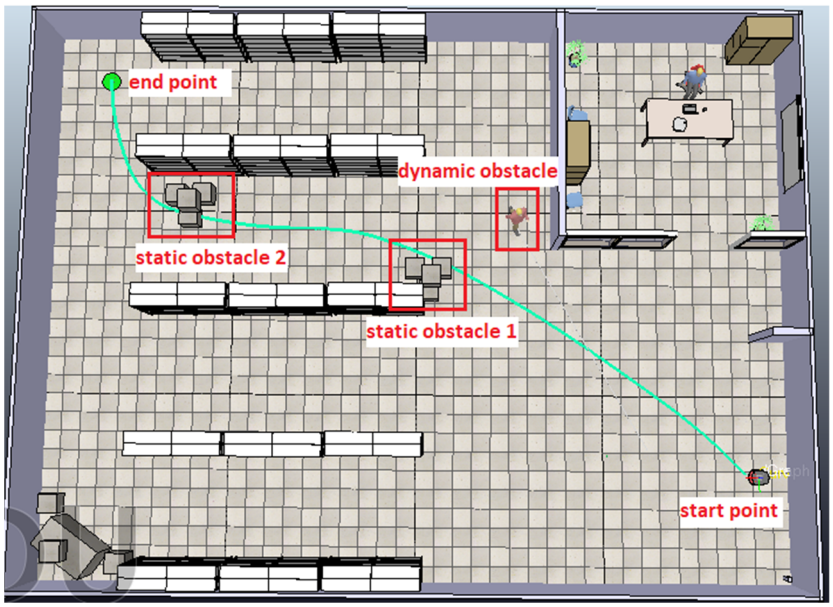

Figure 18. A scene with the optimal path, static obstacles and a dynamic obstacle.

For the robot tracking process, a vector P is formulated containing the (x, y) coordinates necessary for the robot's path to the end point. To this end, the ANFIS tracking controller is deployed. This controller facilitates the robot's movement from one point to another within vector P. As the robot reaches each point within the vector, it proceeds towards the next point. This process continues until the robot reaches the final end point represented by a green circle in Figure 18.

In the examined scenario, apart from the obstacles handled using the Bump-Surface concept to find the shortest path for the robot towards the end point while avoiding them, three additional obstacles have been incorporated into the scene. These include two static obstacles, represented by paper boxes, and one dynamic obstacle, represented by a human moving in the environment, as shown in Figure 18. These obstacles obstruct the path-planning process, requiring the robot to maneuver around them to reach the end point.

The robot navigation starts by following the predefined path using the ANFIS tracking controllers. The ANFIS tracking controller is employed to guide the robot along the predefined path (green line). When the robot senses that there is an obstacle blocking its way (within 1-m range), the obstacle avoidance controller is enabled. The robot then executes a maneuver, temporarily diverting from its pre-established path to evade a collision with the obstacle. Once the obstacle is avoided, the robot returns back to the predefined path.

The first obstacle encountered is the dynamic obstacle (human); thus prompting the activation of the ANFIS avoidance controllers. The robot then executes a maneuver to avoid the human moving close to the robot, as illustrated in Figure 19. In all Figures, the green line depicts the path generated by the path-planning algorithm, whereas the yellow line indicates the actual path taken by the robot to maneuver around obstacles and reach the end point.

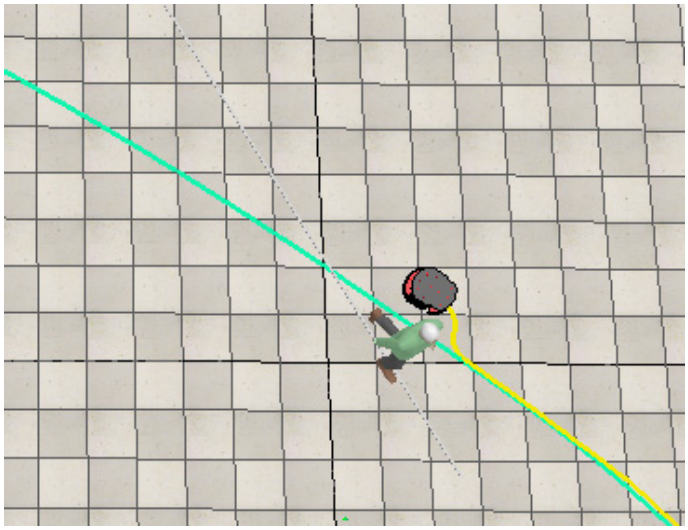

Figure 19. The robot executes a maneuver to avoid the dynamic obstacle.

Following the robot's maneuver to avoid the human, it resumes its journey along the predefined path. However, it encounters the first static obstacle, represented by paper boxes, obstructing its trajectory. Consequently, the robot executes a second maneuver to navigate around this obstacle, as depicted in Figure 20.

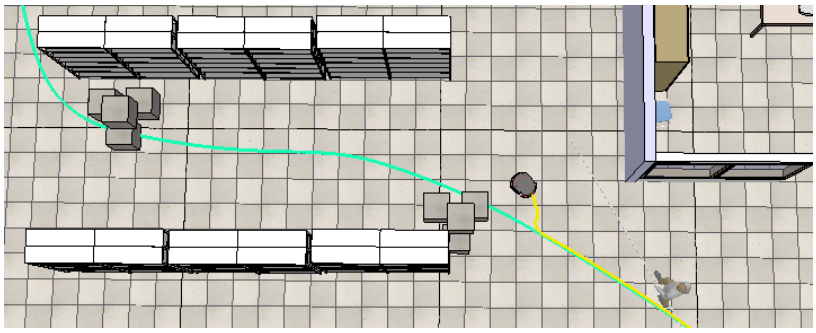

Figure 20. The robot executes a maneuver to avoid the first static obstacle.

After navigating around the first static obstacle (paper boxes), the robot encounters a second static obstacle of the same type. In response, it executes a third maneuver to avoid this obstacle, as depicted in Figure 21.

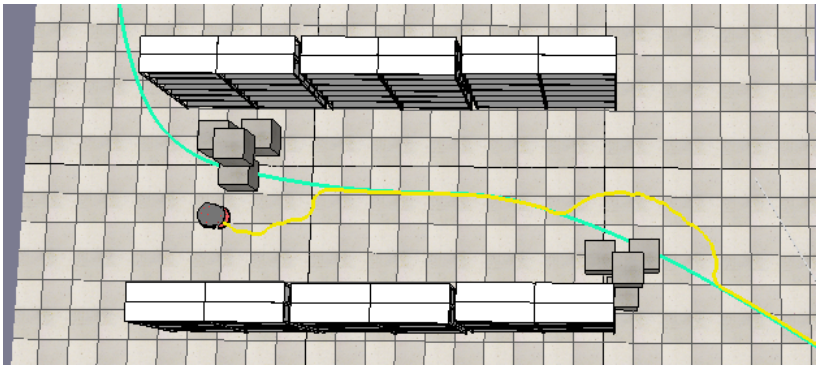

Figure 21. The robot executes a maneuver to avoid the second static obstacle.

Figure 22 illustrates the entire path that the robot follows to reach the end point, encompassing the maneuvers made to navigate around obstacles encountered along the way.

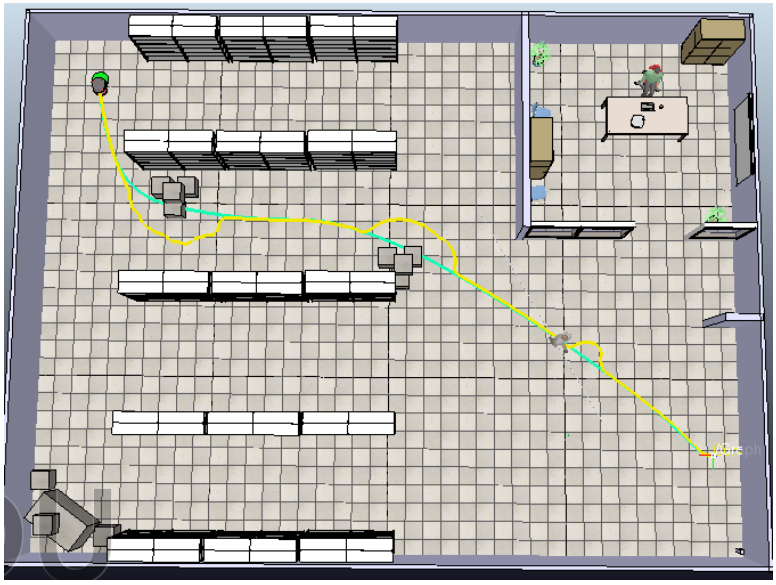

Figure 22. The robot effectively navigates by avoiding all encountered obstacles to successfully reach its designated end point.

After the simulation test is completed, the robot's performance is evaluated, revealing a completion time of 71.4 s and a total of 13 turns, as depicted in Figure 23. In this study, a

targeted approach is implemented to simplify paths and detect turns, aiming to enhance computational efficiency and the significance of our findings in robotic navigation. Specifically, a 'significant turn' in the robot's path is defined as a directional change of 0.2 radians or more. This threshold is established through extensive preliminary experimentation, which assessed the impact of various turn magnitudes on path efficiency and navigational accuracy. Our findings indicate that turns measuring under 0.2 radians generally had negligible impacts on the overall path trajectory, making them insignificant for our analysis.

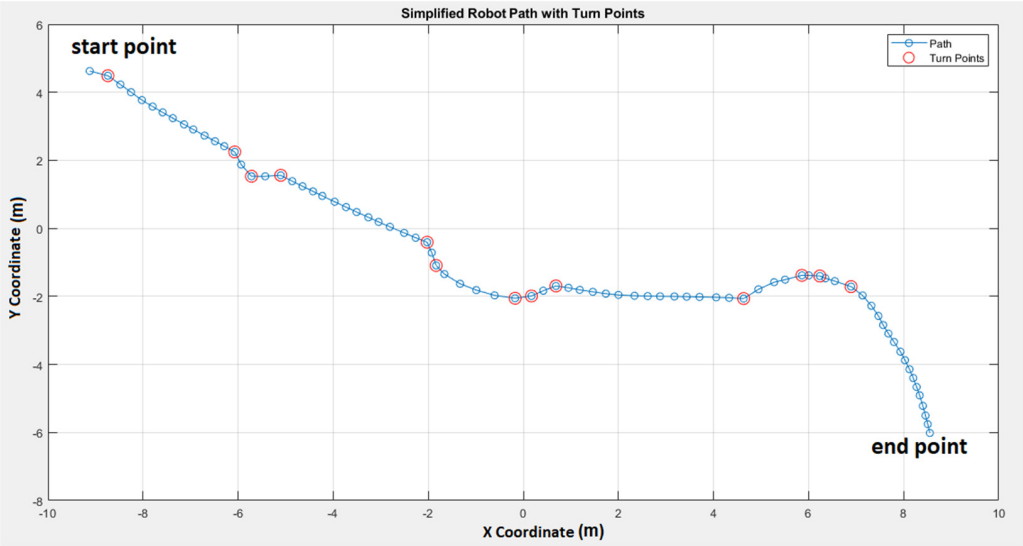

Figure 23. Number of turns using ANFIS controllers.

Aiming to compare the ANFIS strategy and the fuzzy logic controller strategy, Table 5 provides a summary of key performance indicators. These indicators include the time taken to reach the end point (in seconds), the number of turns executed, and the count of rules for each strategy: ANFIS and fuzzy logic controller.

Table 5. Performance comparison between ANFIS and fuzzy logic controller strategies.

Strategy	Time to Reach the End Point (s)	Number of Turns	Number of Rules for Tracking Controller	Number of Rules for Avoidance Controller
ANFIS	71.4	13	9	16
Fuzzy system	73.2	18	15	25

The comparison shows that the ANFIS strategy outperforms the fuzzy logic strategy in several aspects. The ANFIS strategy takes a shorter time to reach the end point with 71.4 s compared to 73.2 s for the fuzzy logic controller. Furthermore, the ANFIS strategy exhibits fewer turns, totaling only 13 compared to 18 observed in the fuzzy logic controller (Figure 24), leading to a rule reduction of 33.3%. Additionally, the ANFIS strategy employs simpler rules, with 9 rules for the tracking controller and 16 for the avoidance controller (a rule reduction of 40%), while fuzzy logic uses 15 rules for tracking and 25 for avoidance (a rule reduction of 36%). These results indicate that the ANFIS strategy provides smoother navigation paths, making it more effective for robotic navigation tasks.

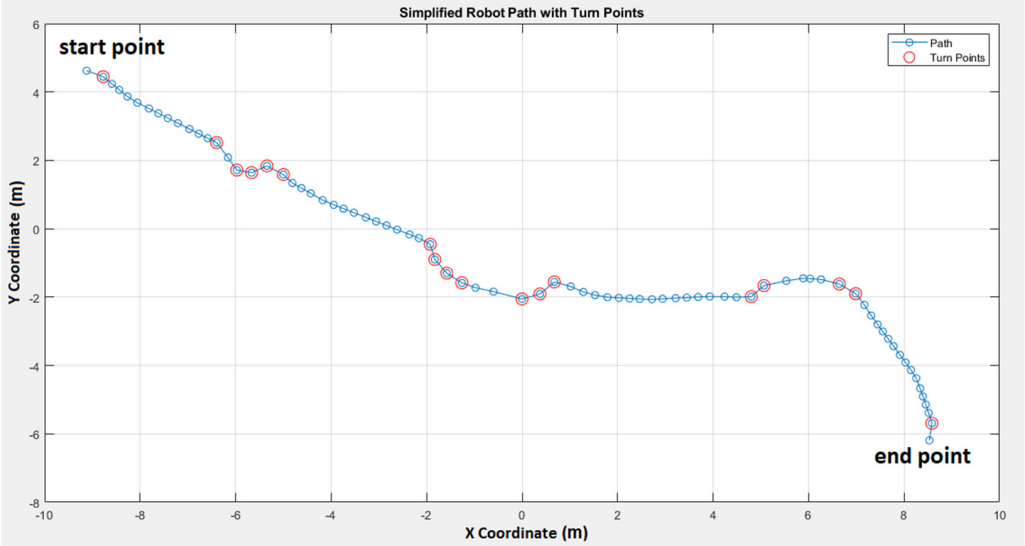

Figure 24. Number of turns using fuzzy logic controllers.

Reducing the number of rules offers several key benefits that enhance the system's overall performance and efficiency. Firstly, a smaller number of rules leads to a simpler fuzzy model, which is generally easier to understand, interpret, and manage. This simplification is crucial for maintaining the transparency of the system, particularly valuable in control applications where understanding the logic behind decision-making processes is important. Secondly, reducing the number of rules decreases the necessary computations required for decision-making, thereby lowering the overall computational load [36]. This is especially beneficial in real-time applications where decision speed is critical, allowing the ANFIS controller to operate more efficiently and respond faster in dynamic environments typically encountered by mobile robots. Lastly, models with fewer rules are easier to maintain and update, offering better scalability in practical applications, where the environment or operational conditions may evolve over time, necessitating model adjustments or updates.

In all test cases, we evaluated the human obstacle at walking speeds ranging from 0.45 to 0.8 m/s to assess the effectiveness of our algorithm. These velocities were chosen as initial benchmarks. If the environment necessitates a faster walking speed, adjustments to the robot's velocity or sensor range can be made.

To ensure the broad applicability of our method, we conducted an additional simulation with the human obstacle programmed to walk at an average speed of approximately 1.42 m/s, reflecting typical human walking speed. By extending the range of the ultrasonic sensors to 2 m, the robot was able to detect and react to the faster-moving obstacle without altering its maximum velocity.

The results of this simulation, depicted in Figure 25, demonstrated that the robot could successfully navigate the environment, avoiding all static and dynamic obstacles, including the faster-moving human, and reach the end point effectively. This further underscores the robustness and adaptability of our control strategy under varying dynamic conditions, confirming that the obstacle's velocity does not compromise the method's effectiveness and applicability.

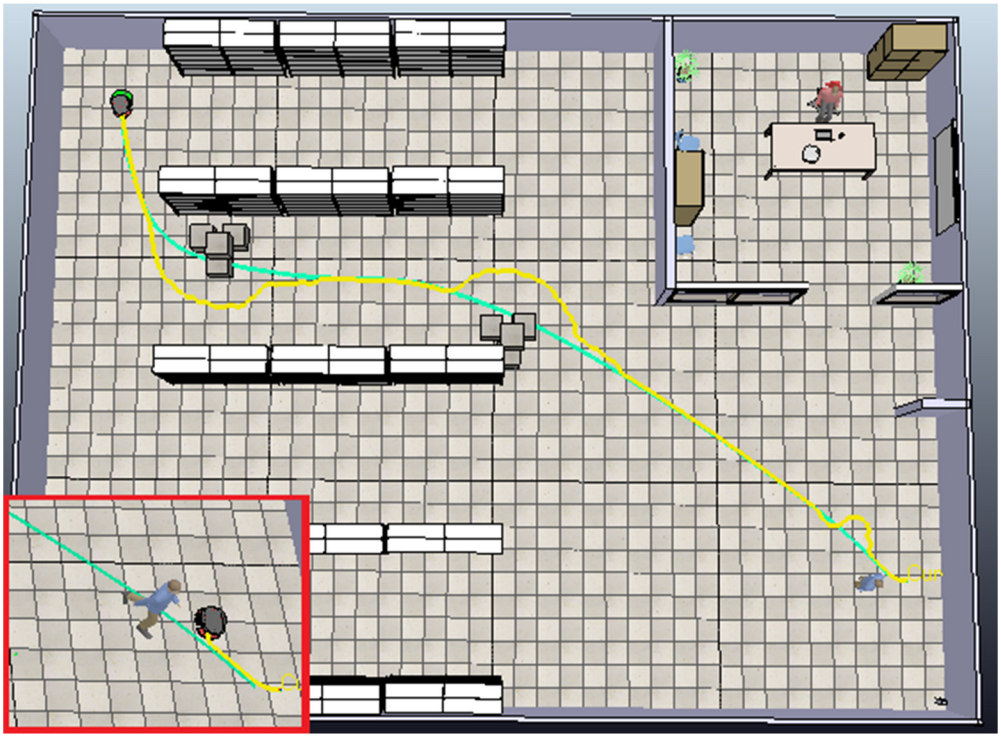

Figure 25. Simulation results showing the robot avoiding a faster-moving human obstacle with extended sensor range.

Additionally, while the moving obstacle in our tests was represented by a human worker, it could equally represent any moving entity, such as another robot. In all tests, we assume that the moving obstacle does not sense the environment, providing a consistent and challenging target for our navigation system.

Next, another scene is established, as shown in Figure 26, depicting both the provided path after using the path-planning algorithm (green line) and the actual path (yellow line). The obstacles are positioned to intersect the planned path, requiring the robot to change its direction to avoid them. The robot navigates by following the path and succeeds in reaching the end point, avoiding collisions with both static and dynamic obstacles. Figure 26 illustrates the static and two dynamic obstacles (humans) in the environment. The numbered images (1–4) on the left are zoomed-in screenshots that highlight specific points of interest in the central image, that focus on particular obstacles within the environment, showcasing key details related to the navigation paths depicted in the central layout.

Figure 26 also provides snapshots where the robot effectively maneuvers around obstacles. The robot consistently follows the planned path, adjusting its trajectory as needed to avoid collisions. It is worth noting that the success rate of the proposed system was 100% across all test cases, demonstrating full reliability for reaching the end point without collisions with static or dynamic obstacles. This success rate is critical for assessing the effectiveness of the developed navigation algorithms.

A limitation was identified which concerns sensor position and range. Obstacles located beyond the sensor's effective range were occasionally not detected early enough, resulting in potential collisions or sudden maneuvers. It mainly occurs in rare scenarios, where a moving object approaches from beyond the sensor range without altering its path. However, such scenarios are unlikely to occur in real-world applications, as humans would notice the robot and either stop or change direction, while other robots would use their sensors to avoid collisions. Despite this limitation, the robot successfully navigated around both static and dynamic obstacles to reach its end point in all tested scenarios. Although a few instances involved late detection of objects, leading to sudden maneuvers, the robot consistently managed to reach the end point successfully.

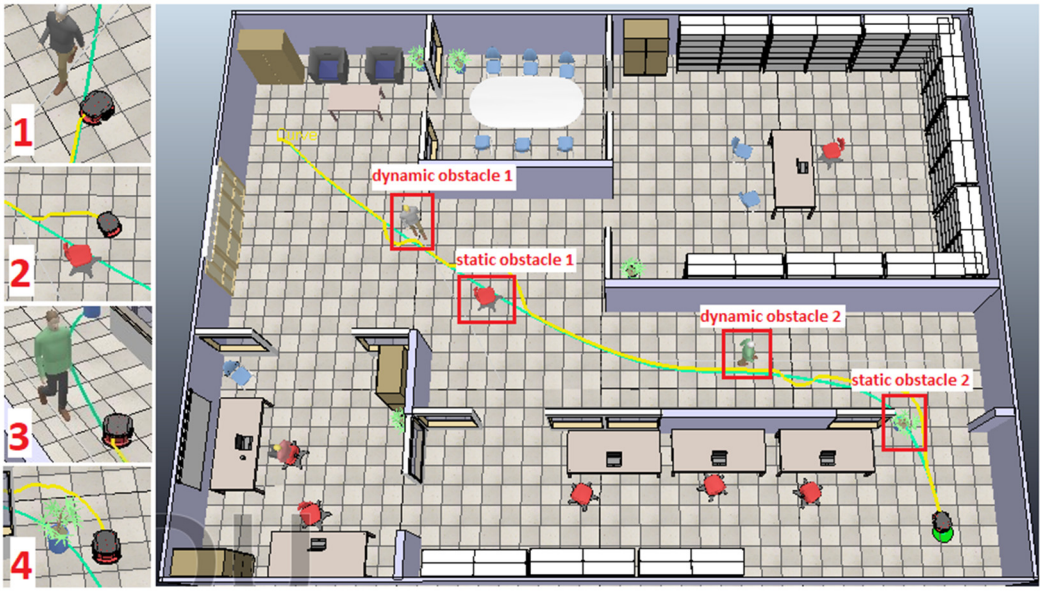

Figure 26. Another scene including static and dynamic obstacles with detailed snapshots of the robot navigating around obstacles.

To compare the CPU times, 10 test cases were executed in each of the three environments both using ANFIS controllers and fuzzy controllers for comparison. Figure 27 presents the average computation times (CPU times) for all test cases conducted within the three environments. It is clear that the average CPU time using the ANFIS strategy is lower than its counterpart using the fuzzy system strategy. Consequently, the reduction in the number of rules not only maintains the system's performance but also achieves less computational time.

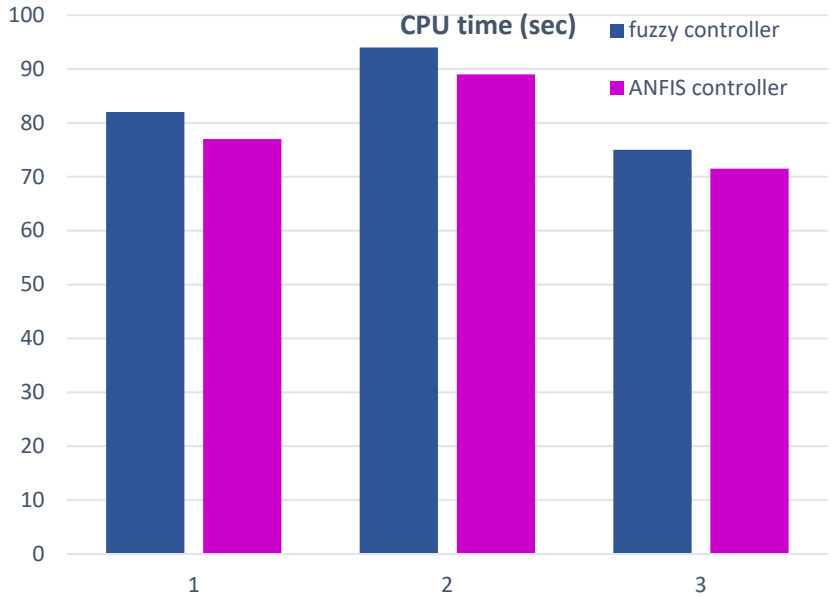

Figure 27. The average CPU times for all test cases within the three environments.

In our simulations, conducted on a Windows 10 system with an Intel(R) Core(TM) i5-7400 CPU @3.00 GHz and 8 GB of RAM, we evaluated CPU resource utilization to assess computational efficiency. The ANFIS-based method showed an average CPU load of 40% and memory usage of 1.9 GB, whereas the fuzzy logic approach exhibited a 50% CPU load and 2.1 GB memory usage. This indicates the optimized nature of our ANFIS-based method.

As a concluding remark, the primary advantage of using the ANFIS model over a fuzzy strategy model lies in its adaptive learning capabilities. ANFIS refines and adjusts the initial rules, which is particularly important as some rules may be weakly triggered, contributing minimally to the decision-making process under certain conditions. By adapting the rules, ANFIS ensures that only the most effective rules are applied, enhancing the system's overall efficiency and accuracy.

7. Conclusions

In this study, we introduce an innovative ANFIS-based control strategy, representing a notable advancement in robotics, especially concerning autonomous mobile robot navigation within dynamic environments. The core issue of our research work lies in the integration of the Adaptive Neuro-Fuzzy Inference System (ANFIS) with a path-planning algorithm. This integration establishes a new standard for realizing collision-free navigation, simultaneously enhancing both the safety and efficiency of the process.

Our method focuses on a sophisticated combination of four ANFIS controllers, each one designed to handle specific aspects of the navigation process. Two of these controllers are responsible for guiding the robot towards its end point, ensuring precise and reliable path planning. The remaining two controllers are activated in the presence of obstacles, enabling the robot to perform avoidance maneuvers without deviating from its designated path to the end point.

Our strategy was rigorously tested using the Pioneer p_3dx mobile robot equipped with 16 ultrasonic sensors in a simulation environment utilizing the CoppeliaSim platform. The simulation results demonstrated the robot's capability to navigate through an environment cluttered with both static and dynamic obstacles, utilizing ultrasonic sensors to detect and react to these challenges in real-time.

The simulation outcomes highlight the superiority of the ANFIS-based approach over the conventional fuzzy logic methods, demonstrating not only improved computational efficiency but also enhanced navigational smoothness. The robot successfully completed its paths, successfully avoiding obstacles and reaching its end point in a shorter time frame and with fewer turns compared to strategies employing conventional fuzzy logic controllers.

This study presents a novel ANFIS-based control strategy and also paves the way for future research in autonomous robot navigation. Our approach's adaptability and efficiency in dynamic environments hold vast potential for a wide array of applications. A noted limitation was the sensor range: obstacles beyond the effective range sometimes were not detected early enough, leading to collisions or abrupt maneuvers. This highlights the need for future improvements in sensor capabilities and processing power for earlier obstacle detection and smoother navigation adjustments.

Our future work will be dedicated to enhancing the robustness and intelligence of the ANFIS-based navigation framework using machine learning models. We aim to extend the system's efficacy across diverse and challenging operational scenarios, including outdoor terrains and dynamically changing environments. A significant focus will also be placed on sectors like autonomous delivery systems, search and rescue operations, and smart mobility solutions, ensuring a broader societal impact and real-world utility.

Author Contributions: Conceptualization, S.S. and P.Z.; methodology, S.S.; software, S.S.; validation, S.S.; formal analysis, S.S. and P.Z.; investigation, S.S. and P.Z.; resources, S.S.; data curation, S.S.; writing—original draft preparation, S.S. and P.Z.; writing—review and editing, S.S. and P.Z.; visualization, S.S. and P.Z.; supervision, P.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author/s.

Acknowledgments: The authors would like to thank Elias Xidias for his essential support and guidance in assisting with the path planning approach.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Pratama, P.S.; Jeong, S.K.; Park, S.S.; Kim, S.B. Moving Object Tracking and Avoidance Algorithm for Differential Driving AGV Based on Laser Measurement Technology. *Int. J. Sci. Eng.* **2013**, *4*, 11–15. [CrossRef]
2. Ito, S.; Hiratsuka, S.; Ohta, M.; Matsubara, H.; Ogawa, M. Small Imaging Depth LIDAR and DCNN-Based Localization for Automated Guided Vehicle. *Sensors* **2018**, *18*, 177. [CrossRef]
3. Rozsa, Z.; Sziranyi, T. Obstacle Prediction for Automated Guided Vehicles Based on Point Clouds Measured by a Tilted LIDAR Sensor. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 2708–2720. [CrossRef]
4. Lee, J.; Hyun, C.-H.; Park, M. A Vision-Based Automated Guided Vehicle System with Marker Recognition for Indoor Use. *Sensors* **2013**, *13*, 10052–10073. [CrossRef]
5. Miao, Z.; Zhang, X.; Huang, G. Research on Dynamic Obstacle Avoidance Path Planning Strategy of AGV. *J. Phys. Conf. Ser.* **2021**, *2006*, 12067. [CrossRef]
6. Haider, M.H.; Wang, Z.; Khan, A.A.; Ali, H.; Zheng, H.; Usman, S.; Kumar, R.; Usman Maqbool Bhutta, M.; Zhi, P. Robust mobile robot navigation in cluttered environments based on hybrid adaptive neuro-fuzzy inference and sensor fusion. *J. King Saud Univ. Comput. Inf. Sci.* **2022**, *34*, 9060–9070. [CrossRef]
7. Farahat, H.; Farid, S.; Mahmoud, O.E. Adaptive Neuro-Fuzzy control of Autonomous Ground Vehicle (AGV) based on Machine Vision. *Eng. Res. J.* **2019**, *163*, 218–233. [CrossRef]
8. Jung, K.; Lee, I.; Song, H.; Kim, J.; Kim, S. Vision Guidance System for AGV Using ANFIS. In Proceedings of the 5th International Conference on Intelligent Robotics and Applications, Montreal, QC, Canada, 3–5 October 2012; Volume I, pp. 377–385. [CrossRef]
9. Khelchandra, T.; Huang, J.; Debnath, S. Path planning of mobile robot with neuro-genetic-fuzzy technique in static environment. *Int. J. Hybrid Intell. Syst.* **2014**, *11*, 71–80. [CrossRef]
10. Faisal, M.; Hedjar, R.; Al Sulaiman, M.; Al-Mutib, K. Fuzzy Logic Navigation and Obstacle Avoidance by a Mobile Robot in an Unknown Dynamic Environment. *Int. J. Adv. Robot. Syst.* **2013**, *10*, 37. [CrossRef]
11. Anish, P.; Abhishek, K.K.; Dayal, R.P.; Patle, B.K. Autonomous mobile robot navigation between static and dynamic obstacles using multiple ANFIS architecture. *World J. Eng.* **2019**, *16*, 275–286. [CrossRef]
12. Brahim, H.; Mohammed, R.; Abdelwahab, N. An intelligent ANFIS mobile robot controller using an expertise-based guidance technique. In Proceedings of the 14th IEEE International Conference on Intelligent Systems: Theories and Applications (SITA), Casablanca, Morocco, 22–23 November 2023; pp. 1–6. [CrossRef]
13. Singh, M.K.; Parhi, D.R.; Pothal, J.K. ANFIS approach for navigation of mobile robots. In Proceedings of the IEEE International Conference on Advances in Recent Technologies in Communication and Computing, Kottayam, India, 27–28 October 2009; pp. 727–731. [CrossRef]
14. Muhammad, H.H.; Hub, A.; Abdullah, A.K.; Hao, Z.; Usman, M.B.; Shaban, U.; Pengpeng, Z.; Zhonglai, V. Autonomous mobile robot navigation using adaptive neuro fuzzy inference system. In Proceedings of the IEEE International Conference on Innovations and Development of Information Technologies and Robotics (IDITR), Chengdu, China, 27–29 May 2022; pp. 93–99. [CrossRef]
15. Malika, L.; Nacéra, B. Intelligent system for robotic navigation using ANFIS and ACOr. *Appl. Artif. Intell.* **2019**, *33*, 399–419. [CrossRef]
16. Marichal, G.; Acosta, L.; Moreno, L.; Méndez, J.A.; Rodrigo, J.; Sigut, M. Obstacle avoidance for a mobile robot: A neuro-fuzzy approach. *Fuzzy Sets Syst.* **2001**, *124*, 171–179. [CrossRef]
17. Vaidhehi, V. The role of dataset in training ANFIS system for course advisor. *Int. J. Innov. Res. Adv. Eng.* **2014**, *1*, 249–253.
18. Mishra, D.K.; Thomas, A.; Kuruvilla, J.; Kalyanasundaram, P.; Ramalingeswara Prasad, K.; Haldorai, A. Design of mobile robot navigation controller using neuro-fuzzy logic system. *Comput. Electr. Eng.* **2022**, *101*, 108044. [CrossRef]
19. Zacharia, P.T. An Adaptive Neuro-fuzzy Inference System for Robot Handling Fabrics with Curved Edges towards Sewing. *J. Intell. Robot. Syst.* **2010**, *58*, 193–209. [CrossRef]
20. Sy-Hung, B.; Soo, Y.Y. An efficient approach for line-following automated guided vehicles based on fuzzy inference mechanism. *J. Robot. Control* **2022**, *3*, 395–401. [CrossRef]
21. Kovács, S.; Kóczy, L.T. Application of an approximate fuzzy logic controller in an AGV steering system, path tracking and collision avoidance strategy. *Fuzzy Set Theory Appl. Tatra Mt. Math. Publ.* **1999**, *16*, 456–467.
22. Dudek, G.; Jenkin, M. *Computational Principles of Mobile Robotics*, 2nd ed.; Cambridge University Press: Cambridge, UK, 2010.
23. Gul, F.; Rahiman, W.; Sahal Nazli Alhady, S. A comprehensive study for robot navigation techniques. *Cogent Eng.* **2019**, *6*, 1–25. [CrossRef]
24. Juang, C.-F.; Chou, C.-Y.; Lin, C.-T. Navigation of a Fuzzy-Controlled Wheeled Robot Through the Combination of Expert Knowledge and Data-Driven Multiobjective Evolutionary Learning. *IEEE Trans. Cybern.* **2022**, *52*, 7388–7401. [CrossRef]
25. Al-Mallah, M.; Ali, M.; Al-Khawaldeh, M. Obstacles Avoidance for Mobile Robot Using Type-2 Fuzzy Logic Controller. *Robotics* **2022**, *11*, 130. [CrossRef]
26. Oliveira, L.D.; Neto, A.A. Comparative Analysis of Fuzzy Inference Systems Applications on Mobile Robot Navigation in Unknown Environments. In Proceedings of the 2023 Latin American Robotics Symposium (LARS), 2023 Brazilian Symposium on Robotics (SBR), and 2023 Workshop on Robotics in Education (WRE), Salvador, Brazil, 9–11 October 2023; pp. 325–330. [CrossRef]

27. Al-Mahturi, A.; Santoso, F.; Garratt, M.A.; Anavatti, S.G. A Novel Evolving Type-2 Fuzzy System for Controlling a Mobile Robot under Large Uncertainties. *Robotics* **2023**, *12*, 40. [CrossRef]
28. Mohd Romlay, M.R.; Mohd Ibrahim, A.; Toha, S.F.; Toha, S.F.; De Wilde, P.; Venkat, I.; Ahmad, M.S. Obstacle avoidance for a robotic navigation aid using Fuzzy Logic Controller-Optimal Reciprocal Collision Avoidance (FLC-ORCA). *Neural Comput. Appl.* **2023**, *35*, 22405–22429. [CrossRef]
29. Hong, T.S.; Nakhaeinia, D.; Karasfi, A. Application of Fuzzy Logic in Mobile Robot Navigation. In *Fuzzy Logic—Controls, Concepts, Theories and Applications*; InTechOpen: London, UK, 2012; pp. 21–36. [CrossRef]
30. Brahami, A. Virtual navigation of mobile robot in V-REP using hybrid ANFIS-PSO controller. *Control Eng. Appl. Inform.* **2024**, *26*, 25–35. [CrossRef]
31. Yousfi, N.; Bououden, S.; Fergani, B. Sliding Mode Control with Neural State Observer for enhanced trajectory tracking in mobile robots. *Sens. Mater.* **2024**, *36*, 1405–1418. [CrossRef]
32. Kowalski, P.; Nowak, J. Sensor fusion for outdoor mobile robot localization using ANFIS. *Meas. Autom. Monit.* **2023**, *68*, 116–120. [CrossRef]
33. Apriaskar, E.; Fahmizal, F.; Cahyani, I.; Mayub, A. Autonomous Mobile Robot based on Behaviour-Based Robotic using V-REP Simulator–Pioneer P3-DX Robot. *J. Robot. Eng.* **2020**, *16*, 15–22. [CrossRef]
34. Azariadis, P.; Aspragathos, N. Obstacle representation by Bump-Surfaces for optimal motion-planning. *J. Robot. Auton. Syst.* **2005**, *51*, 129–150. [CrossRef]
35. Jang, J.-S.R. ANFIS Adaptive-Network-based Fuzzy Inference System. *IEEE Trans. Syst. Man Cybern.* **1993**, *23*, 665–685. [CrossRef]
36. Hung, C.-C.; Fernandez, B. Minimizing rules of fuzzy logic system by using a systematic approach. In Proceedings of the 2nd IEEE International Conference on Fuzzy Systems, San Francisco, CA, USA, 28 March–1 April 1993; Volume 1, pp. 38–44. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.