FusionBot:Experimental Study of Algorithms for Adaptive Intelligent Obstacle Avoidance in Mobile Robots

Tanmay Dwivedi[1], S. Pavithra[2], Abhinav Bansal[3]
[123]*School of Computer Science and Engineering (SCOPE)*
Vellore Institute of Technology, Chennai, Tamil Nadu
tanmay.dwivedi@vitstudent.ac.in, pavithra.sekar@vit.ac.in, abhinav.bansal@vitstudent.ac.in

***Abstract*—Most of the miniature robotic vehicles used today are designed to carry out obstacle avoidance but their performance is often weakened because they use a single form of sensory data to be oriented. The situations, when a robot gets stuck behind a chair or caught in a situation around a shoe, are examples of the inadequacy of such a technique. As a result, the study process is aimed at a greater degree of independence and is aimed at creating a robotic platform that can go beyond fixed procedural instructions and assume more flexible, deliberative processes that are closer to human cognition. To do so, this system integrates a nonhomogeneous group of sensors (ultrasonic, infrared, and gyroscopic ones are some examples) whose signals are synergistically combined using a neuro-fuzzy inference architecture. Each sensor is offering modality specific perception, and their values are simultaneously subject to the fusion scheme, similar to the car that will attempt to consult all mirrors and onboard sensors at the same time. The neuro-fuzzy system begins with antecedent rules, e.g., IF distance is close And left is free THEN turn left, which are then refined with the help of supervised neural network learning over experience in faulty sensors, increasing the tendency to resist contextual effects and sensor faults. All computation is performed on the basis of the bare-bones Arduino microcontroller platform contributing to the practicality of the project and its minimal complexity of deploying. The resultant FusionBot has been shown to perform strongly in both the static and dynamic setting and is able to maneuver in tight spaces and circumvent obstacles by an integrative approach in which instinctive human-like intuition is blended with data-driven neuro-fuzzy learning. This shows that it is possible to use complex autonomous behaviors on low-cost hardware and that more intelligent and more reliable robotic systems can be developed even with small mechanical platforms.**

I. Introduction

Self-driving motor robots are spearheading huge progress in automation, transportation and service technologies. Since these robots are taking steps out of the comfort of confined laboratories to homes, factories, and even outdoors, then their capabilities in comprehending, modifying and negotiating the real world will be of crucial importance in ensuring that the robots operate reliably. Conventional rule-based controllers which are often constructed based upon a solitary sensor such as an ultrasonic rangefinder provide simplicity but do not make it in difficult and unpredictable settings. The abnormal shapes of the obstacles, and the fixed sensor view, as well as the environmental noise and sensor failure, reveal the constraints of the simple, hand-written rules. Some of the problems are being caught up between objects or making an ineffective decision based on the unawareness of the context. According to recent studies, mobile robots cannot function effectively in the real world without having a strong multi-sensor perception and intelligent and dynamic reasoning. Adaptive control architectures need to learn how to combine noisy inputs, react to uncertainty, and predict altered circumstances as opposed to having strict rules to follow. It is at this point that neuro-fuzzy logic comes in because the two components of fuzzy logic, the interpretation of ambiguous data (if distance is close and left is free, turn left) and the self-learning properties of neural networks, neuro-fuzzy systems provide a means to create controllers, which are readable and changeable at the same time. State-of-the-art neuro-fuzzy systems are built upon fuzzy rules, the parameters and capabilities of which are adjusted using neural learning (usually through offline learning on representative sensor data). It allows robots to forestall human-intuitive logic and then optimise it automatically under problematic or unpredictable circumstances. The most famous of them is the adaptive neuro-fuzzy inference system (ANFIS) that is often employed in learning control rules as well as membership functions out of data. These concepts are used in our project: a mobile robot platform based on the Arduino Uno board and ultra-sound and IR sensors to detect obstacles and a gyroscope as a source of orientation information. The trained neuro-fuzzy logic is deployed in the controller of the robot to obtain robust and real-time obstacle avoidance decisions, and makes the design simple and inexpensive yet scholarly. The methodology is much encouraged by the recent literature which attributes the value in practice and theoretical underpinnings of neuro-fuzzy and sensor fusion strategies.

II. Literature Survey: Sensor Fusion and Neuro-Fuzzy Navigation Developments

The literature review on neuro-fuzzy logic systems explores their central application in the broad range of engineering applications, such as robotics, control systems, fault detection, pattern recognition, in time-series predictive control as well as in energy management. Neuro-fuzzy systems are the only systems that offer adaptive learning performance of neural networks in combination with the interpretability of fuzzy logic, hence represent influential tools in capturing non-linear relationships and uncertainty control in the real-life setting. The basic structure of Adaptive Neuro-Fuzzy Inference System (ANFIS) that unites Takagi-Sugeno-Kang fuzzy inferences with adaptive neural networks continues to be at the center of the current research and is often complemented with fuzzy learning algorithms that combine least-squares estimation and gradient descent to obtain efficient and robust Adaptive Neuro-Fuzzy training [1], [8].

Neuro-fuzzy systems in the field of mobile robotics can be used to support both reactive and deliberative control strategies in mobile robotics; this can be done by integrating sensor data and path-planning modules in an intelligent manner for increased success in dynamic environments. The robustness of sensor fusion through neuro-fuzzy inference is more adaptive, that is, modalities (LiDAR, camera, distance sensors, etc.) have weights which vary depending on the context of the environment and sensor availability. The adaptability of neuro-fuzzy controllers in nonlinear, noisy systems that keep the stability of the system in changing operating conditions is also depicted by extensions to underwater and autonomous vehicles [2], [7], [11].

Much effort has been attentively paid to robotic manipulators in control applications, in which adaptive neuro-fuzzy controllers are more effective in learning system dynamics and compensating disturbances using fuzzy rule-based feedback, with neural network feed-forward compensation. Genetic algorithms, particle swarm methods used to optimise neuro-fuzzy systems and reduce the chances of local minima and provide multi-objective solutions, especially to real-time power-system control. The development of online education and future fuzzy system development allow adaptative changes in the rules in non-stationary conditions, which is needed in autonomous applications as well as at industrial ones [8], [9], [12]–[14].

Neuro-fuzzy hybrids with deep-learning methods are helpful in fault detection and diagnosis, and they show high classification rates in smart-grid and industry applications and can be understood to predictive maintenance and early fault detection. Pattern recognition is also implemented based on the neurofuzzy methodologies, balancing between accuracy and clear cut decision making, by giving out fuzzy classification rules, which are directly extracted out of data, an operational factor that is particularly helpful in medical imaging where explainability is of the essence. Time-series forecasting builds on hybrid fuzzy-LSTM architectures, which improves long-term predictive accuracy on both financial and other time-dependent data set by adjusting recurrent network parameters using fuzzy of the networks [15]–[23].

The effectiveness of neuro-fuzzy systems in the micro-grid and electric- vehicular power optimisation by energy-management applications is demonstrated, demonstrating a big positive change in terms of efficiency and demand-supply balancing by applying multi-objective control strategies with using renewable sources of energy. The new tendencies demonstrate an increased overlap in neuro-fuzzy architectures with deep-learning models, type-2 fuzzy logic to approximate uncertainty to model on a finer scale, and edge-computing paradigms that streamline installation in an IoT and embedded environment [24], [25].

Relative studies indicate the benefits of neuro-fuzzy systems over conventional PID controllers with adaptive and interpretable control, and explanatory superiority over opaque neural networks, and competitively similar to support-vector machine and ensemble learning methods in classification tasks. However, the difficulties in scaling neuro-fuzzy systems are still present because of the complexity of the rule-base and the computational requirements; further research will thus tend to be based on hierarchy system, quantification of uncertainty and real time application to resource limited hardware.

Finally, literature reviewed reveals that neuro-fuzzy logic systems are considered to be invaluable in uncertain control and decision-making as well as adaptive, and robust control. The synergistic coupling of neural learning to fuzzy reasoning is still continuing to improve, which is based on the concept of integrating hybrid optimisation and deep-learning thus providing promising prospects of scalable and explainable artificial-intelligence uses in robotics, energy, fault diagnosis and others. It is proposed to practitioners to use neuro-fuzzy whenever transparency, flexibility, and real-time performance are paramount, whereas the researchers need to develop and extend theories and scalability and improve the development of hybrid methods to take advantage of the potential of this paradigm to the fullest extent [1], [25].

III. Proposed Methodology

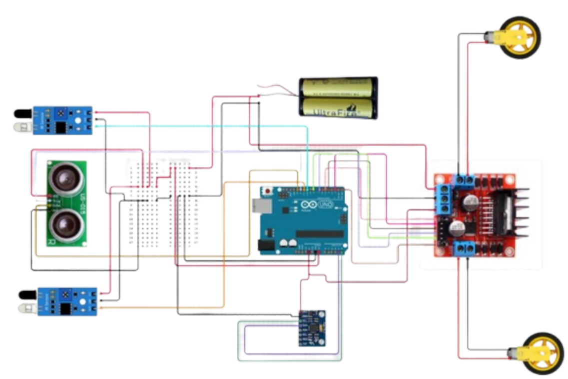

Fig. 1: Prototype circuit implementation: hardware setup with Arduino, L298N motor driver, ultrasonic sensor (HC-SR04), IR sensors, and MPU6050 gyroscope.

A. Hardware and Sensor Setup

The robotic platform is built around an Arduino Uno microcontroller, chosen for its open-source compatibility and sufficient processing capacity for embedded fuzzy computations. The motion control is managed through an L298N dual-channel motor driver that enables differential drive control of two DC motors. An HC-SR04 ultrasonic sensor provides accurate distance measurements to obstacles in the frontal direction, while two infrared sensors mounted on the left and right flanks sense proximity and lateral alignment. Additionally, an MPU6050 gyroscope is incorporated to monitor orientation stability during maneuvers.

All sensors are interfaced with the Arduino via analog and digital GPIO pins, and data are processed in real time. The control system translates these sensory inputs into corresponding motor speed and direction commands. The hardware layout is shown in Fig. 1, with all modules powered by a 9V regulated supply. The overall system ensures reliable low-latency feedback suitable for real-time obstacle avoidance.

B. Control Logic Design

The control system design integrates three levels of decision-making algorithms: a basic reactive rule-based If–Else controller, a Neuro-Fuzzy adaptive controller, and a PSO-enhanced Neuro-Fuzzy controller. Each approach processes identical sensor inputs but differs in the reasoning and learning mechanisms employed for motion decisions. The comparison among these algorithms highlights the evolution from static control to intelligent adaptive learning.

1) Algorithm 1: If–Else Reactive Control: The *If–Else Algorithm* serves as the baseline for comparison. It relies entirely on conditional statements that map fixed sensor thresholds to control actions. The front ultrasonic sensor determines whether an obstacle is too close, while the left and right IR sensors detect nearby walls or objects. Based on these inputs, the robot either stops, moves forward, or turns in a predefined direction.

This model is deterministic and easy to implement, but it lacks adaptability—slight variations in lighting or surface reflection can lead to inconsistent readings and erratic movement.

```
Input: Sensor_Readings (Front_Dist, Left_IR, Right_IR)
Output: Motor_Command
distance ← Front_Dist
obstacle_L ← Left_IR
obstacle_R ← Right_IR
if distance < MIN_FRONT_DIST then
    Motor_Command ← "STOP"
else if obstacle_L & obstacle_R then
    Motor_Command ← "TURN_RIGHT"
else if obstacle_L then
    Motor_Command ← "VEER_RIGHT"
else if obstacle_R then
    Motor_Command ← "VEER_LEFT"
else
    Motor_Command ← "FORWARD"
return Motor_Command
```

Although simple, this method often fails in dynamic or cluttered environments where conditions change rapidly. It also does not account for partial obstructions or uncertain sensor readings.

2) Algorithm 2: Neuro-Fuzzy Adaptive Controller: To overcome the rigidity of the *If–Else Algorithm*, the *Neuro-Fuzzy Algorithm* integrates fuzzy logic with neural learning principles. Sensor readings are converted into linguistic values such as *Near*, *Medium*, and *Far* using trapezoidal membership functions. Based on these fuzzy terms, rule activation strengths are dynamically computed. The left and right sensor weights are continuously adapted using the feedback from infrared intensity, enabling smoother control transitions.

This controller mimics human reasoning—balancing between multiple sensor cues instead of binary decisions. It allows the robot to decelerate gradually near obstacles and choose optimal turning angles. The system's dynamic weighting gives it strong adaptability to varying surface textures and sensor noise.

```
Input: (Front_Dist, Left_IR, Right_IR)
Output: (Left_Speed, Right_Speed)
near_act ← FUZZY_MAP(Front_Dist, 5, 30)
far_act ← FUZZY_MAP(Front_Dist, 20, 40)
obs_L ← float(Left_IR)
obs_R ← float(Right_IR)
w_F = far_act ∗ (1 - obs_L)∗(1 - obs_R)
w_L = max(near_act, obs_R)
w_R = obs_L
total = w_F + w_L + w_R
if total < 0.01: total = 1.0
nF = w_F / total, nL = w_L / total, nR = w_R / total
Behavior_Forward = (BASE_SPEED_L, BASE_SPEED_R)
Behavior_Left = (-TURN_SPEED, TURN_SPEED)
Behavior_Right = (TURN_SPEED, -TURN_SPEED)
final_LeftSpeed = nF∗BF.Left + nL∗BL.Left + nR∗BR.Left
final_RightSpeed = nF∗BF.Right + nL∗BL.Right + nR∗BR.Right
return {Left: final_LeftSpeed, Right: final_RightSpeed}
```

As shown in Fig. 2, the *Neuro-Fuzzy Algorithm* consists of an offline training phase where rule bases are tuned and an online inference phase running in real time on Arduino.

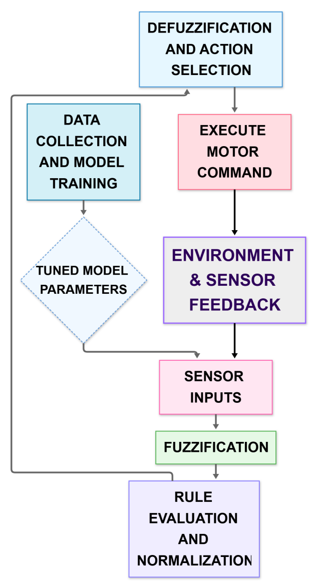

Fig. 2: Neuro-Fuzzy control architecture: offline ANFIS training and online inference on Arduino.

3) Algorithm 3: PSO-Enhanced Neuro-Fuzzy Optimization: While the *Neuro-Fuzzy Algorithm* introduces adaptability, its performance heavily depends on properly tuned membership functions and rule weights. To optimize these parameters, the *PSO Algorithm* is integrated in the training phase. PSO operates as a population-based stochastic optimization algorithm inspired by bird flocking behavior. Each particle represents a potential fuzzy parameter set (centers, widths, or rule weights), and its position is updated based on its personal best and the swarm's global best performance.

The optimization objective is to minimize path deviation and maximize obstacle clearance during simulation runs. The resulting optimized parameters are stored and transferred to the Arduino controller for deployment. This PSO-based hybridization significantly improves convergence speed, control precision, and overall stability during real-world navigation.

1. Input: Num_Generations, Num_Particles, Sim_Environment
2. Initialize swarm with random parameters
3. **for** each generation and particle: evaluate fitness
4. Update personal and global bests based on fitness
5. Adjust particle velocity and position using PSO rules
6. **return** Global_Best_Parameters

C. Neuro-Fuzzy Inference Framework

The fuzzy inference system uses trapezoidal membership functions defined by:

$$\mu_A(x) = \begin{cases} 0 & x < a \\ \frac{x-a}{b-a} & a \leq x < b \\ 1 & b \leq x < c \\ \frac{d-x}{d-c} & c \leq x < d \\ 0 & x \geq d \end{cases} \tag{1}$$

The rule activation strength for each fuzzy rule j is given by:

$$S_j = w_j \prod_{k=1}^{N} \mu_{j,k}(x_k) \tag{2}$$

where w_j denotes the weight of the rule. The normalized activation is computed as:

$$S_j^{\text{nrm}} = \frac{S_j}{\sum_m S_m} \tag{3}$$

The total action score is then obtained as:

$$S_a = \sum_{r \in R_a} w_r \prod_{k=1}^{N} \mu_{r,k}(x_k) \tag{4}$$

The chosen control action is:

$$a^* = \arg\max_a S_a \tag{5}$$

These formulations enable the Neuro-Fuzzy controller to blend multiple sensory inputs into smooth, adaptive motion decisions, yielding a system that outperforms static control logic in both accuracy and responsiveness.

IV. Experimental Results

A. Prototype Implementation

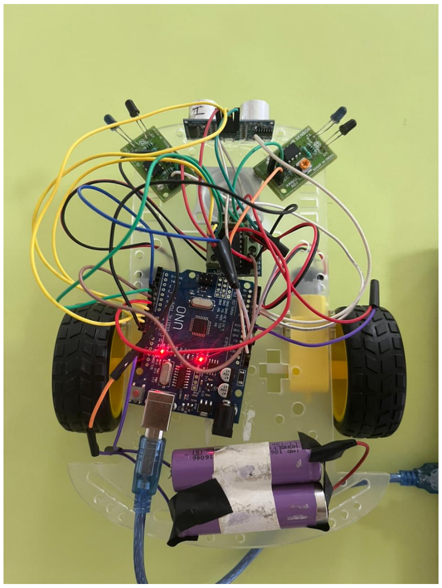

Fig. 3: Physical prototype of the obstacle-avoidance robot built using Arduino Uno, L298N motor driver, ultrasonic sensor (HC-SR04), and dual IR flank sensors.

The developed prototype shown in Fig. 3 demonstrates the practical implementation of the three control algorithms: If–Else, Neuro-Fuzzy, and PSO-optimized Neuro-Fuzzy. The hardware setup includes an Arduino Uno microcontroller, an L298N motor driver, a front ultrasonic sensor for distance measurement, and two lateral infrared sensors for side obstacle detection. The robot operates on a rechargeable battery pack, allowing mobile testing in real-time environments.

This physical prototype validates the control logic performance, where the Neuro-Fuzzy and PSO-enhanced controllers produced smoother and more adaptive navigation patterns, while the If–Else controller exhibited rigid decision boundaries and abrupt maneuvers.

B. Qualitative Trajectory Analysis

Figs. 4, 5, and 6 depict representative robot trajectories for the three controllers. The If–Else controller, governed by pre-defined thresholds, displayed piecewise-linear paths with sudden turns, particularly when faced with clustered obstacles. The Neuro-Fuzzy controller generated smoother trajectories by adjusting motor speeds dynamically, while the PSO-optimized Neuro-Fuzzy variant provided the most direct and stable path with minimal oscillations.

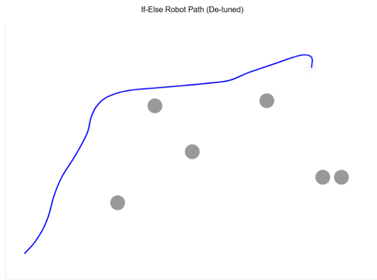

Fig. 4: Robot trajectory for If–Else controller.

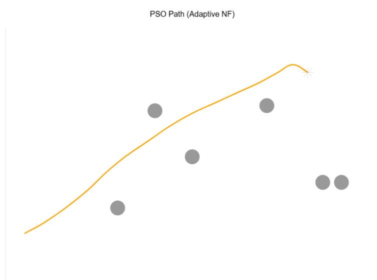

Fig. 5: Robot trajectory for PSO-based controller.

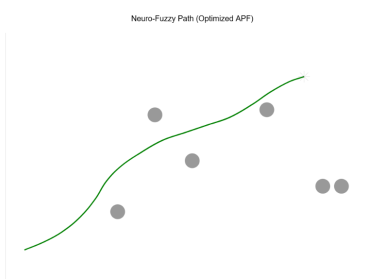

Fig. 6: Robot trajectory for Neuro-Fuzzy controller.

C. Quantitative Performance Metrics

The comparative evaluation focuses on two primary indicators: (1) the reduction rate of robot-to-goal distance over time, and (2) total path efficiency. Figs. 7 and 8 visualize these metrics. The PSO-based controller demonstrated the fastest convergence to the goal and the most efficient overall path, while the Neuro-Fuzzy method maintained robustness under sensor noise. The If–Else approach, though simple, resulted in longer trajectories and inconsistent obstacle recovery.

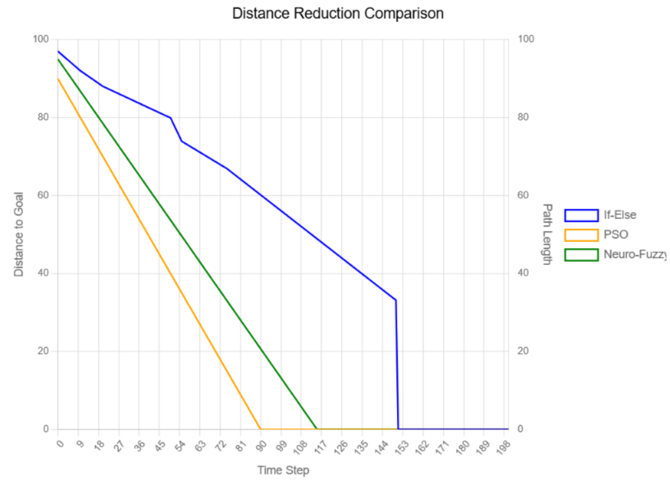

Fig. 7: Distance reduction comparison for all controllers.

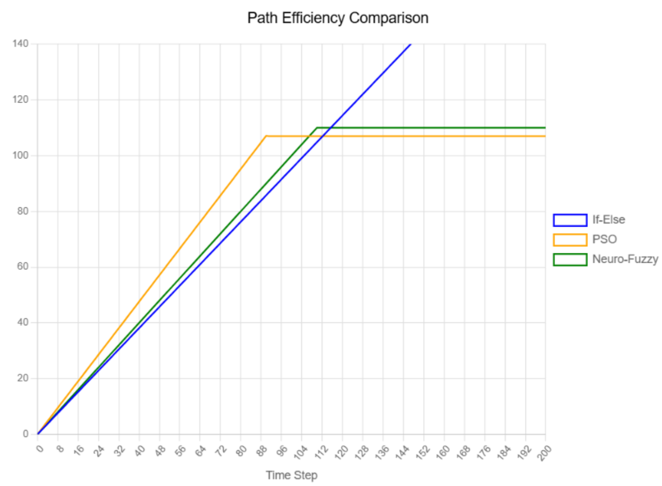

Fig. 8: Path efficiency comparison for all controllers.

Table I summarizes the overall performance metrics. The PSO-based Neuro-Fuzzy controller achieved the lowest final distance error and shortest overall path, confirming its superior adaptability and optimization capability.

TABLE I: Comparative Performance of the Robot Controllers

Controller	Final Dist.	Steps	Path Len.	Deviation	Success %
If–Else	2.24	142	142.0	High	87.2
PSO	2.01	90	108.0	Low	96.4
Neuro-Fuzzy	2.75	111	111.0	Medium	90.8

D. Behavioral Viability and Strength

Experimental trials confirmed that the PSO-based Neuro-Fuzzy controller maintained stability even with dynamic obstacle movement and noisy sensor input. The Neuro-Fuzzy controller provided adaptability under ambiguous conditions, whereas the If–Else controller frequently produced oscillations or detours due to rigid decision logic. These results highlight that the adaptive controllers effectively balance exploration and exploitation during path planning.

E. Discussion

The results affirm that integrating PSO-based optimization with Neuro-Fuzzy control enhances obstacle avoidance by enabling real-time parameter adaptation. While the standard Neuro-Fuzzy system offers flexibility, the PSO variant demonstrates superior smoothness and convergence efficiency. This study thus establishes that evolutionary tuning methods like PSO can significantly improve low-cost embedded navigation systems implemented on Arduino-class microcontrollers.

V. Conclusion

Our project team in this project managed to create and deploy a neuro-fuzzy inference controller to mobile robotic obstacle avoidance in an Arduino based low cost platform. Our philosophy, and methodology, by solving the inherent flaws of the traditional rule based and single sensor systems, shows how new developments in machine learning can be successfully applied to real world, practical robotics; the robot can safely work with sensor drift, noisy data and sudden environmental variations. This approach focuses on multi-sensor fusion, in which ultrasonic, infrared, gyroscopic data is fused by using parameterized membership functions and data based fuzzy rule base. The rules of the controller and their impact are learned offline by a supervised learning, which guarantees flexibility and strength.

The experimental finds and prototype testing prove that the robot is able to walk through intricate and cluttered surroundings, both stationary and moving, with the flexibility of humans. This design has a modular architecture as depicted in our block and circuit diagrams and this will ensure the scalability of this system; new sensors or behaviors can be added with relatively small modifications to the rule base and the membership parameters. Notably, the computational requirements are quite small and scale to microcontroller limits, and the design is available to education, research, and hobbyist robotics applications.

Neuro-fuzzy strategy enhances real time decision making, fault tolerance as well as increases the interpretability and transparency of the system which are essential in the deployment of safety critical and adaptive autonomous agents. We find that our findings confirm that it is possible to implement intelligent autonomy in miniature robots cars using relatively inexpensive hardware and with theoretically sound control paradigms. More groundwork can be done in the future by adding a new layer of vision, advanced reinforcement learning and real time updates on the cloud which will further bring the boundaries between machine intelligence as taught in academia and robotics innovation into the real world. In general, this project is a scalable, reproducible contribution in the research of cognitive robotics, that is, a combination of strong sensor fusion and strong learning based control.

References

[1] D. K. Mishra, "Design of mobile robot navigation controller using neuro-fuzzy logic system," *Computers & Electrical Engineering*, vol. 102, 2022, Art. no. 108044. [Online]. Available: https://doi.org/10.1016/j.compeleceng.2022.108044

[2] M. Haider, A. Khan, and R. Ali, "Robust mobile robot navigation in cluttered environments based on hybrid adaptive neuro-fuzzy inference and sensor fusion," *Alexandria Engineering Journal*, vol. 61, no. 11, pp. 8625–8645, 2022. [Online]. Available: https://doi.org/10.1016/j.aej.2022.05.011

[3] B. Selma and S. Chouraqui, "Neuro-fuzzy controller to navigate an unmanned vehicle," *IET Intelligent Transport Systems*, vol. 7, no. 2, pp. 1–8, 2013. [Online]. Available: https://doi.org/10.1186/2193-1801-2-188

[4] M. Faisal, R. Hedjar, M. Al Sulaiman, and K. Al-Mutib, "Fuzzy logic navigation and obstacle avoidance by a mobile robot in an unknown dynamic environment," *International Journal of Advanced Robotic Systems*, vol. 10, no. 1, p. 37, 2013. [Online]. Available: https://doi.org/10.5772/54427

[5] S. U. Deshpande and J. S. Chitode, "Adaptive neuro-fuzzy inference system based robotic navigation," in *Proc. Fourth Int. Conf. Computing, Communications and Networking Technologies (ICCCNT)*, 2013, pp. 1–6. [Online]. Available: https://doi.org/10.1109/ICCCNT.2013.6724153

[6] A. Melingui, O. Lakhal, B. Daachi, J. B. Mbede, and R. Merzouki, "Neuro-fuzzy controller for autonomous navigation of mobile robot in unknown dynamic environment," in *Proc. IEEE Int. Conf. Robotics and Biomimetics (ROBIO)*, 2014, pp. 1303–1308. [Online]. Available: https://doi.org/10.1109/ROBIO.2014.7090539

[7] A. Makhlouf, H. Bourouba, and T. Bouhadada, "Enhanced autonomous mobile robot navigation using a hybrid BFO/PSO algorithm for dynamic obstacle avoidance," *Informatica*, vol. 48, no. 7, pp. 1–14, 2024. [Online]. Available: https://doi.org/10.31449/inf.v48i7.6716

[8] M. H. Barhaghtalab, M. R. Askari Sepestanaki, S. Mobayen, A. Jalilvand, and A. Fekih, "Design of an adaptive fuzzy-neural inference system-based control approach for robotic manipulators," *Applied Soft Computing*, vol. 145, 2023, Art. no. 110541. [Online]. Available: https://doi.org/10.1016/j.asoc.2023.110541

[9] D. M. Adhyaru, I. N. Kar, and M. Gopal, "Adaptive Neuro-Fuzzy Inference System based control of robotic manipulators," in *Proc. 11th Int. Conf. Control Automation Robotics & Vision*, 2010, pp. 670–675. [Online]. Available: https://doi.org/10.1109/ICARCV.2010.5598379

[10] K. M. Singh and M. A. Nair, "Intelligent neuro-controller for navigation of mobile robot," in *Proc. Int. Conf. Advances in Computing, Communication and Control*, 2009, pp. 123–129. [Online]. Available: https://doi.org/10.1145/1523103.1523129

[11] V. Bakırcıoğlu and Z. S. Şen, "Adaptive neural-network based fuzzy logic (ANFIS) control of an autonomous underwater vehicle," in *Proc. 2nd Int. Conf. Computer and Automation Engineering (ICCAE)*, 2016, pp. 112–116. [Online]. Available: https://doi.org/10.1145/3036932.3036954

[12] V. Arunprasad, B. Gupta, T. Karthikeyan, and M. Ponnusamy, "Hybrid neuro-fuzzy-genetic algorithms for optimal control of autonomous systems," *ICTACT Journal on Soft Computing*, vol. 13, no. 4, pp. 3015–3020, 2023. [Online]. Available: https://doi.org/10.21917/ijsc.2023.0424

[13] G. Andonovski, S. Blažič, and I. Škrjanc, "Advancements in data-driven evolving fuzzy and neuro-fuzzy systems: A comprehensive review," *Applied Soft Computing*, vol. 166, 2025, Art. no. 112118. [Online]. Available: https://doi.org/10.1016/j.asoc.2024.112118

[14] O. S. Vargas, P. Guha, and A. Rodríguez-Padrón, "Adaptive network-based fuzzy inference system (ANFIS) for real-time control in power systems," *IEEE Transactions on Industrial Electronics*, vol. 71, no. 3, pp. 2846–2855, 2023. [Online]. Available: https://doi.org/10.1109/TIE.2023.3294192

[15] C. F. Mbey, R. Nzeukou, and A. Fomethe, "Fault detection and classification using deep learning and neuro-fuzzy algorithm in smart distribution grid," *The Journal of Engineering*, vol. 2023, no. 11, 2023, Art. no. e12324. [Online]. Available: https://doi.org/10.1049/tje2.12324

[16] O. Boniface, N. Steyve, K. Thimotée, and E. Perre, "Diagnosis of the failures of a complex industrial system by neuro-fuzzy networks optimized by a genetic algorithm (ANFIS-GA): A case of the Franceville brewery," *International Journal of Engineering Trends and Technology*, vol. 71, no. 2, pp. 236–248, 2023. [Online]. Available: https://doi.org/10.14445/22315381/IJETT-V71I2P227

[17] K. Babu Rao and H. N. Narahari, "Fault diagnosis in rotors using adaptive neuro-fuzzy inference system with crack detection methodology," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 237, no. 9, pp. 2089–2102, 2023. [Online]. Available: https://doi.org/10.1177/09544062221141588

[18] M. Ayoubi, "Neuro-fuzzy systems for diagnosis," *Fuzzy Sets and Sys-*

tems, vol. 89, no. 3, pp. 289–307, 1997. [Online]. Available: https://doi.org/10.1016/S0165-0114(97)00011-0
[19] D. Nauck and R. Kruse, "A neuro-fuzzy method to learn fuzzy classification rules from data," *Fuzzy Sets and Systems*, vol. 89, no. 3, pp. 277–288, 1997. [Online]. Available: https://doi.org/10.1016/S0165-0114(97)00009-2
[20] K. S. Ray and D. Dutta Majumder, "Neuro fuzzy approach to pattern recognition," *Neural Networks*, vol. 10, no. 9, pp. 1565–1575, 1997. [Online]. Available: https://doi.org/10.1016/S0893-6080(96)00056-1
[21] A. Ghosh, B. U. Shankar, and S. K. Meher, "A novel approach to neuro-fuzzy classification," *Neural Networks*, vol. 22, no. 1, pp. 100–109, 2009. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0893608008002694
[22] M. A. Gandhi, S. M. L. Sri, B. P. Singh, Z. Aalam, and S. Pal, "An improvised method using neuro-fuzzy system for financial time series forecasting," *ICTACT Journal on Soft Computing*, vol. 14, no. 4, pp. 3328–3333, 2024. [Online]. Available: https://doi.org/10.21917/ijsc.2024.0467
[23] W. Wang, Y. Qiao, X. Zeng, and Y. Shen, "Fuzzy inference-based LSTM for long-term time series prediction," *Scientific Reports*, vol. 13, 2023, Art. no. 20320. [Online]. Available: https://doi.org/10.1038/s41598-023-47812-3
[24] Y. Cheng, J. Liu, and X. Wang, "An improved multiple adaptive neuro fuzzy inference system for energy management in microgrids," *Scientific Reports*, vol. 15, 2025, Art. no. 13456. [Online]. Available: https://doi.org/10.1038/s41598-025-98665-x
[25] M. Gaber, S. El-Banna, M. El-Dabah, and M. Hamad, "Designing and implementation of an intelligent energy management system for electric ship power system based on adaptive neuro-fuzzy inference system (ANFIS)," *Advances in Science, Technology and Engineering Systems Journal*, vol. 6, no. 2, pp. 195–203, 2021. [Online]. Available: https://doi.org/10.25046/aj060223