International Journal of Research in Communication and Embedded Electronics
Volume1, Issue 1 2025

Intelligent Robotics Navigation System: Problems, Methods, and Algorithm

1 Mr.Naushad Alam, 2 Khaja Ehtesham Uddin, 3 Hablullah Burhan Uddin, 4 Abdullah Ibrahim

1Assistent Professor,Department of CSE-AIML, Lords Institute of Engineering & Technology.
234 Student Department of CSE-AIML, Lords Institute of Engineering & Technology.

Research Article

Date of Submission: 10-05-2025 Date of Acceptance: 12-05-2025 Date of Publication: 09-07-2025

Abstract

By using insights from these biological systems, this work aimed to provide a concise and understandable overview of the advanced progress in the field of navigation systems, beginning with a single robot, multi-robot, and swarm robots from a specific viewpoint. Observing humans and social animals—which are thought to be very advantageous for this purpose—provides inspiration from nature. Based on a social animal's biological structure or an individual trait, the intelligent navigation system was created. This paper's discussion will center on how the structure of basic agents makes use of flexible and prospective outcomes to maneuver through an unstructured and productive environment. An important field of study in intelligent robotic systems is the integration of the navigation system and biologically inspired approach, which has garnered a lot of interest. The implementation, which is the outcome of simulation carried out by the embodiment of robots functioning in actual surroundings, is examined in this work overall.

INTRODUCTION

The navigation system is one of the most important and crucial topics in the study of mobile robots as it must be precisely identified at the design stage. Notably, a number of problems have been resolved, including those related to perception, cognition, action, human-robot interaction, and control systems [1]–[5]. When a mobile robot is confined to a limited area, such as a home, workplace, or factory, the issues are often readily resolved. For a mobile robot to create great navigation, it must have many characteristics, such as control ability, obstacle avoidance, trajectory planning, and safe distance to the target. To guarantee that all activities can be completed, any navigation system must take into account the previously listed typical designs. A stiff model with many constraints is one of the methods that conventional control systems have suggested to address the current problems [6]–[9]. However, it is well recognized that it is difficult to represent in mathematical models the interactions of mobile robots with their environment, sensor, and actuator [10]–[14]. Consequently, it is thought to limit the applicability of such a control system design [15], [16]. Furthermore, mobile robot behaviors are often broken down into a sense model-plan-act type using traditional control techniques [17]. It is then found to generate complicated equations for hard computation and environmental mapping. Other than that, the model is only suitable for a certain kind of environmental circumstance. Therefore, the mobile robot may be piled at a local minimum or halt at one location due to the uncertainty and imprecision of the surroundings. Furthermore, adding several jobs to the control system in order to accomplish the goal increases the complexity of mobile robot behavior [18]–[20].

The navigation system is built using learning strategies that provide the robot the ability to reason in an uncertain environment and to observe from various angles, both of which are critical for controlling and producing high-quality results. However, a number of issues, including inherent uncertainties in the disorganized environment, inaccurate actuators, and inadequate perceptual information, will make the design challenging to construct. Accordingly, the design of the navigation system should be able to: (i) respond to unforeseen circumstances in an efficient manner as soon as they occur; (ii) take into account many concurrent needs throughout the process; and (iii) attain the goal based on a specified object. The majority of existing studies mostly focused on single robot navigation challenges. In the meanwhile, there has been a noticeable increase in interest in cooperative techniques, which makes communication a crucial area of concentration [21]–[23]. The navigation functions might be quite challenging to overcome with a single robot under certain environmental conditions. The environmental conditions that provide a postdisaster aid and target finding challenge in military applications are often caused by the vast area for sensing. Therefore, to guarantee that all the functions are in order, a single robot has to be developed with strong structures and hardware [24]. In this instance, overcoming this problem will need a higher design cost, greater processing power, and more memory. It is crucial to remember that the whole system might be impacted if the robot malfunctions.

Currently, the simplest form of a group of robots based on a network developed by many researchers [25]-[28]. The robotics system is made to function based on a cooperative approach in order to communicate with each other for the purpose of conducting tasks that are difficult to be performed on their own. Generally, counterparts and with system solution that is highly dependent on three characteristics, namely self-organization, self-adaptiveness, and emergence [29]-[31]. The characteristics are based on the fact that the swarm's organization originates "from within the system not imposed from outside or it comes from local interactions between individual robot in a decentralized way" [29],[30]. In some cases, they are required to move between two places whereby the collective navigation has made it possible to function [22],[29]-[33]. The swarm formation must be controlled due to the facts that all robots work in a particular group with one target. The swarm formation control is performed without a designated leader; hence, the control and communication system are highly desirable. On top of that, the swarm robots are simple hardware, which explains the limited computational cost. All requirements are very vital parameters in the design of the robot. However, it needs to be known that this is very difficult to compute in reality and may not be relevance to all possible surroundings. Therefore, these challenges must be overcome by developing simple and robust algorithms for the purpose of controlling and coordinating these very large groups of robots. The overall structure of this paper takes the form of five sections described as follows: Section 2 offers a brief overview of the mobile robot navigation issue. Section 3 is concerned with the concise methods used by the navigation system. Section 4 presents a review on single robot compare to swarm robots navigation algorithm. Finally, Section 5 provides a concise summary of the entire findings of this study.

1. MOBILE ROBOT NAVIGATION PROBLEMS

2. Three systems may be distinguished among the types of robots used in navigation research: single robots, multi-robots, and swarm robots. The distinctions between swarm and multi-robot robots often depend on the structure and function of a given system. A small group of robots with various forms and functions that may cooperate to accomplish a common objective is referred to as a multi-robot. Swarm robots, on the other hand, are defined as a significant number of basic robots with comparable shapes and functions. To complete the duties, they often collaborate via local coordination and communication. In this instance, all mobile robots have a navigation system that enables them to use their sensing, processing, and actuation skills to make control decisions. In order to move from one starting point to another until it reaches the goal destination, which stays the same in the case of single robot navigation, the mobile robot in this system must select a path with a lower chance of collision. To account for the scenario when the robots move together in an unknown environment to locate the objective, the barriers in multi-robot and swarm robots are built to be both static and dynamic. This implies even more that the robot may provide a dynamic challenge. All single robots, multi-robots, and swarm robots should thus be able to move in a real-time trajectory while overcoming many obstacles from the environment to reach the designated objective. Nonetheless, a number of factors pertaining to implementation, uncertainty, imprecision, and insufficient information in an unstructured real-world setting were taken into account. Perception and cognition are essential for learning about the surroundings and executing control orders that are carried out by a number of sensors and actuators throughout the navigation process. Four categories of

mobile robot navigation systems may be distinguished depending on the

3. These four types of navigation—map-based, behavior-based, learning-based, and communication-based—describe the interplay between perception in the sensor process and control in the actuators. This section goes into additional detail about each of the navigational tasks, behaviors, and robot types used in the navigation system shown in Figure 1.

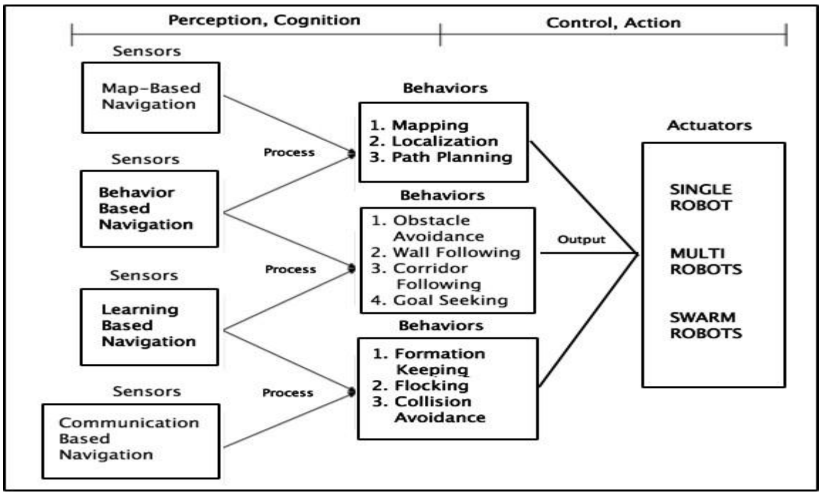

Figure 1. Navigation system based on perception-actuation

3.1. Mapping-based Navigation

The three key skills of mapping, localization, and route planning may be combined to form the navigation system with environmental mapping [34]–[38]. To enable the mobile robot to travel freely and investigate its surroundings, environmental mapping is integrated into it utilizing a memorization technique. Furthermore, the robot's current location inside the ambient mapping is ascertained via the localization procedure. The route planning process is the average of determining a specific movement to attain the target utilizing the map and localization procedure. All abilities often provide the robot the capacity to determine where it is in relation to its reference structure, which enables it to design a route to go to certain target places. Furthermore, the navigation models that rely on this method have to make trade-offs between maps and the robot's predicted location. Making ensuring the robot can manage its approximate position in every condition and environment is vital to the procedure. Such a technique does have several limitations, however, such as the fact that it only depends on the local environment and sensing. Therefore, it is essential that it also have strong sensors or combinations. Lastly, the algorithm has to make sure that all of the sensor elements' uncertainties might result in imprecision and unpredictability when they operate [15], [39].

3.2. Behavior-based Navigation

4. Behavior-based navigation encompasses four competencies: target seeking, wall following, corridor following, and obstacle avoidance. It is feasible to seamlessly integrate behavior resulting from many concurrent objectives into a dynamic control action sequence. Furthermore, it is anticipated that the design of the navigation system would exhibit acceptable behavioral qualities that were established as the potential control action. In addition, it may lead to

conflict in the mobile robot's mobility, especially if it operates in a real chaotic setting. Robot behaviors and environmental mapping are two methods that may be used to provide behavior-based navigation. The map specifically depicts environmental circumstance, while the robot's behavior makes use of its movements. If just one of the two processes is employed, the system should be used in two circumstances in another scenario. This greatly relies on how it is implemented and how it interacts with other concurrent robot actions. However, using this specific strategy would inevitably lead to two significant issues: (i) the integration of more than two behaviors; and (ii) the combining of two simple behaviors to generate a complex one.

4.1. Learning-based Navigation

The traditional approaches require that the robot be powerfully constructed with several sensors, actuators, and a controller without taking into account the potential problems brought on by the environment. Therefore, the navigation system may use a variety of strategies to get past these obstacles and problems. Prior research has suggested artificial intelligence techniques to address the navigation challenge, which is associated with the learning-ability-based process. To include artificial intelligence into mobile robot navigation, however, a number of important considerations must be made, including incomplete issues, imprecision, inaccurateness, and uncertainty conditions when they interact with potential surrounds. Only single and multi-robot behaviors are compatible with this specific system. However, there are certain circumstances, such a complicated environment, that make it very difficult for a robot to do every duty, which leads to additional control process failures. Furthermore, if some system components and the robot's operation were subpar, it indicates that the fault tolerance feature is not supported in a large-scale setting. In order to find innovative approaches that may aid in overcoming the current difficulties, a great deal more study on intelligent navigation on mobile robots will be required.

Communication-based Navigation

A novel platform in the field of mobile robot intelligence navigation is communication based on robotic systems. Local communication modification and sensing are used in the navigation process, which involves the placement of a large number of simple physical robots. They are specifically referred to as swarm robots in the field of robotic systems. Their functioning necessitates a variety of techniques, some of which include decentralized control, a basic autonomous platform, a number of works that draw inspiration from biology, and the significance of coordination and collaboration [30], [33], [40]-[42]. Swarm robots are distinct in that they communicate with one another rather than depending on external infrastructure [44], map-building techniques [36], or maps [43]. However, the issues with swarm robots' competing limitations are exceedingly difficult to resolve, especially when a dynamic environment necessitates routing an optimum path in real time and when a new restriction arises. In addition, swarm robots must navigate to their destination while attempting to conform and adapt to their courses while taking into account the possibility of collisions with other robots and stationary barriers. This presents a challenge. Furthermore, the robots have to calculate their movements separately and decentralizedly due to the large number of them and real-time limitations. In this instance, animals with behavioral programs—such as insect colonies [45], [46], bird flocks [45], [47], schools of fish [45], [48], and amoeba groups [49]—are shown to be adaptable enough to deal with environmental changes. Two goals of the algorithm, which is based on the basic behavioral rule, are (i) to reduce the amount of complexity required in the information-processing system and (ii) to enable behavior creation in order to maximize energy expenditures.

Individual interactions with one another are the root cause of the system's whole character. Without the need to specifically choose certain characteristics, natural selection tends to favor optimization techniques that make use of basic laws and inherent flexibility. As a result, the navigation output will be reliable, allowing for the handling of error-prone situations, such as noise from sensors and actuators, as well as the capability of fault tolerance. Nevertheless, there are a number of drawbacks to optimization based on the nature of animal social approaches, including: (i) the inability to control the motion of the robots; (ii) the ability to be optimal only locally; (iii) the production of sloppy global movements when multiple robots maneuver in a complex environment; and (iv) the potential for the robots to be a trap in a local minimum [50]-50]. As a result, it is essential to address the current issues that are crucial to the design requirement, such as the real-time, chaotic, and dynamic environment as well as the issues of imprecision, incompleteness, and uncertainty in the navigation systems of individual and swarm robots. Furthermore, poor algorithms and unsuccessful communication are addressed as specific issues throughout the navigation system's development. Finally, once all the factors have been examined, the overall performance of the robotic navigation systems may either be enhanced or at least not worse.

5. METHODS OF NAVIGATION SYSTEM

The navigation systems are thoroughly examined in this part in order to provide many crucial details for the investigation of a single robot becoming a swarm robot. Figure 2 compares the three approaches—conventional artificial intelligence, soft computing, and swarm intelligence—that are associated with mobile robot navigation systems. This part will also include a detailed explanation of the research's definitions and methodology, as well as a thorough analysis of its results.

5.1. Conventional Artificial Intelligence

To create robots that can complete several tasks given to them, the most basic navigation technique was created [53]–[55]. However, a specific method must be used in conjunction with the adaptive technique. Furthermore, it seems that the navigation only works in the local area. As a result, if the surroundings are constantly changing, the robot will find it difficult to detect and regulate its movements, which would ultimately hinder the mission's success. The current limitations also require a great deal of traditional artificial intelligence (AI) techniques, such as artificial potential field techniques [56], virtual target techniques [57], landmark learning [58], tangent graphs [59], path velocity decomposition techniques [60], accessibility graphs [61], space–time concepts [62], incremental planning [63], relative velocity techniques [64], reactive control schemes [17], curvature-velocity techniques [65], dynamic window techniques [9], and Simultaneous Localization and Mapping (SLAM). Regretfully, the aforementioned traditional AI techniques appear to suffer from the following drawbacks: (i) local controller [66], (ii) high computational resources brought on by a high number of states [67], [68], (iii) no optimization module [69], (iv) frequent dead-lock situations because of local minimum [70], and (v) no passage between closely spaced obstacles, which causes oscillations [56]. Consequently, it is strongly advised that the control strategy be created in order to provide a workable solution for the navigation issues of mobile robots.

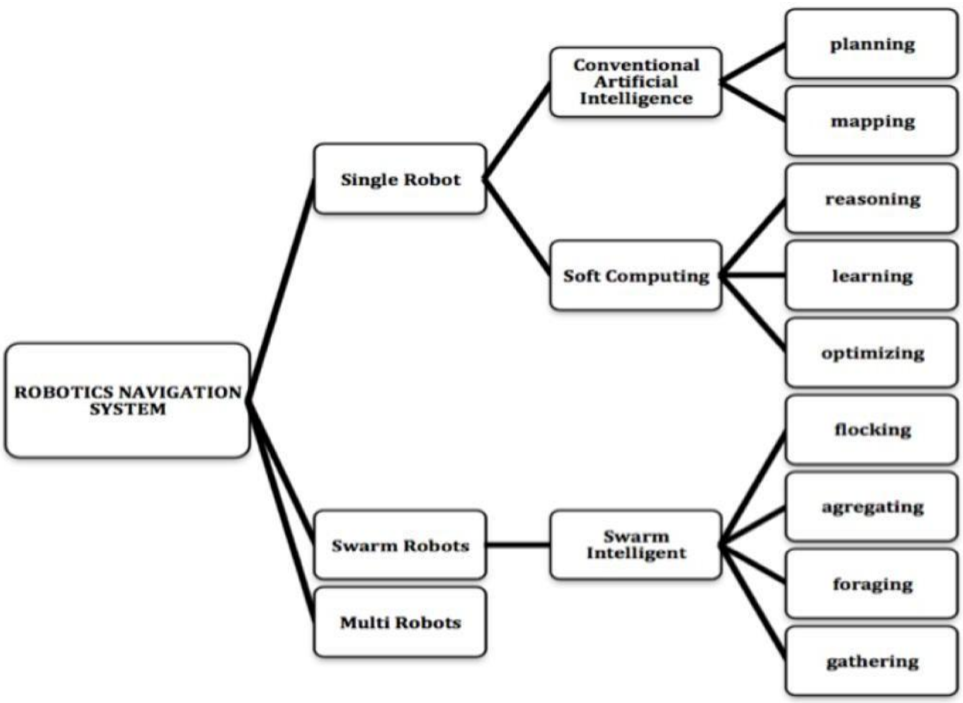

Figure 2. Intelligent robotics navigation system algorithms

5.2. Soft Computing

The goal of Soft Computing (SC) techniques is to provide a reliable and affordable solution. As a result, this approach has suggested a technique that makes use of several pieces of information on the extraordinary capacity of the human mind for learning and reasoning [71]. Additionally, it offers a different approach to address some of the previously described navigation issues. Because it avoids imprecision, ambiguity, partial truth, and approximation, this specific approach differs from traditional AI techniques. This method has produced high performance and has been widely used in the design of mobile robot applications [72]–[75]. This approach is a good way to cope with robotic control and navigation problems since it can handle messy and unfamiliar situations. These strategies are thought to increase the intelligence of mobile robot navigation and provide practical approaches. Neural networks, genetic algorithms, and fuzzy logic systems are a few examples of soft computing approaches. The intrinsic imprecision of fuzzy logic systems, particularly type-1 fuzzy logic systems (T1FLS), makes them a great option for mobile robot navigation. However, due to the limited modeling of T1FLS membership functions (Mfs) in reducing the influence of uncertainty, T1FLS is unable to adequately handle the uncertainties [76]. When MFs are at least substantially supplied, the uncertainty value will vanish [77]. Under such circumstances, the mistake will still happen, and even a little error might have a big impact on navigation performance [78]. form-2 fuzzy logic system (T2FLS), a new form of fuzzy logic, has recently been developed as an enhanced variant of T1FLS and has shown itself to beproficient in the navigation of mobile robots [79]–[81]. Specifically, when it is difficult to do the actual measurement, T2FLS are quite helpful [77]. In addition, T2FLS are computationally demanding and challenging to develop for real-time use, particularly in mobile robot navigation [82], [83]. To make the calculation simpler, the interval type-2 fuzzy logic system (IT2FLS) is suggested. IT2FLS has the potential to overcome the limitations of T1FLS and create a new fuzzy system generation with enhanced navigation system performance [84], [85]. In contrast to T1FLS, IT2FLS is still computationally in great demand [86].

An imprecise, wide, dynamic, and unstructured environment is shown by another example of navigation study. Therefore, in order to reach the destination without colliding, it is crucial for a mobile robot to be able to comprehend the specific environment [10]. Perception is also necessary for the processing, recognition, learning, reasoning, interpretation, decision-making, and action of data. Neural networks (NNs) are capable of observing the conditions and simulating the exceptional perception and pattern recognition for each environment, which is necessary to develop a mobile robot's adaptive navigation system. Over the last several years, NNs and their use to help mobile robots increase their operating skills in new environments have been the subject of previously published research [87]–[89]. Since NNs are known to be very noise-tolerant, their processing of flawed or noisy data is more useful than that of conventional AI approaches [90]. Furthermore, the NNs approach has been effectively used in several research to construct models relevant to mobile robot navigation. The main drawback of the traditional NNs approach, however, is the need for real-time training data presentation, which sometimes leads to a very lengthy learning period. Although learning, reasoning, and decision-making have been suggested as components of SC approaches, all outcomes must be adjusted to get superior navigation system performance, particularly when attempting to determine the ideal goal position value. The Genetic Algorithm (GA) has been identified as one of the most effective algorithms for difficult optimization problems. Furthermore, GA is introduced as a new optimization technique, and its basic characteristics make it a desirable option for resolving the mobile robot navigation issue [91]–[93]. In addition, GA can address the following problems with gradient-based and other classic search methods: (1) high computational cost, (2) vast memory areas, and (3) time consumption [94]. However, the GA algorithm's application in mobile robot navigation produces delayed convergence and struggles to achieve a global optimal solution [93]. The capacity to understand and describe the process of choosing the best course of action for a specific kind of issue without extrapolating it to others is one of the unique qualities shared by all intelligent soft computing systems. Neural networks are very good at identifying patterns and have a variety of learning capabilities. Neural networks, however, are not capable of elucidating the decision-making process [95]. Furthermore, fuzzy logic systems excel in making their own conclusions and resolving the causes of ambiguity and erroneous information [96]. They struggle to get the guidelines that are established to produce the optimal judgments right away, however. With a high success rate, evolutionary algorithms (EA) provide outstanding results throughout the optimization process and have been used in a wide range of applications. The algorithm imitates the manner evolution acts, which then allows the performance of controllers to be improved or be adapteds to different systems. On the other hand, GA is linked to probabilistic, locally optimal, and slowly convergent random numbers [97]. Table 1 describes a number of features of the soft computing method used in mobile robot applications. However, there are a number of restrictions related to soft computing techniques which makes it hard for navigation tasks to be performed in large-scale environment. Finally, they are unable to guarantee the robustness and fault tolerance characteristic because they are related to centralized control architecture and does not support self-organization.

Table 1. Soft computing performance in intelligent navigation

Algorithm	Process	Behavior	Adaptability	Computational	Word problems
Type-1 Fuzzy Logic	reasoning and decision- making	perception to action	low	low rate	uncertainty and imprecision environment
Type-2 Fuzzy Logic	reasoning and decision- making	perception to action	medium	high rate	uncertainty and imprecision environment
Interval Type-2 Fuzzy Logic	reasoning and decision- making	perception to action	medium	medium rate	uncertainty and imprecision environment
Neural Networks	learning and adapting	human capabilities to learn and adapt	high	high rate	The dynamic environment under varying conditions
Evolutionary Algorithm	searching and optimizing	human capabilities to learn and adapt	high	medium rate	the dynamic environment under varying conditions

5.3. Swarm Intelligence

Because its algorithms are based on individual human knowledge, the Soft Computing approach is in opposition to the Swarm Intelligence (SI) approach. SI's algorithms, on the other hand, are based on the behaviors of social creatures, such as insects and animals that live in groups. Therefore, Swarm Intelligence (SI), which may be further divided into autonomy, dispersed functioning, and self-organizing for the aim of creating various artificial systems, is based on the behavior of social insects [30]. The use of the SI technique in navigation systems, particularly in multi-robot and swarm robotics, is the outcome of distributed functioning, autonomous communication, and coordination and collaboration of self-organization among all robots in the group. Consequently, the aforementioned needs will need to be taken into account by the suggested approaches. Swarm robotics and multi-robotic systems seem to have comparable characteristics to SI, which analyzes the cooperative behaviors of robots interacting locally with their surroundings. Additionally, the navigation system performs very well, especially for complicated systems operating in predetermined environments. Particle swarm optimization (PSO) [104], ant colony optimization (ACO) [105], [106], bee colony optimization (BCO) [107], and firefly algorithm (FA) [108] are among the several kinds of SI techniques. Every technique used in the SI approach has a variety of advantages and disadvantages, much like other approaches. Nevertheless, there isn't a single optimal optimization method that can be applied to all issues. It is crucial to remember that the quality of the swarm robots' navigation solution is determined by the appropriate procedures and the set of parameters.

Kennedy and Eberhart proposed the PSO algorithm, a population-based optimization technique, in 1995. The social behavior of a school of fish and a flock of birds provides the method's insight. Because of its unique searching mechanism, straightforward idea, computational efficiency, and ease of implementation, PSO is often used in a wide range of optimization domains [98]. Because of this, its simplicity has given rise to a number of robotics navigation issues, which are resolved by using the PSO algorithm to achieve high performance [24], [99]-[102]. During the navigation process, the data is collected from sensors on a real-time robot. There are three steps in this navigation method. The first step is to transform the navigation problem into an optimization problem. The appropriate objective function is then created with the purpose and challenges in mind. Lastly, quick convergence in a variety of intricate optimizations and search problems is the primary benefit of PSO [103], [102]. Due to their greater dependence on function values rather than subordinate data, population-based heuristics are more costly. PSO is susceptible to partial convergence, nevertheless, particularly when there are several potential outcomes or optimizable dimensions that are likely to fall into local optima. [109]–[111].

One of the heuristic methods is the Ant Colony Systems (ACS). Therefore, the ACS process—which was carried out in line with the ants' natural nature, namely in the mechanism of cooperation and practice—is the answer to the combinatorial optimization issue [112]. In the meanwhile, Ant Colony Optimization (ACO) is another colony strategy that pertains to heuristic algorithms. Finding the basic cost route in a graph by idealizing a problem is the fundamental notion behind ACO. In the form of pheromone trails, the ACO contradicts the ACS [105], [113], and [114]. To modify the pheromone level on the edges that are allocated to the best ant tour currently in existence, ACO upgrades the pheromone in two ways: locally and globally. The research claims that both ACS and ACO provide strong and adaptable abilities to handle a range of optimization problems. Furthermore, ACO has also been modified to account for the location of various odor sources [41], [115], and [116]. In addition, two robot foraging algorithms inspired by ants are presented, which improves the robots' organization [117]. Overall, great optimization results are often obtained when the ACO method is used in swarm robot applications [31], [118], and [119]. However, because of the dependent process of ACO, which leads to an unknown period of convergence, the ACO algorithm in navigation systems seems to have some drawbacks. Numerous natural systems have shown that even the most basic individual organisms can communicate dynamically with one another to develop systems capable of doing very challenging tasks. The communities of artificial bees are thought to behave similarly to those of real bees, but they are thought to vary significantly. Therefore, it is thought that artificial bee colony optimization (BCO) may resolve limited optimization problems. In addition, the BCO is capable of solving deterministic combinatorial problems, including uncertainty-based combinatorial problems [107]. Apart from that, BCO has been used to help mobile robots figure out their routes [120]–[122]. The work required to determine the robots' motion trajectory is one of the study's obstacles. With the ultimate goal of minimizing the route distance for every robot, this procedure starts from a predetermined beginning place and moves to a permanent target position on the globe map. The program includes a navigation strategy to traverse in a new environment as well as a recruiting approach to share the established discoveries with other robots in the swarm. The BCO method is helpful in coming up with a workable solution to the problems of route design, which subsequently shortens the path's emergence time. However, due to their complete randomness, the algorithm's local convergence cannot provide global convergence [125]–[126]. Therefore, to show an environment with certainty, dynamic, and stochastic properties, the algorithm has to be modified. It is well known that the Firefly algorithm (FF) produces brief, rhythmic flashes. In instance, flash designs are often specific to a single species. Such flashes serve two fundamental purposes: they attract prospective prey and mating partners (communication) [108]. Fireflies are used in the process of developing fault tolerance and route in the context of swarm robots [127] [129]. Path planning has emerged as a significant obstacle in mobile robot navigation, with the goal of determining the optimal course with the lowest possible chance of collision in a certain environment. In general, there are several paths that may help the robot reach a certain goal, but it's crucial to remember that the optimal way must be selected in accordance with the stated guidelines.

Table 2. Swarm intelligence performance in intelligent navigation

Algorithm	Process	Behavior	Computational	Word problems
Particle Swarm Optimization	aggregating and flocking	coordinate motion and collective exploration	low rate	target seeking, path planning, localization
Ant Colony Optimization	foraging and trailing	collective transport, task allocation and consensus achievement	medium rate	path planning, obstacle avoidance, trail avoidance, mapping
Bee Colony Optimization	foraging	task allocation and consensus achievement	low rate	path planning, localization
Firefly Algorithm	gathering	collective fault detection and group size regulation	medium rate	path planning and fault tolerance

The automated subdivision and the ability to negotiate with multimodality are cited as the two main advantages of FA [123]. Finding the ideal route with the following attributes—shortest path, least energy consumption, or shortest time—is the result of mobile robot navigation. However, the firefly algorithm's incapacity to remember or learn any past events with a better situation can lead to the swarm robots being trapped in multiple local optimums. This can result in missing conditions as the robots move without remembering their previous better situation [124]. Table 2 summarizes all the methods that may be used to compare the application of swarm intelligence algorithms to real-world issues.

Table 3 presents the main differences among conventional artificial intelligence, soft computing, and swarm intelligence approaches, particularly in terms of software, hardware, and algorithm requirements.

Table 3. Comparison of three approaches in navigation system

Performance	Conventional AI	Soft Computing	Swarm Intelligence
Processing time	slow	medium	fast
Computational	high	medium	low
Complexity	high	medium	low
Scalability	low	low	high
Adaptability	nil	low	high
Typical application	single agent	single agent/multi-agent	multi-agent
Environment	known	known/unknown	unknown
Algorithm design	human experience	human and animal behavior	social animal
Control architecture	centralized	centralized	decentralized
Design characteristic	powerful hardware	powerful hardware	simple hardware
Cost	high	high	low

SINGLE ROBOT VS SWARM ROBOTS NAVIGATION ALGORITHMS

5.4. Single Robot

There have been thorough comparisons between research on mobile robot navigation systems using Soft Computing (SC) and traditional AI. A review of the advantages and disadvantages of comparable strategies is given in Table 4. Additionally, a number of relevant technology comparisons are shown, particularly with respect to the use of soft computing techniques to address the navigation system's inherent constraints. Table 4 clearly compares the traditional AI technique with the Soft Computing approach in the context of mobile robot applications. The attempt to mimic human intelligence using symbol manipulation and symbolically structured systems seems to get all of the focus from classical AI.

. This specific intelligence is shown by software or machines. However, this method limits the circumstances in which traditional AI may be used. In the meanwhile, the significance of planning and mapping in regulating the motion of mobile robots seems to provide both additional benefits and drawbacks. It is important to remember that the mobile robot's environment is real-time, dynamic, and disorganized. Because it requires a properly described analytical model and often a significant amount of calculation time, it therefore restricts implementation.

SC is recognized as a component of the computational intelligence methodology. More precisely, it refers to a collection of computational methods and strategies that draw inspiration from nature in order to concentrate on challenging real-world problems. The impacts of traditional AI and the human mind, which serves as a paradigm for soft computing, are both at odds with SC. It is very forgiving of approximation, partial truth, uncertainty, and imprecision. Because of the existence of imprecision in sensor detection, uncertainty in a dynamic environment, and actuator error, these benefits are thus very advantageous in the design of intelligent navigation systems. Furthermore, SC is used to get around the difficulties and create a navigation system with great performance, including a simple and adaptable algorithm for communication and navigation. To ensure excellent functionality, the robots must possess several qualities, including the ability to avoid potential collisions, efficiently cover the terrain, divide the task, help each other with additional data via multiple sensors, and the ability to generate an unfixed redistribution to comply with the situation in case the robot is unable to function [29]. Therefore, it is obvious that the process of managing the robot teams has to be given careful consideration. However, since the process might make the system more complicated, it is crucial to recognize its difficulties [130]. Furthermore, a number of traditional centralized methods have been used [130], [131], but no notable drawbacks have been found, therefore they cannot be regarded as a general-purpose solution. Centralized control has many drawbacks, such as excessive resilience, lack of flexibility, and significant computational and communication complexity [132].

Table 4. Strengths and limitations of conventional AI and soft computing approach in navigation systems

Approach	Strengths	Limitations	References
Simultaneous localization and map building (SLAM)	able to eliminate the need for artificial infrastructures	Complexity sub-optimal map-building high computational cost requires a consistent map	[6]; [133];[134].
Potential Field	quickly observe efficient mathematical analysis and simplicity	path sub-optimal high computational cost trap situations due to local minimum no passage between closely spaced obstacles. oscillations in the presence of obstacles and narrow passages. the global workspace must be known	[56];[7].
Curvature velocity method	high accuracy computational efficiency generalizes well to arbitrary simple to implement real-time computation	Complexity path sub-optimal trap in local minima	[65];[8].
The dynamic window approach (DWA)	Accuracy, consistency, efficiency, correctly and in a rigorous way ncorporates the dynamics of the robot	Complexity, path sub-optimal, trap in local minima	[135];[9].
Type-1 Fuzzy Logic System	constant sensitivity requires expert knowledge to incorporate in the control of the system	difficult to construct fuzzy rule base , high computational cost involving larger numbers of input and output.	[72];[136];[96].
Type-2 Fuzzy Logic System	better performance compared to T1FLS reduce the rule base number increase accuracy	high computational cost even with few input difficult to construct fuzzy rule base	[137];[138];[82];[81];[84].
Neural Networks	provide mathematical modeling to approximate continuous real-valued functions.	require large memory and high-speed processor. High computational cost slow convergence	[139];[90].
Hybrid Fuzzy-GA	process the online learning and adaptation of the controller possess the competency to dynamically adjust to new surrounding and update its knowledge	produce high computational cost. require huge amount of iterations to develop a good controller	[140];[141].
Hybrid Neural-Fuzzy	possess the ability to automatically extract the fuzzy rules and MFs.	requires complex training and limited implementation in dedicated hardware	[142];[143].

5.5. Swarm Robots

It is well recognized that one of the most important applications of swarm intelligence is swarm robotics. In this instance, the assigned control strategy is seen to be better suitable for controlling systems with a large number of robots, especially those in which the robots themselves are able to gather or sense environmental information. Swarm robots draw inspiration from biological evolution, which also develops behaviors from animals to build robust and efficient navigation and optimum collective judgments about the navigation of individual robots. Each robot in this scenario prefers to choose its own course of action by identifying its current environment and using a variety of preset control rules. The main idea is to create control principles that will enable the complete robot system to achieve the desired goals, such navigating without colliding or constructing a spatial structure. Many academics have proposed many swarm intelligence-based optimization algorithms to replace the conventional centralized algorithms, such as PSO, ACO, BCO, and FA. Without a coordinator, they proceed methodically. They are processed using modest computing resources and simple code, and when the computational cost is significant, it is thought that the individual system may change its movement mode. Swarm intelligence is suggested for mobile robot navigation in order to determine the best and most collision-free path from a starting point to the destination in an unknown and dynamic environment [104]–[108]. Proximity, quality, diversified response, stability, and flexibility are some of the essential principle qualities that are present in the common swarm intelligence system. Last but not least, this study offers some comparisons of similar algorithms, particularly with regard to the use of the swarm intelligent algorithm to address the intrinsic constraints of mobile robot navigation.

Table 5. Strengths and limitations of swarm intelligence approach in navigation system

Algorithm	Strengths	Limitations	References
Particle Swarm Optimization	easy to implement few parameter control low computation great optimization ability fast convergence good implementation in swarm robots	premature convergence slow convergence optimality convergence influenced by inertia weight low flexibility trap in local minima	[50];[144];[145].
Ant Colony Optimization	distributed computation dynamic application good result in swarm robots search and exploration	difficult analysis slow convergence uncertain time to converge	[104];[113];[114];[105].
Bee Colony Optimization	Fast convergence high flexibility global optimization support implementation of parallel processing	high computational cost poor convergence local optimization	[107];[121].
Firefly Algorithm	the high convergence rate low computational cost less number of iterations and floating point suitable for parallel processing	slow convergence speed the algorithm inflexible algorithm parameters do not change with the time Local optimization	[127]; [128].

Swarm intelligence is appropriate for simple creatures with simple behavior and awareness, as Table 5 demonstrates. The lack of global knowledge in the system causes the control structure to be distributed. Furthermore, when mobile robots move dynamically in every changing environment, the collapse of an individual agent is accepted. Regarding the deployment of swarm robots, it is challenging to build the navigation system with regard to the parameters since they might have a significant impact on the development of collective behavior. The individual behavior, however, seems to be noise. Furthermore, there is no analytical method, and under certain circumstances, the behavior of a swarm of robots cannot be deduced from the behavior of a single robot. The uncertainty, limitations, and errors that result from the processing technique of sensor input must also be taken into account by the coordination forms used in swarm robots. Apart from that, each of the current algorithms for swarm robot navigation has advantages and disadvantages that are connected to a particular objective and take into account the significance of priority among various performances. Ultimately, a number of algorithms were able to be succinctly described from their different perspectives.

CONCLUSION

The issues, approaches, and use of the sophisticated mobile robot navigation system have all been investigated in this work. Along with identifying areas that need further investigation, this study also aimed to provide fresh data to the body of literature in this specific field. Due to its many potential uses, mobile robot navigation is regarded as one of the primary application areas that has attracted a lot of interest. Furthermore, it has been noted that generic algorithms encounter development barriers in this domain, such as intricate computation and a significant reliance on high-precision sensors. In mobile robot applications, the ability to navigate in any environment is essential to preventing dangerous scenarios like accidents and dangerous circumstances. In order to reach the target quickly, it is necessary to maintain the trajectory and formation's stability. In essence, three combination algorithms—self-localization, route planning, and map building—can be used to accomplish navigation. As a result, an environmental model has been used to suggest a number of traditional environment methods. Analyzing the mobile robot's sensing, actuation, and interaction in a specific environment is challenging, however. In addition, it was shown that the algorithm does not guarantee error, impression, or ambiguity in the dynamic environment. Therefore, an intelligent method is suggested to enhance performance and address the drawbacks associated with mobile robot navigation. Furthermore, the computational intelligent algorithm—which incorporates swarm intelligence and soft computing—is regarded as a potent method that may solve the problem without requiring environment modeling. Three competencies—reasoning, learning, and optimizing—were developed in relation to this. Since every activity must adhere to its own unique set of requirements, it is regrettably best to admit that not all algorithms are appropriate for the general purpose. Finally, if the algorithms are combined, a good performance in the single robot, the multi-robot, and the swarm robots system is achieved.

REFERENCES

[1] S. Thrun and A. Bücken, "Integrating grid-based and topological maps for mobile robot navigation," in *Proceedings of the National Conference on Artificial Intelligence*, pp. 944-951, 1996.

[2] A. Elfes, "Using occupancy grids for mobile robot perception and navigation," *Computer*, vol/issue: 22(6), pp. 46- 57, 1989.

[3] M. A. Goodrich and A. C. Schultz, "Human-robot interaction: a Survey," *Foundations and trends in human- computer interaction*, vol/issue: 1(3), pp. 203-275, 2007.

[4] E. Gat, "Integrating planning and reacting in a heterogeneous asynchronous architecture for controlling real-world mobile robots," in *AAAi,* vol. 1992, pp. 809-815, 1992.

[5] R. C. Arkin, "Integrating behavioral, perceptual, and world knowledge in reactive navigation," *Robotics and autonomous systems*, vol/issue: 6(1), pp. 105-122, 1990.

[6] Y. Yingmin and X. Hu, "Robot Simultaneous Localization and Mapping Based on Self-Detected Waypoint," *Cybernetics and Information Technologies*, vol/issue: 16(2), pp. 212-221, 2016.

[7] S. S. Ge and Y. J. Cui, "Dynamic motion planning for mobile robots using potential field method," *Autonomous Robots*, vol/issue: 13(3), pp. 207-222, 2002.

[8] J. L. Fernández, *et al.*, "Improving collision avoidance for mobile robots in partially known environments: the beam curvature method," *Robotics and Autonomous Systems*, vol/issue: 46(4), pp. 205-219, 2004.

[9] D. Fox, *et al.*, "The Dynamic Window Approach to Collision Avoidance," *IEEE Robotics and Automation Magazine*, vol. 23, pp. 23–33, 1997.

[10] A. Saffiotti, "The uses of fuzzy logic in autonomous robot navigation," *Soft Computing*, vol/issue: 1(4), pp. 180- 197, 1997.

[11] O. Motlagh, *et al.*, "An expert fuzzy cognitive map for reactive navigation of mobile robots," *Fuzzy Sets and Systems*, vol. 201, pp. 105-121, 2012.

[12] M. R. Akbarzadeh-T, *et al.*, "Soft computing for autonomous robotic systems," *Computers & Electrical Engineering*, vol/issue: 26(1), pp. 5-32, 2000.

[13] K. Tanaka, *et al.*, "Novel method of estimating surface condition for tiny mobile robot to improve locomotion performance," in *Intelligent Robots and Systems. IEEE/RSJ International Conference on*, pp. 6515-6520, 2015.

[14] M. Algabri, *et al.*, "Comparative study of soft computing techniques for mobile robot navigation in an unknown environment," *Computers in Human Behavior*, vol. 50, pp. 42-56, 2015.

[15] S. Nurmaini, *et al.*, "An embedded interval type-2 neuro-fuzzy controller for mobile robot navigation," in *Systems, Man and Cybernetics, SMC. IEEE International Conference on.*, pp. 4315-4321, 2009.

[16] A. Saxena and A. Saxena, "Review of Soft Computing Techniques used in Robotics Application," *International Journal of Information and Computation Technology*, vol/issue: 3(3), pp. 101-106, 2013.

[17] R. A. Brooks, "A robust layered control system for a mobile robot," *Journal of Robotics and Automation*, vol/issue: 2(1), pp. 14-23, 1986.

[18] L. P. Kaelbling, "Learning in embedded systems," MIT press, 1993.

[19] S. Nurmaini, "Intelligent Navigation in Unstructured Environment by using Memory-Based Reasoning in Embedded Mobile Robot," *European Journal of Scientific Research*, vol/issue: 72(2), pp. 228-244, 2012.

[20] U. Nehmzow, "Physically embedded genetic algorithm learning in multi-robot scenarios," *The pega algorithm*, pp. 115-123, 2002.

[21] R. Fierro, *et al.*, "Cooperative control of robot formations," in *Cooperative control and optimization*, Springer US, pp. 73-93, 2002.

[22] F. Ducatelle, *et al.*, "Cooperative navigation in robotic swarms," *Swarm Intelligence*, vol/issue: 8(1), pp. 1-33, 2014.

[23] J. V. Hook, *et al.*, "Algorithms for cooperative active localization of static targets with mobile bearing sensors under communication constraints," *Robotics, IEEE Transactions on*, vol/issue: 31(4), pp. 864-876, 2015.

[24] S. Nurmaini, *et al.*, "Pattern Recognition Approach for Swarm Robots Reactive Control with Fuzzy-Kohonen Networks and Particle Swarm Optimization Algorithm," *Journal of Telecommunication, Electronic and Computer Engineering*, vol/issue: 8(3), pp. 155-160, 2016.

[25] P. Corke, *et al.*, "Localization and navigation assisted by networked cooperating sensors and robots," *The International Journal of Robotics Research*, vol/issue: 24(9), pp. 771-786, 2005.

[26] G. S. Sukhatme, *et al.*, "Design and development of a wireless robotic networked aquatic microbial observing system," *Environmental Engineering Science*, vol/issue: 24(2), pp. 205-215, 2007.

[27] V. Kumar, *et al.*, "Networked robots," in Springer Handbook of Robotics, Springer Berlin Heidelberg, pp. 943-958, 2008.

[28] K. Kamei, *et al.*, "Cloud networked robotics," *Network, IEEE*, vol/issue: 26(3), pp. 28-34, 2012.
[29] E. Sahin, "Swarm robotics: From sources of inspiration to domains of application," in *Swarm robotics*, Springer Berlin Heidelberg, pp. 10-20, 2004.
[30] M. Brambilla, *et al.*, "Swarm robotics: a review from the swarm engineering perspective," *Swarm Intelligence*, vol/issue: 7(1), pp. 1-41, 2013.
[31] M. Senanayake, *et al.*, "Search and tracking algorithms for swarms of robots: A survey," *Robotics and Autonomous Systems*, vol. 75, pp. 422-434, 2016.
[32] M. Dorigo, *et al.*, "Swarmanoid: a novel concept for the study of heterogeneous robotic swarms," *Robotics and Automation Magazine, IEEE*, vol/issue: 20(4), pp. 60-71, 2013.
[33] A. V. Savkin, *et al.*, "Distributed formation building algorithms for groups of wheeled mobile robots," *Robotics and Autonomous Systems*, vol. 75, pp. 463-474, 2016.
[34] T. S. Levitt and D. T. Lawton, "Qualitative navigation for mobile robots," *Artificial intelligence*, vol/issue: 44(3), pp. 305-360, 1990.
[35] J. A. Meyer and D. Filliat, "Map-based navigation in mobile robots: a review of map-learning and path-planning strategies," *Cognitive Systems Research*, vol/issue: 4(4), pp. 283-317, 2003.
[36] H. D. Whyte and T. Bailey, "Simultaneous localization and mapping: part I," *Robotics and Automation Magazine, IEEE*, vol/issue: 13(2), pp. 99-110, 2006.
[37] B. Tovar, *et al.*, "Planning exploration strategies for simultaneous localization and mapping," *Robotics and Autonomous Systems*, vol/issue: 54(4), pp. 314-331, 2006.
[38] J. Clemens, *et al.*, "An evidential approach to SLAM, path planning, and active exploration," *International Journal of Approximate Reasoning*, vol. 73, pp. 1-26, 2016.
[39] S. Nurmaini and S. Z. M. Hashim, "Motion planning in unknown environment using an interval fuzzy type-2 and neural network classifier," in *Computational Intelligence for Measurement Systems and Applications, IEEE International Conference on,* pp. 50-55, 2009.
[40] A. J. Sharkey and N. Sharkey, "The application of swarm intelligence to collective robots," *Advances in applied artificial intelligence*, pp. 157, 2006.
[41] G. Beni, "From swarm intelligence to swarm robotics," in *Swarm robotics*, Springer Berlin Heidelberg, pp. 1-9, 2004.
[42] F. Mondada, *et al.*, "SWARM-BOT: A new distributed robotic concept," *Autonomous Robots*, vol/issue: 17(2-3), pp. 193-221, 2004.
[43] E. Trulls, *et al.*, "Autonomous navigation for mobile service robots in urban pedestrian environments," *Journal of Field Robotics*, vol/issue: 28(3), pp. 329-354, 2011.
[44] K. J. O'Hara, *et al.*, "Physical path planning using a pervasive embedded network," *IEEE Transactions on Robotics*, vol/issue: 24(3), pp. 741-746, 2008.
[45] P. Miller, "The smart swarm: How understanding flocks, schools, and colonies can make us better at communicating, decision making, and getting things done," Avery Publishing Group, Inc., 2010.
[46] N. Palmieri and S. Marano, "Discrete Firefly Algorithm for Recruiting Task in a Swarm of Robots," in *Nature- Inspired Computation in Engineering,* Springer International Publishing, pp. 133-150, 2016.
[47] R. Poli, "Analysis of the publications on the applications of particle swarm optimisation," *Journal of Artificial Evolution and Applications*, vol. 3, 2008.
[48] G. Lee and N. Y. Chong, "Flocking controls for swarms of mobile robots inspired by fish schools," INTECH Open Access Publisher, 2008.
[49] V. Trianni, *et al.*, "Evolving aggregation behaviors in a swarm of robots," in *Advances in artificial life*, Springer Berlin Heidelberg, pp. 865-874, 2003.
[50] S. Doctor, *et al.*, "Optimal PSO for collective robotic search applications," in *Evolutionary Computation*, *CEC2004. Congress on,* vol. 2, pp. 1390-1395, 2004.
[51] L. S. Marcolino and L. Chaimowicz, "No robot left behind: Coordination to overcome local minima in swarm navigation," in *Robotics and Automation, ICRA 2008. IEEE International Conference on*, pp. 1904-1909, 2008.
[52] M. S. Couceiro, *et al.*, "A novel multi-robot exploration approach based on particle swarm optimization algorithms," in *Safety, Security, and Rescue Robotics, 2011 IEEE International Symposium on*, pp. 327-332, 2011.
[53] O. Trullier, *et al.*, "Biologically based artificial navigation systems: Review and prospects," *Progress in neurobiology*, vol/issue: 51(5), pp. 483-544, 1997.
[54] S. Thrun, "Probabilistic algorithms in robotics," *Ai Magazine*, vol/issue: 21(4), pp. 93, 2000.
[55] J. A. Meyer and D. Filliat, "Map-based navigation in mobile robots:: Ii. a review of map-learning and path-planning strategies," *Cognitive Systems Research*, vol/issue: 4(4), pp. 283-317, 2003.
[56] Koren and Borenstein, "Potential field methods and their inherent limitations for mobile robot navigation," in *Robotics and Automation, Proceedings., 1991 IEEE International Conference on*, pp. 1398-1404, 1991.
[57] S. X. Yang, *et al.*, "A layered goal-oriented fuzzy motion planning strategy for mobile robot navigation," *IEEE transactions on systems, man, and cybernetics—part b: cybernetics,* vol/issue: 35(6), pp. 1214-1224, 2005.
[58] K. M. Krishna and P. K. Kalra, "Perception and remembrance of the environment during real-time navigation of a mobile robot," *Robot. Automation System,* vol. 37, pp. 25–51, 2001.
[59] Y. H. Liu and S. Arimoto, "Path planning using a tangent graph for mobile robots among polynomial and curved obstacles," *The International Journal of*

Robotics Research, vol/issue: 11(4), pp. 376–82, 1992.

[60] Kant K. and Zucker S. W., "Towards efficient planning: the path velocity decomposition," *The International Journal of Robotics Research*, vol/issue: 5(1), pp. 72–89, 1986.

[61] K. Fujimura and H. Samet, "Accessibility: a new approach to path planning among moving obstacles," *Proceedings of IEEE conference on computer vision and pattern recognition, Ann Arbor, MI*, pp. 803–810, 1988.

[62] J. G. Lamadrid and M. L. Gini, "Path tracking through uncharted moving obstacles," *International Journal IEEE Transactions on Systems, Man and Cybernetics,* vol/issue: 20(6), pp. 1408–1422, 1990.