Received 17 March 2022; revised 13 May 2022; accepted 24 May 2022. Date of publication 6 June 2022;
date of current version 15 June 2022. The review of this paper was arranged by Associate Editor Roberto Oboe.

Digital Object Identifier 10.1109/OJIES.2022.3179617

Past, Present and Future of Path-Planning Algorithms for Mobile Robot Navigation in Dynamic Environments

H. S. HEWAWASAM (Student Member, IEEE), M. YOUSEF IBRAHIM, AND GAYAN KAHANDAWA APPUHAMILLAGE (Member, IEEE)

School of Engineering, Information Technology and Physical Sciences, Federation University Australia, Churchill, VIC 3842, Australia

CORRESPONDING AUTHOR: H. S. HEWAWASAM (e-mail: h.hewawasam@federation.edu.au).

ABSTRACT Mobile robots have been making a significant contribution to the advancement of many sectors including automation of mining, space, surveillance, military, health, agriculture and many more. Safe and efficient navigation is a fundamental requirement of mobile robots, thus, the demand for advanced algorithms rapidly increased. Mobile robot navigation encompasses the following four requirements: perception, localization, path-planning and motion control. Among those, path-planning is a vital part of a fast, secure operation. During the last couple of decades, many path-planning algorithms were developed. Despite most of the mobile robot applications being in dynamic environments, the number of algorithms capable of navigating robots in dynamic environments is limited. This paper presents a qualitative comparative study of the up-to-date mobile robot path-planning methods capable of navigating robots in dynamic environments. The paper discusses both classical and heuristic methods including artificial potential field, genetic algorithm, fuzzy logic, neural networks, artificial bee colony, particle swarm optimization, bacterial foraging optimization, ant-colony and Agoraphilic algorithm. The general advantages and disadvantages of each method are discussed. Furthermore, the commonly used state-of-the-art methods are critically analyzed based on six performance criteria: algorithm's ability to navigate in dynamically cluttered areas, moving goal hunting ability, object tracking ability, object path prediction ability, incorporating the obstacle velocity in the decision, validation by simulation and experimentation. This investigation benefits researchers in choosing suitable path-planning methods for different applications as well as identifying gaps in this field.

INDEX TERMS Dynamic environment, mobile robot, navigation, obstacle avoidance, path-planning.

I. INTRODUCTION

Mobile robots play an important role in many sectors such as mining [1], surveillance [2], space [3], military [4] and agriculture [5] in the contemporary world. They are used in many applications to improve the efficiency of the processes, increase the accuracy, and to reduce the risk for humans. Most of the time, mobile robots' environment is unknown and dynamic. Therefore, mobile robot navigation is a complicated task. This has sparked the interest of many researchers in developing methodologies to address the mobile robot navigation problem. The primary task in navigation is either to reach a predetermined goal or to follow a predetermined path without any collisions.

Autonomous navigation is subdivided into four main subtasks [6] (see Fig. 1):

1) The sensory system captures the robot's surrounding environment (Perception).
2) Identification of robot's location in the environment (localisation).
3) The robot decides how to manoeuvre to reach the goal without collision (path-planning).
4) The robot's motions are controlled to follow the desired path (motion control).

Among the above four tasks, path-planning is one of the most important areas that are the focus of this research. This paper discusses the different up-to-date types of robot

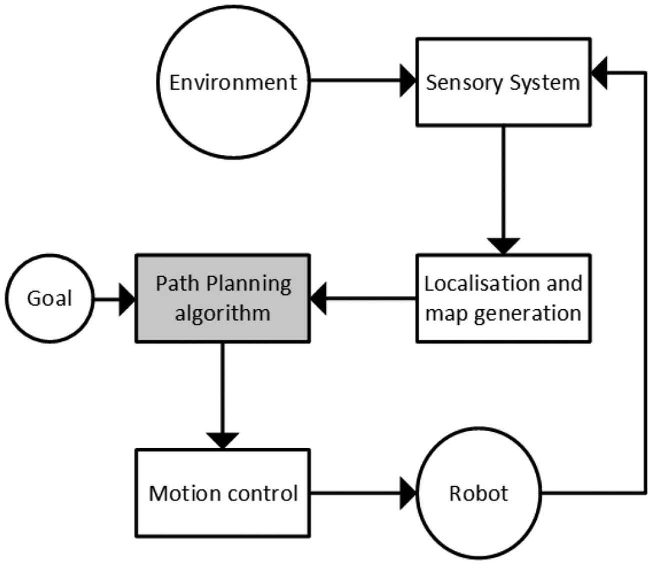

FIGURE 1. Basic steps needed for mobile robot navigation.

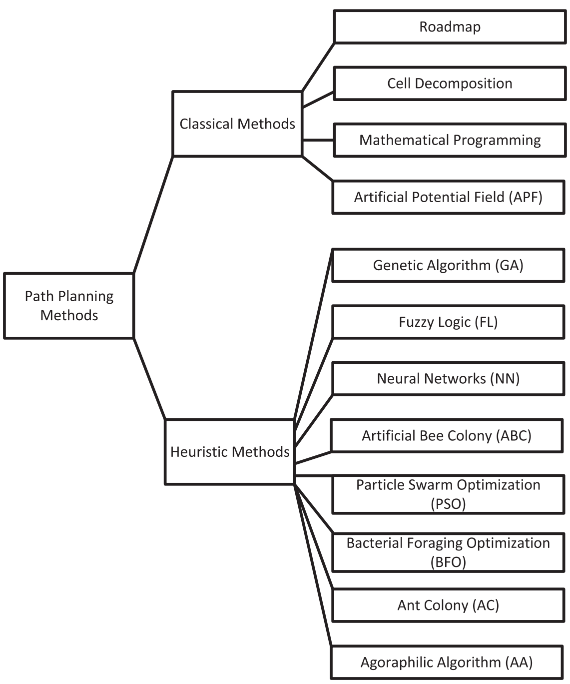

FIGURE 2. Basic path-planning methods.

path-planning methods used for navigating mobile robots in dynamic environments.

In general, the path-planning methodologies are classified into two main groups: classical approaches and heuristic approaches. In the literature, the roadmap approach, cell decomposition, mathematical programming and the Artificial Potential Field (APF) method can be identified as frequently used classical path-planning methods (see Fig. 2).

However, most of the classical methods, except the APF method, failed to handle the high uncertainties in dynamic environments. Therefore, the APF method is the only classical method that became famous among researchers developing navigation algorithms suitable for dynamic environments [7].

Conversely, heuristic methods such as genetic algorithm (GA), Fuzzy Logic (FL), Artificial Neural Networks (ANNs), Artificial Bee Colony (ABC), Particle Swarm Optimisation (PSO), Bacterial Foraging Optimisation (BFO), Ant Colony Optimisation (ACO) and Agoraphilic Algorithm (AA) are becoming famous among the researchers in the field (see Fig. 2).

There have been many surveys in the field of mobile robot navigation [8]–[12]. However, they were mostly limited to the static environments or are insufficient with an in-depth analysis of current algorithms that are capable of navigating robots, specifically, in dynamic environments. The review study conducted by Patel *et al.* [13] was focussed mainly on static environment. Also, the reviews presented in [10] and [11] are limited to the reinforcement learning-based navigation algorithms. Pandey *et al.* [12] have discussed algorithms capable of navigating robots in dynamic environments. However, it has not provided any comparative or qualitative analysis of algorithms.

This study was conducted to analyse the major aspects of the mobile robot navigation methods in dynamic environments with classifications of the diverse approaches that characterizes recent trends with their advantages and disadvantages. Furthermore, in this study, we introduced a set of key performance criteria that should be included to indicate the level of success of an Algorithm's capability of navigating an autonomous agent in an unknown dynamic environment. The algorithms discussed in this paper are analysed and compared based on those criteria.

The remainder of the paper is arranged in the following order: Section II discusses popular navigation techniques capable of navigating robots in dynamic environments. Section III presents an in-depth comparative analysis and discussion of the current state-of-the art techniques. The final section presents the conclusions and remarks of the study.

II. POPULAR NAVIGATION TECHNIQUES CAPABLE OF NAVIGATING ROBOTS IN DYNAMIC ENVIRONMENTS

A. ARTIFICIAL POTENTIAL FIELD

In the APF model, the robot is considered a point mass while the goal and obstacles are modelled as force fields. The goal creates an attractive force, and the obstacles create repulsive forces on the robot. Those forces push and pull the robot towards the goal while avoiding obstacles. The direction and the magnitude of the resultant force vector denote the direction of the robot's velocity vector and magnitude, respectively. The original research focused only on static obstacles.

However, the concept was also extended to dynamic environments. In a dynamic environment, the algorithm assumes that there is only one obstacle close to the robot. The possibility of colliding with the obstacle is calculated by a function of the distance between the robot and the obstacle and the relative velocity of the obstacle with respect to the robot [14].

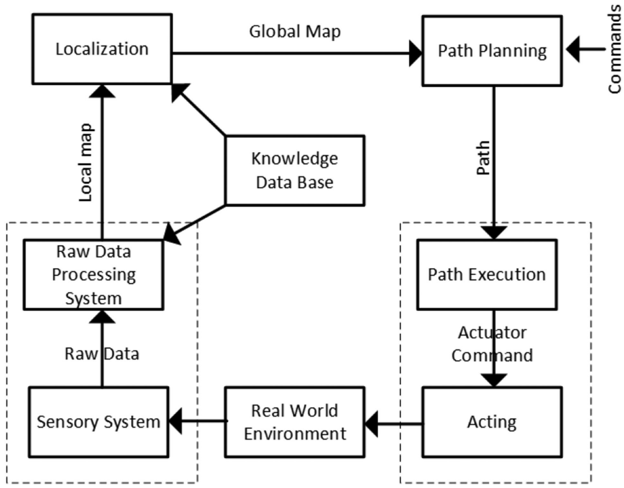

FIGURE 3. Main steps of an improved APF-based algorithm [14].

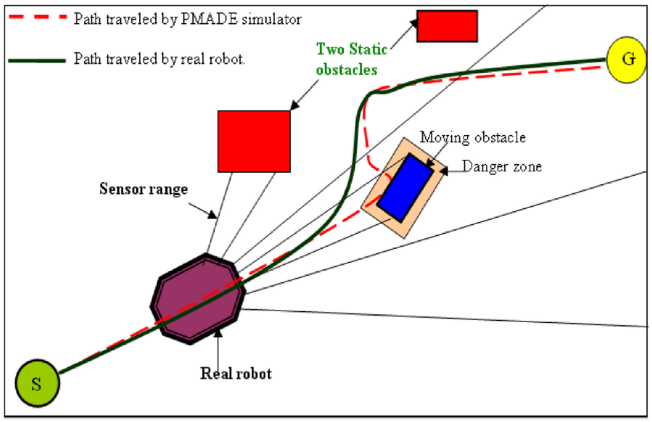

FIGURE 4. Experimental test results from an improved APF-based algorithm [14].

When the distance between the robot and the obstacle increases and the relative velocity decreases, the avoidability measure increases. Nak *et al.* [14] used a virtual distance function as the avoidability measure, which emphasises the distance metric over the speed. The algorithm can make the robot avoid obstacles that are closer or further away by tuning this function. A potential force is generated accordingly to match with the original potential field method. Numerous studies have used this concept to navigate robots in dynamic environments. Further, some studies have used this concept when the goal is moving.

A path-planning algorithm that is capable of navigating robots in static and dynamic environments is introduced in [15]. This algorithm uses a potential field-based method for path-planning. The main steps of that algorithm are shown in Fig. 3. Also, that algorithm is experimentally tested using a research robot platform (see Fig. 4).

A modified APF-based navigation method (APF-elaborated resistance approach) was proposed in [16]. In this method, different potential functions are assigned to different obstacles and road boundaries. The drivable area is meshed, and resistance values are added based on the corresponding potential functions. The collision-free paths are found using a local current comparison method. In this method, the motion planning process is divided into virtual and actual spaces.

1) VIRTUAL SPACE

In the virtual space, the robot's trajectory is predicted and executed step by step over a short horizon with the robot's current speed. The predicted trajectory is analysed, and the robot's speed is decided. Based on the analysis, the robot may change the speed. Then the decided speed is sent to the actual space.

2) ACTUAL SPACE

In actual space, a 'CarSim' model is used to track the planned path. This model has been developed based on the predictive controller model, and it was reported [16] that this model has been experimentally validated.

Another improved APF-based path-planning methodology for autonomous vehicle navigation was proposed by Wang *et al.* [17]. A safety model was also used in the proposed methodology. The safety model was developed based on an analysis of human drivers' path-planning behaviours. As reported in [17], this safety model is used to improve the repulsive forces generated by the obstacles. Those improved repulsive forces are eventually used in the potential field function. Further, a collision-free path is generated based on the improved APF. The presented simulation and experimental results in [17] show the ability to navigate vehicles in real time using the proposed method. Further, it was reported in [17] that the approach optimises the generated path according to driver habits, which is helpful in improving the riding comfort of autonomous vehicles.

B. GENETIC ALGORITHM

GA is an optimisation methodology that is commonly used to generate solutions for combined optimisation and search problems. This method follows the basic principles of genetic and natural selection. Applications of GA were mainly focused on the field of computer science. However, GA-based methods are also used in the field of mobile robot navigation. The GA starts without any knowledge of the correct solution and depends completely on the responses of the environment and the evolutionary operators to arrive at the best solution [18] (see Fig. 5).

Patle *et al.* [19] discussed a modified GA for mobile robot navigation in static and dynamic environments [19]. The proposed method of the Matrix-Binary Codes based Genetic Algorithm (MGA) is also identified as an optimisation tool for searching the best-fit path for mobile robot navigation. This method transforms any unknown robot's surrounding into an array. Subsequently, the algorithm develops a path between the robot and the goal.

In the MGA, a matrix trace-based controller is used to arrange the operation throughout the navigation process. The

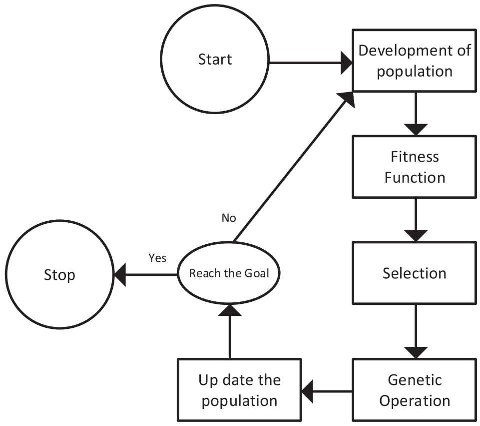

FIGURE 5. **Process flowchart of a basic genetic algorithm.**

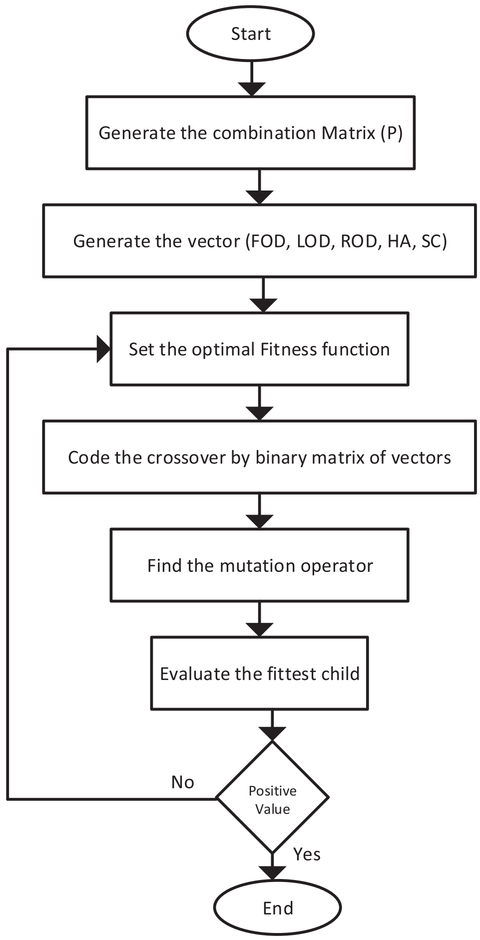

FIGURE 6. **Main steps of MGA.**

GA is used to search for the goal by avoiding the obstacles. The used searching mechanism is a nonlinear iterative method. The algorithm takes three sensory inputs: i) distance to the left obstacle, ii) distance to the right obstacle and, iii) distance to the front obstacle. Based on the input data the algorithm creates the heading angle as its output. The main steps of the MGA are shown in Fig. 6.

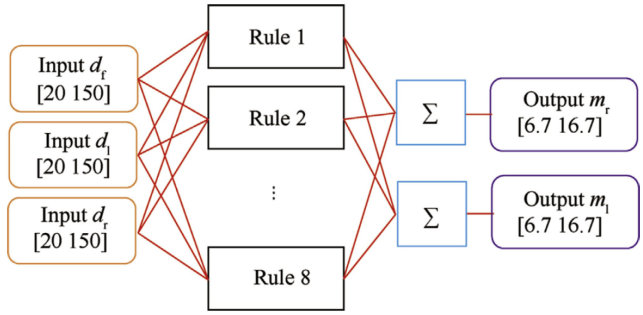

FIGURE 7. **The structure of a proposed FL controller [22].**

Also, a different path-planning algorithm using a knowl edge-based GA, for navigating mobile robots in dynamic environments was proposed in [20]. In this method, a problem-specific GA was used instead of a standard one. Further, in this algorithm, domain knowledge is attached to specialised operators. This allows the proposed method to handle multiple robots in dynamic environments. When the robot's environment is relatively simple, or the environment does not change rapidly, the algorithm evolves a near-optimal path. Further, the authors reported that a feasible path could be obtained in a dynamic environment.

C. *FUZZY LOGIC*

FLCs are either a rule- or knowledge-based system. Those systems comprise a set of fuzzy rules. These fuzzy rules are generated based on the domain knowledge or human experts. The simplicity of rule-based systems, the low computational cost, and the capability to perform a wide variety of tasks make FL-based methods quite popular among researchers [21]. FL is a very versatile AI-based technique for mobile robot navigation, with the ability to represent the fuzzy rules using linguistic terms. Further, FL systems are identified as reliable decision-making systems under high uncertainty and with incomplete information.

Pandey *et al.* [22] proposed a FL-based system to navigate robots in unknown dynamic environments. In this system a singleton type-1 FL controller and a Fuzzy-Wind Driven Optimization (WDO) algorithm were used. The Fuzzy-WDO algorithm was used to optimise the input and output membership functions of the FL controller. The motion behaviour of tiny air parcels over an N-dimensional search domain was used in the fundamental concept for WDO. The Type-1 FL controller plays the main role of avoiding collisions and navigating the robot in static and dynamic environments. The controller takes sensory data as its input and generates two output signals to control the left and right motors of the robot. The three sensory data inputs are distance to the i) front obstacle, ii) left obstacle and iii) right obstacle. The inputs are logically connected to the outputs through eight fuzzy rules,(see Fig. 7).

Also, in [23] a wireless FL control system was proposed to navigate wheeled robot in a warehouse environment. In this

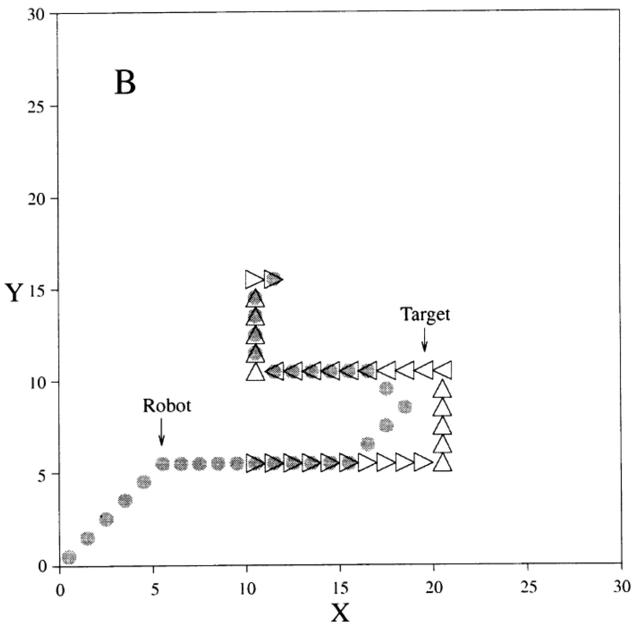

FIGURE 8. Hunting a moving goal by the ANN-based algorithm proposed by Simon et.al. [24].

approach two fuzzy logic controllers were used to navigate the robot from the start point to the goal.

D. ARTIFICIAL NEURAL NETWORKS

The ANN techniques were initially used for computer science-based work such as image processing and pattern recognition. ANNs are inspired by the mechanism of the human brain. ANNs are complicated networks of artificial neurons, commonly known as nodes. These networks are capable of solving AI based problems. ANNs can handle high levels of uncertainties that exist in dynamic environments. There have been number of attempts of developing ANN-based navigation algorithms capable of navigating robots in dynamic environments.

Simon *et al.* [24] proposed an ANN-based, real-time path-planning method to navigate robots in dynamic environments [24]. The authors reported that the algorithm does not use an explicit optimization of a global cost function in static environments. Furthermore, the presented test results in [24] show the algorithm's capability of navigating robots in dynamic environments with a moving goal, Fig. 8.

The ANN-based algorithm proposed by Engedy *et al.* in [25] is also capable of navigating robots in dynamic environments. This uses ANNs to control the motion of a car type robot on 2D environment. The method uses an extended Back Propagation Through Time (BPTT) algorithm to train the ANN. This training process is identified as an online training method. The training algorithm uses a potential field-based obstacle avoidance method to train the ANN. The trained ANNs are also capable of predicting the paths of moving

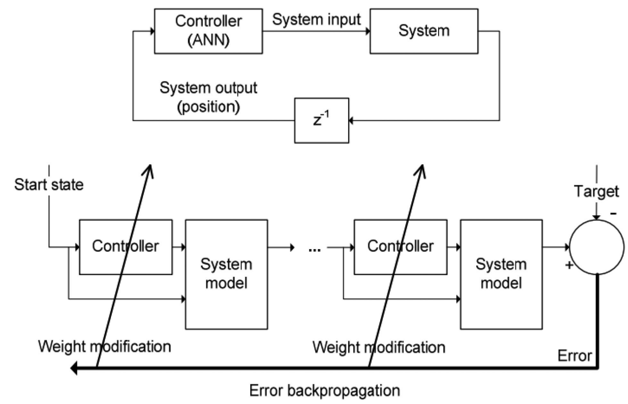

FIGURE 9. BPTT system used for training in an iteration [25].

obstacles. This prediction system has improved the performance of the algorithm. The proposed BPTT training system is described in Fig. 9.

E. ARTIFICIAL BEE COLONY

The ABC algorithm was introduced by Dervis Karaboge in 2005. This is a swarm-based search algorithm that adapts the foraging behaviour of honeybees. There are three main elements involved in Bees' cooperative aptitude collection. They are,

1) A food sources.
2) Worker bees.
3) Non-worker bees.

In the ABC algorithm, artificial bees are working in one half of the colony. The other half of the colony consists of onlookers. In the ABC algorithm, there is an employed bee for each food source. The employed bee whose food source is exhausted by the employed and onlooker bees become scouts. A possible solution of the optimization problem is optimized by the position of a food source [26], [27]. The fitness of the associated solution is represented by the viability of a food source. The ABC algorithm's flowchart is shown in Fig. 10.

Faridi *et al.* [28] developed a navigation algorithm based on the ABC technique. This algorithm is capable of solving the multi-agent, multi-target navigation problem in an unknown dynamic environment. Liang *et al.* [29] also developed an online multiple mobile robots navigation system based on the ABC method.

F. PARTICLE SWARM OPTIMIZATION

PSO is a population-based optimisation technique that adopts the motion of schooling fish and bird flocks. The PSO technique was introduced in 1995 by Eberhart *et al.* [30]. It has some similarities to other evolutionary computation techniques. The process starts with a population of random solutions and seeks the optimal solution. In PSO, potential solutions are known as particles. In the process, a set of current optimum particles is identified. The other partials follow the current optimal partials. PSO's high efficiency in

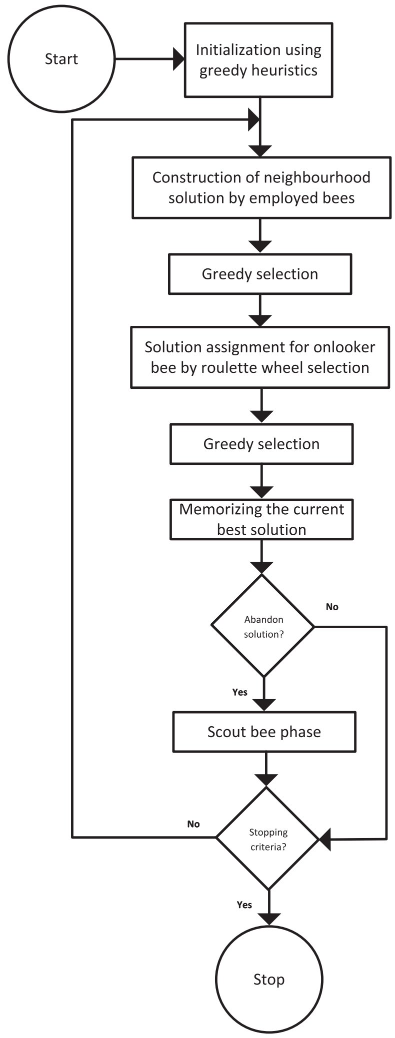

FIGURE 10. Flowchart of original ABC algorithm.

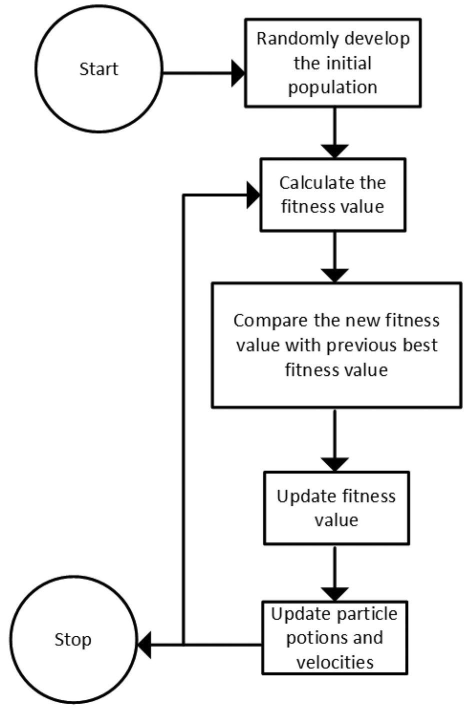

FIGURE 11. Steps of original PSO.

terms of speed and memory requirements inspired most of the researchers to adapt this technique to use for mobile robot navigation in dynamic environments. The decision-making process of the algorithm is shown in Fig. 11.

Wang *et al.* [31] proposed a path-planning algorithm based on PSO for a soccer robot. This algorithm takes the shape of the robot and obstacles into account when generating the path planner. In this method a fitness function is defined with a simple coding scheme.

In this approach, the robot moves in 2D restricted workspace. When considering a particular robot all other robots are considered as moving obstacles. A safety boundary is applied to each of the obstacles to make ensure the planned path is safe. With the safety boundary, the robot can be assumed to be a particle in the problem space. The path from the current location of the robot (S) to the goal (G) is generated via -points defied as (P_1, P_2, ..., P_m). The robot's shortest and safe path to the goal is defined as P= (S, p_j,...,p_m, G) where, P_j is a non-obstacle point and there is no obstacle between P_j and P_{j+1} and its neighbours.

This path-planning algorithm (finding P_j locations and corresponding velocities) can be described in five main steps.

S1: The robot's and goal's coordinates are calculated according to a transform formulation in a 2D coordinate system.

S2: Based on the constrains of position and velocity particle Xi is initialized.

S3: Based on the comprehensive fitness function the fitness values of particles are generated.

S4: Local optimal position (P) and goal optimal position (P_g) is found.

S5: Velocity and position of the particle are updated accordingly.

S6: Go back to S3 and iterate until the stop criteria is achieved.

Rath *et al.* [32] also proposed a path-planning algorithm for mobile robots in unknown dynamic environments using the PSO technique. Similar to most other navigation algorithms, this one also tries to find the shortest path to the target while avoiding obstacles. An objective function is used in this approach in developing the path-planning algorithm. The objective function is optimised to determine the best global particle among the swarm. This allows the robot to move towards the global best position. The robot keeps moving towards the goal via these intermediate positions.

G. BACTERIAL FORAGING OPTIMIZATION

The BFO algorithm was introduced by Passino in 2002 [8]. This optimisation algorithm adapts the behaviour of Escherichia coli bacteria. E. coli is a cell body with eight to

10 rotating flagella attached to the body randomly. E. coli bacteria move in their surrounding environment by performing two main actions: 'run' and 'tumble'. The bacteria run or tumble by rotating their flagella in an anticlockwise or clockwise direction. In addition to motion, E. coli bacteria propagate to renew their colony and eliminate the weaker or older individuals. There are three main parts of the behaviour of the bacteria.

1) *Chemotaxis:* Bacteria use flagellation to move.
2) *Reproduction:* Reproduction is a way of renewing the colony, as well as removing the inefficient individuals. The efficient individuals are used for reproduction. Reproduction is also an optimisation behaviour of BFO.
3) *Elimination and dispersal:* One major variation in the environment is the reduction of food concentration. At certain reproduction rounds in the bacterial lifecycle, the diffusion processes take place [33].

Hossain *et al.* [9] developed an algorithm based on the BFO technique. This algorithm is also capable of navigating robots in dynamic environments. The algorithm uses randomly distributed particles around the robot to find the path towards the goal while avoiding obstacles. It was reported that the best particles are selected based on two factors:

1) distance to the goal.
2) the particle's gaussian cost function.

Then, a high-level decision strategy is used for the selection and consequently proceeds for the result. It was reported that this method does not generate a map for navigation. It only uses a robot sensory system to capture the robots surrounding. It was reported that this method can be implemented without a requirement for tuning algorithm and complex calculations. The model flowchart of this algorithm is shown in Fig. 12

H. ANT COLONY OPTIMIZATION

ACO is used to solve the combinatorial optimisation problem. The population-based ACO algorithm adopts the behaviour of ants. Ants are skilled to discover the shortest path to their food source from their nest [13]. The ACO algorithm finds the solutions to the Travelling Salesman Problem (TSP) using agents called ants. Ants use a special pheromone, which they deposit on the edge of the TSP graph to communicate with others [34]. This optimisation algorithm is also used as the base algorithm to develop navigation algorithms capable of navigating robots in dynamic environments.

Zheng *et al.* [35] proposed a path-planning algorithm capable of navigating robots under dynamic environments based on ACO. The proposed method has three main steps.

Step 1: Develop a free-space model for the mobile robot using the 'MAKLINK' graph theory.
Step 2: Using Dijkstra's algorithm, a sub-optimal path with no collisions is found.
Step 3: Generate the globally optimal path by optimising the locations of sub-optimal paths using the ACO algorithm.

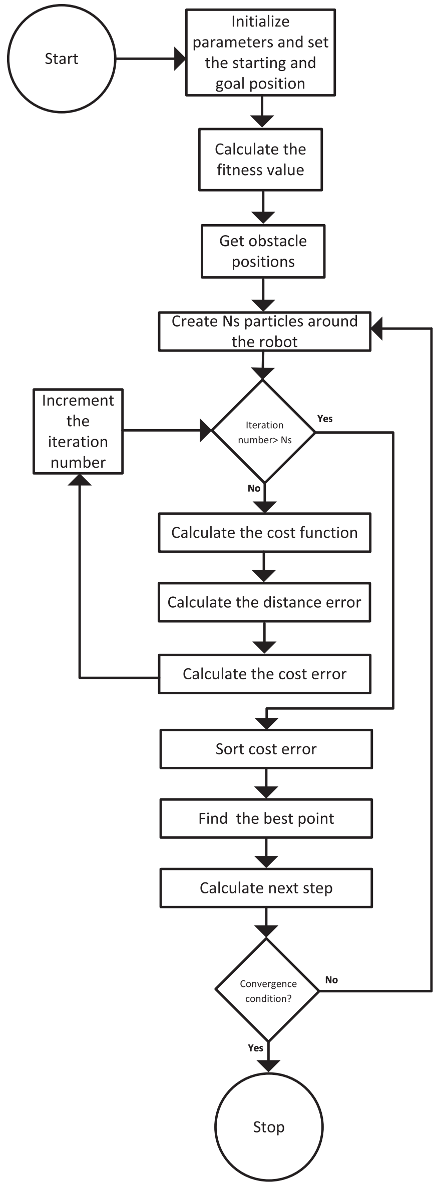

FIGURE 12. Flowchart of a BFO-based algorithm.

Another navigation algorithm using ACO and capable of navigating mobile robots under dynamic environments is discussed in [36]. In this work, the standard ACO algorithm was modified by introducing age to ants (agents). This improved method is called aging-based ACO. This modified ACO is combined with grid-based modelling to develop the path-planning algorithm. The process of the proposed algorithm is shown in Fig. 13.

I. AGORAPHILIC ALGORITHM

AA introduces the new free-space attraction concept. This algorithm does not look for obstacles to avoid but rather look for free space to follow, because of this nature Agoraphilic algorithm is considered as an optimistic navigation algorithm [37]. In this algorithm four main module can be identified.

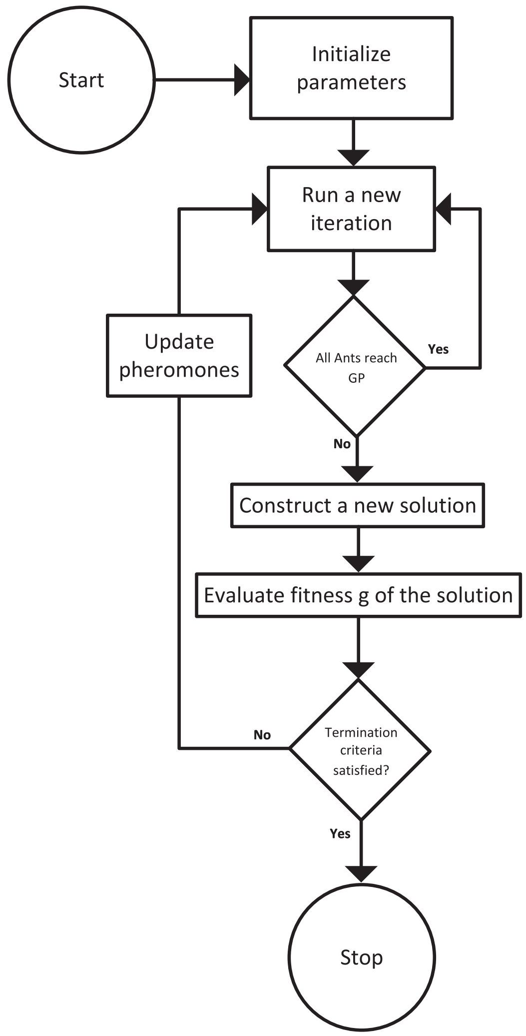

FIGURE 13. **Flowchart of an ACO-based algorithm.**

1) *Free-Space Histogram (FSH) generation module:* This module converts the robots surrounding environment into a histogram based on the distance to neighboring objects.
2) *Free-Space Force (FSF) generation module:* Based on the generated free-space histogram this module crates a corresponding force vector (sector force) to each sector of the histogram. This force vector is directly related to the amount of free-space in that section. A group of sector forces are call as the corresponding free-space forces for a particular iteration.
3) *Force shaping module:* In navigation problem robot needs to be navigated to the goal in a safe root. The task of the force shaping module is to forces the free-space forces towards the goal. This process is conducted by a weighting system.
4) *Driving force generation module:* This module generates the robots driving force based on the shaped forces. This driving force pulls the robot towards the goal through free-space. In each iteration a new driving force is generated.

The Agoraphilic algorithm was initially designed only for static environments. However, the free-space attraction concept (Agoraphilic) is used to develop new navigation algorithms capable of navigating robots in dynamic environments.

Navigation algorithms capable of navigating robots in dynamic environments are proposed in [38]–[40]. The algorithm presented in [40] uses modified free-space attraction concept. As shown in Fig. 14, this algorithm consists of seven main modules: object tracking module, prediction module, FSH generation module, FSF generation module, force shaping module, instantaneous driving force component generation module and instantaneous driving force component weighing module. As shown in Fig. 14 these modules are repeatedly used to generate the robot's driving force based on the robot's current environment (Current Global Map (CGM)) and the predicted future environments also known as Future Global Maps (FGM).

III. DISCUSSION

Based on the methodologies presented in this study, it was apparent that navigation algorithms can be classified as classical and heuristic approaches. There are a number of classical methods such as APF, cell decomposition, the roadmap approach and mathematical programming. Those methods were popular in the early days in developing navigation algorithms. The classical methods are simple, have a low computational cost and are easy to implement. Therefore, they were chosen for many real-time path-planning applications. Also, classical methods have obtained good results, especially in static environments. However, most of these methods couldn't handle the high uncertainties in dynamic environments (see Fig. 15).

Conversely, heuristic methods are considered more intelligent and effective compared to classical methods as they can adapt to the uncertainty of constantly changing environments. Consequently, heuristic-based methods are used in most navigation algorithms for robots in dynamic environments. However, most of the heuristic approaches need a learning phase with high computational cost (see Table 1)

Table 2 provides a detailed analysis of the algorithms used to date for navigating robots in dynamic environments. The analysis was based on some key parameters, such as the basic concept used, the ability to navigate through multiple moving objects, the ability to track a moving goal and the ability to track and estimate the states of moving objects. The analysis also considered the algorithms that considered the velocities of moving objects in decision-making and the ability of the algorithm to predict the future environments of the robot and included validation of the algorithm via simulation and experiments. This analysis showed the number of classical methods suitable for dynamic environments is extremely low compared to the heuristic methods that are suitable (see Fig. 16).

According to this study (study was conducted using 62 papers with 75% papers in last ten years), among the popular algorithms in the field of navigation, APF is the only

TABLE 1. General Advantages and Disadvantages of Base Algorithms

Base-Algorithm	Advantages	Disadvantages
Artificial Potential Field	1. Less computational cost 2. Simple to implement 3. High adaptability	1. Tedious parameter tuning 2. Local minima problem 3. Oscillations 4. Inability to reach a goal when a large goal is nearby
Genetic Algorithm	1. Generate high quality accurate solutions	1. Can develop oscillations 2. Hard to implement in dynamic environments
Fuzzy Logic	1. Capable of representing the human thinking. 2. Can mimic the actions of a manual robot operator. 3. Fuzzy rules are transparent and clear. (As using the linguistic variables) 4. Can handle high uncertainties	1. Selecting right rules and membership functions are critical
Artificial Neural Networks	1. Nonlinear mapping capability 2. Capable of parallel processing 3. Learning ability 4. System can be retrained when the conditions are changed	1. Needs to go through a huge learning process 2. Computational cost is high 3. Impossible to come up with a transfer function
Artificial Bee Colony	1. Simplicity, flexibility and robustness 2. Ability to explore local solutions	1. Slow when in sequential processing 2. Higher number of objective function evaluation
Particle Swarm Optimization	1. Less computational complexity 2. PSO's high efficiency in terms of speed and memory requirements	1. Possibilities of getting tarped in local minimums in complex environments 2. Less accurate and practical than GA
Bacterial Foraging Optimization	1. Ability to solve the multi difficulty scheduling problem effectively	1. High computational cost minimums in complex environments
Ant Colony	1. Require less control parameters	1. Slow convergence
Agoraphilic	1. Take the optimistic approach of obstacle avoidance 2. No local minima problem 3. Less computational cost	1. Needs parameter tunning for tracking

TABLE 2. Comparison of Published Algorithms Based on Identified Key Parameters

Base-methodology	Algorithms can navigate with multiple moving objects	Algorithms capable of hunting a moving goal	Algorithms use a tracking method to track moving obstacles	Algorithms use a prediction method to predict the motion of moving obstacles	Algorithms use velocity/ relative velocities of moving obstacles in decision making	Algorithms validated by simulation tests	Algorithms validated by experimental tests
APF	[7], [15], [17], [42], [43], [44]	[42], [43]	[7], [17]	[17]	[7], [17], [42], [43]	[7], [15], [17], [42], [43], [44]	[7], [15], [17]
GA	[19], [20], [45]					[19], [20], [45]	[19]
FL	[46], [47], [23], [43], [48], [49], [50], [51]				[47]	[46], [47], [23], [43], [48], [49], [50], [51]	[23], [48], [49], [51]
ANN	[52], [53], [24], [25]	[24], [53]	[52]	[25]	[52], [25], [53]	[52], [24], [25], [53]	[52], [25]
ABC	[28], [29], [54]				[29], [54]	[28], [29], [54]	
PSO	[46], [31], [32]					[46], [31], [32]	
BF	[9], [44]					[9], [44]	
ACO	[36], [47], [35]					[36], [47], [35]	
AA	[38], [40], [55], [39]	[55]	[38], [40], [55], [39]	[40], [39], [55]	[39], [40], [55]	[38], [40], [55], [39]	[40]
Others	[41], [56], [57], [58], [59], [60]	[41], [57], [58], [60]	[41], [57]	[41], [56], [57], [59]	[41], [56], [57], [59]	[41], [56], [57], [58], [59], [60]	[57], [56], [60]

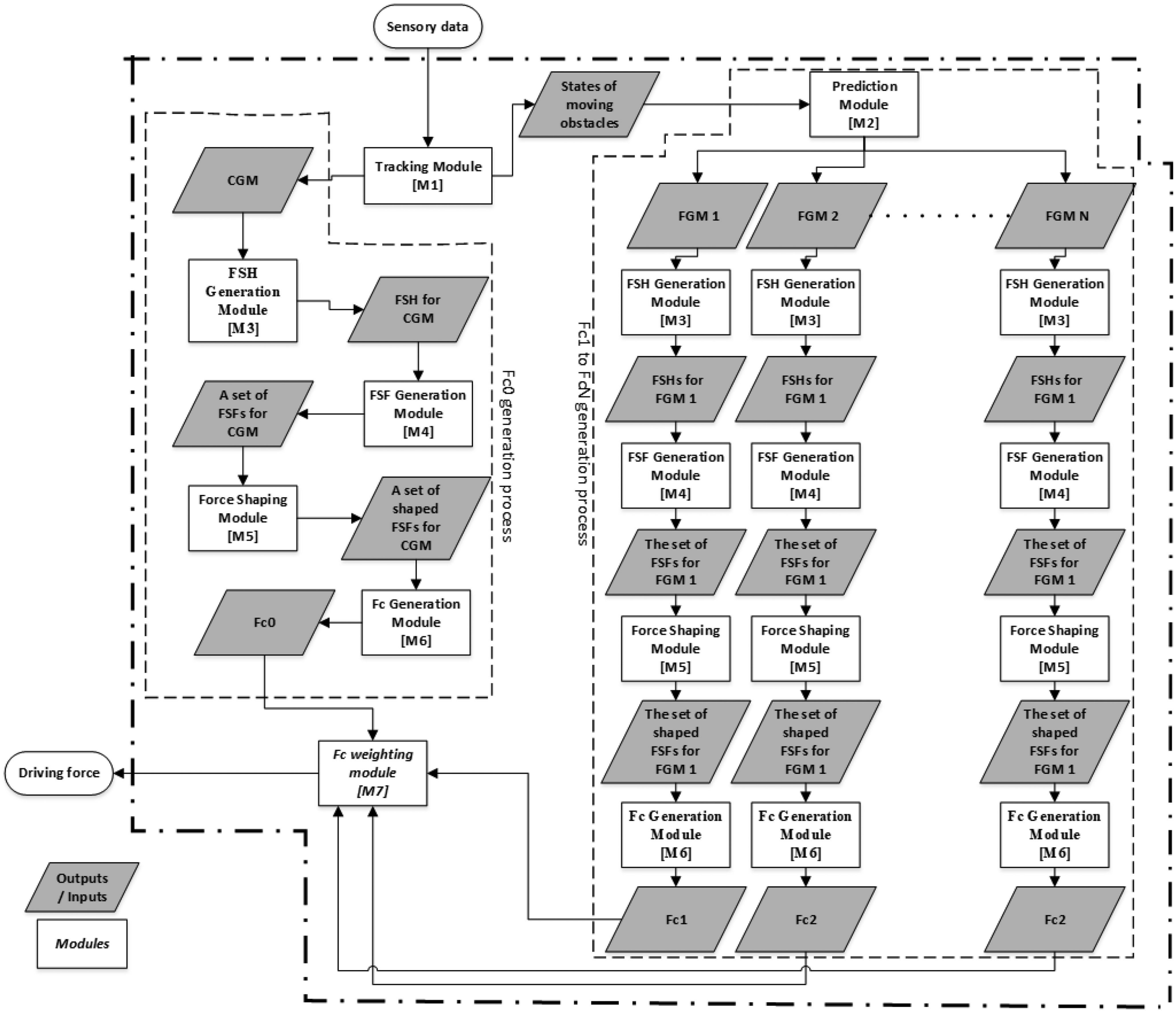

FIGURE 14. **Schematic diagram of Robot's driving force generation of the Agoraphilic Algorithm [40].**

classical method used for developing navigation algorithms for dynamic environments. Among the considered algorithms, 19% use FL, 7% use GA and 15% use the APF algorithm as the base algorithm (see Fig. 15). It was also found that there has also been a considerable number (12%) of successful navigation algorithms developed using other methods such as Gaussian Process Motion Control, improved Lyapunov Guidance Vector Field, A* Algorithm and D* Algorithm.

It is important to take the velocity of the moving objects into account in the decision-making for navigation in dynamic environments [41]. However, there are only a few algorithms that incorporate the velocity information for decision-making (see Fig. 17). This is a major drawback of most of the algorithms. Most of the existing sensory systems do not give velocity information; they only give distance information. This is one of the main reasons for not incorporating velocity information in navigation. Some algorithms have estimated the velocities by simply finding the rate of change of displacement based only on distance information. In some studies, it has been argued that noisy sensory information can give inaccurate velocity estimations. A feasible solution for this issue would be fusing a low computational cost tracking system to estimate the velocities of moving objects with the basic navigation algorithms.

Only a few algorithms use a prediction system in unknown dynamic environments, Table 2. A short-term prediction system could improve the efficiency of navigation algorithms. Further, a prediction system can help the robot navigate in critical conditions with multiple MOs challenging the robot at the same time [39]. However, there are few algorithms that can perform the navigation task successfully when there is a moving goal (see Fig. 18).

Almost all the algorithms have been verified using simulation tests (see Fig. 19). However, only 37% of the navigation

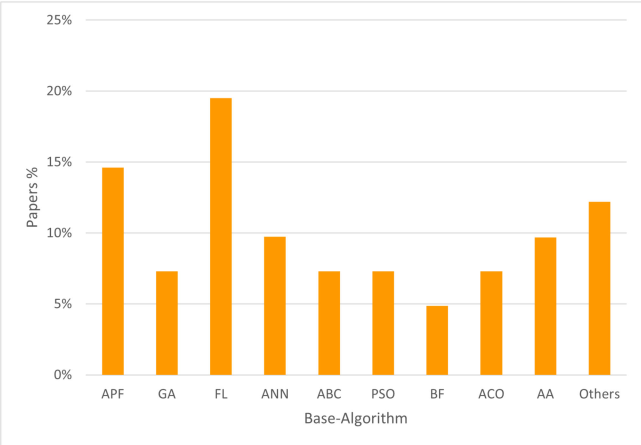

FIGURE 15. Number of published papers per base methodology as a percentage (study was conducted using a sample of 62 papers with 70% papers in last ten years).

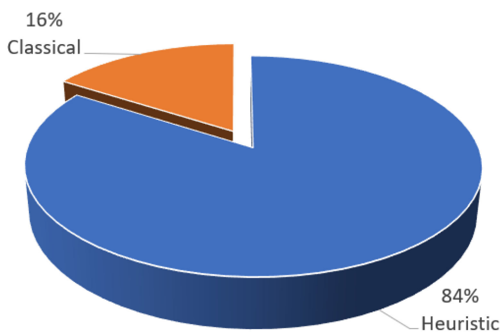

FIGURE 16. Statistics of classical and heuristic approaches used in dynamic environments based on published papers (study was conducted using a sample of 62 papers with 70% papers in last ten years).

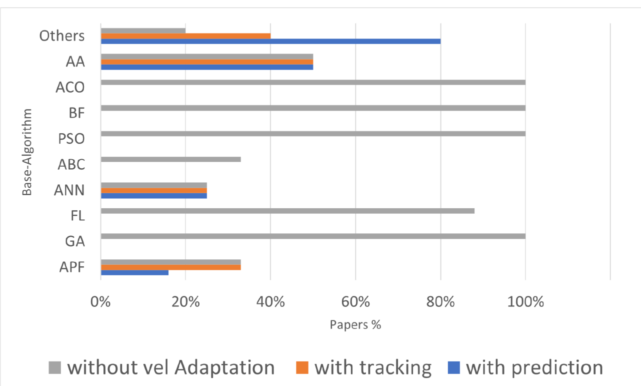

FIGURE 17. Navigation techniques based on velocity adaptation and moving object prediction.

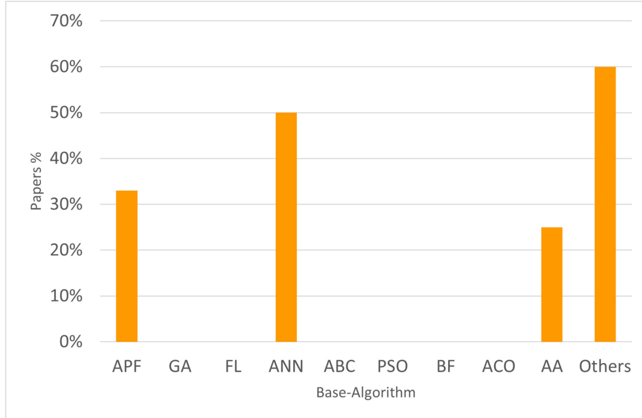

FIGURE 18. Percentage of state-of-the-art algorithms of each base methodology that are capable of hutting a moving goal in a dynamic environment.

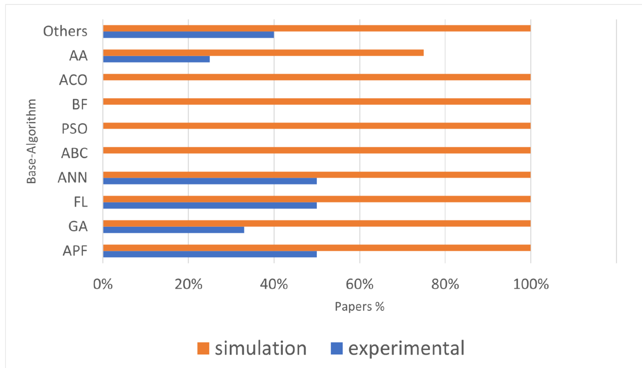

FIGURE 19. Methods used for validation (simulations and experiments) of state-of-the-art algorithms of each base methodology as a percentage.

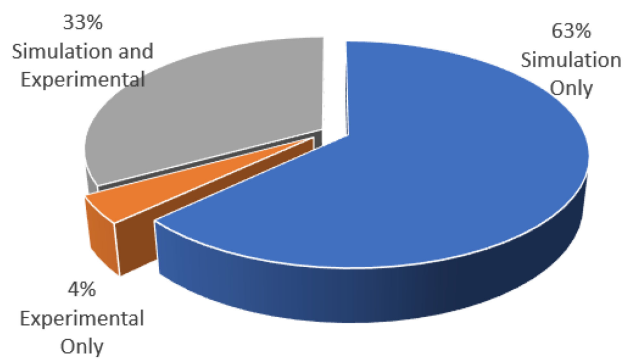

FIGURE 20. Methods used for validation of state-of-the-art algorithms in general.

algorithms surveyed have been validated via real experiments (see Fig. 20). Moreover, very few algorithms have been tested for their behaviour when multiple moving objects challenge the robot at the same time and for their decision-making when the robot is heading to a trap.

IV. CONCLUSION

This paper presented a comprehensive summary of the up-to-date methodologies attempted for mobile robot navigation in dynamic environments. The discussed methods were divided into two main groups i) classical methods and ii) heuristic methods. Path-planning approaches such as Artificial Potential Field (APF), Genetic Algorithm (GA), Fuzzy Logic (FL), Artificial Neural Networks (ANN), Artificial Bee Colony

(ABC), Particle Swarm Optimization (PSO), Bacterial Foraging Optimization (BFO), Ant Colony Optimization (ACO) and Agoraphilic Algorithm (AA) are considered for this study. This paper also provides the pros and cons of each of these methods. Furthermore, this study discussed the different path-planning algorithms developed based on the above- mentioned methods. A survey was conducted on path-planning algorithms based on identified key features. This investigation helps to understand the common drawbacks of existing algorithms and the research gaps in the field of interest.

In general, it was noticed that many algorithms do not have a proper method to adapt to the velocities of moving objects in the navigation decision making process. Also, the study has shown there is only a few navigation algorithms which are capable of navigating robots in dynamic environments with a moving goal. Almost all of the algorithms used in this study have been verified by simulation tests. However, only 37% of the navigation algorithms surveyed have been validated via real experiments. This study also proves, although there are a number of algorithms capable of navigating robots in static environments, there are a lot of research gaps exist in dynamic environments.

REFERENCES

[1] W. Liu *et al.*, "Design a novel target to improve positioning accuracy of autonomous vehicular navigation system in GPS denied environments," *IEEE Trans. Ind. Informat.*, vol. 17, no. 11, pp. 7575–7588, Nov. 2021, doi: 10.1109/TII.2021.3052529.

[2] I. Rangapur, B. S. Prasad, and R. Suresh, "Design and development of spherical spy robot for surveillance operation," in *Proc. Comput. Sci., 3rd Int. Conf. Comput. Netw. Commun.*, 2020, vol. 171, pp. 1212–1220.

[3] Y. Gao and S. Chien, "Review on space robotics: Toward top-level science through space exploration," *Sci. Robot.*, vol. 2, no. 7, 2017, Art. no. eaan5074, doi: 10.1126/scirobotics.aan5074.

[4] D. Patil, M. Ansari, D. Tendulkar, R. Bhatlekar, V. N. Pawar, and S. Aswale, "A survey on autonomous military service robot," in *Proc. Int. Conf. Emerg. Trends Inf. Technol. Eng.*, 2020, pp. 1–7, doi: 10.1109/ic-ETITE47903.2020.78.

[5] E. Kayacan, E. Kayacan, H. Ramon, O. Kaynak, and W. Saeys, "Towards agrobots: Trajectory control of an autonomous tractor using type-2 fuzzy logic controllers," *IEEE/ASME Trans. Mechatronics*, vol. 20, no. 1, pp. 287–298, Feb. 2015, doi: 10.1109/TMECH.2013.2291874.

[6] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza, *Introduction to Autonomous Mobile Robots*. Cambridge, MA, USA: MIT Press, 2011.

[7] W. Yaonan *et al.*, "Autonomous mobile robot navigation system designed in dynamic environment based on transferable belief model," *Measurement*, vol. 44, no. 8, pp. 1389–1405, 2011.

[8] V. Sharma, S. Pattnaik, and T. Garg, "A review of bacterial foraging optimization and its applications," in *Proc. Nat. Conf. Future Aspects Artif. Intell. Ind. Automat.*, 2012, pp. 9–12.

[9] M. A. Hossain and I. Ferdous, "Autonomous robot path planning in dynamic environment using a new optimization technique inspired by bacterial foraging technique," *Robot. Auton. Syst.*, vol. 64, pp. 137–141, 2015.

[10] M. A. Kareem Jaradat, M. Al-Rousan, and L. Quadan, "Reinforcement based mobile robot navigation in dynamic environment," *Robot. Comput.- Integr. Manuf.*, vol. 27, no. 1, pp. 135–149, 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0736584510000700

[11] G. G. Yen and T. W. Hickey, "Reinforcement learning algorithms for robotic navigation in dynamic environments," *ISA Trans.*, vol. 43, no. 2, pp. 217–230, 2004. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0019057807600329

[12] A. Pandey, S. Pandey, and D. Parhi, "Mobile robot navigation and obstacle avoidance techniques: A review," *Int. Robot. Automat. J*, vol. 2, no. 3, 2017, Art. no. 00022.

[13] B. Patle *et al.*, "A review: On path planning strategies for navigation of mobile robot," *Defence Technol.*, vol. 15, no. 4, pp. 582–606, 2019.

[14] N. Y. Ko and B. H. Lee, "Avoidability measure in moving obstacle avoidance problem and its use for robot motion planning," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 1996, pp. 1296–1303.

[15] D. Bodhale, N. Afzulpurkar, and N. T. Thanh, "Path planning for a mobile robot in a dynamic environment," in *Proc. IEEE Int. Conf. Robot. Biomimetics*, 2009, pp. 2115–2120.

[16] Y. Huang *et al.*, "A motion planning and tracking framework for autonomous vehicles based on artificial potential field elaborated resistance network approach," *IEEE Trans. Ind. Electron.*, vol. 67, no. 2, pp. 1376–1386, Feb. 2020.

[17] P. Wang, S. Gao, L. Li, B. Sun, and S. Cheng, "Obstacle avoidance path planning design for autonomous driving vehicles based on an improved artificial potential field algorithm," *Energies*, vol. 12, no. 12, 2019, Art. no. 2342.

[18] N. Leena and K. Saju, "A survey on path planning techniques for autonomous mobile robots," *IOSR J. Mech. Civil Eng.*, vol. 8, pp. 76–79, 2014.

[19] B. Patle, D. Parhi, A. Jagadeesh, and S. K. Kashyap, "Matrix-binary codes based genetic algorithm for path planning of mobile robot," *Comput. Elect. Eng.*, vol. 67, pp. 708–728, 2018.

[20] S. X. Yang, Y. Hu, and M. Q.-h. Meng, "A knowledge based ga for path planning of multiple mobile robots in dynamic environments," in *Proc. IEEE Conf. Robot., Automat. Mechatronics*, 2006, pp. 1–6.

[21] T. S. Hong, D. Nakhaeinia, and B. Karasfi, "Application of fuzzy logic in mobile robot navigation," *Fuzzy Logic-Controls, Concepts, Theor. Appl.*, 2012, pp. 21–36.

[22] A. Pandey and D. R. Parhi, "Optimum path planning of mobile robot in unknown static and dynamic environments using fuzzy-wind driven optimization algorithm," *Defence Technol.*, vol. 13, no. 1, pp. 47–58, 2017.

[23] M. Faisal, R. Hedjar, M. Al Sulaiman, and K. Al-Mutib, "Fuzzy logic navigation and obstacle avoidance by a mobile robot in an unknown dynamic environment," *Int. J. Adv. Robot. Syst.*, vol. 10, no. 1, 2013, Art. no. 37.

[24] S. X. Yang and M. Meng, "An efficient neural network approach to dynamic robot motion planning," *Neural Netw.*, vol. 13, no. 2, pp. 143–148, 2000.

[25] I. Engedy and G. Horváth, "Artificial neural network based mobile robot navigation," in *Proc. IEEE Int. Symp. Intell. Signal Process.*, 2009, pp. 241–246.

[26] Y. Tusi and H.-Y. Chung, "Using abc and RRT algorithms to improve mobile robot path planning with danger degree," in *Proc. 5th Int. Conf. Future Gener. Commun. Technol.*, 2016, pp. 21–26.

[27] H. Hu, K. Ng, and Y. Qin, "Robust parallel machine scheduling problem with uncertainties and sequence-dependent setup time," *Sci. Program.*, vol. 2016, 2016, Art. no. 5127253 .

[28] A. Q. Faridi, S. Sharma, A. Shukla, R. Tiwari, and J. Dhar, "Multi-robot multi-target dynamic path planning using artificial bee colony and evolutionary programming in unknown environment," *Intell. Serv. Robot.*, vol. 11, no. 2, pp. 171–186, 2018.

[29] J.-H. Liang and C.-H. Lee, "Efficient collision-free path-planning of multiple mobile robots system using efficient artificial bee colony algorithm," *Adv. Eng. Softw.*, vol. 79, pp. 47–56, 2015.

[30] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proc. 6th Int. Symp. Micro Mach. Hum. Sci.*, 1995, pp. 39–43.

[31] L. Wang, Y. Liu, H. Deng, and Y. Xu, "Obstacle-avoidance path planning for soccer robots using particle swarm optimization," in *Proc. IEEE Int. Conf. Robot. Biomimetics*, 2006, pp. 1233–1238.

[32] M. K. Rath and B. Deepak, "PSO based system architecture for path planning of mobile robot in dynamic environment," in *Proc. Glob. Conf. Commun. Technol.*, 2015, pp. 797–801.

[33] L. Tan, H. Wang, C. Yang, and B. Niu, "A multi-objective optimization method based on discrete bacterial algorithm for environmental/economic power dispatch," *Natural Comput.*, vol. 16, no. 4, pp. 549–565, 2017.

[34] M. Dorigo and L. M. Gambardella, "Ant colony system: A cooperative learning approach to the traveling salesman problem," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 53–66, Apr. 1997.

[35] T. Guan-Zheng, H. Huan, and A. Sloman, "Ant colony system algorithm for real-time globally optimal path planning of mobile robots," *Acta Automatica Sinica*, vol. 33, no. 3, pp. 279–285, 2007.

[36] F. H. Ajeil, I. K. Ibraheem, A. T. Azar, and A. J. Humaidi, "Grid-based mobile robot path planning using aging-based ant colony optimization algorithm in static and dynamic environments," *Sensors*, vol. 20, no. 7, 2020, Art. no. 1880.

[37] H. S. Hewawasam, Y. Ibrahim, G. Kahandawa, and T. Choudhury, "Agoraphilic navigation algorithm under dynamic environment," *IEEE/ASME Trans. Mechatronics*, to be published, doi: 10.1109/TMECH.2021.3085943.

[38] H. Hewawasam, M. Y. Ibrahim, G. Kahandawa, and T. A. Choudhury, "Development and bench-marking of agoraphilic navigation algorithm in dynamic environment," in *Proc. IEEE 28th Int. Symp. Ind. Electron.*, 2019, pp. 1156–1161, doi: 10.1109/ISIE.2019.8781352.

[39] H. Hewawasam, M. Y. Ibrahim, G. Kahandawa, and T. A. Choudhury, "Agoraphilic navigation algorithm in dynamic environment with and without prediction of moving objects location," in *Proc. 45th Annu. Conf. IEEE Ind. Electron. Soc.*, 2019, pp. 5179–5185, doi: 10.1109/IECON.2019.8927145.

[40] H. S. Hewawasam, M. Y. Ibrahim, G. Kahandawa, and T. A. Choudhury, "Agoraphilic navigation algorithm in dynamic environment with obstacles motion tracking and prediction," *Robotica*, vol. 40, no. 2, pp. 329–347, 2022, doi: 10.1017/S0263574721000588.

[41] F. Kamil, T. S. Hong, W. Khaksar, M. Y. Moghrabiah, N. Zulkifli, and S. A. Ahmad, "New robot navigation algorithm for arbitrary unknown dynamic environments based on future prediction and priority behavior," *Expert Syst. Appl.*, vol. 86, pp. 274–291, 2017, doi: 10.1016/j.eswa.2017.05.059. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417417303809

[42] L. Huang, "Velocity planning for a mobile robot to track a moving target – A potential field approach," *Robot. Auton. Syst.*, vol. 57, no. 1, pp. 55–63, 2009. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0921889008000183

[43] M. A. K. Jaradat, M. H. Garibeh, and E. A. Feilat, "Autonomous mobile robot dynamic motion planning using hybrid fuzzy potential field," *Soft Comput.*, vol. 16, no. 1, pp. 153–164, 2012.

[44] O. Montiel, U. Orozco-Rosas, and R. Sepúlveda, "Path planning for mobile robots using bacterial potential field for avoiding static and dynamic obstacles," *Expert Syst. Appl.*, vol. 42, no. 12, pp. 5177–5191, 2015.

[45] P. Shi and Y. Cui, "Dynamic path planning for mobile robot based on genetic algorithm in unknown environment," in *Proc. Chin. Control Decis. Conf.*, 2010, pp. 4325–4329.

[46] A. Aouf, L. Boussaid, and A. Sakly, "A PSO algorithm applied to a PID controller for motion mobile robot in a complex dynamic environment," in *Proc. Int. Conf. Eng.*, 2017, pp. 1–7.

[47] Z. Bi, Y. Yimin, and X. Yisan, "Mobile robot navigation in unknown dynamic environment based on ant colony algorithm," in *Proc. WRI Glob. Congr. Intell. Syst.*, 2009, pp. 98–102.

[48] M. Faisal, K. Al-Mutib, R. Hedjar, H. Mathkour, M. Alsulaiman, and E. Mattar, "Multi modules fuzzy logic for mobile robots navigation and obstacle avoidance in unknown indoor dynamic environment," in *Proc. Int. Conf. Syst., Control Inform.*, 2013, pp. 371–379.

[49] V. M. Peri and D. Simon, "Fuzzy logic control for an autonomous robot," in *Proc. Annu. Meeting North Amer. Fuzzy Inf. Process. Soc.*, 2005, pp. 337–342.

[50] A. Zhu and S. X. Yang, "A fuzzy logic approach to reactive navigation of behavior-based mobile robots," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2004, vol. 5, pp. 5045–5050.

[51] F. Abdessemed *et al.*, "A hierarchical fuzzy control design for indoor mobile robot," *Int. J. Adv. Robot. Syst.*, vol. 11, no. 3, 2014, Art. no. 33.

[52] N. H. Singh and K. Thongam, "Mobile robot navigation using MLP-BP approaches in dynamic environments," *Arabian J. Sci. Eng.*, vol. 43, no. 12, pp. 8013–8028, 2018.

[53] X. Chen and Y. Li, "Smooth formation navigation of multiple mobile robots for avoiding moving obstacles," *Int. J. Control, Automat., Syst.*, vol. 4, no. 4, pp. 466–479, 2006.

[54] P. Goel and D. Singh, "Efficient ABC algorithm for dynamic path planning," *Int. J. Comput. Appl.*, vol. 88, no. 2, pp. 15–18, 2014.

[55] H. Hewawasam, M. Y. Ibrahim, G. Kahandawa, and T. A. Choudhury, "The agoraphilic navigation algorithm under dynamic environment (ANADE) with a moving goal," in *Proc. IEEE 29th Int. Symp. Ind. Electron.*, 2021, pp. 1–8.

[56] S. Choi, E. Kim, and S. Oh, "Real-time navigation in crowded dynamic environments using Gaussian process motion control," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2014, pp. 3221–3226.

[57] P. Yao, H. Wang, and Z. Su, "Real-time path planning of unmanned aerial vehicle for target tracking and obstacle avoidance in complex dynamic environment," *Aerosp. Sci. Technol.*, vol. 47, pp. 269–279, 2015.

[58] B. Zhang and H. Duan, "Three-dimensional path planning for uninhabited combat aerial vehicle based on predator-prey pigeon-inspired optimization in dynamic environment," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 14, no. 1, pp. 97–107, Jan./Feb. 2017.

[59] R. Bis, H. Peng, and A. G. Ulsoy, "Velocity occupancy space: Autonomous navigation in an uncertain, dynamic environment," *Int. J. Veh. Auton. Syst.*, vol. 10, no. 1–2, pp. 41–66, 2012.

[60] L. Zhang, Y. Zhang, M. Zeng, and Y. Li, "Robot navigation based on improved A* algorithm in dynamic environment," *Assem. Automat.*, vol. 41, no. 4, pp. 419–430, 2021.

H. S. HEWAWASAM (Student Member, IEEE) received the B.Sc. Engineering degree from the Department of Electrical and Electronic Engineering, University of Peradeniya, Peradeniya, Sri Lanka, and the M.Sc. degree in electrical and electronic engineering from the Swinburne University of Technology, Melbourne, VIC, Australia, in 2017. He received the Ph.D. degree from Federation University, from VIC, Australia, in 2022. He is also a Research Associate with Federation University. During the time in Peradeniya University, he represented Sri Lanka in ABU Robocon 2013. He was the Vice President of the young member's section, Institute of Engineers in Sri Lanka. He started the Ph.D. research in the field of robotics with Federation University in the year 2018.

M. YOUSEF IBRAHIM received the B.Sc., M.Tech., and Ph.D. degrees. He is currently a Professor of engineering (adjunct) with Federation University Australia, Churchill VIC, Australia. He is also the IEEE-IES Vice-President for Memberships. In addition, he was on the Ministerial Advisory Council, Minister for Innovation, Government of Victoria, Australia, for 10 years. Due to his leadership in higher education he was inducted as an Honorary Ambassador of Melbourne in 2015. He was appointed by Australian Research Council as a Reviewer of International Standing for ARC competitive grants scheme and recently as a Performance Reviewer of ARC-funded research centres.

He founded the Mechatronics degree program with the Gippsland Campus of Monash University, Melbourne, VIC, Australia, which is currently offered at Monash University. He was also the Mechatronics subprogram Leader for AusAID program to the Royal Thai Government to establish Mechatronics programs in six Thai Universities. More recently, he established another Mechatronics degree with Federation University Australia. His main research interests include mechatronics, robotics, and industrial automation. He has also interest in maintenance and reliability engineering. Yousef was the general chair of several successful IEEE International conferences and Guest Editor of IEEE-TIE, IEEE-TII, and IEEE-IEM. He is also a Fellow of the Institution of Engineers, Australia.

GAYAN KAHANDAWA APPUHAMILLAGE (Member, IEEE) received the B.Sc., M.Sc., and Ph.D. degrees. He is currently a Senior Lecturer in mechatronics with Federation University, Churchill VIC, Australia. During the Ph.D., he was with the Boeing company to develop monitoring system to detect damage in advanced composite structures using embedded fibre optic sensors. His research projects include health monitoring of structures, west gate bridge, and truck trailers for performance and structural enhancement. He also holds two patents and authored or coauthored edited book, two book chapters, and more than 40 research papers. His research interests include mechatronic systems, instrumentation, sensors and signal processing, artificial intelligence, and mobile robot navigation.