Journal of Intelligent & Robotic Systems (2024) 110:120
https://doi.org/10.1007/s10846-024-02149-5

REVIEW PAPER

Advancements and Challenges in Mobile Robot Navigation: A Comprehensive Review of Algorithms and Potential for Self-Learning Approaches

Suaib Al Mahmud[1] · **Abdurrahman Kamarulariffin**[1] · **Azhar Mohd Ibrahim**[1] · **Ahmad Jazlan Haja Mohideen**[1]

Received: 14 July 2023 / Accepted: 11 July 2024

Abstract

Mobile robot navigation has been a very popular topic of practice among researchers since a while. With the goal of enhancing the autonomy in mobile robot navigation, numerous algorithms (traditional AI-based, swarm intelligence-based, self-learning-based) have been built and implemented independently, and also in blended manners. Nevertheless, the problem of efficient autonomous robot navigation persists in multiple degrees due to the limitation of these algorithms. The lack of knowledge on the implemented techniques and their shortcomings act as a hindrance to further development on this topic. This is why an extensive study on the previously implemented algorithms, their applicability, their weaknesses as well as their potential needs to be conducted in order to assess how to improve mobile robot navigation performance. In this review paper, a comprehensive review of mobile robot navigation algorithms has been conducted. The findings suggest that, even though the self-learning algorithms require huge amounts of training data and have the possibility of learning erroneous behavior, they possess huge potential to overcome challenges rarely addressed by the other traditional algorithms. The findings also insinuate that in the domain of machine learning-based algorithms, integration of knowledge representation with a neuro-symbolic approach has the capacity to improve the accuracy and performance of self-robot navigation training by a significant margin.

Keywords Mobile robot · Navigation · Challenges · Self-learning · Advancement

1 Introduction

Mobile robots has been a sector of great interest due to their extraordinary ability to provide assistance in numerous industrial and even personal tasks. Their effectiveness is significantly enhanced when they act autonomously and can make decisions on their own when deployed in an environment. In this process of automatically accomplishing tasks, navigation is an extremely important variable and is the precursor for task execution using mobile robots.

The application of mobile robots has undergone a remarkable expansion, integrating various aspects of daily life. Previously confined to static warehouse environments or simple straight-line movements, robots are now venturing into complex and dynamic settings, navigating through frequent changes caused by human activities, transport operations, wildlife movements, and more. From delivering goods in urban areas to responding to wildlife emergencies, these robots must interact with multiple agents, including autonomous ones [1]. As a consequence, multiple techniques for their efficient navigation have been developed and tested accordingly. These mobile robot navigation algorithms can be classified into three prime categories such as traditional AI-based algorithms which rely on predefined rules, logic and training data to guide the robot's navigation, the swarm intelligence-based algorithms which have been inspired by natural and biological substances, and finally, the self-learning based algorithms which enable the robot to learn and adapt its navigation behavior based on training data. The self-learning algorithms are being employed quite frequently to address the problem of robotic navigation systems in recent years.

✉ Azhar Mohd Ibrahim
azhar_ibrahim@iium.edu.my

1 Advanced Multi-Agent System Lab, Department of Mechatronics Engineering, International Islamic University Malaysia, Kuala Lumpur, Malaysia

Published online: 17 August 2024

Figure 1 depicts a mind map of the classification of these strategies.

Advances in navigation technologies are already helping in the improvement of mobile robots' effectiveness and in expanding their applications across various industries [2]. When focusing on mobile robot navigation techniques, two types of navigation are in play, namely global and local navigation. Both are critical for successful operation of robotic navigation systems and have seen a few overlaps in terms of applied algorithms. For global navigation, Voronoi diagram [3, 4], artificial potential field approach [5, 6], Dijkstra theory [7], visibility diagram [8], grids/framework techniques [9], and cell decomposition approach [10] have been employed in applications ranging from self-driving automobiles to warehouse robots. Their efficacies vary according to the surroundings and based on the unique requirements of the assigned tasks. Hence, researchers continue to investigate new strategies and enhancements of existing techniques in order to improve the performance of global navigation systems of robots. On the other hand, local navigation methods such as the science of fuzzy logic [11], neural networks [12], neuro-fuzzy theory [13], genetic algorithms [14], particle swarm optimization [15], ant colony optimization [16], and simulated annealing algorithms [17] have been proven useful for assuring the robots' safety, avoiding accidents, and moving effectively in crowded situations. To some extent, local navigation is comparatively a more critical component of mobile robot navigation and is frequently utilized in applications such as autonomous guided vehicles, warehouse automation, and delivery robots.

With the growing need for autonomous systems in a variety of industries, there is an increased interest in constructing differential kinematic drive mobile robots that can move and navigate independently. Among the various categories of mobile robots, wheeled mobile robots are one of the most widely used robot categories for task execution. Hence, motion control for autonomous wheeled mobile robots is an important topic of research that has seen tremendous progress in recent years. For this purpose, incorporating sensors, developing efficient controllers and augmenting the decision-making processes of mobile robots have attracted a great deal of attention, explicitly for navigation purposes. Perhaps a few examples could clarify the idea.

Hui et al. [18] improved the decision-making process of autonomous neuro-fuzzy wheeled mobile robots by combining the flexibility and interpretability of fuzzy logic and neural networks. This resulted in a more efficient and effective navigation system that allowed the robot to avoid moving impediments in its surroundings by recognizing its motion restrictions. Abadi and Khooban [19] proposed a method to improve trajectory tracking for non-holonomic wheeled mobile robots by combining optimization algorithms with fuzzy control. To accomplish steady and precise navigation on non-flat terrain, Chakraborty [20] presented a hybrid control system based on a backstepping controller and a fuzzy logic controller. By developing an error dynamic system, the backstepping controller delivers precise tracking performance whereas the fuzzy logic controller tackles the uncertainty of the terrain conditions by modifying the control input based on the robots' inclination angle and angular velocity. Wang and Yang [21] designed a controller for a non-holonomic differential drive robotic navigation system with the application of neuro-fuzzy system. Four precise infrared sensors were utilized to calculate obstacle distances

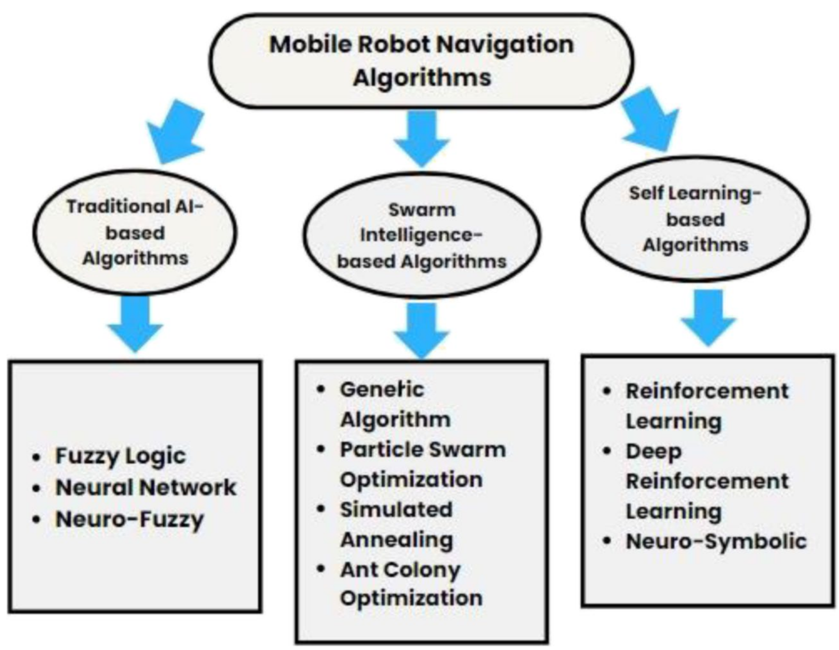

Fig. 1 Mind map of mobile robot navigation techniques

and to adjust the speed of the two motors. Type-2 fuzzy logic enables the control system to manage uncertain and ambiguous input, which is vital in mobile robotics because there might be many variables influencing robot mobility. The fuzzy logic system takes inputs from various sensors and determines the actions for optimal control based on a set of rules defined by Martinez et al. [22]. On the other hand, the training method of Al-Araji [23] used an adaptive control algorithm and a neural network which allowed it to continuously update its' parameters and improve its' performance over time. The neural network takes inputs from various sensors on the robot, such as its position, velocity, and orientation. Then it generates control signals that steer the robot towards its desired motion and orientation. Liang et al. [24] discussed the kinematics of a differential drive robotic navigation system with a double-wheeled model, which is the study of its motion and position based on the parameters of its wheels. It is apparent that the so far-seen blended multidimensional development of mobile robots hugely includes the application of computational methods based on artificial intelligence. Along with them, the evolutionary and self-learning algorithms have also made their way into the problems of mobile robot navigation and obstacle avoidance. Whereas some algorithms addressed both problems, some just focused on one of them only.

Though proved extremely effective in case to case, these approaches have limitations from multiple dimensions. Acknowledging and understanding these restrictions properly is extremely important for the further development of efficient robot navigation systems that can perform with the highest expected accuracy along with eliminating the risks and uncertainties associated with task execution using mobile robots. This article aims to conduct a comprehensive study on the available mobile robot navigation techniques in order to find out the shortcomings of these approaches and to lay a groundwork pointing to the methods that have the potential to overcome these limitations accordingly. Table 1 shows that the previously conducted surveys on mobile robot and obstacle avoidance does not incorporate the self-learning techniques with traditional AI and swarm intelligence let alone the opportunities for their integration. Thus the previous surveys left a gap in the understanding of how the combination of these algorithms would in turn enhance the achievement of objectives related to both robot navigation and obstacle avoidance, which this review paper of ours intends to fill out.

The rest of the article is organized into 5 sections. In section 2 the computational intelligence methods for robot navigation have been discussed followed by a detailed analysis of previous work on mobile robot navigation using the evolutionary algorithms and self-learning algorithms in section 3 and section 4 respectively. Section 5 provides a discussion on evaluation metrics for navigation along with implementation tools and potential research scopes. Finally, in section 6, the conclusion of the review has been presented.

2 Traditional AI-Based Algorithms

Soft computing techniques have emerged as viable methods for addressing the issues of robotic navigation systems. These techniques have the potential to improve the productivity, precision, and safety of mobile robot applications in a variety of industries including manufacturing, logistics, transportation, and disaster management. To address these domains of application, researchers have investigated numerous soft computing solutions, such as neuro-fuzzy algorithms, particle swarm optimization, evolutionary algorithms, artificial neural networks, and fuzzy logic.

Since the traditional mathematical models have been widely used in the domain of control system since the beginning of excavation on this scope of research, a question arises regarding the necessity of the introduction of soft computing replacing the traditional techniques. This is known that the traditional mathematical models and analytical algorithms are known as hard computing and are quite convenient for solving problems requiring deterministic and precise calculation and for problems those have well formulated mathematical solutions whereas the soft computing techniques are based on the notion of approximation and suitable for handling issues for which finding an exact solution is problematic. And unlike the

Table 1 Previous surveys in soft computing methods for navigation in mobile robots

Reference	Applications		Techniques (algorithms)		
	Navigation	Obstacle avoidance	Traditional AI	Swarm intelligence	Self-learning
Bijli and Kumar [25]	✓	✓	✓	N/A	N/A
Loganathan and Ahmad [26]	✓	✓	✓	✓	N/A
Liu et al. [27]	✓	✓	✓	✓	N/A
Patle et al. [28]	✓	✓	✓	✓	N/A
Naema et al. [29]	✓	✓	✓	✓	N/A
Zhu and Zhang [30]	✓	✓	N/A	N/A	✓

hard computing techniques, the soft computing approaches are flexible in terms of forbearing imprecision, uncertainty and partial truth. Furthermore, these techniques are widely popular and expected to expand for the rapid growth, cost-effectiveness and extraordinary performance capabilities of digital processor and memory chips [31]. As the nature of the mobile robot navigation research is observed, it is apparent that even though in the past the use of mobile robots were limited and restricted to static simple environments, in this era the nature has completely changed. Now, mobile robots are expected to navigate efficiently in dynamic and rapidly changing environments that might be structured, semi-structured or unstructured [32]. This requires quick decision making, rapid optimization, selective robustness and strong resiliency. The fickle nature of environments mobile robots are expected to navigate in, does not leave much scopes for the traditional hard computing techniques (mathematical, analytical, algorithmic) to be implemented. Because the precise and exact control method are unable to account for the sudden uncertainties and changes in the environment and adjust their parameters accordingly. In many cases the robot environment is prone to unpredictable changes. The traditional techniques have limitations in terms of designing intelligent robots due to these uncertainties existing specially in unstructured environments, perceptual information about the environment which are incomplete and not reliable as well as due to imprecise actuators because these techniques break down the control method in sense-plan-act paradigm [33]. This is where the specialty of the soft-computing techniques hit since they are particularly expert at handling environmental issues by offering flexibility and accounting for the unstable nature and uncertainties of environments. By offering human-like reasoning, these approaches equip the robots with the capability of learning from the environmental conditions, adapt to the sudden disturbances and optimize the task completion efficiency according to the learnt environmental patterns where applicable. Which in turns enable the robots to deal dynamic and rapidly changing environments much more efficiently than when equipped with preset rules and strict operation algorithms. Which is why these approaches are very ideal for mobile robot navigations for today's circumstances.

These approaches have been utilized to improve mobile robot control, obstacle recognition, path planning, and decision-making. Soft computing techniques are projected to be used more in robotic navigation systems in the future as they continue to demonstrate their usefulness in dealing with complex and dynamic settings. These control mechanisms are determined by their unique application and the required performance objectives. Model Predictive Control (MPC), adaptive control, the science of fuzzy logic, and the neural network control approach are examples of sophisticated control methods that can increase the accuracy, efficiency, and safety of mobile robot applications in a number of sectors.

2.1 Soft Computing Algorithms

2.1.1 Fuzzy Logic

One extensively utilized approach is fuzzy logic, which was presented by Zadeh [34], and has been shown to be quite useful in operating mobile robots. Researchers have used human-like thinking and perception to construct clever fuzzy logic controllers for differential drive robotic navigation systems in new and changing settings. Fuzzy logic controllers are designed to emulate human reasoning and decision-making processes, allowing robots to navigate in dynamic environments. Yousfi [35] for example, developed a style of fuzzy architecture for a differential drive robotic navigation system in unfamiliar settings. To control the robot's movement in the proper direction, these architectures use a set of basic behaviors such as targeting location-seekers, detecting and avoiding objects, and tracking the movement as well. These behaviors are combined and prioritized using a fuzzy logic controller to generate a final control signal.

Furthermore, sensor-based controllers work by using sensors' data to calculate the appropriate motor commands to control the robot's movement in the appropriate direction. Buragohain et al. [36] explained that hybrid approaches combining fuzzy controllers with optimization algorithms have shown promise in improving differential drive robotic navigation systems in dynamic and unique environments. By optimizing fuzzy controller parameters, these methods can enhance the robots' ability to navigate in complex and uncertain environments, leading to improved efficiency and effectiveness in a variety of applications.

Comparative studies have been carried out [37] between the choice of Mamdani and Takagi-Sugeno (TS) fuzzy models depending on the specific application requirements and performance trade-offs. While Mamdani-type fuzzy models offer interpretability and smoothness, TS models provide faster response times and more efficient control in real-time applications.

2.1.2 Neural Networks

Neural networks may be utilized for a range of tasks in mobile robot navigation including sensor fusion, environment mapping, obstacle recognition and avoidance as well as path planning. They are also effective for learning and adapting to new locations and situations which makes them beneficial for navigating in dynamic and unexpected contexts. Zou et al. [38] conducted a discussion on various neural network models, such as multilayer perceptions, recurrent neural networks, self-organizing maps, and adaptive resonance theory with their applications in mobile robot navigation and control. They also provided insights into the advantages and limitations of neural network-based approaches and suggested possible future research directions. In Xiaos' [39] work, a simulation of a mobile robot navigating through a maze-like environment showed better performance compared to traditional algorithms such as the A*. Rai and Rai [40] trained a neural network controller using the backpropagation algorithm to adjust the weights and biases to achieve the desired speed of the robot. The PID controller was also implemented to ensure stability and improve the accuracy of the system. The results of the study showed that the combination of both controllers resulted in a more efficient and accurate control system for the mobile robot. Patino and Carelli [41] designed the system to adapt to the driver's behavior and preferences and trained it using a supervised learning algorithm. The neural network was designed to take inputs from various sensors, such as a speed sensor, a steering angle sensor, and a camera sensor with the goal of producing the appropriate steering angle for the vehicle. The research of Yang and Meng [42] was based on the concept of self-organized behaviors observed in social insects such as ants and bees. The neural network model used in the approach was called the AntNet model which incorporated a combination of local pheromone trails and global pheromone trails to navigate the robot towards the goal while avoiding obstacles. The wall-following differential drive robotic navigation system presented by Nichols [43] is a great example of this approach where the robot uses a neural network to detect and follow walls in its' environment. Such robots have potential applications in industries such as manufacturing and warehousing where they can navigate autonomously and perform tasks such as material handling and transportation. However, there is still much work to be done in refining the algorithms and techniques used for neural network-based differential drive robotic navigation systems for optimizing the performance and reliability of these systems for real-world applications. The proposed system consists of a feedforward neural network that takes in sensor data from the environment and outputs control commands to the robot's motors. The network is taught via reinforcement learning which involves rewarding or punishing the robot depending on its behaviors in the environment. Motlagh et al. [44] used neural networks with reinforcement learning to teach the neural network to correlate sensor data with behaviors that result in rewards and to avoid actions that result in penalties. To solve the challenge of mobile robot navigation in unfamiliar surroundings, Gavrilova and Lee [45] devised a hybrid neural network design that combines a radial basis function neural network with a multilayer perceptron neural network. For obstacle avoidance, the radial basis function neural network was employed whereas the multilayer perceptron neural network was used for goal-seeking. To provide suitable control signals for the robot's motions, the two networks were merged in a fuzzy logic system. The suggested hybrid neural network technique was tested on a real mobile robot, and the findings showed that the robot effectively navigates across unexpected areas and avoids obstacles while arriving at its destination. Singh and Parhi [46] also used a multilayer feedforward neural network to manage the robots' steering angle.

2.1.3 Adaptive Neuro-Fuzzy Inference System

Neuro-fuzzy techniques have been put into a good amount of use in differential drive robotic navigation systems to improve the performance of traditional fuzzy controllers by combining them with neural networks. The combination of these two techniques results in a more robust and adaptive controller that can handle complex and dynamic environments. Zhu & Yang [13] proposed a model that uses 48 fuzzy rules to adjust input signals, and then a neural network is used to generate output signals. The proposed model was tested in a simulation environment and it showed promising results for obstacle detection-avoidance and target-seeking tasks. Al Mutib and Mattar [47] suggested a system with a neuro-fuzzy algorithm for differential drive robotic navigation that blends fuzzy logic with neural networks for a robust system. The system detects obstructions with ultrasonic sensors and generates an appropriate control signal. It was tested on a real mobile robot. It showed that it can successfully travel in an unfamiliar area and has the ability to avoid a recognized object when required. Godjevac and Steele [48] used a Takagi-Sugeno to map the sensory input to the necessary robot control output. For this purpose, the RBFNN was utilized, and the Takagi-Sugeno fuzzy controller was used to assure the system's resilience in unpredictable settings. Li [49] established a structure that includes four behaviors: obstacle avoidance, goal-seeking, wall-following, and wandering. The behaviors were designed based on the sciences of fuzzy logic and theories from neural networks. Joshi & Zaveri [50] developed a system that uses a similar combination to provide an optimized solution for a differential drive robotic navigation system. The system was tested successfully in simulations and real-world experiments, demonstrating its effectiveness in handling complex navigation tasks in dynamic environments. Marichal et al. [51] proposed a design that integrates a fuzzy controller, a neural network, and a real-time trajectory generator to improve obstacle avoidance and navigation accuracy. The RAM-based neuro-fuzzy method proposed by Zhang et al. [52] is a real-time adaptive learning algorithm that uses a neural network and a fuzzy approach to differentially drive robotic navigation systems in dynamic and inconsistent environments. Baturone et al. [53] proposed a system that was tested on a real mobile robot. The outcome was a differential drive robotic navigation system that can perform successfully in an indoor environment. Combining fuzzy inference systems and neural networks Ma et al. [54] mentioned that it can help mobile robots navigate in unknown environments with faster learning and decision-making capabilities. Fuzzy inference systems can handle uncertain and vague information while neural networks can learn and generalize from data. Imen et al. [55] suggested a controller that was tested on a Pioneer 3-DX mobile robot, and the findings demonstrated that it could achieve high tracking performance even in the face of system disturbances and uncertainties. This technology has the potential to increase the precision and resilience of mobile robot navigation and control, making them more trustworthy in a variety of applications. Ganapathy [56] designed two controllers for obstacle detectors and wall-following mobile robots. Zhao & Wang [57] utilized sonar sensors and neural networks for autonomous mobile robot navigation. Kumar & Dhama [58] proposed a control approach that combines fuzzy logic and neural networks for mobile robot navigation in crowded and unfamiliar environments. Their approach uses fuzzy logic to handle the uncertain and imprecise nature of sensory data and incorporates neural networks to learn and adapt to changing environmental conditions. Song et al. [59] proposed a method using fuzzy logic to determine the direction and distance of obstacles A neural network was also employed to compute the optimal velocity for the robot to navigate safely in the environment. The experimental results showed that the proposed approach significantly improved the obstacle detection and navigation performance of the robot in difficult and inconsistent environments. Lee's [60] work seems to be focused on improving the path-tracking stability for robots. This type of system combines fuzzy logic and neural networks to create a powerful control system that can adapt to changing environments and conditions. By using a recurrent system, the controller is able to remember past data and make more informed decisions about future actions. Deshpande & Bhosale [61] explored the navigation of a wheeled mobile robot utilizing an ANFIS controller. Rusu & Petriu [62] used fuzzy logic to represent the robots' knowledge of their environment and to generate motion control commands based on sensor data. The neural network was used to learn the relationships between sensor readings and control commands. The proposed controller was tested on a mobile robot navigating through a maze. It demonstrated improved performance compared to a conventional fuzzy controller. Pothal & Parhi [63] proposed a controller that utilizes inputs from sensors such as sonar, LIDAR, and camera to generate appropriate control signals for the robots to move towards the goal while avoiding obstacles. ANFIS system was used to learn the relationship between the input sensor data and the desired control signals. LIDAR was also relied upon with indoor GPS sensor by Chikurtev et al. [64] for localization and navigation of mobile robot via ROS operating system. The method was proved to be successful in terms of the mobile robot passing through narrow spaces and reaching its' goal position. For creating a depth map for mobile robot navigation, the combination of Canny detector with a three-level fuzzy model method was employed by Bobyr et al. [65]. This method for improvement of poor-quality stereo image can significantly enhance the efficiency of vision based mobile robot navigation systems. Ng & Trivedi [66] designed a neural network-integrated fuzzy controller for differential drive robotic navigation system and wall-following control. Demirli & Khoshnejad [67] developed a controller that uses a combination of ultrasonic and infrared sensors to detect obstacles and determine the distance to the parking spot. The neuro-fuzzy controller then generates the appropriate steering and speed commands to park the robot parallel to the curb. The proposed method was tested in a simulation environment and experimentally. The results showed that the robot could park autonomously with a high success rate. Al-Mayyahi [68] developed a system that detects obstructions and determines the robot's location as well as orientation using a mix of ultrasonic sensors and a wheel encoder. The ANFIS controller is then used to create control signals for the robot to travel to a desired point while avoiding obstacles in its route. Pradhan [69] created a controller based on a cooperative control method that allows many robots to traverse in unison. The suggested method is based on a mix of fuzzy logic and neural network approaches, which result in a robust and efficient solution for operating many robots in dynamic and unpredictable settings. Algabri et al. [70] taught ANFIS controllers using data from different sensors put on the robot such as the ultrasonic and infrared sensors. The suggested controllers were tested using simulations, and the findings revealed that the ANFIS-based technique outperformed existing approaches.

2.2 Hybridization of Traditional AI-Based algorithms with Soft Computing Techniques

2.2.1 Hybridization of Fuzzy Logic with Traditional AI-Based Algorithms

Algabri et al. [32] mentioned how to improve the performance of a differential drive robotic navigation system with a fuzzy logic approach by allowing the robot to adapt to changing environmental conditions which enables them to navigate more efficiently in complex and uncertain environments. By developing fundamental fuzzy logic algorithms for mobile robots, such as task-to-target, obstacle detection and avoiding, their navigation capabilities in a variety of environments can be enhanced. In another study, Hui [71] and fellow researchers combined genetic algorithms, fuzzy logic, and neural networks that can adapt to dynamic environments and enable car-like mobile robots to navigate safely and efficiently around obstacles while reaching their desired target locations. The fuzzy membership function and neural network weights as chromosomes and apply the method of genetic operators like mutation and crossover in order to generate new alternative answers. The fitness of each solution is evaluated based on its ability to minimize navigation errors. Abdessemed et al. [72] suggested a genetic algorithm that generates new candidate solutions based on their capacity to provide the best-selected path for a differential drive robotic navigation system using a population-based search. Selekwa et al. [73] created a fuzzy behavior controller that combines two fuzzy controllers, an object pathway detection controller and a goal-seeking controller. The object pathway detection controller adjusts the robot's direction to avoid impediments while the goal-seeking controller guides the robot to its destination. Furthermore, Pratihar et al. [74] developed an approach for motion planning problems for differential drive robotic navigation systems in dynamic environments by combining genetic algorithms and fuzzy logic. Babalou and Seifiours' [75] method was tested on a simulation environment and showed promising results in terms of the best-selected path planning and obstacle avoidance. The method uses a combination of a fuzzy logic controller and a particle swarm optimization algorithm to plan and optimize the robot's path. Li and Chang [76] employed fuzzy logic to create a control technique for tracking a moving object that combines sensory input. The suggested technology outperformed previous methods in terms of performance, and it may be used in applications such as surveillance, security, and transportation. Additionally, Dongshu et al. [77] created a behavior-based fuzzy logic controller to navigate differential drive robotic navigation systems in unknown inconsistent environments. Antonelli et.al [78] proposed a method that involves creating a reference trajectory for the robot and utilizing fuzzy logic to derive the proper steering angle and velocity instructions to precisely trace the path. When the suggested strategy was evaluated in both simulated and real-world scenarios, it outperformed standard control methods. Meanwhile, Ayari et al. [79] created a system that uses sensory data and decision-making algorithms to allow robots to adapt and make judgements depending on changes in the environment. The method showed promising results in allowing robots to explore complicated landscapes while avoiding obstacles and adjusting to unforeseen events.

2.2.2 Hybridization of Neural Networks with Traditional AI-Based Algorithms

The controller presented by Rossomando and Soria [80] was made up of a traditional PID controller with an adaptive neural network. To understand the system's dynamics and predict the robot's unknown parameters, the neural network was trained online using a backpropagation technique. The experimental findings revealed that the suggested controller outperformed the traditional PID controller in terms of mobile robot tracking performance. Al-Jarrah et al. [81] employed a hybrid architecture that used local and global data to design collision-free pathways for robots in a dynamic environment. The method performed well in simulation experiments and has the potential to be used in real-world applications such as surveillance and exploration. Janglova [82] demonstrated a strategy through simulations that efficiently allows mobile robots to move independently while avoiding obstacles in complicated situations. This method demonstrates the use of neural network approaches for mobile robot navigation and obstacle avoidance. It points in the right path for further study in this field. Glasius et al. [83] used a way to store a map of the surroundings in the robot system and construct a route avoiding obstacles as the robot arrives at the target point. The technique performed well in complicated situations with various barriers and could alter the trajectory in real-time. They also presented a stability controller to solve the issue of position stability, which assures that the robot stays stable while following the created trajectory. Kim [84] suggested a type-2 fuzzy neural network controller. Even in the presence of uncertainty, simulations indicated that the controller was capable of avoiding obstacles and stabilizing the robots' location. The technique was also shown to be robust to environmental changes, indicating the potential for combining fuzzy logic with neural networks to handle complex mobile robot navigation difficulties. Mahmud et al. [85] proposed a network that was trained using sensory input from a robot's onboard camera. It was then used to produce maps of the robots' surroundings. These maps were then utilized to plot the best pathways for the robot to take. The authors proved the usefulness of their technique by having a differential drive robotic navigation system to navigate through a maze-like environment using the trained neural network. Chohra et al. [86] developed a hierarchical structure consisting of three neural networks for different tasks. The first network was responsible for perception and obstacle detection using a range finder sensor, the second network was used for obstacle detection, avoidance and path planning, and the third network was responsible for the control of the vehicle's motion. Brahmi et al. [87] designed a system that combines an RNN with a global positioning system (GPS) to control a differential drive robotic navigation system. The RNN was used to learn the relationships between the robot's current position, its desired destination as well as the obstacles in its' path, and then to generate a path plan accordingly. The GPS was used to correct the position estimation errors and to ensure accurate localization. Finally, Yang [88] utilized a neural network architecture to control the torque dynamics of non-holonomic mobile robots.

3 Swarm Intelligence-Based Algorithms

Evolutionary algorithms refer to the approaches that have been derived from the behavior of organisms of natural substances and applied to the engineering domain for executing certain tasks. These techniques have been proved quite useful in terms of application in the domain of mobile robot control.

3.1 Genetic Algorithm

Genetic algorithms are popular optimization techniques used in robotics for path-planning problems. They use population-based search algorithms to generate potential paths for a robot to follow, with the goal of finding the optimal path that minimizes the cost function. Ghorbani et al. [14] suggested a genetic algorithm for generating optimum pathways for mobile robots based on a fitness function that takes into account the shortest distance between the robots' starting and finishing sites as well as obstacle avoidance. Elshamli et al. [89] created a comparable evolutionary algorithm that mitigates inconsistency detection-obstacle avoidance by modifying the pathways in real time based on sensor inputs. Tuncer and Yildirim [90] proposed a novel mutation operator called "direction-based mutation" for genetic algorithms in differential drive robotic navigation systems in inconsistent environments. The direction-based mutation operator provides a new way of generating offspring by considering the direction of the robot's movement in addition to the standard mutation operator. Ming et al. [91] described a method where a genetic algorithm is used to optimize the fuzzy controllers' membership functions and rules, which are then used to determine the robots' steering angle based on its current position and the positions of obstacles in the environment. Meanwhile, Hu et al. [92] created an algorithm that uses prior knowledge of the environment to guide the path-planning process and reduce the search space for the genetic algorithm. The effectiveness of this approach was demonstrated through simulations. In simulations, both techniques yielded promising outcomes. Liu et al. [93] proposed an approach that uses a fuzzy logic controller to generate feasible paths, which are then evaluated and optimized using a genetic algorithm to find the best path. The proposed method was validated through simulations and experimental results showing that it performed better than the existing path planning methods. Li et al. [94] proposed a dynamic mutation operator and a dynamic crossover operator to improve the performance of the genetic algorithm in dynamic environments. The proposed approach was tested and validated through simulations in both static and dynamic environments and was found to be effective in generating optimal paths for mobile robots. Qu et al. [49] developed an approach focused on optimizing the selection and crossover operations in genetic algorithms to improve the efficiency and effectiveness of path planning for multiple robots. They tested their algorithm on various scenarios and demonstrated its ability to generate optimal paths for multiple robots in dynamic environments, while Algabri et al. [70] implemented and tested the approach on a mobile robot and demonstrated better performance compared to other controllers. Castillo et al. [95] designed a system that helps to find the optimal path that is not only shorter but also smoother and easier for the robot to traverse. Arora et al. [96] presented an algorithm that aims to find an optimal path for a mobile robot to navigate through obstacles while minimizing the distance travelled. These approaches have been tested and validated through simulations.

3.2 Simulated Annealing Approach

Simulated annealing is an optimization technique that mimics the metallurgical annealing process in order to determine the optimal solution by minimizing a cost function. Miao and Tian [97] developed a heuristic function to guide the simulated annealing algorithm for searching the optimal path. The heuristic function estimated the remaining distance to the goal node from each node in the graph, which helped the algorithm to prioritize nodes that were more likely to lead to the goal node. They tested their algorithm on several scenarios including a maze as well as a room with obstacles, and found that it outperformed the Dijkstra algorithm in terms of processing time. Martinez-Alfaro and Gomez Garciz et al. [98] presented a method that searches for an optimal path between fixed obstacles while simultaneously adjusting the robots' velocity using fuzzy rules. The algorithm aims to find a collision-free path with minimum time and distance by iteratively updating the temperature of the simulated annealing process and adjusting the fuzzy logic control parameters accordingly. Zhu et al. [99] paired the Artificial Potential Field (APF) technique, which generates a virtual attractive and repulsive force field to steer the robot towards its objective while avoiding obstacles, with the Simulated Annealing Algorithm, a metaheuristic optimization process that finds the best path. The combination of these two strategies yielded promising results in terms of path planning efficiency and accuracy. Precup et al. [100] optimized the settings of a fuzzy controller for a servo system using simulated annealing and fuzzy logic. As compared to the typical PID controller, the suggested technique enhanced tracking performance and reduced the systems' steady-state error. To link the error and its rate of change to the output control signal, the optimized fuzzy controller employed a set of rules and membership functions. The settings of the membership functions were adjusted using a simulated annealing approach to reduce the difference between the planned and actual system responses. For mobile robot navigation in unfamiliar situations, Janabi-Sharifi and Vinke [101] presented a combined technique using the theory of Artificial Potential Field and the Simulated Annealing Algorithm approach. The APF approach was utilized to avoid obstacles and lead the robot to the target, while the SA algorithm was employed for global path planning. The suggested method was evaluated in both simulated and real-world situations, yielding encouraging results in collision avoidance and path planning. The adaptive simulated annealing approach developed by Tavares et al. [102] was shown to be more efficient and resilient than previous algorithms in discovering optimal pathways. It can be used for path planning in mobile robot applications. The optimal controller developed by Nakamura and Kehtarnavaz [103] was utilized to discover the best mix of rules and membership functions, while the SAA was used to optimize the scaling factors. Simulations were used to evaluate the proposed approach, and the results proved its' effectiveness in improving the accuracy and speed of the differential drive robotic navigation system. Hussein et al. [104] implemented algorithms based on several criteria such as path length, processing time, and collision avoidance. ACO was found to be more effective in terms of collision avoidance. The authors concluded that the selection of the optimization algorithm depends on the specific requirements of the application and a combination of multiple algorithms could be used to achieve better performance. A heuristic-based simulated annealing method was developed that outperformed the Dijkstra algorithm in terms of processing time for robot path planning. They proposed a simulated annealing-based intelligent navigation controller that can adaptively adjust the parameters of the controller to improve the search efficiency and quality of the optimal path. Zhang et al. [105] proposed a hybrid algorithm that combines SA and ACO to improve the navigation efficiency of mobile robots. In their work, they utilized SA to find the initial solution and then used ACO to refine the solution and find the optimal path. Synodinos and Aspragathos [106] integrated a method that uses the simulated annealing algorithm to refine the generated paths and avoid deadlocks or local minima in the artificial potential field. The method was tested in various simulated scenarios, and the results showed that the proposed method can generate optimal paths with a higher success rate compared to other traditional methods. Lastly, Zhao and Zu [107] developed a Modified Particle Swarm Optimization technique for differential drive robotic navigation systems in fluctuating and inconsistent environments.

3.3 Particle Swarm Optimization

Particle Swarm Optimization (PSO) is an optimization technique that evaluates particle performance and updates it repeatedly to find the optimal solution. It has been updated and refined throughout time and has been utilized in a variety of industrial applications including mobile robot navigation. Despite its simplicity, PSO has demonstrated promising outcomes in terms of solving complicated problems. Ahmadzadeh and Ghanavati [15] conducted a study considering each robot as a particle and the swarm as a group of robots. The PSO algorithm was used to find the optimal path for each robot by considering the position of other robots in the swarm as well as obstacles in real-life situations. Castillo et al. [22] built an optimal controller by improving the fuzzy controller's membership function. PSO was used to optimize the scaling factors of the membership functions, while ACO was utilized to optimize the membership functions' center and breadth. Zhang et al. [108] proposed an algorithm for balancing two competing objectives: distance to target and distance to barriers. The MOPSO (Modified Particle Swarm Optimization) method was proven to be successful in identifying a trade-off between the two goals when compared to other multi-objective optimization algorithms such as NSGA-II and MOEA/D, while Zhang and Li [109] presented a mobile robot's best path. Raja and Pugazhenthi [110] used the Particle Swarm Optimization (PSO) method for path planning in mobile robots with a fitness function that takes into account several criteria such as the robots' distance from the destination, the density of obstacles in the environment, and the robots' speed. Meanwhile, Masehian and Sedighizadeh [111] defined the path planning issue as a multi-objective optimization problem with the goal of decreasing the robots' distance travelled and avoiding obstacles. Such techniques highlight the utility of applying optimization algorithms for path planning in mobile robots, taking into consideration a variety of elements and objectives in order to increase navigation efficiency and safety. Wong et al. [112] created an optimal fuzzy controller based on a fitness function that takes into account the distance to the objective and the obstacles in the surroundings. The PSO algorithm was utilized to estimate the ideal wheel velocities for the robot while the fuzzy controller provides a stable and adaptable control system capable of dealing with the ambiguity and complexity of the robots' surroundings. Li et al. [76] developed an algorithm using PSO to search for the optimal path that minimized the distance travelled by the robot while avoiding obstacles. The algorithm was able to generate smooth and collision-free paths, even in complex environments. Chung et al. [113] integrated the PSO algorithm that was used to avoid dead-end conditions, while the fuzzy algorithms controlled turn angles. The system was able to navigate the robot through complex environments while avoiding obstacles and dead-ends. The modification in the PSO algorithm known as Modified Particle Swarm Optimization (MPSO) by Shiltagh and Jalal [114] which includes a mutation operator, was able to generate high-quality paths for mobile robots, even in complex and dynamic environments. To determine the best fuzzy rules for mobile robot control, Juang and Chang [115] created a fuzzy rule-based system and a PSO algorithm. When compared to standard wall-following control algorithms, their technique outperformed them. Lu [116] introduced a route planning approach that turns the robot path planning issue into an optimization problem for fitness functions. The PSO method was applied to minimize the fitness function, resulting in an optimum path that avoided obstacles while reaching the destination. PSO was also used by Allawi and Abdalla [117] to optimize the fuzzy type-2 controller that controlled the robot's motions using sensor data, allowing it to attain its goal while avoiding collisions during navigation.

3.4 Ant Colony Optimization

Ant Colony Optimization (ACO) is a metaheuristic optimization technique inspired by the behavior of ants in their search for food. Researchers have found that this method is effective in solving various problems related to mobile robot navigation and path planning, especially in environments with multiple obstacles and complex terrain. The bio-inspired ACO algorithm was first presented in 1999 by Dorigo et al. [118]. It has subsequently undergone multiple improvements. For instance, Guan-Zheng et al. [119] developed a cutting-edge global path mapping method for a differential drive robotic navigation system by combining the Dijkstra and ACO algorithms. The ACO algorithm and a fuzzy controller were utilized by Purian and Sadeghian [120] to find the best route for mobile robots in unpredictable and dynamic situations. An improved Ant Colony System (ACS) was also created by Bi et al. [121] to increase the efficiency of path-seeking in dynamic situations. Ganapathy [56] introduced an Enhanced Ant Colony Optimization (EACO) algorithm that surpassed classic ACO and other heuristic approaches in terms of navigation time and path length. Sariff & Buniyamin [122] compared the performance of ACO and GA algorithms for robot path planning in a global static environment and found that the ACO algorithm was faster in finding the optimal path. Ganganath [123] developed an offline path mapping for a non-holonomic differential drive robotic navigation system using the ACO algorithm. Juang & Hsu [124] designed the Reinforcement Ant Optimized Fuzzy Controller (RAOFC) for wheeled mobile robots' wall-following control in reinforcement learning environments. Additionally, Hsu & Juang [125] proposed a novel approach for differential drive robotic navigation systems with wall-following capability by integrating the ACO algorithm with a type-2 fuzzy controller (IT2FC). The ACO algorithm was used to determine the optimal path for the differential drive robotic navigation system, while the IT2FC was used to adjust the robot's velocity to improve its' wall-following performance. In another study, Juang [126] presented a navigation method incorporating a fuzzy controller to guide the leader robot to follow a predefined path while avoiding obstacles in its path. The follower robot then tracks the position of the leader robot using a fuzzy controller. The idea of Chen et al. [127] was to optimize a potential field function to generate an optimal path for the differential drive robotic navigation system. The algorithm was tested in a simulation environment and compared to other path-planning algorithms. The results showed that the ACO algorithm produced better paths with fewer collisions. Bacterial Foraging Optimization (BFO) is an approach that mimics the foraging behavior of bacteria. It was utilized by Hossain and Ferdous [128] to improve the control parameters of a differential drive robotic navigation system in fluctuating situations. The algorithm performed well in terms of selecting the best path and avoiding obstacles. To clarify more, the bacterial foraging algorithm developed by Liang et al. [129] is a bio-inspired optimization system that searches for optimal solutions in a problem space using chemotaxis, swarming, and reproduction mechanisms. It may be used to determine the optimum route for a robot to take from its' starting point to its destination while avoiding obstacles and reducing travel time. The Firefly Algorithm has been shown to be an alternative in robot path mapping, particularly in 2D moving and non-moving environments, and has been found to outperform other algorithms such as ACO in terms of efficiency and path length. The algorithm was inspired by the flashing patterns of fireflies as they communicate with one another to locate potential mates. This strategy is also used to optimize solutions to complex problems as mentioned by Brand and Yu [130]. The RPOA algorithm was designed by Mohajer et al. [131]. This is for local path planning which means that it focuses on finding a feasible path for the robot in the immediate vicinity of the robot, rather than planning the entire path from starting to goal position. It uses a particle swarm optimization approach to search for the best path selection while detecting-avoiding obstacles detected by the robots' sensors. The algorithm also incorporates a random element to ensure the exploration of the environment and prevent getting stuck in local optima. Fan's [132] work covered various methods of sensor fusion, including Bayesian probability theory, fuzzy logic, and artificial neural networks, and their applications in robotics, automation, and control systems.

4 Self Learning-Based Algorithms

Because of their independence, mobile robots are becoming more and more useful in a variety of industries, including manufacturing and healthcare. The capability to move around their surroundings while avoiding impediments is essential to their operation. Self-learning is one of the approaches to accomplish this [133]. Self-learning can enable robots to adapt to their environment and gain experience.

Three common self-learning approaches are available for mobile robots: reinforcement learning (RL), deep reinforcement learning (DRL) and neuro-symbolic approach. With the use of reinforcement learning, a robot's performance may be improved by rewarding it for behaving correctly according to the task given and incentivizing it to repeat it [30]. A modification of traditional reinforcement learning known as deep reinforcement learning, which uses neural networks, is an advanced technique that enhances a robot's learning capabilities and enables it to make informed decisions [134]. Furthermore, the neuro-symbolic method is another strategy that blends symbolic thinking with neural networks to provide more resilient and efficient navigation and obstacle avoidance [135]. This technique enables the robot to reason about its surroundings using symbolic reasoning and to make judgments using neural networks that learn from experiences. In Table 2, a summary of several review papers on common self-learning approaches has been listed accordingly. It is summarized that these techniques are essential for the development of mobile robots that can function autonomously and adapt to a broad range of activities and situations. By combining these techniques, mobile robots can navigate and avoid obstacles in unknown and dynamic environments which makes them suitable for a variety of applications. As mobile robots become more advanced, we can expect to see them used in a wide range of industrial activities and contributing to increased efficiency and safety in many areas of our lives.

4.1 Reinforcement Learning

From the subtopics of the machine learning field, reinforcement learning has received a lot of attention recently. It is built based on trial-and-error learning, in which the robot begins with a blank slate and interacts with its surroundings through actions. The robot learns to correlate certain behaviors with specific rewards over time and adapts its behavior appropriately, making itself incredibly versatile and adaptive. Nevertheless, there are certain hurdles in the way of employing reinforcement learning in

Table 2 Common self-learning approaches in robot navigation problems

Approach	Architecture Design Sampled	Function	Model Properties	Applications	Example References
Reinforcement Learning Deep Reinforcement Learning	Q-learning and DQN are machine learning algorithms applied for reinforcement learning, while DDPG is for continuous control problems.	Sequential decision-making problems	RL and DRL are useful for modeling complex environments and intelligent agents	To train an agent how to make decisions in an environment based on feedback in the form of incentives for path planning, use trial-and-error learning	[134, 136, 137] [30, 131, 138–140]
Neuro-Symbolic	Perception Module, Convolutional Neural Network (CNN), Symbolic Reasoning Module, Neural-Symbolic Integration Layer and Motor Control Module	Spatial and Sequential data modelling	Ideal for visual missions where convolution layers demand significant training and massive calculations	Identifying and classifying barriers and characteristics. Moreover, activity recognition and mobility forecasting	[135, 141–146]

robotics applications such as the requirement for a large volume of data to efficiently train the robot and establishing a reward system that appropriately represents the robot's expected performance. A few works have been conducted on this lately. Fatemeh et al. [147] used supervised learning and fuzzy reinforcement learning to train a controller for obstacle avoidance. When compared to traditional approaches, the findings revealed considerable gains in terms of learning time and failure rate. This method is especially useful when the environment is complicated and difficult to precisely model. Fuzzy reinforcement learning enables the agent to adjust its behavior to changing environments in real-time, making it more efficient and resilient. Similarly, Shiying Sun et al. [148] showed how reinforcement learning might help robots interact with people in a more natural and intuitive way. Robots can change their behavior to better match that of people by learning from human behavior, resulting in more efficient and successful human-robot interactions. This concept has uses outside human-robot interaction too such as in autonomous driving where the robot may learn from the actions of other drivers in order to enhance its' own driving. To increase the generalization power of reinforcement learning models in mobile robot navigation, Bruce et al. [149] presented an interactive replay buffer approach. In a simulated robot navigation test, the technology displayed increased performance and faster convergence rates when compared to existing methods. This method considerably enhanced performance in both training and validation settings, and was critical for achieving dependable zero-shot transfer to modifications in the environment that were not encountered during training [150]. Although the promise of reinforcement learning is obvious, putting this theory into practice can be difficult, particularly in real-world settings with dynamic surroundings. Yet, the research discussed above shows that reinforcement learning is a potential strategy for constructing mobile robots that can navigate in congested settings while avoiding obstacles. The previous studies have shed light on the potential of reinforcement learning for robotics applications and set the way for future study in this field.

4.2 Deep Reinforcement Learning

Embracing the science of deep learning techniques employed in this approach to represent the robots' surroundings and make choices, whilst reinforcement learning allows the robot to learn from experience by doing tasks and obtaining rewards or punishments [134]. This method has yielded encouraging results in a variety of applications including autonomous driving and robot navigation in difficult situations. Deep reinforcement learning can assist robots in adapting and learning from their environment, allowing them to be more efficient and successful in accomplishing tasks. Unfortunately, this strategy necessitates the collection of vast volume of data, which may be time-consuming and costly. Yet, deep reinforcement learnings' potential benefits make it a desirable field of research for mobile robot navigation and control. Evidenced can be found in the work of P. Zielinski and U. Markowska-Kaczmar [151] where artificial intelligence has been utilized by Autonomous Underwater Vehicles (AUVs). Their research created a 3D robotic navigation system that effectively guided an AUV between two predefined places in a 3D environment using both deep reinforcement learning and vision-based methodologies. The authors investigated numerous methods of visual feature extraction and demonstrated how they affected the models' training and overall performance. This work demonstrates the neural-symbolic approach's potential for improving the navigation skills of autonomous underwater vehicles. Pararth Shah et al. [152] employed solely visual and depth information to follow natural language commands. Deep reinforcement learning was used in this work to teach the robot to follow directions and detect landmarks. The results showed that the robot can generalize these instructions and travel to the destination successfully. This technique demonstrates the neuro-symbolic approach's potential for developing robots that can understand and follow natural language instructions. Deep reinforcement learning in robotics has opened up new avenues for constructing autonomous robots in dynamic and complicated situations. Hartmut Surmann et al. [153] developed a sensor data fusion technique with RGB-D cameras to allow robots to travel in actual surroundings with true 3D obstacle avoidance. This method reduces the need to tailor the environment to the robots' sensory capabilities, making it more adaptive to complex and changing situations. The authors proved the efficacy of their method in a real-world situation in which a robot navigated through a congested area while avoiding obstacles in real time. The potential of deep reinforcement learning in mobile robot navigation and obstacle avoidance has been demonstrated by several studies including one where the authors evaluated their approach on a simulated mobile robot navigating in a cluttered environment [30]. The study showed that the robot successfully avoided obstacles and reached its goal using a policy learned through deep reinforcement learning. These findings highlight the effectiveness of deep reinforcement learning in enabling mobile robots to navigate in cluttered environments while avoiding obstacles. However, it is important to note that deep reinforcement learning can be computationally intensive and may require large amounts of data to train effectively. Despite these challenges, the potential of deep reinforcement learning in robotics applications, particularly in mobile robot navigation and obstacle avoidance, is promising.

4.3 Liquid Neural Networks

A very recent research led to discovery of a new kind of neural network, namely Liquid Neural Network (LNN) that is able to learn from the continuous streams of data in an environment while deployed to execute a task instead of learning only at the training phase. The nodes/neurons of the LNN communicate with each other with dynamic synapses which enables it to develop a dynamic architecture to generate diverse responses to various data inputs equipping the network with robust adaptability and flexibility.

Even though LNN is more compatible in processing continuous sequential data handling, hence is more suitable for pattern recognition, data processing and forecasting of data, they can indeed be employed for self-learning due to the significant advantage and leverage in computing it can provide. However, the LNN is bound by some limitations. Firstly, when trained with gradient descent, LNN may suffer from the vanishing gradient problem of liquid time constant on which LNN is dependent. This prevents neural networks to be able to reach their optimum weights and might lead LNN to inefficiency in terms of learning long time dependencies [154–156]. However, the vanishing gradient problem might be tackled via appropriate selection of activation functions in the hidden layer and by using appropriate gradient descent optimization algorithms [157]. Secondly, the parameter tuning in LNN is critical since it involves choice of ordinary differential equation solver along with network architecture and regularization parameters[155]. Coming up with the combination of suitable parameters is time consuming and if not done properly can lead to suboptimal performance. Parallel computing might be useful to rectify this issue of computational expense in terms of time. As mentioned in literature, in application domains like control of robots in continuous time observation and action space casual structures like liquid time constants can help improve reasoning [155]. Should Neuro-Symbolic techniques be combined with LNN to bolster the reasoning capability of time continuous systems and subsidize the sub-optimal performance due to gradient descent, the results might turn out very promising.

4.4 Neuro-Symbolic Approach

The neuro-symbolic approach to artificial intelligence combines symbolic thinking with artificial neural networks. With neural networks being effective at pattern recognition and symbolic reasoning being good at logical inference, this method combines their respective strengths [158]. To address the issue of global localization, Coraggio and Gregorio [159] created a neuro-symbolic method that makes use of landmarks. They were able to utilize the characteristics of each technique by merging symbolic thinking with artificial neural networks. This strategy significantly reduced computation time for both the robot decision and vision systems. They also employed a virtual neural sensor to assess a piece of the input image, which sped up the visual processes. This novel method has shown promising results in terms of boosting the efficiency of robot navigation and obstacle avoidance. An automatic method for collecting image data for neuro-symbolic task planning for robot navigation was presented by Rakhman [135] et al. Traditional methods of collecting images for robot task planning can be laborious, requiring the operator to manually operate the robot, adjust the environment, and capture high-quality images repeatedly. Moon [153] designed a policy that can translate sensor data into actions. The researchers used a hybrid technique using neural networks and symbolic reasoning. They tried their method on a simulated mobile robot moving through a congested environment and discovered that the robot effectively avoided obstacles and arrived at its destination. These findings indicate the neuro-symbolic approaches' promise for the future of robotics in navigation systems [160]. This approach should be emphasized. However, because of its complexity, efficiently executing it may need substantial

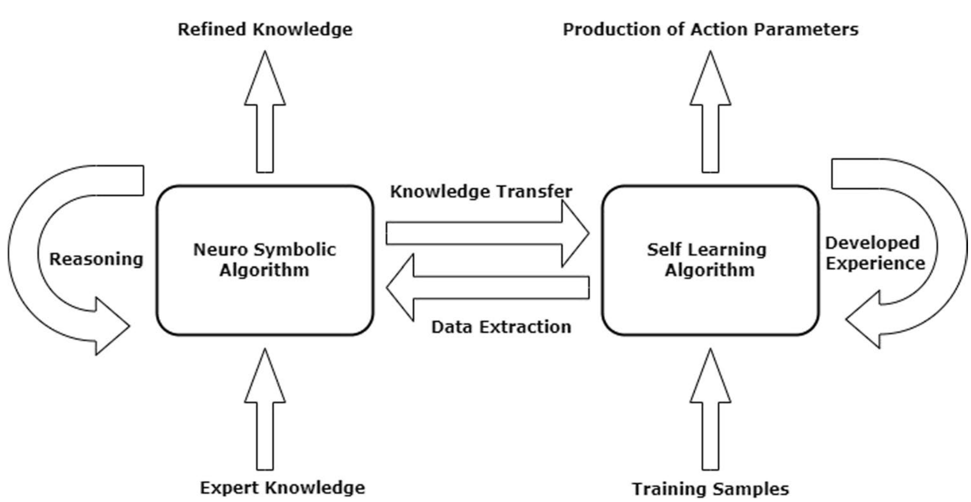

Fig. 2 Working Mechanism of the fusion of Self-Learning and Neuro-Symbolic Approach

quantities of data and computer resources. Yet, the investigations on this so far have provided vital insights into the neuro-symbolic approaches' possible implications in robotics. Figure 2 represents graphical representation of the working mechanism of self-learning techniques combined with neuro symbolic approach:

4.4.1 Examples of Advantages of Using Self-Learning and Neuro-Symbolic Approach For Several Application Areas

The application of autonomous mobile robots in industrial and commercial arena is multifaceted. In industries, the robots are used for a variety of tasks including multiple task accomplishment using a group of robots, surveillance for security purpose, monitoring manufacturing cycles, cleaning, delivering raw materials at different places, etc. where the fusion of self-learning algorithms with neuro-symbolic techniques can be particularly beneficial [161, 162]. For example, the reasoning capabilities of the robots would allow them to automatically allocate tasks among themselves based on the evaluation of importance and individual expertise of each robot where applicable. Furthermore, in warehouse environments where delivery of particular materials is necessary, this fusion might lead to efficient collaborative localization and path planning in terms of shortest path generation. Whereas in manufacturing cycles the reasoning capability might allow the robots to take over tasks depending on the battery level of the already assigned robot, for cleaning and surveillance application it can maximize the area coverage based on relative localization and movements. In commercial application, the environment may differ significantly then the industrial ones. Whereas most industrial environments are indoor environments, for commercial applications like cargo and medicine delivery and interaction with human agents, the environment is highly dynamic and noisy such as in urban streets. In these kinds of situations, the robots might benefit from the logical reasoning where the relative velocity and direction of multiple moving obstacles need to be measured. Furthermore, in other applications like military operations, search and rescue, explorations the fusion might be helpful for evaluating circumstances in GPS denied environments, recognizing geological patterns and landmarks for navigation, understanding the seriousness of situation from multiple dimensions while assisting humans in evacuating hazardous locations, etc.

4.4.2 Limitations and Challenges of Integrating Neuro-Symbolic Techniques with Self-Learning

Even though the combination of Neuro-Symbolic approach with self-learning algorithms possess extraordinary promise in terms of successful operations using mobile robots, some challenges are associated with it that needs to be taken care of properly [163–165]:

i) Since neural systems and symbolic systems differ quite a much in terms of data representation and individual problem-solving approaches, designing unified framework combining both of them according to the parameters of particular task objectives might be tricky. This might be a little more critical when heterogenous tasks are to be performed by homogenous robots in the realm of multi agent systems.
ii) Some of these approaches tend to adjust the design parameters on their own upon training which might lead to over simplification and over synapsis of data samples when large amount of training datasets are involved which needs to be handled carefully.
iii) When these approaches are employed for augmentation of DRL techniques for continuous space problems, the primary values of input parameters need to be set up properly.
iv) The computation of symbolic reasoning and handling the issue with its' brittleness and rigidity would require heavy focus on architecture development of Neuro-Symbolic approaches. Furthermore, the efficiency of task performance using these algorithms might be contingent upon research sections which are still under development such as

(a) Reconciliation first order logic learning and reasoning.
(b) Implementation of semi-supervised and incremental learning.
(c) Evaluation and analysis large scale gains of massive parallelism (particularly sensitive to multi agent system learning)
(d) Implementation of learning-reasoning-acting cycles in cognitive agents.
(e) Extraction of rules and interpretation for networks with large number of neurons.
(f) Application of learning using fibring functions.
(g) Development of theoretical understanding on the differences in application behavior between connectionist and symbolic method.
(h) Development of analogical and abductive neuro-symbolism.
(i) Investigation on Neuro-Symbolic models for modeling utility functions.

5 Discussion on Evaluation Metrics, Implementation Tools and Potential Research Scopes

The efficiency of mobile robot navigation techniques is typically evaluated by examining fee metrics typically known as cost functions. However, though based on specific applications and task completions these cost functions vary, some metrics are ubiquitous. The typically categorized metrics are security metrics, dimension metrics and smoothness metrics. The general performance evaluation criteria are as follows [166]:

i. Path length: distance to reach the goal point from starting point.
ii. Time: time taken to accomplish a task
iii. Collision: number of collisions per execution, per distance and per time
iv. Obstacle Clearance: minimum and maximum distance from obstacle at certain stage
v. Narrow space robustness: number of successful traverses in narrow paths
vi. Smoothness of trajectory: relative to control effort
vii. Mission success: number of successful missions

This is to note that, each of these metrics are relative to specific, particular applications and they are selectively picked for analysis of robot navigation efficiency in each application. For this reason, it's often problematic to draw a comparison of these metrics between navigation techniques when multiple different research works are in play. Table 3 offers a comprehensive and insightful comparison of diverse navigation methods, with a particular emphasis on machine learning techniques, showcasing their real-time applicability in dynamic environments. Additionally, the table provides a detailed overview of the robots employed, the environments in which the navigation tasks were conducted, and the details of obstacle avoidance parameters, ensuring a holistic understanding of the surveyed papers' contributions. Overall, the surveyed papers highlight the importance of selecting the appropriate navigation method and performance evaluation metrics based on the specific application and environment in which the mobile robot will operate.

Given the immense potential of the fusion of self-learning and Neuro-Symbolic approaches, numerous windows of innovation can be opened. Some of the potential researches which can be undertaken in this niche are:

a) Investigation on the performance of such fusion for task completion using mobile robots in extremely dynamic environment (e.g: cargo delivery and interaction with human agents in urban settings).
b) Assessment of the efficacy of the integration in environments containing high level of noise (captured by the sensor) and uncertainty (e.g: navigation of automated underwater vehicles, robot navigation and task completion in agricultural fields, etc.)
c) Implementation of self-learning and Neuro-Symbolic techniques in multi-agent settings with shared experiences (Federated AI) and check its' efficiency.
d) Training of homogenous mobile robots using self-learning and Neuro-Symbolic approach for performing heterogenous navigational tasks.
e) Assessment of time and parameter sensitivity of different self-learning and neuro-symbolic algorithm architectures for single and multi-robot navigation problems.

6 Conclusion

A detailed review of mobile robot navigation techniques was conducted and the key findings are as follows:

a) While soft computing approaches were widely applied in static contexts for differential drive robotic navigation systems and obstacle detection-avoidance, research in inconsistent and dynamically changing environments was very limited. However, recent advances in perceptual systems, sensor technologies, and processing power, on the other hand, have resulted in the creation of unique strategies for dealing with dynamic settings.
b) Hybridization of self-learning-based algorithms and swarm intelligence algorithms is a viable strategy for robotic navigation systems. Good navigation and avoidance strategies are keys to their success. Furthermore, fuzzy logic (a traditional AI algorithm) and self-learning-based algorithms, in particular, have shown significant potential in addressing mobile robot navigation challenges in dynamic environments.
c) The use of self-learning techniques- reinforcement learning, deep reinforcement learning, and the neuro-symbolic method- enables mobile robots to navigate and avoid obstacles successfully by gaining experience in the environment. These techniques can enable mobile robots to carry out various tasks independently while adapting to their surroundings which the other traditional techniques fail to accomplish even though the self-learning strategies need large computational resources and training time.
d) Most of the studies were conducted in simulation environments, and there is a necessity for more experimental studies to validate the effectiveness of these techniques in real-world scenarios.

Table 3 Works on mobile robot navigation and obstacle avoidance with simulation tools and environments

References	Navigation	Obstacle Avoidance	ML Techniques	Simulation Techniques Platform	Indoor/Outdoor
Fatemeh et al. [147]	✓	Using 9 static obstacles	Establishes the angle of robot motion at each time step for the robot to proceed towards the target while avoiding obstacles.	E-puck robot	Both
Sun et al. [148]	✓	It involves obstacle avoidance but the number of obstacles used for modeling was not mentioned.	Learn from human demonstrations' pathways and construct a path that corresponds to both their kinematics and human behavioral norms.	Rviz ROS	Both
Bruce et al. [149]	✓	N/A	The use of pre-trained visual features, the employment of an interactive replay buffer approach, and the addition of stochastic observations to the training environment.	Simultaneous localization and mapping (SLAM)	Both
Zielinski & Markowska [151]	✓	N/A	The model is trained and supervised using data from the simulation, allowing it to outperform an end-to-end solution that just uses built-in, trainable layers.	YOLO network	Outdoor
Shah et al. [152]	✓	It involves obstacle avoidance but the number of obstacles used for modeling was not mentioned.	The model is trained in a supervised way using data from the simulation, allowing it to produce much better outcomes than an end-to-end solution that employs just built-in, trainable layers.	FollowNet	Indoor
Surmann et al. [153]	✓	It involves obstacle avoidance but the number of obstacles used for modeling was not mentioned.	The model is trained in a supervised way using simulation data, allowing it to produce much better outcomes than an end-to-end system that just employs built-in, trainable layers.	RViz ROS	Indoor
Coraggio & Gregorio [158]	✓	N/A	The initial contribution to speeding up visual processing was made by virtual neural sensors that analysed a portion of the incoming image.	WiSARD map	N/A
Rakhman et al. [135]	✓	Tested from 3 to 7 obstacles	In a continuous space, it creates a number of random planning instances.	Gazebo	N/A
Surmann et al. [153]	✓	N/A	MAS is a multi-agent system framework that uses COP and 20 agents to mimic a surveillance situation. It is made up of two kinds of agents: COP and surveillance agents.	OpenAI Gym	N/A
Botteghi et al [160]	✓	It involves obstacle avoidance but the number of obstacles used for modeling was not mentioned.	DRL and SLAM are map-less route planners for navigation in unfamiliar situations, using training environment map information into the reward function.	SLAM ROS OpenAI	Both
Su et al. [150]	✓	It involves obstacle avoidance but the number of obstacles used for modeling was not mentioned.	Create a fuzzy rule for navigation to help you find the right activities in unfamiliar places.	N/A	N/A

Table 3 (continued)

References	Navigation	Obstacle Avoidance	ML Techniques	Simulation Techniques Platform	Indoor/Outdoor
Okal & Arras [167]	✓	N/A	To tackle this challenge, a Bayesian strategy based on particle filtering to preserve the posterior over states and models, and online planning employing trajectory sampling to discover the optimum action is used.	N/A	N/A
Ciou et al [168]	✓	It involves obstacle avoidance but the number of obstacles used for modeling was not mentioned.	The suggested system can socially navigate in settings with a 10% higher success rate and a 10% higher average payout.	Microsoft Kinect	Indoor
Chen et al. [127]	✓	It involves obstacle avoidance but the number of obstacles used for modeling was not mentioned.	To extract the pairwise interaction characteristic between the robot and each person, utilize a multi-layer perceptron.	Python ORCA	Both
Wöhlke et al. [169]	✓	It involves obstacle avoidance but the number of obstacles used for modeling was not mentioned.	Structured common domain information accessible in autonomous navigation tasks, such as a basic floor layout of the surroundings, into assumptions.	N/A	Both
Vasquez et al. [170]	✓	N/A	Applied as a set of learning algorithms and three ROS modules to help in the simulations transition to trials with real robots.	ROS	N/A
Dong et al. [171]	✓	It involves obstacle avoidance but the number of obstacles used for modeling was not mentioned.	To blend standard learning algorithms and quantum processing approaches, and perhaps certain problems will be handled in a novel way.	Visual C++	Indoor
Zhu et al. [172]	✓	It involves obstacle avoidance but the number of obstacles used for modeling was not mentioned.	Illustrate that RuRL can increase navigation performance while remaining scalable.	SLAM ROS Kinetic MATLAB	N/A
Burattini et al. [173]	✓	N/A	eNSBL, an expanded form of cNSBL generated through fibring neural networks, and neuro-symbolic modelling language for behavior-based systems.	N/A	N/A
Silver et al. [142]	✓	It involves obstacle avoidance but the number of obstacles used for modeling was not mentioned.	In object-centric robotics contexts, a neuro-symbolic method is used as a decision-making framework.	N/A	N/A
Kraetzschmar et.al [174]	✓	It involves obstacle avoidance but the number of obstacles used for modeling was not mentioned.	In mobile robots, hybrid spatial representations offer several chances to research neuro-symbolic integration.	N/A	N/A
Liu et al [93]	✓	It involves obstacle avoidance but the number of obstacles used for modeling was not mentioned.	Created by combining a static obstacle depiction with the use of an angle map vs the use of an occupancy grid.	Python ORCA	Outdoor
Hu et al. [92]	✓	It involves obstacle avoidance but the number of obstacles used for modeling was not mentioned.	A deep reinforcement learning-based system based on depth photographs is proposed for navigating in an unfamiliar environment.	Gazebo	N/A

Table 3 (continued)

References	Navigation	Obstacle Avoidance	ML Techniques	Simulation Techniques Platform	Indoor/Outdoor
Zhang et al. [175]	✓	It involves obstacle avoidance but the number of obstacles used for modeling was not mentioned.	Created a system that uses depth pictures, elevation maps, and 3D orientation as inputs to allow optimum navigation behaviors for mobile robots using a network architecture.	Gazebo ROS	N/A
Chen et al [139]	✓	It involves obstacle avoidance but the number of obstacles used for modeling was not mentioned.	To improve our performance, we focused experience, response, and incorporated curricular learning methodologies based on fighting double DQN.	Gazebo	Indoor

Future works in this domain can incorporate the blended application machine learning-based algorithms with the various evolutionary algorithms for mobile robot navigation in dynamically changing environments. While the machine learning-based algorithms would provide the robot with an overall idea and experience of the environment, the evolutionary techniques would augment its capability of finding the optimal solution to a problem.

Acknowledgements Not applicable.

Authors' Contributions Suaib Al Mahmud and Abdurrahman Kamarulariffin were behind literature collection, conduction of review, generation of figures, curation of data, refinement of contents and overall organization. Azhar Mohd Ibrahim was responsible for conceptualization, supervision, fund management, revision and administration. Ahmad Jazlan Haja Mohideen helped with guidance and contributed to the revision.

Funding This research is supported by the FRGS 2023 Grant: FRGS/1/2023/TK02/UIAM/02/1 awarded by the Ministry of Education Malaysia.

Data Availability There is no sharable data associated with this paper.

Declaration

Ethical Approval Not applicable. This study does not include any human participants or animals.

Consent to Participate Not applicable.

Consent for Publication All authors have consented to publish the manuscript.

Conflict of Interest The authors declare that they have no competing interest.

References

1. Rehman, B., Yagfarov, R., Klimchik, A.: Interactive mobile robot in a dynamic environment. In: IFAC-PapersOnLine, pp. 354–359. Elsevier B.V. (2018). https://doi.org/10.1016/j.ifacol.2018.11.331
2. Ni, J., Wu, L., Fan, X., Yang, S.X.: Bioinspired intelligent algorithm and its applications for mobile robot control: A survey. Comput. Intell. Neurosci. **2016**. Hindawi Publishing Corporation, (2016). https://doi.org/10.1155/2016/3810903

3. Takahashi, O., Schilling, R.J.: Motion Planning in a Plane Using Generalized Voronoi Diagrams. IEEE Trans. Robot. Autom. **5**(2), 143–150 (1989). https://doi.org/10.1109/70.88035
4. Bhattacharya, P., Gavrilova, M.L.: Roadmap-based path planning - Using the voronoi diagram for a clearance-based shortest path. IEEE Robot. Autom. Mag. **15**(2), 58–66 (2008). https://doi.org/10.1109/MRA.2008.921540
5. E. J. Gómez, F. M. Santa, and F. H. M. Sarmiento, "A comparative study of geometric path planning methods for a mobile robot: Potential field and Voronoi diagrams," in 2013 2nd International Congress of Engineering Mechatronics and Automation, CIIMA 2013 - Conference Proceedings, 2013. doi: https://doi.org/10.1109/CIIMA.2013.6682776.
6. Abiyev, R., Ibrahim, D., Erin, B.: Navigation of mobile robots in the presence of obstacles. Adv. Eng. Softw. **41**(10–11), 1179–1186 (2010). https://doi.org/10.1016/j.advengsoft.2010.08.001
7. Soltani, A.R., Tawfik, H., Goulermas, J.Y., Fernando, T.: Path planning in construction sites: Performance evaluation of the dijkstra, a*, and GA search algorithms. Adv. Eng. Inform. **16**(4), 291–303 (2002). https://doi.org/10.1016/S1474-0346(03)00018-1
8. Masehian, E., Amin-Naseri, M.R.: A voronoi diagram-visibility graph-potencial field compound algorith for robot path planning. J. Robot. Syst. **21**(6), 275–300 (2004). https://doi.org/10.1002/rob.20014
9. M. Weigl, B. Siemiatkowska, K. A. Sikorski, and A. Borkowski, "Grid-based mapping for mobile robot autonomous."
10. Park, K.-H., Kim, Y.-J., Kim, J.-H.: Modular Q-learning based multi-agent cooperation for robot soccer. Rob. Auton. Syst. **35**(2), 109–122 (2001). https://doi.org/10.1016/S0921-8890(01)00114-2
11. E. L. Evi, M. B. Montaner, and A. Ramirez-Serrano, "Expert Systems with Applications Fuzzy knowledge-based controller design for autonomous robot navigation," 1998.
12. I. Engedy and G. Horváth, "Artificial neural network based local motion planning of a wheeled mobile robot," in 11th IEEE International Symposium on Computational Intelligence and Informatics, CINTI 2010 - Proceedings, 2010, pp. 213–218. doi: 10.1109/CINTI.2010.5672245.
13. Zhu, A., Yang, S.X.: Neurofuzzy-based approach to mobile robot navigation in unknown environments. IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. **37**(4), 610–621 (2007). https://doi.org/10.1109/TSMCC.2007.897499
14. A. Ghorbani, S. Shiry, and A. Nodehi, "Using genetic algorithm for a mobile robot path planning," in Proceedings - 2009 International Conference on Future Computer and Communication, ICFCC 2009, 2009, pp. 164–166. doi: https://doi.org/10.1109/ICFCC.2009.28.
15. Ahmadzadeh, S., Ghanavati, M.: Navigation of mobile robot using the PSO particle swarm optimization. J. Acad. Appl. Stud. **2** (2012). Available: https://api.semanticscholar.org/CorpusID:18077178
16. Garcia, M.A.P., Montiel, O., Castillo, O., Sepúlveda, R., Melin, P.: Path planning for autonomous mobile robot navigation with ant colony optimization and fuzzy cost function evaluation. Appl. Soft Comput. J. **9**(3), 1102–1110 (2009). https://doi.org/10.1016/j.asoc.2009.02.014
17. Miao, H., Tian, Y.C.: Dynamic robot path planning using an enhanced simulated annealing approach. Appl. Math. Comput. **222**, 420–437 (2013). https://doi.org/10.1016/j.amc.2013.07.022
18. Hui, N.B., Mahendar, V., Pratihar, D.K.: Time-optimal, collision-free navigation of a car-like mobile robot using neuro-fuzzy approaches. Fuzzy Set. Syst. **157**(16), 2171–2204 (2006). https://doi.org/10.1016/j.fss.2006.04.004
19. Abadi, D.N.M., Khooban, M.H.: Design of optimal Mamdani-type fuzzy controller for nonholonomic wheeled mobile robots. J. King Saud Univ. Eng. Sci. **27**(1), 92–100 (2015). https://doi.org/10.1016/j.jksues.2013.05.003
20. Chakraborty, N., Ghosal, A.: Kinematics of wheeled mobile robots on uneven terrain. Mech. Mach. Theory. **39**(12 SPEC ISS), 1273–1287 (2004). https://doi.org/10.1016/j.mechmachtheory.2004.05.016
21. X. Wang and S. X. Yangt, "A Neuro-Fuzzy Approach to Obstacle Avoidance of a' Nonholonomic Mobile Robot*."
22. Castillo, O., Martínez-Marroquín, R., Melin, P., Valdez, F., Soria, J.: Comparative study of bio-inspired algorithms applied to the optimization of type-1 and type-2 fuzzy controllers for an autonomous mobile robot. Inf. Sci. (N Y). **192**, 19–38 (2012). https://doi.org/10.1016/j.ins.2010.02.022
23. Al-Araji, A.S., Abbod, M.F., Al-Raweshidy, H.S.: Applying posture identifier in designing an adaptive nonlinear predictive controller for nonholonomic mobile robot. Neurocomputing. **99**, 543–554 (2013). https://doi.org/10.1016/j.neucom.2012.06.033
24. Liang Y, Xu L, Wei R, et al. Adaptive Fuzzy Control for Trajectory Tracking of Mobile Robot. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). 2010. 4755–4760.
25. Bijli, M., Kumar, N.: Autonomous navigation of mobile robot with obstacle avoidance: a review. In: Singh, P.K., Veselov, G., Pljonkin, A., Kumar, Y., Paprzycki, M., Zachinyaev, Y. (eds.) Futuristic Trends in Network and Communication Technologies, pp. 305–316. Springer, Singapore (2021)
26. Loganathan, A., Ahmad, N.S.: A systematic review on recent advances in autonomous mobile robot navigation. Eng. Sci. Technol. Int. J. **40**, 101343 (2023). https://doi.org/10.1016/j.jestch.2023.101343
27. Liu, L., Wang, X., Yang, X., Liu, H., Li, J., Wang, P.: Path planning techniques for mobile robots: Review and prospect. Expert Syst. Appl. **227**, 120254 (2023). https://doi.org/10.1016/j.eswa.2023.120254
28. Patle, B.K., Babu, G.L., Pandey, A., Parhi, D.R.K., Jagadeesh, A.: A review: On path planning strategies for navigation of mobile robot. Def. Technol. **15**(4), 582–606 (2019). https://doi.org/10.1016/j.dt.2019.04.011
29. Naema, A., Rafai, A., Adzhar, N., Jaini, N.I.: A review on path planning and obstacle avoidance algorithms for autonomous mobile robots, J. Robot. **2022** (2022). https://doi.org/10.1155/2022/2538220
30. Zhu, K., Zhang, T.: Deep reinforcement learning based mobile robot navigation: a review. Tsinghua Sci. Technol. **26**(5), 674–691 (2021). https://doi.org/10.26599/TST.2021.9010012
31. Ibrahim, D.: An Overview of Soft Computing. Procedia Comput. Sci., Elsevier B.V. 34–38 (2016). https://doi.org/10.1016/j.procs.2016.09.366
32. Algabri, M., Mathkour, H., Ramdane, H., Alsulaiman, M.: Comparative study of soft computing techniques for mobile robot navigation in an unknown environment. Comput. Human Behav. **50**, 42–56 (2015). https://doi.org/10.1016/j.chb.2015.03.062
33. Hoffmann, F.: Soft computing techniques for the design of mobile robot behaviors. Inf. Sci. (NY). **122**(2), 241–258 (2000). https://doi.org/10.1016/S0020-0255(99)00120-6
34. L. A. Zadeh, "The Concept of a Linguistic Variable and its Application to Approximate Reasoning-I," 1975.
35. Allagui, N.Y., Salem, F.A., Aljuaid, A.M.: Artificial Fuzzy-PID Gain Scheduling Algorithm Design for Motion Control in Differential Drive Mobile Robotic Platforms. Comput. Intell. Neurosci. 1–13 (2021, 2021). https://doi.org/10.1155/2021/5542888
36. Buragohain, M., Mahanta, C.: A Novel Approach for ANFIS Modeling Based on Full Factorial Design. Appl. Soft Comput. **8**, 609–625 (2008). https://doi.org/10.1016/j.asoc.2007.03.010
37. U. Farooq, K. M. Hasan, G. Abbas, and M. U. Asad, "Comparative analysis of zero order Sugeno and Mamdani fuzzy

logic controllers for obstacle avoidance behavior in mobile robot navigation," in Proceedings of the 2011 International Conference and Workshop on the Current Trends in Information Technology, CTIT'11, 2011, pp. 113–119. doi: 10.1109/CTIT.2011.6107945.
38. A.-M. Zou, Z.-G. Hou, S.-Y. Fu, and M. Tan, "LNCS 3972 - Neural Networks for Mobile Robot Navigation: A Survey," 2006.
39. H. Xiao, L. Liao, and F. Zhou, "Mobile Robot Path Planning Based on Q-ANN," 2007.
40. Rai, N., Rai, B.: Neural network based closed loop speed control of DC motor using Arduino Uno. Available: https://api.semanticscholar.org/CorpusID:18809816 (2013)
41. H. D. Patiño and R. Carelli, "Neural network-based optimal control for autonomous mobile vehicle navigation," in IEEE International Symposium on Intelligent Control - Proceedings, 2004, pp. 391–396. doi: https://doi.org/10.1109/isic.2004.1387715.
42. S. X. Yang and M. Meng, "Neural Network Approaches to Dynamic Collision-Free Trajectory Generation," 2001.
43. Nichols, E., McDaid, L.J., Siddique, N.: Biologically inspired SNN for robot control. IEEE Trans. Cybern. **43**(1), 115–128 (2013). https://doi.org/10.1109/TSMCB.2012.2200674
44. Motlagh, O., Nakhaeinia, D., Tang, S.H., Karasfi, B., Khaksar, W.: Automatic navigation of mobile robots in unknown environments. Neural Comput. Appl. **24**(7–8), 1569–1581 (2014). https://doi.org/10.1007/s00521-013-1393-z
45. A. v. Gavrilov and S. Lee, "An Architecture of Hybrid Neural Network Based Navigation System for Mobile Robot," Apr. 2008, pp. 587–590. doi: https://doi.org/10.1109/isda.2007.49.
46. Singh, M.K., Parhi, D.R.: Path optimisation of a mobile robot using an artificial neural network controller. Int. J. Syst. Sci. **42**(1), 107–120 (2011). https://doi.org/10.1080/00207720903470155
47. K. al Mutib and E. Mattar, "Neuro-fuzzy controlled autonomous mobile robotics system," in Proceedings - 2011 UKSim 13th International Conference on Modelling and Simulation, UKSim 2011, 2011, pp. 1–7. doi: https://doi.org/10.1109/UKSIM.2011.11.
48. J. Godjevac and N. Steele, "Neuro-fuzzy control of a mobile robot," 1999.
49. W. Li, C. Ma, and F. M. Wahl, "A neuro-fuzzy system architecture for behavior-based control of a mobile robot in unknown environments 1," 1997.
50. M. M. Joshi and M. A. Zaveri, "Neuro-fuzzy based autonomous mobile robot navigation system," in 11th International Conference on Control, Automation, Robotics and Vision, ICARCV 2010, 2010, pp. 384–389. doi: https://doi.org/10.1109/ICARCV.2010.5707354.
51. Marichal, G.N., Acosta, L., Moreno, L., Méndez, J.A., Rodrigo, J.J., Sigut, M.: Obstacle avoidance for a mobile robot: a neuro-fuzzy approach. Fuzzy Sets Syst. **124**(2), 171–179 (2001). https://doi.org/10.1016/S0165-0114(00)00095-6
52. N. Zhang, D. Beetner, D. C. Wunsch, B. Hemmelman, and A. Hasan, "An embedded real-time neuro-fuzzy controller for mobile robot navigation," in IEEE International Conference on Fuzzy Systems, 2005, pp. 319–324. doi: 10.1109/fuzzy.2005.1452413.
53. Baturone, I., Gersnoviez, A., Barriga, Á.: Neuro-fuzzy techniques to optimize an FPGA embedded controller for robot navigation. Appl. Soft Comput. J. **21**, 95–106 (2014). https://doi.org/10.1016/j.asoc.2014.03.001
54. X. Ma, X. Li, H. Qiao, and P. O. Box, "Fuzzy neural network-based real-time self-reaction of mobile robot in unknown environments."
55. M. Imen, M. Mansouri, and M. A. Shoorehdeli, "Tracking control of mobile robot using ANFIS," in 2011 IEEE International Conference on Mechatronics and Automation, ICMA 2011, 2011, pp. 422–427. doi: https://doi.org/10.1109/ICMA.2011.5985695.
56. V. Ganapathy, C. Y. Soh, and J. Ng, "Fuzzy and neural controllers for acute obstacle avoidance in mobile robot navigation," in IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM, 2009, pp. 1236–1241. doi: 10.1109/AIM.2009.5229761.
57. T. Zhao and Y. Wang, "A neural-network based autonomous navigation system using mobile robots," in 2012 12th International Conference on Control, Automation, Robotics and Vision, ICARCV 2012, 2012, pp. 1101–1106. doi: https://doi.org/10.1109/ICARCV.2012.6485311.
58. D. Kumar and K. Dhama, "Neuro-fuzzy control of an intelligent mobile robot," in Proceedings - 2012 2nd International Conference on Advanced Computing and Communication Technologies, ACCT 2012, 2012, pp. 106–111. doi: https://doi.org/10.1109/ACCT.2012.77.
59. Song, K.-T., Sheen, L.-H.: Heuristic fuzzy-neuro network and its application to reactive navigation of a mobile robot. Fuzzy Sets Syst. **110**(3), 331–340 (2000). https://doi.org/10.1016/S0165-0114(97)00401-6
60. Lee, C.H., Chiu, M.H.: Recurrent neuro fuzzy control design for tracking of mobile robots via hybrid algorithm. Expert. Syst. Appl. **36**(5), 8993–8999 (2009). https://doi.org/10.1016/j.eswa.2008.11.051
61. S. U. Deshpande and S. S. Bhosale, "Adaptive neuro-fuzzy inference system based robotic navigation," in 2013 IEEE International Conference on Computational Intelligence and Computing Research, IEEE ICCIC 2013, 2013. doi: https://doi.org/10.1109/ICCIC.2013.6724153.
62. C. G. Rusu, I. T. Birou, and E. Szöke, "Fuzzy based obstacle avoidance system for autonomous mobile robot," in 2010 IEEE International Conference on Automation, Quality and Testing, Robotics, AQTR 2010 - Proceedings, 2010, vol. 1, pp. 337–342. doi: https://doi.org/10.1109/AQTR.2010.5520862.
63. Pothal, J.K., Parhi, D.R.: Navigation of multiple mobile robots in a highly clutter terrains using adaptive neuro-fuzzy inference system. Rob. Auton. Syst. **72**, 48–58 (2015). https://doi.org/10.1016/j.robot.2015.04.007
64. Chikurtev, D., Chivarov, N., Chivarov, S., Chikurteva, A.: Mobile robot localization and navigation using LIDAR and indoor GPS. IFAC-PapersOnLine. **54**(13), 351–356 (2021). https://doi.org/10.1016/j.ifacol.2021.10.472
65. Bobyr, M., Arkhipov, A., Emelyanov, S., Milostnaya, N.: A method for creating a depth map based on a three-level fuzzy model. Eng. Appl. Artif. Intell. **117**, 105629 (2023). https://doi.org/10.1016/j.engappai.2022.105629
66. K. C. Ng, M. M. Trivedi, and S. Member, "A Neuro-Fuzzy Controller for Mobile Robot Navigation and Multirobot Convoying," 1998.
67. M. Khoshnejad and K. Demirli, "Autonomous parallel parking of a car-like mobile robot by a neuro-fuzzy behavior-based controller," in Annual Conference of the North American Fuzzy Information Processing Society - NAFIPS, 2005, vol. 2005, pp. 814–819. doi: 10.1109/NAFIPS.2005.1548645.
68. Al-Mayyahi, A., Wang, W., Birch, P.: Adaptive neuro-fuzzy technique for autonomous ground vehicle navigation. Robotics. **3**(4), 349–370 (2014). https://doi.org/10.3390/robotics3040349
69. Pradhan, S.K., Parhi, D.R., Panda, A.K.: Fuzzy logic techniques for navigation of several mobile robots. Appl. Soft Comput J. **9**(1), 290–304 (2009). https://doi.org/10.1016/j.asoc.2008.04.008
70. M. Algabri, H. Mathkour, and H. Ramdane, "Mobile Robot Navigation and Obstacle-avoidance using ANFIS in Unknown Environment," 2014.
71. Hui, N.B., Pratihar, D.K.: A comparative study on some navigation schemes of a real robot tackling moving obstacles. Robot.

Comput. Integr. Manuf. **25**(4–5), 810–828 (2009). https://doi.org/10.1016/j.rcim.2008.12.003
72. Abdessemed, F., Benmahammed, K., Monacelli, E.: A fuzzy-based reactive controller for a non-holonomic mobile robot. Rob. Auton. Syst. **47**(1), 31–46 (2004). https://doi.org/10.1016/j.robot.2004.02.006
73. Selekwa, M.F., Dunlap, D.D., Shi, D., Collins, E.G.: Robot navigation in very cluttered environments by preference-based fuzzy behaviors. Rob. Auton. Syst. **56**(3), 231–246 (2008). https://doi.org/10.1016/j.robot.2007.07.006
74. D. Kumar Pratihar, K. Deb, and A. Ghosh, "A genetic-fuzzy approach for mobile robot navigation among moving obstacles," 1999.
75. A. Babalou and N. Seifipour, "Application of fuzzy decision making in mobile robot navigation in dynamic environments," in IEEE International Conference on Fuzzy Systems, 2009, pp. 877–881. doi: https://doi.org/10.1109/FUZZY.2009.5277189.
76. Li, T.H.S., Chang, S.J., Chen, Y.X.: Implementation of human-like driving skills by autonomous fuzzy behavior control on an FPGA-based car-like mobile robot. IEEE Trans. Ind. Electron. **50**(5), 867–880 (2003). https://doi.org/10.1109/TIE.2003.817490
77. "II. SYSTEM DESCRIPTION."
78. Antonelli, G., Chiaverini, S., Fusco, G.: A fuzzy-logic-based approach for mobile robot path tracking. IEEE Trans. Fuzzy Syst. **15**(2), 211–221 (2007). https://doi.org/10.1109/TFUZZ.2006.879998
79. E. Ayari, S. Hadouaj, and K. Ghedira, "A fuzzy logic method for autonomous robot navigation in dynamic and uncertain environment composed with complex traps," in Proceedings - 5th International Multi-Conference on Computing in the Global Information Technology, ICCGI 2010, 2010, pp. 18–23. doi: https://doi.org/10.1109/ICCGI.2010.47.
80. Rossomando, F.G., Soria, C.M.: Design and implementation of adaptive neural PID for non linear dynamics in mobile robots. IEEE Lat. Am. Trans. **13**(4), 913–918 (2015). https://doi.org/10.1109/TLA.2015.7106337
81. Al-Jarrah, R., Shahzad, A., Roth, H.: Path planning and motion coordination for multi-robots system using probabilistic neuro-fuzzy. IFAC-PapersOnLine. **28**(10), 46–51 (2015). https://doi.org/10.1016/j.ifacol.2015.08.106
82. D. Janglová, "Neural Networks in Mobile Robot Motion."
83. R. Glasius, A. Komoda, and S. C. A. M. Gielen, "0893-6080(94) E0045-M Neural Network Dynamics for Path Planning and Obstacle Avoidance," 1995.
84. Kim, C.J., Chwa, D.: Obstacle avoidance method for wheeled mobile robots using interval type-2 fuzzy neural network. IEEE Trans. Fuzzy Syst. **23**(3), 677–687 (2015). https://doi.org/10.1109/TFUZZ.2014.2321771
85. F. Mahmud, A. Arafat, and S. T. Zuhori, "Intelligent autonomous vehicle navigated by using artificial neural network," in 2012 7th International Conference on Electrical and Computer Engineering, ICECE 2012, 2012, pp. 105–108. doi: https://doi.org/10.1109/ICECE.2012.6471496.
86. A. Chohra, C. Benmehrez, and A. Farah, "Neural Navigation Approach for Intelligent Autonomous Vehicles (IAV) in Partially Structured Environments," 1998.
87. H. Brahmi, B. Ammar, and A. M. Alimi, "Intelligent path planning algorithm for autonomous robot based on recurrent neural networks," in 2013 International Conference on Advanced Logistics and Transport, ICALT 2013, 2013, pp. 199–204. doi: https://doi.org/10.1109/ICAdLT.2013.6568459.
88. S. X. Yang, T. Hu, X. Yuan, P. X. Liu, and M. Meng, "A neural network based torque controller for collision-free navigation of mobile robots," in Proceedings - IEEE International Conference on Robotics and Automation, 2003, vol. 1, pp. 13–18. doi: https://doi.org/10.1109/robot.2003.1241566.
89. A. Elshamli, H. A. Abdullah, and S. Areibi, "Genetic algorithm for dynamic path planning," in Canadian Conference on Electrical and Computer Engineering, 2004, vol. 2, pp. 0677–0680. doi: https://doi.org/10.1109/ccece.2004.1345203.
90. Tuncer, A., Yildirim, M.: Dynamic path planning of mobile robots with improved genetic algorithm. Comput. Electr. Eng. **38**(6), 1564–1572 (2012). https://doi.org/10.1016/j.compeleceng.2012.06.016
91. L. Ming, G. Zailin, and Y. Shuzi, "Mobile robot fuzzy control optimization using genetic algorithm," 1996.
92. Hu, H., Zhang, K., Tan, A.H., Ruan, M., Agia, C.G., Nejat, G.: A Sim-to-Real Pipeline for Deep Reinforcement Learning for Autonomous Robot Navigation in Cluttered Rough Terrain. IEEE Rob. Autom. Lett. **6**(4), 6569–6576 (2021). https://doi.org/10.1109/LRA.2021.3093551
93. Liu, B., Wang, L., Liu, M.: Lifelong Federated Reinforcement Learning: A Learning Architecture for Navigation in Cloud Robotic Systems. IEEE Rob. Autom. Lett. **4**(4), 4555–4562 (2019). https://doi.org/10.1109/LRA.2019.2931179
94. Q. Y. Bao, S. M. Li, W. Y. Shang, and M. J. An, "A fuzzy behavior-based architecture for mobile robot navigation in unknown environments," in 2009 International Conference on Artificial Intelligence and Computational Intelligence, AICI 2009, 2009, vol. 2, pp. 257–261. doi: https://doi.org/10.1109/AICI.2009.125.
95. Castillo, O., Trujillo, L., Melin, P.: Multiple objective genetic algorithms for path-planning optimization in autonomous mobile robots. Soft. Comput. **11**(3), 269–279 (2007). https://doi.org/10.1007/s00500-006-0068-4
96. T. Arora, Y. Gigras, and V. Arora, "Robotic Path Planning using Genetic Algorithm in Dynamic Environment," 2014.
97. H. Miao and Y. C. Tian, "Robot path planning in dynamic environments using a simulated annealing based approach," in 2008 10th International Conference on Control, Automation, Robotics and Vision, ICARCV 2008, 2008, pp. 1253–1258. doi: https://doi.org/10.1109/ICARCV.2008.4795701.
98. H. Martínez-Alfaro and S. Gómez-García, "Mobile robot path planning and tracking using simulated annealing and fuzzy logic control."
99. Q. Zhu, Y. Yan, and Z. Xing, "Robot path planning based on artificial potential field approach with simulated annealing," in Proceedings - ISDA 2006: Sixth International Conference on Intelligent Systems Design and Applications, 2006, vol. 2, pp. 622–627. doi: https://doi.org/10.1109/ISDA.2006.253908.
100. R. E. Precup, M. B. Radac, C. A. Dragos, S. Preitl, and E. M. Petriu, "Simulated annealing approach to fuzzy modeling of servo systems," in 2013 IEEE International Conference on Cybernetics, CYBCONF 2013, 2013, pp. 267–272. doi: https://doi.org/10.1109/CYBConf.2013.6617449.
101. F. Janabi-Sharifi and D. Vinke, "Integration of the artificial potential field approach with simulated annealing for robot path planning," in Proceedings of 8th IEEE International Symposium on Intelligent Control, pp. 536–541. doi: https://doi.org/10.1109/ISIC.1993.397640.
102. Tavares, R.S., Martins, T.C., Tsuzuki, M.S.G.: Simulated annealing with adaptive neighborhood: A case study in off-line robot path planning. Expert Syst. Appl. **38**(4), 2951–2965 (2011). https://doi.org/10.1016/j.eswa.2010.08.084
103. E. Nakamura and N. Kehtarnavaz, "Optimization of fuzzy membership function parameters," in Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, 1995, vol. 1, pp. 1–6. doi: https://doi.org/10.1109/icsmc.1995.537723.
104. A. Hussein, H. Mostafa, M. Badrel-Din, O. Sultan, and A. Khamis, "Metaheuristic optimization approach to mobile robot path planning," in International Conference on Engineering and Technology, ICET 2012 - Conference Booklet, 2012. doi: https://doi.org/10.1109/ICEngTechnol.2012.6396150.

105. Zhang, Y., Gong, D.W., Zhang, J.H.: Robot path planning in uncertain environment using multi-objective particle swarm optimization. Neurocomputing. **103**, 172–185 (2013). https://doi.org/10.1016/j.neucom.2012.09.019
106. A. Synodinos and N. A. Aspragathos, "Path planning of a mobile robot using solid modeling techniques on potential fields," in Proceedings of 2010 IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications, MESA 2010, 2010, pp. 549–553. doi: https://doi.org/10.1109/MESA.2010.5552011.
107. Y. Zhao and W. Zu, "Real-time obstacle avoidance method for mobile robots based on a modified particle swarm optimization," in Proceedings of the 2009 International Joint Conference on Computational Sciences and Optimization, CSO 2009, 2009, vol. 2, pp. 269–272. doi: https://doi.org/10.1109/CSO.2009.196.
108. Q. Zhang, J. Ma, and Q. Liu, "Path planning based quadtree representation for mobile robot using hybrid-simulated annealing and ant colony optimization algorithm," in Proceedings of the World Congress on Intelligent Control and Automation (WCICA), 2012, pp. 2537–2542. doi: https://doi.org/10.1109/WCICA.2012.6358300.
109. Q. Li, W. Zhang, Y. Yin, and Z. Wang Guangjun Liu, "An Improved Genetic Algorithm of Optimum Path Planning for Mobile Robots*," 2006.
110. P. Raja and S. Pugazhenthi, "Path planning for mobile robots in dynamic environments using particle swarm optimization," in ARTCom 2009 - International Conference on Advances in Recent Technologies in Communication and Computing, 2009, pp. 401–405. doi: https://doi.org/10.1109/ARTCom.2009.24.
111. E. Masehian and D. Sedighizadeh, "A multi-objective PSO-based algorithm for robot path planning," in Proceedings of the IEEE International Conference on Industrial Technology, 2010, pp. 465–470. doi: https://doi.org/10.1109/ICIT.2010.5472755.
112. Wong, C.-C., Wang, H.-Y., Li, S.-A.: PSO-based motion fuzzy controller design for mobile robots. Available: https://api.semanticscholar.org/CorpusID:126337032 (2008)
113. H. Y. Chung, C. C. Hou, and S. C. Liu, "Automatic navigation of a wheeled mobile robot using particle swarm optimization and fuzzy control," in IEEE International Symposium on Industrial Electronics, 2013. doi: https://doi.org/10.1109/ISIE.2013.6563767.
114. Shiltagh, N.A., Jalal, L.D.: Optimal path planning for intelligent mobile robot navigation using modified particle swarm optimization. Int. J. Eng. Adv. Technol. **2**(4), 260–267 (2013). Available: http://www.ijeat.org/attachments/File/v2i4/D1390042413.pdf
115. Hsu, C.H., Juang, C.F.: Multi-objective continuous-ant-colony-optimized FC for robot wall-following control. IEEE Comput. Intell. Mag. **8**(3), 28–40 (2013). https://doi.org/10.1109/MCI.2013.2264233
116. L. Lu and D. Gong, "Robot path planning in unknown environments using particle swarm optimization," in Proceedings - 4th International Conference on Natural Computation, ICNC 2008, 2008, vol. 4, pp. 422–426. doi: 10.1109/ICNC.2008.923.
117. Z. T. Allawi and T. Y. Abdalla, "A PSO-optimized type-2 fuzzy logic controller for navigation of multiple mobile robots," in 2014 19th International Conference on Methods and Models in Automation and Robotics, MMAR 2014, Nov. 2014, pp. 33–39. doi: https://doi.org/10.1109/MMAR.2014.6957321.
118. M. Dorigo and G. di Caro, "Ant colony optimization: A new meta-heuristic," in Proceedings of the 1999 Congress on Evolutionary Computation, CEC 1999, 1999, vol. 2, pp. 1470–1477. doi: https://doi.org/10.1109/CEC.1999.782657.
119. T. Guan-Zheng, H. E. Huan, and S. Aaron, "Ant Colony System Algorithm for Real-Time Globally Optimal Path Planning of Mobile Robots."
120. F. K. Purian and E. Sadeghian, "Mobile robots path planning using ant colony optimization and Fuzzy Logic algorithms in unknown dynamic environments," in CARE 2013 - 2013 IEEE International Conference on Control, Automation, Robotics and Embedded Systems, Proceedings, 2013. doi: https://doi.org/10.1109/CARE.2013.6733718.
121. B. Zeng, Y. Yang, and Y. Xu, "Mobile robot navigation in unknown dynamic environment based on ant colony algorithm," in Proceedings of the 2009 WRI Global Congress on Intelligent Systems, GCIS 2009, 2009, vol. 3, pp. 98–102. doi: https://doi.org/10.1109/GCIS.2009.274.
122. N. B. Sariff and N. Buniyamin, "Comparative study of genetic algorithm and ant colony optimization algorithm performances for robot path planning in global static environments of different complexities," in Proceedings of IEEE International Symposium on Computational Intelligence in Robotics and Automation, CIRA, 2009, pp. 132–137. doi: 10.1109/CIRA.2009.5423220.
123. N. Ganganath, C. T. Cheng, and C. K. Tse, "An ACO-based off-line path planner for nonholonomic mobile robots," in Proceedings - IEEE International Symposium on Circuits and Systems, 2014, pp. 1038–1041. doi: https://doi.org/10.1109/ISCAS.2014.6865316.
124. Juang, C.F., Hsu, C.H.: Reinforcement ant optimized fuzzy controller for mobile-robot wall-following control. IEEE Trans. Ind. Electron. **56**(10), 3931–3940 (2009). https://doi.org/10.1109/TIE.2009.2017557
125. Hsu, C.H., Juang, C.F.: Evolutionary robot wall-following control using type-2 fuzzy controller with species-DE-activated continuous ACO. IEEE Trans. Fuzzy Syst. **21**(1), 100–112 (2013). https://doi.org/10.1109/TFUZZ.2012.2202665
126. Juang, C.F., Chang, Y.C.: Evolutionary-group-based particle-swarm-optimized fuzzy controller with application to mobile-robot navigation in unknown environments. IEEE Trans. Fuzzy Syst. **19**(2), 379–392 (2011). https://doi.org/10.1109/TFUZZ.2011.2104364
127. Chen, X., Kong, Y., Fang, X., Wu, Q.: A fast two-stage ACO algorithm for robotic path planning. Neural Comput. Applic. **22**(2), 313–319 (2013). https://doi.org/10.1007/s00521-011-0682-7
128. M. A. Hossain and I. Ferdous, "Autonomous robot path planning in dynamic environment using a new optimization technique inspired by Bacterial Foraging technique," in 2013 International Conference on Electrical Information and Communication Technology, EICT 2013, 2014. doi: https://doi.org/10.1109/EICT.2014.6777884.
129. Liang, X.D., Li, L.Y., Wu, J.G., Chen, H.N.: Mobile robot path planning based on adaptive bacterial foraging algorithm. J. Cent. South Univ. **20**(12), 3391–3400 (2013). https://doi.org/10.1007/s11771-013-1864-5
130. M. Brand and X. H. Yu, "Autonomous robot path optimization using firefly algorithm," in Proceedings - International Conference on Machine Learning and Cybernetics, 2013, vol. 3, pp. 1028–1032. doi: https://doi.org/10.1109/ICMLC.2013.6890747.
131. Mohajer, B., Kiani, K., Samiei, E., Sharifi, M.: A new online random particles optimization algorithm for mobile robot path planning in dynamic environments. Math. Probl. Eng. **2013**, (2013). https://doi.org/10.1155/2013/491346
132. X. Fan, X. Luo, S. Yi, S. Yang, and H. Zhang, "Optimal path planning for mobile robots based on intensified ant colony optimization algorithm," in RISSP 2003, 2003, vol. 2003-October, pp. 131–136. doi: 10.1109/RISSP.2003.1285562.
133. M. Algabri, H. Mathkour, H. Ramdane, M. Alsulaiman, and K. Al-Mutib, "Self-learning Mobile Robot Navigation in Unknown Environment Using Evolutionary Learning."

134. Lee, M.-F.R., Yusuf, S.H.: Mobile Robot Navigation Using Deep Reinforcement Learning. Processes. **10**(12), 2748 (2022). https://doi.org/10.3390/pr10122748
135. U. Rakhman, J. Ahn, and C. Nam, "Fully automatic data collection for neuro-symbolic task planning for mobile robot navigation," in Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics, 2021, pp. 450–455. doi: https://doi.org/10.1109/SMC52423.2021.9658822.
136. Altuntas, N., Imal, E., Emanet, N., Öztürk, C.N.: Reinforcement learning-based mobile robot navigation. Turk. J. Electr. Eng. Comput. Sci. **24**(3), 1747–1767 (2016). https://doi.org/10.3906/elk-1311-129
137. A. v. Bernstein, E. v. Burnaev, and O. N. Kachan, "Reinforcement learning for computer vision and robot navigation," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2018, vol. 10935 LNAI, pp. 258–272. doi: https://doi.org/10.1007/978-3-319-96133-0_20.
138. L. Liu, D. Dugas, G. Cesari, R. Siegwart, and R. Dube, "Robot navigation in crowded environments using deep reinforcement learning," in IEEE International Conference on Intelligent Robots and Systems, Oct. 2020, pp. 5671–5677. doi: https://doi.org/10.1109/IROS45743.2020.9341540.
139. G. Chen et al., "Robot Navigation with Map-Based Deep Reinforcement Learning," in 2020 IEEE International Conference on Networking, Sensing and Control, ICNSC 2020, Oct. 2020. doi: https://doi.org/10.1109/ICNSC48988.2020.9238090.
140. H. Hase et al., "Ultrasound-guided robotic navigation with deep reinforcement learning," in IEEE International Conference on Intelligent Robots and Systems, Oct. 2020, pp. 5534–5541. doi: https://doi.org/10.1109/IROS45743.2020.9340913.
141. Dang-Nhu, R.: PLANS: neuro-symbolic program learning from videos. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H. (eds.) Advances in neural information processing systems. Curran Associates, Inc., pp. 22445–22455. Available: https://proceedings.neurips.cc/paper_files/paper/2020/file/fe131d7f5a6b38b23cc967316c13dae2-Paper.pdf (2020)
142. Silver, T., Athalye, A., Tenenbaum, J.B., Lozano-Pérez, T., Kaelbling, L.P.: Learning neuro-symbolic skills for bilevel planning. Proc. Mach. Learn. Res. **205**(CoRL), 701–714 (2023)
143. Lu, Y. et al.: Neuro-symbolic procedural planning with commonsense prompting **1**(2009), 1–34 (2022). Available: http://arxiv.org/abs/2206.02928
144. Zheng, K., et al.: JARVIS: a neuro-symbolic commonsense reasoning framework for conversational embodied agents. Available: http://arxiv.org/abs/2208.13266 (2022)
145. Zellers, R., Holtzman, A., Peters, M., Mottaghi, R., Kembhavi, A., Farhadi, A., Choi, Y.: PIGLeT: language grounding through neuro-symbolic interaction in a 3D world. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pp. 2040–2050. Association for Computational Linguistics (2021). https://doi.org/10.18653/v1/2021.acl-long.159
146. Moon, J.: Plugin framework-based neuro-symbolic grounded task planning for multi-agent system. Sensors. **21**(23), (2021). https://doi.org/10.3390/s21237896
147. Fathinezhad, F., Derhami, V., Rezaeian, M.: Supervised fuzzy reinforcement learning for robot navigation. Appl. Soft Comput. **40**, 33–41 (2016). https://doi.org/10.1016/j.asoc.2015.11.030
148. Sun, S., Zhao, X., Li, Q., Tan, M.: Inverse reinforcement learning-based time-dependent A* planner for human-aware robot navigation with local vision. Adv. Robot. **34**(13), 888–901 (2020). https://doi.org/10.1080/01691864.2020.1753569
149. Bruce, J., Suenderhauf, N., Mirowski, P., Hadsell, R., Milford, M.: One-shot reinforcement learning for robot navigation with interactive replay. In: Proceedings of the NIPS Workshop on Acting and Interacting in the Real World: Challenges in Robot Learning. Available: http://arxiv.org/abs/1711.10137 (2017)
150. M. C. Su, D. Y. Huang, C. H. Chou, and C. C. Hsieh, "A reinforcement-learning approach to robot navigation," in Conference Proceeding - IEEE International Conference on Networking, Sensing and Control, 2004, vol. 1, pp. 665–669. doi: https://doi.org/10.1109/icnsc.2004.1297519.
151. Zieliński, P., Markowska-Kaczmar, U.: 3D robotic navigation using a vision-based deep reinforcement learning model. Appl. Soft Comput. **110**, (2021). https://doi.org/10.1016/j.asoc.2021.107602
152. Shah, P., Fiser, M., Faust, A., Kew, J.C., Hakkani-Tur, D.: FollowNet: robot navigation by following natural language directions with deep reinforcement learning [Online]. Available: http://arxiv.org/abs/1805.06150 (2018)
153. Surmann, H., Jestel, C., Marchel, R., Musberg, F., Elhadj, H., Ardani, M.: Deep reinforcement learning for real autonomous mobile robot navigation in indoor environments. Available: http://arxiv.org/abs/2005.13857 (2020)
154. R. Hasani, A. Amini, M. Lechner, F. Naser, R. Grosu, and D. Rus, "Response Characterization for Auditing Cell Dynamics in Long Short-term Memory Networks," in 2019 International Joint Conference on Neural Networks (IJCNN), 2019, pp. 1–8. doi: https://doi.org/10.1109/IJCNN.2019.8851954.
155. R. Hasani, M. Lechner, A. Amini, D. Rus, and R. Grosu, "Liquid Time-Constant Networks," 35th AAAI Conf. Artif. Intell. AAAI 2021, vol. 9A, pp. 7657–7666, 2021, doi: https://doi.org/10.1609/aaai.v35i9.16936.
156. Pascanu, R., Mikolov, T., Bengio, Y.: On the difficulty of training recurrent neural networks. In: Proceedings of the 30th International Conference on Machine Learning, vol **28**(3), pp. 1310–1318. PMLR (2013). Available: http://proceedings.mlr.press/v28/pascanu13.pdf
157. Ruder, S.: An overview of gradient descent optimization algorithms, pp. 1–14. Available: http://arxiv.org/abs/1609.04747 (2016)
158. P. Coraggio and M. de Gregorio, "A Neurosymbolic Hybrid Approach for Landmark Recognition and Robot Localization."
159. Coraggio, P., De Gregorio, M., Forastiere, M.: Robot navigation based on neurosymbolic reasoning over landmarks. Int. J. Pattern Recognit. Artif. Intell. **22**(05), 1001–1014 (2008). https://doi.org/10.1142/S0218001408006624
160. Botteghi, N., Sirmacek, B., Mustafa, K.A.A., Poel, M., Stramigioli, S.: On reward shaping for mobile robot navigation: a reinforcement learning and SLAM based approach. Available: http://arxiv.org/abs/2002.04109 (2020)
161. M. Shneier, "Literature Review of Mobile Robots for Manufacturing Literature Review of Mobile Robots for Manufacturing".
162. Sharma, N., Pandey, J.K., Mondal, S.: A Review of Mobile Robots: Applications and Future Prospect. Int. J. Precis. Eng. Manuf. **24**(9), 1695–1706 (2023). https://doi.org/10.1007/s12541-023-00876-7
163. Yu, D., Yang, B., Liu, D., Wang, H., Pan, S.: A survey on neural-symbolic learning systems. Neural Netw. **166**, 105–126 (2023). https://doi.org/10.1016/j.neunet.2023.06.028
164. Yang, X., et al.: Neuro-Symbolic Computing: Advancements and Challenges in Hardware-Software Co-Design. IEEE Trans. Circuits Syst. II Express Briefs. 1 (2023). https://doi.org/10.1109/TCSII.2023.3336251
165. Hitzler, P., Eberhart, A., Ebrahimi, M., Sarker, M.K., Zhou, L.: Neuro-symbolic approaches in artificial intelligence. Natl. Sci. Rev. **9**(6), (2022). https://doi.org/10.1093/nsr/nwac035
166. Ceballos, N.D.M., Valencia, J.A., Ospina, N.L.: Quantitative performance metrics for mobile robots navigation, Ch. 24. In: Barrera, A. (ed.) Mobile Robots Navigation. IntechOpen, Rijeka. https://doi.org/10.5772/8988 (2010)

167. B. Okal and K. O. Arras, "Learning socially normative robot navigation behaviors with Bayesian inverse reinforcement learning," in Proceedings - IEEE International Conference on Robotics and Automation, Jun. 2016, vol. 2016-June, pp. 2889–2895. doi: https://doi.org/10.1109/ICRA.2016.7487452.
168. P. Hwai Ciou, Y.-T. Hsiao, Z.-Z. Wu, S.-H. Tseng, and L.-C. Fu, Composite Reinforcement Learning for Social Robot Navigation. 2018.
169. J. Wöhlke, F. Schmitt, and H. van Hoof, "Hierarchies of Planning and Reinforcement Learning for Robot Navigation," in Proceedings - IEEE International Conference on Robotics and Automation, 2021, vol. 2021-May, pp. 10682–10688. doi: https://doi.org/10.1109/ICRA48506.2021.9561151.
170. D. Vasquez, B. Okal, and K. O. Arras, "Inverse Reinforcement Learning algorithms and features for robot navigation in crowds: An experimental comparison," in IEEE International Conference on Intelligent Robots and Systems, 2014, pp. 1341–1346. doi: https://doi.org/10.1109/IROS.2014.6942731.
171. Dong, D., Chen, C., Chu, J., Tarn, T.J.: Robust quantum-inspired reinforcement learning for robot navigation. IEEE/ASME Trans. Mechatr. **17**(1), 86–97 (2012). https://doi.org/10.1109/TMECH.2010.2090896
172. Zhu, Y., Wang, Z., Chen, C., Dong, D.: Rule-Based Reinforcement Learning for Efficient Robot Navigation With Space Reduction. IEEE/ASME Trans. Mechatr. **27**(2), 846–857 (2022). https://doi.org/10.1109/TMECH.2021.3072675
173. Burattini, E., Datteri, E., Tamburrini, G.: Neuro-symbolic programs for robots. Proc. NeSy. **5**, (2005)
174. Kestler, H.A., et al.: Concurrent object identification and localization for a mobile robot. Künstliche Intell. **14**(4), 23–29 (2000)
175. K. Zhang, F. Niroui, M. Ficocelli, and G. Nejat, "Robot Navigation of Environments with Unknown Rough Terrain Using Deep Reinforcement Learning," in 2018 IEEE International Symposium on Safety, Security, and Rescue Robotics, SSRR 2018, Sep. 2018. doi: https://doi.org/10.1109/SSRR.2018.8468643.

Suaib Al Mahmud received his Bachelor of Engineering degree in Mechanical-Automotive Engineering from the International Islamic University Malaysia (IIUM) in 2021. His current work, as a research assistant at the Advanced Multi Agent System Lab, IIUM integrates Game Theory with Multi Agent Reinforcement Learning for multi-robot coordination.

Abdurrahman Kamarulariffin received a B. Eng (Hons) degree in Mechatronics Engineering from International Islamic University Malaysia (IIUM) & is currently pursuing M.Sc from the same institution. The focus of his previous works includes modelling and simulating intelligent agent for evacuation path planning using teacher-student AI-based learning framework. His current research area covers the field of improving deep reinforcement learning training convergence using fuzzy logic for autonomous mobile robot navigation.

Azhar Mohd Ibrahim is an Associate Professor at the Department of Mechatronics Engineering, International Islamic University Malaysia (IIUM). He earned his PhD from the School of Computer Sciences at Universiti Sains Malaysia (USM), specializing in collective intelligence. His research expertise encompasses computational intelligence, reinforcement learning, and multi-agent-based collective intelligence.

Ahmad Jazlan Haja Mohideen's research interests span across smart materials, renewable energy, biomedical instrumentation and signal processing, control theory, and machine learning. He completed an industrial attachment with Texas Instruments Malaysia, where he focused on enhancing traceability of legacy machines and forecasting die bond daily output. He is currently an Assistant Professor at the Mechatronics Engineering Department of the Faculty of Engineering, International Islamic University Malaysia (IIUM), a registered Professional Engineer with the Board of Engineers Malaysia (BEM) and a Member of the Institute of Engineers Malaysia (IEM) with a PhD in Control Systems Engineering from the University of Western Australia.