electronics

MDPI

Article

Path Planning Techniques for Real-Time Multi-Robot Systems: A Systematic Review †

Nour AbuJabal [1], Tamer Rabie [2], Mohammed Baziyad [1,*], Ibrahim Kamel [2] and Khawla Almazrouei [2]

[1] Research Institute of Research and Engineering, University of Sharjah, Sharjah P.O. Box 27272, United Arab Emirates; nabujabal@sharjah.ac.ae
[2] College of Computing & Informatics, University of Sharjah, Sharjah P.O. Box 27272, United Arab Emirates; trabie@sharjah.ac.ae (T.R.); kamel@sharjah.ac.ae (I.K.); u21201560@sharjah.ac.ae (K.A.)
* Correspondence: mbaziyad@sharjah.ac.ae

† This work is an expanded version of the conference paper published in the proceedings of the 2023 15th International Conference on Innovations in Information Technology (IIT), Abu Dhabi, UAE, 20–22 November 2023; pp. 1–6.

Abstract: A vast amount of research has been conducted on path planning over recent decades, driven by the complexity of achieving optimal solutions. This paper reviews multi-robot path planning approaches and presents the path planning algorithms for various types of robots. Multi-robot path planning approaches have been classified as deterministic approaches, artificial intelligence (AI)-based approaches, and hybrid approaches. Bio-inspired techniques are the most employed approaches, and artificial intelligence approaches have gained more attention recently. However, multi-robot systems suffer from well-known problems such as the number of robots in the system, energy efficiency, fault tolerance and robustness, and dynamic targets. Deploying systems with multiple interacting robots offers numerous advantages. The aim of this review paper is to provide a comprehensive assessment and an insightful look into various path planning techniques developed in multi-robot systems, in addition to highlighting the basic problems involved in this field. This will allow the reader to discover the research gaps that must be solved for a better path planning experience for multi-robot systems.

Keywords: systematic review; path planning; multi-robot; deterministic approaches; artificial intelligence approaches; hybrid approaches

check for updates

Citation: AbuJabal, N.; Rabie, T.; Baziyad, M.; Kamel, I.; Almazrouei, K. Path Planning Techniques for Real-Time Multi-Robot Systems: A Systematic Review. *Electronics* **2024**, *13*, 2239. https://doi.org/10.3390/electronics13122239

Academic Editor: Giuseppe Prencipe

Received: 25 March 2024
Revised: 20 April 2024
Accepted: 23 April 2024
Published: 7 June 2024

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

1. Introduction

In recent years, there has been a significant increase in the demand for robots, with widespread application across various sectors and industries. Multi-robot systems have demonstrated their effectiveness across a range of scenarios and applications, showcasing distinct advantages derived from coordinated interactions among multiple robots. The importance of multi-robot systems has grown substantially, particularly in areas such as manufacturing [1], transportation [2], healthcare [3,4], agriculture [5], logistics [6], and construction [7]. In manufacturing, robots are used to automate repetitive tasks, such as welding and assembly, which improves efficiency and reduces the need for human labor [8]. In transportation, self-driving cars and drones are being developed to improve safety and increase the efficiency of delivery systems [2]. In healthcare, robots are being used to assist with surgeries and to provide therapy to patients with physical or cognitive impairments. For example, manipulators play a significant role in the healthcare field in giving medication to patients and rehabilitating the elderly and disabled [9]. In construction, teams of robots can work together to build structures more quickly and with greater precision than humans [7]. In logistics, multi-robot systems can be used to move packages or pallets of goods more efficiently in warehouses [6]. In agriculture, agricultural robotics also benefits from multi-robot systems, particularly in tasks like planting, pesticide application, and harvesting.

Leveraging coordinated movements, robots traverse expansive fields with precision and speed, leading to increased crop yields and reduced labor requirements [5].

The deployment of systems involving multiple collaborating robots offers a range of benefits. Such systems enhance the efficiency and capability of performing tasks that might be challenging or unattainable for an individual robot [10]. For instance, search and rescue operations can be optimized with robot teams capable of covering larger areas more swiftly and efficiently than a solitary robot [11] Additionally, the risk of mission failure due to a single robot's malfunction is significantly reduced in a multi-robot system, as other robots can seamlessly continue the task, thereby enhancing the system's overall resilience.

Despite their numerous advantages, multi-robot systems confront certain limitations and drawbacks that warrant consideration. Coordination complexity poses a significant challenge, particularly in dynamic environments where algorithms for communication and collaboration must contend with uncertainty. Additionally, communication and sensing limitations can impede performance, especially in environments with obstacles or interference, necessitating robust solutions to ensure reliable information exchange among robots. Moreover, resource constraints such as limited energy and processing power may restrict the complexity of tasks that can be undertaken, demanding efficient resource management strategies. Ensuring robustness and fault tolerance is paramount, as failures in individual robots can jeopardize the entire system's performance.

Multi-robot systems may consist of either homogeneous or heterogeneous configurations. The primary distinction between these two types of systems lies in the capabilities and functionalities of the robots comprising the system [12]. Homogeneous systems are composed of identical robots, all equipped to undertake identical tasks, whereas heterogeneous systems are made up of robots each possessing distinct capabilities and functionalities. Homogeneous systems are typically used in applications where a large number of simple tasks need to be performed simultaneously, such as search and rescue, surveillance, and mapping. On the other hand, heterogeneous systems are typically used in applications where a variety of tasks need to be performed, such as construction, exploration, and transportation. Heterogeneous systems can also be used to overcome the limitations of homogeneous systems, such as a lack of redundancy or the ability to adapt to changing environments. Additionally, homogeneous systems are relatively easy to design and control compared to heterogeneous systems. However, heterogeneous systems have a much higher potential for flexibility and adaptability.

Examples of homogeneous systems encompass Unmanned Ground Vehicles (UGVs) [13], Unmanned Aerial Vehicles (UAVs) [14], Unmanned Surface Vehicles (USVs) [15], and Unmanned Underwater Vehicles (UUVs). Examples of heterogeneous systems include Unmanned Aerial Vehicles and Unmanned Ground Vehicle (UAV-UGV) systems [16,17]. These are systems that combine the capabilities of flying and ground robots. The UAVs can be used for tasks such as aerial surveillance, while the UGVs can be used for tasks such as ground-level inspection or traversing difficult terrain. Another example is Unmanned Surface Vehicles (USVs) and Unmanned Underwater Vehicles (UUVs). These are systems that combine the capabilities of surface and underwater robots. The USVs can be used for tasks such as surveillance on the water surface, while the UUVs can be used for tasks such as underwater exploration or inspection. Other heterogeneous systems include the UAV-Manipulator (Aerial Manipulator) [18] and UGV-Manipulator (Mobile Manipulator) [19].

This study seeks to categorize path planning methods according to a variety of criteria, including obstacles, destination, communication, shortest duration, and shortest route. It will assess the types of obstacles encountered during path planning, differentiating among static, dynamic, and scenarios devoid of obstacles. Additionally, the research will analyze if the destination changes or remains consistent with the starting point, and whether the destination is static or subject to change. Another crucial factor to take into account is whether there are capabilities for communication throughout the path planning phase. Furthermore, this document will investigate if the goal is to identify the quickest time or the most direct route to the destination. By organizing path planning strategies according

to these parameters, this research intends to deliver a thorough insight into the varied methodologies employed across different situations, thereby facilitating the choice of the most suitable technique for particular application needs.

The structure of the remainder of this document is outlined as follows: Section 2 provides a concise overview of path planning and offers a summary of the work related to this field. Section 3 identifies the methodology of conducting this review. Section 4 presents a comprehensive taxonomy, along with a review of the methods categorized within each class. Section 6 briefly outlines the main results of evaluating defense mechanisms from the literature and discusses the current challenges of multi-robot systems path planning. Finally, the concluding remarks are presented in Section 7.

2. Background and Related Work

Path planning is a fundamental problem in robotics and autonomous systems, involving the determination of an optimal path for a robot to navigate from a starting point to a desired destination. It considers obstacles and objectives like minimizing travel time or distance. Path planning algorithms focus on high-level decision-making, determining the sequence of waypoints to be followed. In contrast, trajectory planning is concerned with generating a smooth and feasible trajectory that adheres to the chosen path. It considers the dynamics and constraints of the robot, such as velocity and acceleration limits [20]. Trajectory planning computes the motion profile, including the desired velocity and acceleration, to ensure the generated trajectory is feasible and satisfies the constraints. In essence, while path planning focuses on finding the optimal path, trajectory planning focuses on generating the motion commands required to follow that path smoothly and accurately. Both components play crucial roles in enabling robots and autonomous systems to navigate complex environments effectively.

There have been several attempts to produce a review on path planning techniques for multi-robot systems in different types of environments. However, the reviews published recently in the literature have been limted in several ways. For example, they might only review a specific robot type such as Unmanned Ground Vehicle (UGV) systems [21–26], Unmanned Aerial Vehicle (UAV) systems [27,28], Unmanned Surface Vehicles (USV) [29], or Autonomous Underwater Vehicle (AUV) systems [30]. Other reviews neglect communication, which is an essential process while dealing with multi-robot systems and focusing on the challenges of path planning [31]. While others still study the coordination of multiple robot systems [32].

The authors in [21] provided a thorough review of path planning techniques for autonomous mobile robots. They classified path planning algorithms into four main categories: Soft-Computing, Reactive-Computing, C-Space Search, and Optimal Control Algorithms. The main characteristics that the authors considered from each path planning category were the Preliminary Map Model, Deterministic, Replanning, Optimality, and Completeness. The work in [22] reviewed path planning techniques for mobile robots. The review explored various classifications of path planning, taking into account both comprehensive and localized strategies, as well as examining both classical and heuristic methods. It categorized the approaches into global methods such as cell decomposition, local methods such as bug algorithms, and heuristic approaches such as artificial neural networks. Similar work was conducted in [23], where the path planning techniques are classified into classical approaches such as the roadmap approach and heuristics approaches such as particle swarm optimization. In addition, the paper took into consideration three main criteria: path length, smoothness, and safety degree.

Another comprehensive review of path planning strategies for mobile robot navigation was presented in [24]. The work categorized path planning algorithms into classical and reactive approaches, while the path planning algorithms were classified into classical and evolutionary approaches in [25]. Various path planning approaches were analyzed in different environmental conditions, including static and dynamic obstacles for single and multiple mobile robot systems. In addition, the advantages and disadvantages of both

path planning categories were highlighted in [25]. The authors in [33] provided a general overview of trajectory planning algorithms within multiple robot systems. These algorithms were classified into four main categories, including bio-inspired, metaheuristic model, sampling-based, and decomposition graph-based approaches. Moreover, a comparison between these approaches was conducted based on different criteria. These criteria include the applicability, importance, safety, advantages, and disadvantages of these approaches.

In [26], different approaches for multi-robot path planning and decision-making are presented, covering different types of robots, such as UAV, UGV, and AUV systems. The paper categorizes the multi-robot path planning approaches into classical, heuristic, bio-inspired, and artificial intelligence approaches. In terms of decision-making strategies, the focus is primarily on centralized and decentralized approaches. In [32], a detailed study on multi-robot system coordination was conducted. The authors started by discussing the basic terminology, classification, and different application aspects. In addition, they provided an overview and valuable perspectives on the coordination methods suggested for each application aspect. Furthermore, the strengths, limitations, and open research issues were extensively studied and a scope for future work was highlighted.

To the author's best knowledge, this document initiates an unprecedented exploration within its scope, introducing a classification framework that evaluates aspects such as the type of obstacles, the characteristics of the destination, the availability of communication, and the goals of optimization, which have not been tackled before. Despite extensive investigations into path planning methods, a detailed categorization and examination of these strategies according to the specified criteria have yet to be undertaken. This research aims to bridge this gap in the literature, offering valuable perspectives and laying the groundwork for subsequent studies in the realm of path planning and optimization. This paper conducts a comprehensive review following Kitchenham and Charters's guidelines for a Systematic Literature Review (SLR) [34]. Figure 1 shows a general block diagram of the proposed review methodology.

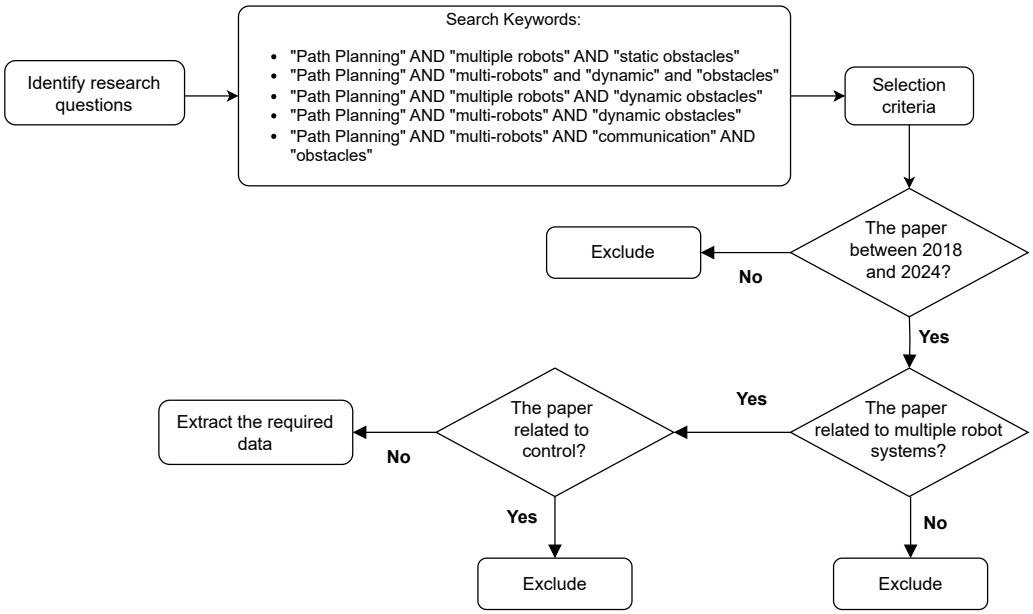


Figure 1. The logic flow of the review methodology.

3. Methodology

This survey follows the SLR guidelines presented in [34]. The presented methodology divides the search into three phases: the planning phase, the conducting phase, and the reporting phase. Through the planning phase, the researchers must identify the need for a systematic survey and the research questions that their work will answer, and they must define their review protocol. Then, the conducting phase includes establishing the search,

selecting primary studies, and finally extracting and analyzing data. The last phase is the reporting phase, in which the main report is formatted and the dissemination mechanism is specified.

The rest of this section includes important information about the employed methodology. First, the research questions are identified, and the search protocol is defined (planning phase). Then, the study selection criteria and the quality assessment tools are presented. Finally, the data extraction strategy is outlined (the conducting phase).

The search was restricted to include literature about path planning techniques for multi-robot systems that were published between 2018 and 2023. The applied search protocol resulted in the inclusion of 46 annually distributed articles, as Figure 2 shows.

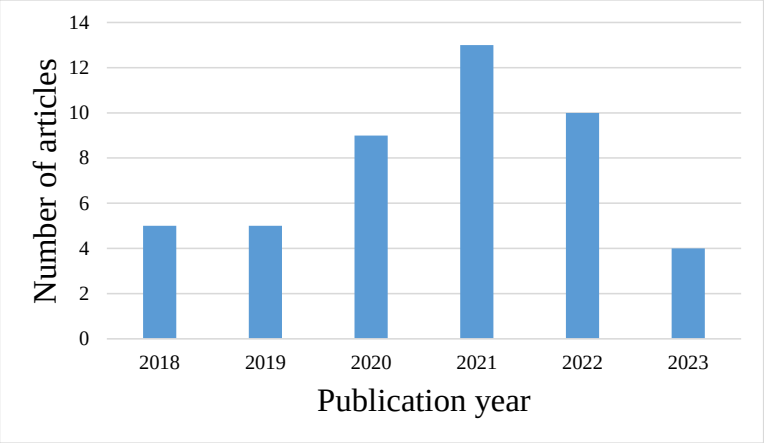


Figure 2. The number of included publications per year.

3.1. Planning Phase—Research Questions

This work was conducted to answer the following questions:

RQ1: What are the predominant path planning methods employed in multi-robot systems?

RQ2: Which path planning techniques are efficient while dealing with a huge number of robots?

RQ3: Do hybrid path planning strategies enhance the conventional methods in aspects such as duration, route distance, and trajectory smoothness?

RQ4: What are the primary obstacles faced by multi-robot systems that impact path planning parameters?

RQ5: Which methods are used in the literature for multi-robot path planning in dynamic environments?

3.2. Conducting Phase—Search Strategy

For this SLR, the following search strategy was implemented based on the objectives and research questions of this review. The search was conducted in three stages: the Term-based search, the Crawling-based search, and applying the inclusion/exclusion criteria.

In the Term-based search, we followed search terms and we conducted the search through some predefined sources using Google Scholar and other search engines. Through the Crawling-based search, we conducted a search of the literature reviewed by previous surveys in this area. Finally, we applied the inclusion/exclusion criteria to make sure that only relative works were involved in the survey. The following shows the search terms and resources that were used to conduct the Term-based search phase.

3.2.1. Search Keywords

Derived from the categories of path planning methods illustrated in Tables 1–3, the following search terms were chosen:

- "Path Planning" AND "multiple robots" AND "static obstacle";

- "Path Planning" AND "multi-robots" AND "dynamic" AND "obstacle";
- "Path Planning" AND "multiple robots" AND "dynamic obstacles" ;
- "Path Planning" AND "multi-robots" AND "dynamic obstacles";
- "Path Planning" AND "multi-robots" AND "communication" AND "obstacle";

Table 1. Summary of deterministic path planning approaches in the literature. The first class denoted as "Cell Decomp." refers to the class of "Cell Decomposition". The column "Dest." refers to the word "Destination". The column "Comm." denotes the word "Communication", which refers to the inner communications between the robots.

					Evaluation			
Class	**Technique**	**Paper**	**Year**	**Obstacles**	**Dest.**	**Comm.**	**Shortest Time**	**Shortest Path**
Cell Decomp.	A*	[35]	2021	Static	Different	Yes	No	Yes
		[36]	2021	Dynamic		No	No	
		[37]	2020	Dynamic		No	No	
		[38]	2023	Dynamic		No	Yes	
	Theta*	[39]	2022	Dynamic	Different	Yes	Yes	Yes
	Greedy algorithms	[40]	2020	Dynamic	Different	Yes	No	No
	Dijkstra	[41]	2021	Dynamic	Different	No	No	Yes
	Ubiquitous Trapezoidal	[42]	2023	Static	Different	Yes	Yes	No
Sampling	PRM	[43]	2023	Dynamic	Different	Yes	Yes	No
	RRT	[44]	2018	Static	Same	Yes	Yes	No
	RRT*	[45]	2021	Dynamic	Same	No	Yes	Yes
APF	-	[46]	2022	Non	Different	Yes	No	Yes
		[47]	2022	Static	Different	Yes	No	Yes
		[48]	2019	Static	Same	No	Yes	No

Table 2. Summary of AI path planning approaches in the literature.

						Evaluation		
Class	**Technique**	**Paper**	**Year**	**Obstacles (Static/Dynamic/ Obstacle Free)**	**Destination**	**Communication (Yes/No)**	**Shortest Time (Yes/No)**	**Shortest Path (Yes/No)**
Artificial neural network	RNN	[49]	2021	Dynamic	Same, Static	Yes	No	No
	Adaptive neural network	[50]	2018	Static	Same, Dynamic	Yes	Yes	Yes
Reinforcement learning	Globally guided reinforcement learning	[51]	2020	Dynamic	Same, Static	No	No	Yes
	Deep Q-learning	[52]	2021	Static	Different, Static	Yes	Yes	No
Bio-Inspired algorithms	GA	[53]	2019	Static	Different, Static	No	No	Yes
		[54]	2018	Obstacle free		No	Yes	No
		[55]	2020	Dynamic		Yes	No	Yes
	BFO	[56]	2020	Dynamic	Same, Static	No	No	Yes
	PSO	[57]	2021	Dynamic	Different, Static	No	No	Yes
	Ant-lion optimization	[58]	2020	Dynamic	Different, Static	Yes	No	Yes
	ACO	[59]	2022	Static	Same, Static	Yes	Yes	Yes
	Grasshopper optimization	[60]	2021	Dynamic	Same, Static & Dynamic	No	Yes	Yes

3.2.2. Study Selection

By applying the mentioned search strategy, 230 publications were obtained. Many of these publications were off-topic, and further filtration was needed to ensure that the included articles served the objectives of this survey. Therefore, the following inclusion/exclusion rules were employed.

- Selection criteria:

- Include studies in the range from January 2018 to December 2023 (6 years);
- Include papers presenting path planning techniques for multiple robot systems;
- Include papers presenting path planning for homogeneous and heterogeneous systems;
- Include papers presenting path planning for static and dynamic obstacles;
- Include papers presenting path planning for the same and different destinations;
- Include papers presenting the recent bio-inspired path planning techniques;
- Include studies in the English language only.

• Exclusion criteria:

- Exclude duplicated papers;
- Exclude papers related to a single robot system;
- Exclude review papers for a single robot system;
- Exclude papers related to control;
- Exclude studies with no clear publication information.

Table 3. Summary of hybrid path planning approaches in the literature.

					Evaluation				
Class	**Technique**	**Paper**	**Year**	**Obstacles (Static/Dynamic/ Obstacle Free)**	**Destination (Same/Different)**	**Destination (Static/Dynamic)**	**Communication (Yes/No)**	**Shortest Time (Yes/No)**	**Shortest Path (Yes/No)**
Hybrid techniques	Deterministic-optimization techniques	[61]	2022	Obstacle-free	Different	Static	No	Yes	Yes
		[62]	2022	Dynamic	Same	Dynamic	No	No	Yes
		[63]	2022	Dynamic	Different	Static	No	No	No
		[64]	2019	Dynamic	Different	Static	Yes	No	Yes
		[65]	2021	Static	Same	Static	Yes	No	Yes
		[66]	2020	Dynamic	Same	Static	No	Yes	Yes
	Multiple deterministic techniques	[67]	2018	Dynamic	Same	Static	Yes	No	Yes
	Multiple optimization techniques	[68]	2020	Dynamic	Same	Static	No	Yes	Yes
		[69]	2022		Same		No		
		[70]	2020		Different		No		
		[71]	2018		Different		Yes		
		[72]	2023		Different		Yes		
		[73]	2023		Different		No		
Other	Optimization algorithms	[74]	2022	Dynamic	Both	Static	No	No	Yes
		[75]	2021	Static	Different	Static	No	Yes	No
		[76]	2021	Static	Different	Dynamic	No	No	Yes
		[77]	2022	Static	Different	Static	Yes	Yes	Yes
		[78]	2021	Dynamic	Different	Static	Yes	No	No
		[79]	2019	Static	Different	Static	No	Yes	Yes
		[80]	2019	Dynamic	Different	Static	Yes	Yes	No
		[81]	2021	Dynamic	Different	Static	No	Yes	No

3.3. Data Extraction Strategy

After the literature passed through all of the above stages, the following data were extracted from the work based on the research questions and the objectives of the review:

- Article title, author(s) name(s), publication date and venue;
- The backbone concept or assumption of the literature;
- The used path planning technique;
- Type of robots;
- Type of obstacles;
- Type of target position;
- Existing communication between the robots;
- The objectives of the optimization process.

3.4. Data Synthesis

The data extracted will be delineated in the following sections of the review. As this information is regarded as the crucial part of this work, each set of data extracted will be illustrated in the most suitable way. For that, tables, figures, and written paragraphs will be used.

4. Path Planning Classification

Path planning approaches can be generally classified into three categories: deterministic approaches, artificial intelligence (AI)-based approaches, and hybrid approaches, as shown in Figure 3. There were specific criteria used to classify a given path planning technique into deterministic, AI-based, and hybrid categories. Deterministic approaches involve traditional algorithms that rely on predefined rules and mathematical models. On the other hand, AI-based methods are those techniques that mimic human-like intelligence in solving problems and include the ability to learn from data or experience. Examples of AI-based algorithms can involve artificial neural networks, reinforcement learning, and bio-inspired algorithms. Finally, hybrid approaches combine elements of both deterministic and AI-based techniques, leveraging the strengths of each. Usually, the core of a hybrid technique is a deterministic technique, while an AI approach is integrated to perform some adaptive tuning to some parameters of the deterministic model. The rest of this section shows a brief background for each category and a summary of several studies included in the literature.

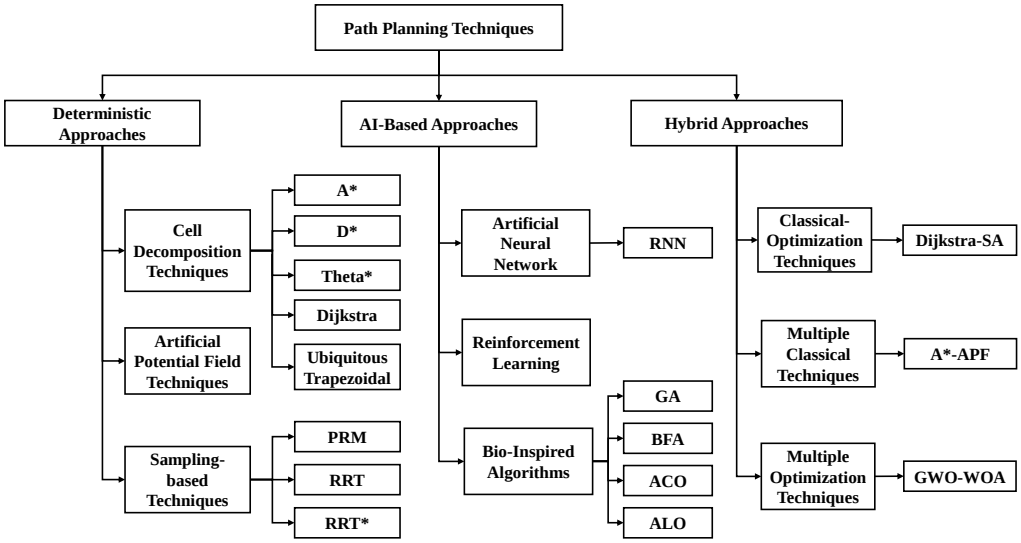


Figure 3. Classification of path planning techniques for multi-robot systems.

Path planning techniques often face a fundamental dilemma: optimizing one aspect often comes at the expense of another. Two critical attributes in path planning are the length of the generated path and the speed at which it can be executed. However, these attributes tend to be inversely related, presenting a challenge in simultaneously enhancing both.

On one hand, minimizing the path length is crucial for efficiency, as it reduces travel time, conserves energy, and often leads to more optimal routes. Shorter paths can be particularly advantageous in scenarios where time is of the essence or resources are limited. However, achieving a minimal path length may involve complex algorithms and meticulous optimization, which can potentially increase the computational overhead and execution time.

On the other hand, optimizing for execution speed is paramount in dynamic environments or real-time applications, where quick decision-making is essential. Faster path planning enables systems to respond promptly to changes in the environment, such as moving obstacles or evolving mission objectives. Yet, prioritizing speed may lead to suboptimal paths, sacrificing efficiency for rapid decision-making.

Balancing these opposing attributes poses a significant challenge in path planning research and development. Engineers and researchers continually strive to devise techniques that strike an optimal balance between path length and execution speed. This pursuit often involves exploring innovative algorithmic approaches, leveraging advancements

in computing hardware, and sometimes resorting to hybrid techniques that combine the strengths of different methodologies.

Ultimately, the ideal path planning technique would seamlessly navigate this trade-off, producing paths that are both efficient in length and swiftly executable. While achieving this balance remains a daunting task, ongoing advancements in artificial intelligence, optimization algorithms, and robotics hold promise for addressing this challenge and unlocking new possibilities in path planning across various domains. Therefore, in this work, each technique will be evaluated in the comparison tables based on its ability to obtain shorter paths in the shortest time possible.

4.1. Deterministic Approaches

Prior to the advent of artificial intelligence (AI) methodologies, traditional path planning strategies were widely adopted for navigating robots. These conventional techniques involved using mathematical models and algorithms to plan the movement of robots from one point to another. While deterministic path planning algorithms offer globally optimal paths and are easy to implement, they face difficulties in handling dynamic and unpredictable environments and can be computationally expensive. Additionally, these approaches may struggle to quickly adapt to unexpected obstacles or changes in the environment [24]. Therefore, these algorithms are not commonly used in real-time implementation. However, classical path planning approaches continue to be relevant and serve as a valuable foundation for modern path planning techniques. A variety of deterministic path planning techniques have been applied, including cell decomposition techniques [35–41], artificial potential field techniques [46–48], and sampling-based techniques [44,45]. The subsequent sections will explore the three subdivisions of traditional methods. Table 1 summarizes most of the work on deterministic path planning for multi-robot systems in the past five years.

4.1.1. Cell Decomposition Methods

Cell decomposition techniques divide the environment into geometric primitives (cell) such as polygons, rectangles, or trapezoids. Each cell corresponds to a distinct region a robot can navigate through or occupy. Grid-based techniques are examples of cell decomposition techniques where a cell is a uniform square forming a grid over the map. These algorithms are suitable for scenarios where the environment can be discretized and represented as a grid, such as indoor environments or structured outdoor spaces. Grid-based algorithms offer several advantages, including simplicity, efficiency, and global optimality. They are easy to implement and provide globally optimal paths, ensuring the robot reaches its destination using the shortest path. Additionally, grid-based algorithms can handle dynamic environments by updating the grid as the robot navigates. However, these algorithms have limitations. They are not well suited for continuous environments or those with complex obstacles that cannot be easily represented by the grid. Furthermore, the grid resolution affects the trade-off between accuracy and computational efficiency, as a finer grid increases precision but also requires more computational resources. In the following, four grid-based techniques including A*, Dijkstra, Theta*, and greedy algorithms will be discussed.

The A* algorithm is a highly favored path planning method that effectively identifies the shortest route between two points on a graph or grid. It is designed to be complete, ensuring it will locate the best solution if it exists. The Dijkstra algorithm, similar to A*, seeks the optimal path within a graph, specifically when the weights of the edges are known. However, unlike the A* algorithm, Dijkstra's method requires an exhaustive search of the entire graph to find the solution. As a result, A* tends to deliver superior performance in scenarios involving expansive environments. Several works utilized the A* algorithm in finding the optimal path for multi-robot systems [35–38]. The authors in [35] used the A* path planning technique in order to find the shortest path in an environment with static obstacles only. Additionally, each mobile robot was supposed to reach a different charging

point which had a fixed position. On the other hand, there is limited work involving the Dijkstra algorithm obtaining the shortest path. In [41], each mobile robot was supposed to reach its target position while taking into consideration dynamic obstacles using the Dijkstra algorithm. In addition, the robots had to avoid collisions while maintaining the path-priority order. Figure 4 demonstrates an example of finding an obstacle-free path using the A* path planning technique.

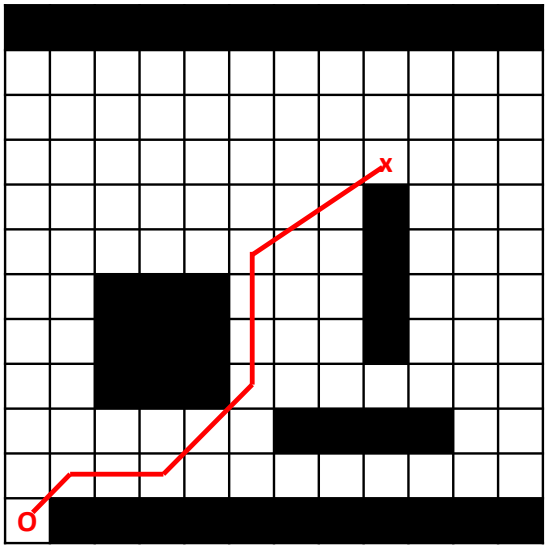


Figure 4. Finding an obstacle-free path using the A* grid-based technique.

The Theta* algorithm builds upon the A* algorithm, integrating further enhancements to boost its efficiency. Contrary to A*, Theta* permits a node's parent to be any other node within the network, not limited to its immediate neighbor, provided there is a direct line of sight connecting them. This approach minimizes the number of nodes that need to be examined by taking into account the direct visibility among consecutive nodes. In [39], the shortest path, which is presented by a set of line segments, was found utilizing Theta* algorithm in presence of dynamic obstacles. Moreover, the main goal was to optimize both the path length and the traveling time from the starting point to different targets. Additionally, the robots were able to communicate and share information about the environment with each other in centralized manner. Lastly, the greedy algorithm, while straightforward, serves as an efficient method in path planning for identifying viable solutions. This technique emphasizes the selection of the best local option at every step, disregarding the broader implications for the path as a whole or the future repercussions of these decisions. It systematically divides the available resources, giving precedence to the utmost immediate access to each resource throughout the process. In [40], a novel approach was proposed for the greedy informative path planning method combined with a novel load-balancing algorithm to recursively repartition the allocated Voronoi components of the mobile robots and achieve a better load-balanced information collection system in an unknown dynamic environment. Furthermore, radio-range communication was utilized for inter-robot communication.

The ubiquitous trapezoidal method is a known technique in path planning [42]. This method involves dividing the environment into trapezoidal regions and generating paths by connecting these regions. The trapezoidal method offers a simple yet effective way to plan paths that navigate around obstacles and reach designated goal locations.

The process typically begins by discretizing the environment into a grid or mesh, with each cell representing a trapezoidal region. These regions are then interconnected to form a network, where each trapezoid serves as a node in the graph. Path planning is accomplished by searching for a path through this network from the starting point to the goal location while avoiding collisions with obstacles.

One of the key advantages of the trapezoidal method is its efficiency in both computation and implementation. By decomposing the environment into simple geometric shapes, the method simplifies the path planning problem and reduces the computational complexity of finding feasible paths. Additionally, the method is relatively easy to implement and can be adapted to various types of environments and robot configurations. However, despite its simplicity, the trapezoidal method has limitations, particularly in handling complex environments with irregular obstacles or dynamic obstacles. In such cases, the method may struggle to generate optimal paths or may require additional techniques to improve path quality and robustness.

In [42], the paper discussed the importance of Coverage Path Planning (CPP) algorithms [82] in optimizing area coverage operations for autonomous robots, particularly in agricultural settings. Despite the repetitive nature of agricultural tasks, the adaptation of CPP algorithms for multi-robot operations remains largely unexplored. The work in [42] addressed this gap by proposing three forms of multi-robot coverage algorithms derived from single-robot CPP methods. Additionally, an optimized CPP algorithm was introduced for multiple in-row robots tasked with weed control in agricultural fields. By using a trapezoidal method, this algorithm, designed for offline planning, minimizes the distance traveled without repeated coverage, comparable to single-robot solutions. The effectiveness of the proposed algorithm is demonstrated through a simulation, showcasing its ability to make online adjustments and optimize coverage with an increasing team size. The quantitative evaluation revealed that, with a team size of 15, the proposed algorithm reduced the average distance traveled per robot by 65% compared to alternative algorithms, with further reductions as the team size increased. Ultimately, this algorithm offers a solution for autonomously covering fields with static obstacles, addressing a common need in agricultural processes.

4.1.2. Artificial Potential Field Methods

Artificial potential field (APF) algorithms represent a foundational path planning approach that models virtual forces within a potential field to direct robot movement. These algorithms generate forces that pull towards the objective while pushing away from obstacles, suitable for environments where obstacles are identifiable. A key strength of the APF approach is its efficiency, achieving desired paths with less computational effort than cell decomposition techniques or sampling-based techniques [83]. Moreover, its capability to adapt to changes in dynamic environments by updating forces in real time allows for the creation of smooth and direct routes towards targets. Nonetheless, APF methods encounter challenges, such as the risk of becoming ensnared in local minima or maxima within the potential field. Fine-tuning the potential field's parameters to optimally navigate around obstacles and efficiently reach goals can also prove difficult. Despite these challenges, the APF approach has been widely applied in existing research [46–48]. For instance, a novel algorithm based on APF was introduced in [47] to enable multiple robots to perform coordinated and track surveillance. The algorithm ensures efficient path planning from source to goal position, considering both two-robot and four-robot scenarios. It optimizes the path and time, effectively avoiding static obstacles along the way. Figure 5 shows an instance of finding an obstacle-free path using the APF approach.

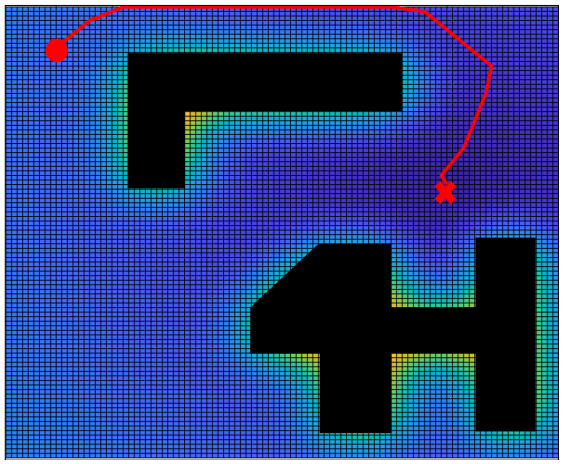


Figure 5. Finding the obstacle-free path using the APF approach.

4.1.3. Sampling-Based Techniques

Sampling-based methods constitute a category of traditional path planning algorithms that utilize random sampling as a means to investigate and move through an environment. These techniques generate a set of points or samples within the environment and connect them to form a graph or tree structure. Sampling-based techniques are particularly useful in scenarios where the environment is complex, continuous, or has high-dimensional state spaces. They are capable of finding feasible paths in high-dimensional spaces and can efficiently explore large state spaces. Additionally, sampling-based techniques are generally computationally efficient and can provide near-optimal solutions. However, sampling-based algorithms are not complete as they may not always guarantee finding the globally optimal solution. Furthermore, the performance of sampling-based techniques can be influenced by the density of samples, the choice of the sampling distribution, and the dimensionality of the problem.

The Rapidly Exploring Random Tree (RRT) [84] and Probabilistic Roadmap (PRM) [43] algorithms are popular path planning algorithms, which aim to sample the environment while considering the starting and ending positions. The RRT (Rapidly Exploring Random Tree) algorithm progressively forms a tree-like framework through the random sampling of points within the configuration space, subsequently integrating these points into the growing tree. This method prioritizes expansion into less explored regions, enabling swift traversal of the search area. RRT is particularly effective in situations where precision in the solution is not paramount, as its main aim is to swiftly produce viable routes. Conversely, the RRT* algorithm enhances RRT by incorporating a rewiring process that optimizes the tree's architecture through the consideration of proximal nodes. This modification permits RRT* to steadily refine and better select the paths it devises, progressively nearing an optimal solution as time progresses. Figure 6 illustrates an instance of finding an obstacle-free path using the PRM algorithm.

In [44], a decentralized version of the well-known RRT algorithm allows each agent to navigate through the environment while avoiding collisions with static obstacles and other agents of the team. Based on the experimental results applied on a heterogeneous system (UAV-UGV), the RRT algorithm is robust to uncertainties and capable of real-time replanning, but it exhibits some limitations related to scalability. The authors in [45] enhance the conventional RRT* algorithm and propose a New Potential Quick-RRT* (NPQRRT*), which incorporates the attitude adjustment angle of the robot into the path planning process. Furthermore, the algorithm is expanded to address the path planning problem in a system with multiple mobile robots. The simulation results indicate that the proposed algorithm exhibits improved dynamic obstacle avoidance capabilities. Additionally, it generates relatively better paths compared to the ordinary RRT planning algorithm.

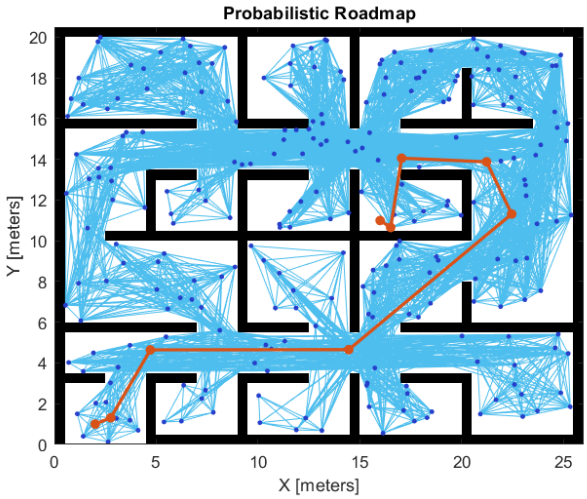


Figure 6. Finding an obstacle-free path using the PRM algorithm.

4.2. AI-Based Approaches

Artificial intelligence (AI)-based path planning methods have marked a substantial advancement beyond traditional strategies. In contrast to deterministic approaches, which depend on established mathematical models and algorithms, AI-driven path planning utilizes machine learning and additional AI technologies. This allows the algorithms to derive insights from data and adjust to environments that are both dynamic and uncertain. One major advantage of AI-based path planning is its ability to handle complex scenarios and make intelligent decisions in real time. It can learn from past experiences, optimize paths based on changing conditions, and quickly adapt to unexpected obstacles or changes in the environment. Furthermore, AI-based path planning approaches can provide more flexible and adaptive solutions, improving the overall efficiency and performance. However, these approaches also have limitations. They require large amounts of training data and computational resources, and their performance heavily relies on the quality and diversity of the training data. Additionally, the interpretability and explainability of AI-based path planning algorithms can be challenging, making it difficult to understand the reasoning behind their decisions. The following subsections discuss the three subcategories of AI-based approaches. Table 2 provides a summary of the recent research conducted in the past five years on path planning for multi-robot systems using artificial intelligence approaches.

4.2.1. Artificial Neural Network

An Artificial Neural Network (ANN) is a machine learning approach utilizing neural networks to determine optimal paths for robots. Some of the characteristics that make ANN-based systems valuable in the field of mobile robotic navigation are their ability to generalize, distributed representation, enormous parallelism, fault tolerance, and learning ability [24]. However, ANN-based techniques have the disadvantage of being time-consuming, and the learning method may not be able to ensure convergence with the optimal solution.

The authors in [50] presented the method for cooperative hunting utilizing a multi-robots system to handle high system dynamics. An adaptive neural network model was proposed based on an existing one. The current model was modified to solve challenges related to the repeated pursuit and search tasks to hunt many evaders. These challenges include the presence of static obstacles and the limited sensing range of the robots. The simulation results using various network types demonstrated the efficiency of the proposed approach in terms of the computational effort required. In [49], a decentralized approach for path planning with multiple quadrotors is presented. The approach utilizes Model Predictive Control (MPC) to incorporate considerations for the interactions of the robots

with obstacles and other robots. This is achieved through the implementation of a Recurrent Neural Network (RNN)-based trajectory prediction model.

4.2.2. Reinforcement Learning

Reinforcement learning (RL) is a method utilized in path planning that enables an agent to sequentially make decisions aimed at optimizing a total reward. Within RL-driven path planning, the agent engages with its surroundings, gains feedback through rewards, and modifies its behavior accordingly. RL proves especially beneficial in situations involving uncertain, intricate, or evolving environments, facilitating the agent's development of optimal strategies via exploration and utilization. One of the key advantages of RL-based path planning is its ability to handle uncertain and changing environments, adapting the agent's behavior accordingly. Therefore, there is no need to have predefined data to train the agent since the agent will learn from experience and observations from the environment. RL can also handle high-dimensional state spaces and provide robust solutions. However, RL also has limitations. It often requires a substantial amount of training time and computational resources. The quality of the solution heavily depends on the reward design and exploration strategy. Additionally, RL may struggle with sample inefficiency and cannot guarantee optimal performance in all scenarios.

In path planning, various reinforcement learning techniques are frequently employed, including Q-learning [85] and Deep Q-Networks (DQN) [52]. In [51], a hierarchical framework that combines global guidance and local RL-based planning to enable end-to-end learning in dynamic environments is presented. The local RL planner exploits both spatial and temporal information within a local area to avoid potential collisions and unnecessary detours. In addition,the authors designed a novel reward structure that provides dense rewards, while not requiring the robot to strictly follow the global guidance at every step, thus encouraging the robot to explore all potential solutions. The experimental results validated the robustness, scalability, and generalizability of the proposed approach. In [52], a Deep Q-learning (DQN) technique for high-level heterogeneous system (UAV-UGV) path planning is introduced. DQN is easily modifiable for learning desired multi-agent behavior and finds global solutions, as was shown through experiments.

4.2.3. Bio-Inspired Algorithms

Bio-inspired methods, often known as biomimicry, include a variety of approaches that are inspired by the natural world. These methods aim to replicate the forms, actions, and functions found in diverse natural systems, such as the human body, animals, and plants. These techniques are employed to devise innovative technologies and systems with the objective of problem-solving and enhancing performance across domains such as robotics and computer science. Bio-inspired techniques in path planning draw inspiration from biological systems and processes to solve navigation problems. Bio-inspired techniques offer advantages in handling complex and dynamic environments, as they can adapt to changing conditions and find optimal solutions. They are particularly more useful than traditional algorithms in handling uncertainties or when there is a need for distributed decision-making. Bio-inspired techniques can provide robustness, scalability, and parallel processing capabilities. However, they also have limitations as they can require significant computational resources and parameter tuning.

The literature features various bio-inspired algorithms, among which genetic algorithms (GAs) stand out [53–55], as do Particle Swarm Optimization (PSO) [57], Ant Colony Optimization (ACO) [59], Bacterial Foraging Optimization (BFO) [56], Ant Lion Optimization (ALO) [58], and the grasshopper algorithm [60].

Genetic algorithms are inspired by the process of natural selection and genetics. They use a population of individuals representing potential solutions, and apply selection, crossover, and mutation operators to evolve and improve the solutions over generations. In [55], a global path planning strategy for multi-robot systems with differential drive and Bluetooth communication solved by a genetic algorithm was developed. The efficiency

of the proposed technique was validated through experimental study implemented in a semi-unknown environment with static and dynamic obstacles.

The grasshopper algorithm mimics the characteristics of grasshoppers to solve optimization problems. Grasshoppers are known for their jumping ability and their ability to locate the best positions for feeding or reproducing. The authors in [60] proposed a novel path planning technique using the grasshopper algorithm in unknown and dynamic environments. The algorithm was implemented in different scenarios. For example, the algorithm was used to obtain the optimal path with the minimum arrival time in the presence of a dynamic target among different types of obstacles (static and dynamic). The simulation results proved the effectiveness of the proposed technique compared with other techniques such as GAs and PSO.

4.3. Hybrid Approaches

Hybrid path planning strategies integrate diverse approaches or techniques to develop the best possible solutions for path planning. These techniques integrate the strengths of different algorithms to overcome the limitations of individual approaches. The advantages of hybrid path planning techniques lie in their ability to handle complex and dynamic environments more effectively. By exploiting the complementary strengths of different methods, they can provide more robust and adaptable solutions. Hybrid techniques can exploit the efficiency of deterministic methods while incorporating the learning capabilities of AI-based approaches. However, the design and implementation of hybrid techniques can be complex and require the careful integration of different algorithms. The performance of hybrid techniques heavily relies on the appropriate selection and combination of individual methods. In this paper, hybrid methods are categorized into three primary groups: deterministic-optimization, various deterministic, and several optimization techniques. Table 3 summarizes the recent work conducted in the past five years on path planning for multi-robot systems using hybrid approaches.

Numerous studies in the literature have amalgamated diverse path planning strategies to create hybrid approaches. Different methodologies have been employed alongside cell decomposition algorithms, including the Dijkstra algorithm and the Simulated Annealing (SA) approach. In [61], A* and potential field [67], A* and reinforcement learning [63], A* and the Dynamic Window Algorithm [66], Theta* and dipole field with the dynamic window approach [64] are explored. Other studies concentrated on merging various optimization strategies to address the complexities in path planning, for instance, integrating the Wolf Swarm Algorithm with the artificial potential field (WSA-APF) [62], the kidney-inspired algorithm and Sine–Cosine Algorithm (KA-SCA) [70], Artificial Bee Colony and Evolutionary Programming (ABC-EP) [71], the Modified Hyperbolic Gravitational Search Algorithm and Dynamic Window Approach (MGSA-DWA) [72], the Self-Organizing Migrating Algorithm and Particle Swarm Optimization (SOMA–PSO) [73], the Grey Wolf Optimizer and Whale Optimizer Algorithm (GWO-WOA) [69], and the Dynamic Window Approach (DWA) and Teaching–Learning-Based Optimization (TLBO) [68]. Finally, the combination of sampling-based and optimization algorithms was employed in [65] in order to solve the path planning problem.

5. Communication Style Classification

Multi-robot systems encounter several challenges, including managing a large number of robots, ensuring energy efficiency, maintaining fault tolerance, and adapting to dynamic targets. Various approaches have been proposed to address these challenges. In centralized approaches, a central controller coordinates the actions of all robots, offering a streamlined method to manage a large number of robots efficiently. This centralization facilitates optimization of energy usage across the entire fleet of robots by considering global factors such as distance, workload distribution, and battery levels. Additionally, centralized systems can incorporate redundancy and robust error-handling mechanisms to detect and recover from robot failures, ensuring uninterrupted operation even in the presence of faults.

Moreover, central controllers can adapt to dynamic targets by continuously updating plans and reassigning tasks based on real-time information, allowing for efficient path planning in dynamic environments.

In contrast, decentralized approaches distribute decision-making among individual robots or small groups, alleviating the burden on any single robot or controller. Decentralized algorithms can promote energy efficiency by allowing robots to communicate and collaborate locally, minimizing unnecessary movements and conserving energy. Furthermore, decentralized architectures inherently provide fault tolerance by distributing decision-making and control, enabling the system to adapt to failures without central intervention. Decentralized systems enable robots to dynamically respond to changing targets or environmental conditions based on local perception and communication with neighboring robots, autonomously adjusting their path planning and behavior to track and reach dynamic targets effectively.

Common strategies employed across different approaches include communication, collaborative decision-making, and distributed task allocation. Communication between robots facilitates collaboration, information exchange, and the synchronization of movements, improving the overall system performance. Collaborative decision-making allows robots to work together to achieve common objectives, utilizing consensus-building, negotiation, or voting mechanisms to reach an agreement on actions and strategies. Distributed task allocation techniques balance workload and optimize resource utilization by distributing tasks efficiently across the robot team, enhancing system efficiency and scalability.

This section provides an overview of the classification approaches based on communication style, focusing on three primary categories: centralized, decentralized, and distributed. Figure 7 shows a comparison between centralized, decentralized, and distributed robotic systems.

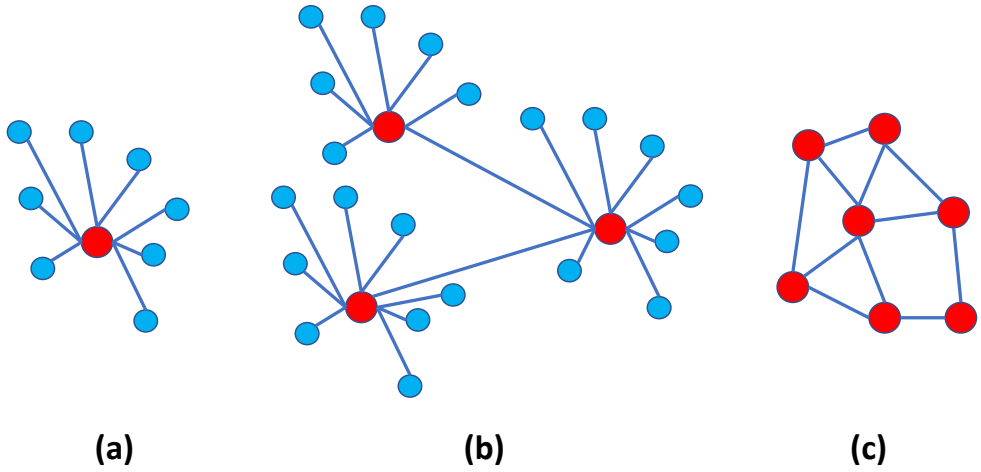


Figure 7. A comparison between (**a**) centralized, (**b**) decentralized, and (**c**) distributed robotic systems.

5.1. Different Robotic Systems

There are various robotic platforms such as Unmanned Ground Vehicles (UGVs), Unmanned Aerial Vehicles (UAVs), Unmanned Surface Vehicles (USVs), and Unmanned Underwater Vehicles (UUVs). Each has its own unique challenges and constraints for path planning methods. Here are some distinctive characteristics and limitations of path planning across these diverse platforms.

5.1.1. UGVs

UGVs operate on land, traversing diverse terrain types including urban environments, off-road landscapes, and structured indoor spaces [61,73]. Path planning for UGVs must contend with obstacles such as buildings, vehicles, pedestrians, and natural terrain features like slopes and rough terrain. Additionally, UGVs face challenges related to dynamic

environmental conditions such as changing weather and lighting conditions, as well as real-time traffic and pedestrian dynamics. Path planning algorithms for UGVs must, therefore, be robust, adaptable, and capable of efficiently navigating complex and dynamic environments while ensuring safety and collision avoidance.

5.1.2. UAVs

UAVs operate in three-dimensional airspace, encountering obstacles such as buildings, trees, power lines, and other aerial vehicles [40,49]. Path planning for UAVs must address airspace regulations, varying weather conditions, and energy constraints. Furthermore, the dynamic nature of aerial environments demands real-time adaptation to changing wind patterns, airspace restrictions, and mission objectives. Path planning algorithms for UAVs must be agile, responsive, and capable of dynamically adjusting flight paths to optimize efficiency, safety, and mission success.

5.1.3. USVs

USVs navigate through waterways, coastal regions, and maritime environments, facing challenges such as water currents, tides, waves, and maritime traffic [29,86]. Path planning for USVs must account for dynamic hydrodynamic effects, maritime regulations, and environmental uncertainties. Additionally, USVs may encounter obstacles such as buoys, ships, and marine infrastructure that pose collision risks. Path planning algorithms for USVs must, therefore, incorporate hydrodynamic modeling, collision avoidance strategies, and real-time adaptation to changing maritime conditions to ensure safe and efficient navigation.

5.1.4. UUVs

UUVs operate in the challenging underwater environment, navigating through depths, currents, and complex underwater terrain [30]. Path planning for UUVs must address limitations in underwater communication, sensing, and navigation accuracy. Additionally, UUVs face constraints related to battery life, mission duration, and underwater visibility. Path planning algorithms for UUVs must be robust, adaptive, and capable of autonomously navigating through underwater obstacles, avoiding collisions, and optimizing energy efficiency to extend mission endurance.

5.2. Centralized Communication

Centralized communication in multi-robot systems path planning involves a single central entity or controller that manages and coordinates all the robots. This controller typically collects information from all robots, processes it centrally, and then provides each robot with a predefined path or set of instructions. This approach offers precise control over the entire robot team, ensuring optimal path planning and collision avoidance. Centralized communication is particularly useful when robots need to execute tightly coordinated movements, such as in warehouse automation or precision manufacturing.

However, there are certain drawbacks associated with a centralized approach. Firstly, the central controller can become a single point of failure, posing a potential risk to the entire system's reliability. Secondly, as the number of robots increases, the overhead in communication may rise, potentially leading to delays in execution. Lastly, the central controller necessitates access to real-time data from all robots, a requirement that might not be feasible in all environments [87].

5.3. Decentralized Communication

Decentralized communication empowers individual robots to make path planning decisions independently or in small groups. Each robot communicates and collaborates with its nearby neighbors to exchange information about their local environments and collectively make path planning decisions. This approach is suitable for scenarios where

robots need to adapt to dynamic environments or when there is no single central entity capable of coordinating all robots effectively.

Decentralized communication offers advantages in terms of fault tolerance and scalability. Since each robot operates independently, there is no single point of failure, and the system can easily accommodate new robots without significant changes. However, achieving global optimization can be challenging, and robots may need to compromise on their individual paths to avoid collisions and conflicts [12].

5.4. Distributed Communication

Distributed communication combines elements of both centralized and decentralized approaches. In a distributed communication system, robots are organized into smaller groups or clusters, each with its own local coordinator or controller. These coordinators communicate with each other to exchange information and coordinate the actions of their respective groups. This approach strikes a balance between centralization and decentralization, offering the benefits of both.

The advantage of employing a distributed approach lies in its ability to effectively handle large-scale systems while offering a considerable level of flexibility and autonomy. Nevertheless, it requires efficient communication protocols and a robust network infrastructure to guarantee reliable and timely communication between the objects [88].

6. Results and Discussion

Up until this stage, this research has examined over 40 studies focused on path planning within multi-robot systems. The growing interest in multi-robot system path planning stems from its complex yet captivating nature, offering vast opportunities to transform numerous sectors. The drive to delve into path planning for multi-robot systems arises from its potential to significantly improve coordination, boost efficiency, and enhance the overall performance of systems. The collaborative effort of multiple robots, steered by sophisticated path planning techniques, can expedite task completion, elevate productivity, and expand system scalability. This section will proceed to outline notable observations regarding the strengths and weaknesses of different path planning strategies mentioned in the literature and will tackle the prevailing challenges and unexplored areas in multi-robot system path planning.

6.1. Method Evaluation

Choosing which method/s to employ is a crucial step when planning to build a system, and, as one of the main objectives of this survey is to guide and assist future research in this area, this section will briefly view the strengths and weaknesses of the main methods employed in the reviewed literature. Figure 8 displays the proportion of articles within each category of path planning.

Based on the work conducted in [47], it is observed from the results of the performance-measuring parameters path length and travel time that the proposed APF algorithm outperforms the existing A*, RRT, and GA algorithms.

The work in [41] showed that the proposed Dijkstra algorithm can complete the task effectively and has better performance in the average trajectory length than those using the benchmark methods of the Shortest Distance Algorithm (SDA) and Reciprocal Orientation Algorithm (ROA). However, it is not guaranteed that the performance of the proposed technique is the best compared to other methods when the number of robots increases or the working environment is changed.

In [76], a smart distance metric-based approach was utilized to find the shortest path in a multi-robot systems for warehouse applications. Based of their findings, the proposed method performs better when compared with other approaches like the A* and Integration Linear Programming (ILP) methods.

In [53], a comparative study between the proposed EGA approach and the A*, PRM, B-RRT, and PSO approaches showed the superiority of the proposed EGA approach in

comparison with the mentioned well-recognized path planning algorithms in terms of path length, path smoothness, runtime, and success rate. The comparative study was performed for 12 environments with various sizes and complexities. The comparative study showed that the success rate of the proposed EGA approach is always 100%, no matter how large and complex the environment is.

In [60], the proposed grasshopper-based technique is compared with PSO, the GA, D*, Neuro-Fuzzy, and RRT*. The simulation results show that the proposed technique successfully guided the robot towards the target, effectively avoided collisions, and found the shortest path in less time compared with the other techniques.

In [70], the robustness and effectiveness of the hybrid KA–SCA algorithm were verified through a comparison with the state of the art, including the Gravitational Search Algorithm, Particle Swarm Optimization (GSA-PSO), Improved Particle Swarm Optimization, and the Gravitational Search Algorithm (IPSO–IGSA). The proposed algorithm minimized the path deviation, the trajectory path length, the number of turns, and the computational cost.

In [71], the proposed hybrid strategy was tested for navigation performance on a collection of benchmark maps against the A* algorithm, PSO with clustering-based distribution factor, the GA, and RRT for path planning. Navigation effectiveness was measured using the smoothness of the feasible paths, the path length, the number of nodes traversed, and the algorithm execution time. The results show that the proposed method produced good results in comparison to the others. For larger maps size, the GA, RRT, and A*-based approaches either took a much longer time or stalled the system due to the amount of computation. The proposed sensor-based approach worked fine with large maps.

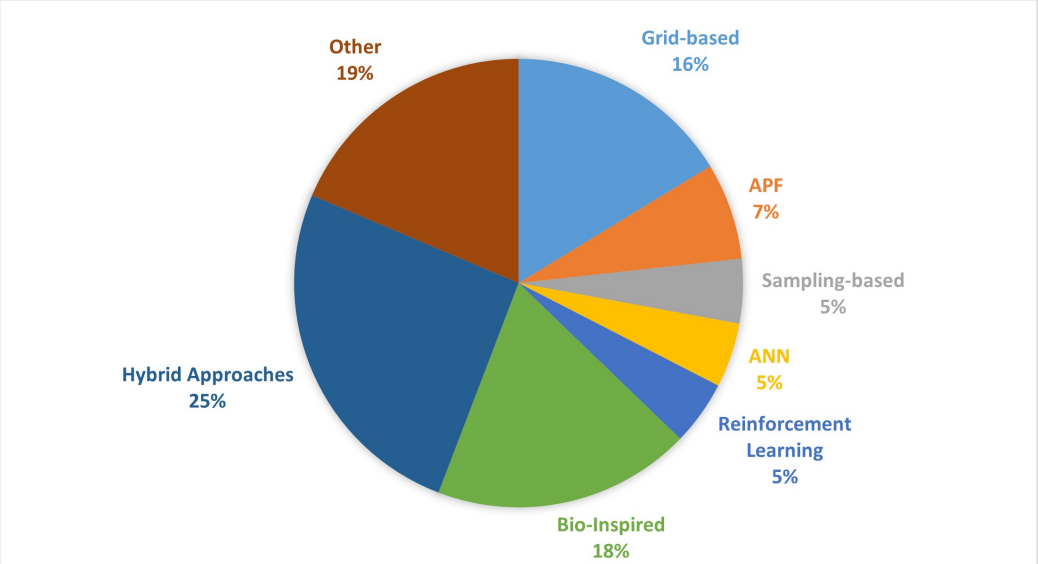


Figure 8. The percentage of articles under each category of path planning

In [63], the authors compared the proposed technique with the DWA algorithm. The results show that there was little variability between the two algorithms in multi-agent navigation without obstacles. However, in scenes with obstacles, the obstacle avoidance effect based on the DWA algorithm was worse in terms of success rate.

In [57], the proposed Jumping mechanism Particle Swarm Optimization (JPSO) algorithm was compared with two other algorithms: Genetic Learning PSO (GLPSO) and the PID-based strategy for PSO (PBS-PSO). It was shown that the proposed algorithm had a faster convergence speed and a higher precision than the other two algorithms.

In [38], the effectiveness of the proposed Bidirectional Alternating Jump Point Search A* (BAJPSA*) algorithm was verified by selecting two sets of maps with scales of 30 × 30 and 100 × 100 for simulation and compared with the A* and Jump Point Search (JPS)

algorithms. BAJPSA* achieved less run times in both map sets. However, the other two algorithms obtained shorter path lengths than BAJPSA* in the 100 × 100 map.

The authors in [52] evaluated the DQN algorithm against other standard 2D algorithms, including RRT, RRT*, and A*, focusing on the computational time required. The findings indicated that, on average, DQN demonstrated the shortest computational time while achieving a path length to the goal that was similar to that of the A* algorithm.

The authors in [59] compared the proposed ACO-based technique with the traditional Ant Colony Algorithm (ACA), Double Layer Ant Colony Algorithm (DLACA), Retraction Mechanism Ant Colony Algorithm (RMACA), and Evolutionary Ant Colony Algorithm (EACA) considering the path length, time consumption, iteration times, and number of turns. The proposed algorithm outperformed the other techniques in all aspects except time consumption.

6.2. Research Gaps

As discussed in the previous sections, many challenges were found to be hindering the development of research in path planning for multi-robot systems. These obstacles encompass the variety of robotic systems, the quantity of robots within the system, energy conservation, fault resilience and durability, underlying assumptions, the configuration of obstacles, and moving targets. In this section, some gaps in the current literature will be identified and discussed.

Firstly, most of the work in the literature focuses on path planning approaches for homogeneous systems. Although considerable work has been conducted on heterogeneous multi-robot systems in recent years, still, the diversity of robots is very limited. Addressing path planning for heterogeneous multi-robot systems presents unique challenges compared to homogeneous systems due to the diversity in capabilities, characteristics, and constraints of the robots involved. The key differences lie in accommodating varying robot types, sensor configurations, mobility capabilities, and task requirements within the same system. Effectively addressing these disparities requires tailored approaches that leverage the strengths of each robot type while mitigating their limitations.

One key difference is in the coordination and communication aspects. Heterogeneous systems often involve robots with different communication protocols, sensing ranges, and processing capabilities. Coordinating their actions and sharing information effectively becomes more complex, requiring adaptive communication strategies and protocols that account for the diverse capabilities of the robots involved. Additionally, the diversity in mobility capabilities can lead to challenges in synchronizing movements and maintaining cohesion within the team, necessitating flexible coordination algorithms that can accommodate different speeds, maneuverability, and traversal capabilities.

Moreover, heterogeneous systems may involve robots with specialized functionalities suited for specific tasks or environments. This introduces the need for task allocation and assignment strategies that consider the capabilities and constraints of each robot type, ensuring the efficient utilization of resources and optimal task performance. Adaptive task allocation algorithms that dynamically assign tasks based on real-time conditions and robot capabilities can help address this challenge.

Furthermore, path planning for heterogeneous systems must consider the trade-offs between optimizing different objectives such as the path length, energy consumption, and task completion time across multiple robot types. Multi-objective optimization techniques that balance these objectives while considering the heterogeneity of the system can lead to more robust and effective path planning solutions.

In future research, addressing the gap in path planning approaches for heterogeneous multi-robot systems will require interdisciplinary efforts combining robotics, artificial intelligence, optimization, and control theory. Developing adaptive algorithms that can adapt to the dynamic nature of heterogeneous systems, learn from experience, and evolve over time will be crucial. Additionally, integrating advanced sensing and communication

technologies to enable seamless interaction and collaboration among heterogeneous robots can enhance system performance and scalability.

Another important parameter to be considered while deploying multi-robot systems is energy efficiency, which has been very much neglected in the presented works. The properties required for a path planning approach to be energy efficient are as follows: minimizing the overall distance traveled by robots, an efficient communication type, and reduced computation requirements. Robustness is another challenge that should be taken into consideration. There are different situations that could degrade the robustness in a multi-robot system such as communication failure between robots, e.g., robots may move out of communication range, on-robot sensors failure, and leader failure. Robustness in terms of communication needs to be addressed and must consider communication range, communication failure, network partition recovery, and low bandwidth, because most of the work assumes a reliable communication medium.

In most of the papers, a few assumptions are made, such as a limited boundary in which robots are moving, a constant velocity of the robots, all sensory data being available, and a restricted angle. Additionally, the variation in obstacle shapes is not addressed in most of the existing literature. Finally, only a few studies propose a single dynamic target, such as [50,60,62,77]. Furthermore, managing multiple dynamic targets will present a greater challenge.

Another challenge is the number of robots in the system since increasing the number of agents will affect the performance and the efficiency of the proposed technique in finding the optimal path. For instance, the PONA algorithm proposed in [74] does not guarantee good performance in a system with more than eight robots.

6.3. Future Research

In light of the rapid advancements and increasing adoption of multi-robot systems in various real-world applications, the pursuit of future research avenues holds immense significance in furthering the capabilities and effectiveness of these systems. As the demand for versatile, scalable, and robust multi-robot solutions continues to grow, it will become imperative for researchers to explore novel approaches, methodologies, and technologies that address the emerging challenges and push the boundaries of our current capabilities.

Regarding the scalability issue, enhancing the scalability of path planning algorithms for larger robot populations requires adopting strategies and methodologies that can efficiently handle the increased computational complexity and coordination challenges. Here are several potential approaches to address this issue.

1. Decentralized and distributed approaches: Moving away from centralized control towards decentralized or distributed approaches can improve scalability by distributing decision-making among individual robots or small groups. Decentralized algorithms allow robots to make local decisions based on local information, reducing the computational burden on any single controller and enabling the system to scale more effectively as the number of robots increases [48];
2. Hierarchical planning: Hierarchical planning divides the path planning problem into multiple levels of abstraction, with higher levels focusing on global planning and coordination and lower levels handling local navigation and obstacle avoidance. This hierarchical structure allows for efficient scalability by delegating specific tasks to different levels of the planning hierarchy, thereby reducing the computational complexity of the overall system [86];
3. Task allocation and coordination: Efficient task allocation and coordination mechanisms are essential for scalability in multi-robot systems. Adaptive algorithms that dynamically allocate tasks based on robot capabilities, workload, and environmental conditions can optimize resource utilization and prevent bottlenecks. Additionally, effective coordination strategies, such as task prioritization and task scheduling, ensure that robots work together cohesively without unnecessary overlap or conflicts [75].

Future research focusing on developing path planning approaches that prioritize energy conservation and robustness in real-world scenarios, particularly in the face of challenges such as communication failures and sensor malfunctions, can adopt several strategies.

1. Energy-aware path planning: Path planning algorithms can be designed to prioritize energy-efficient routes by considering factors such as terrain elevation, surface friction, and the energy consumption rates of individual robots. Techniques such as dynamic programming or optimization-based methods can be employed to minimize energy consumption while satisfying path constraints and mission objectives. Additionally, integrating energy-awareness into task allocation and scheduling algorithms can further optimize resource utilization and prolong the operational lifespan of multi-robot systems [89];
2. Fault-tolerant path planning: Developing fault-tolerant path planning approaches that can adapt to communication failures and sensor malfunctions is crucial for ensuring robustness in real-world scenarios. Techniques such as redundant communication paths, consensus-based decision-making, and decentralized coordination can enable robots to maintain communication and coordination even in the presence of failures. Moreover, path planning algorithms can incorporate contingency plans and alternative routes to mitigate the impact of sensor failures or environmental uncertainties on path execution [90];
3. Distributed sensing and communication: Future research can explore distributed sensing and communication strategies that enhance resilience and adaptability in multi-robot systems. Decentralized sensing techniques, such as collaborative mapping and localization, enable robots to share environmental information and build a collective understanding of the surroundings without relying on a centralized infrastructure. Similarly, distributed communication protocols, such as ad hoc networking and mesh networks, facilitate robust communication among robots by dynamically establishing communication links and adapting to changing network conditions [48].

To address the identified research gaps in heterogeneous systems and dynamic target path planning within the field of multi-robot path planning, several specific recommendations and suggestions can be proposed.

1. Diverse benchmarking and evaluation: Develop standardized benchmarks and evaluation metrics specifically tailored for heterogeneous multi-robot systems and dynamic target scenarios. This would facilitate the fair comparison and rigorous evaluation of different path planning algorithms, enabling researchers to identify strengths, weaknesses, and areas for improvement;
2. Hybrid approaches: Among the three categories, i.e., the deterministic, the AI-based, and the hybrid method, the hybrid methods have still not been studied carefully, as they represent only 25% of the works presented in the literature, as shown in Figure 8. To address the identified research gaps in heterogeneous systems and dynamic target path planning, it is suggested to conduct more investigations into hybrid path planning approaches that seamlessly integrate deterministic techniques with AI-based methods to leverage the strengths of both paradigms. By combining deterministic algorithms with learning-based strategies, hybrid approaches can effectively address the challenges posed by heterogeneous systems and dynamic targets, offering adaptive and robust solutions;
3. Decentralized coordination: Developing decentralized coordination algorithms that enable heterogeneous robots to collaborate and coordinate their actions autonomously without relying on central control. Decentralized approaches promote scalability, fault tolerance, and flexibility, making them well suited for dynamic environments and scenarios with diverse robot capabilities. Based on Table 4, only a few techniques have adopted decentralized coordination.

Table 4. Summary of communication style approaches in the literature.

Communication Style	Robot Type	Papers	Limitations
Centralized	UGV	[36,37,39], [41,45,57], [62,69,71], [47,59,61], [66,67,70], [73]	- Single point of failure. - Scalability issues. - Latency Concerns.
	Humanoid	[64,72]	
Decentralized	UAV	[40,49]	- Difficulty in maintaining consistency. - Difficulty in handling conflicts. - Dependency on local perception.
	UGV	[63,65]	
	Heterogeneous	[44]	
Distributed	UGV	[35,38,71], [48,51,55], [46,56]	- Communication overhead. - Complexity of coordination.
	Heterogeneous	[52]	

Strategies to Move From a Single Robot System to a Multi-Robot System

There are several strategies to move from a single-robot system to a multi-robot system. Here is an overview of these strategies.

1. Transitioning to a distributed system [48]: Transitioning from single to multi-robot path planning through a distributed coordination system involves endowing individual robots with autonomy to make local decisions based on local information. In single-robot path planning, the focus is on generating a path for a single agent based on its perception of the environment and predefined objectives. In contrast, distributed coordination allows multiple robots to interact with their immediate surroundings, exchanging information and adjusting their paths autonomously. This transition is facilitated by enabling each robot to perceive its environment independently and make decisions based on its local observations. Instead of relying on a central controller to dictate the actions of all robots, distributed coordination distributes decision-making authority among individual agents. Through communication protocols or direct sensing, robots share information about their intended paths, obstacles encountered, and other relevant factors, allowing them to coordinate their movements and avoid collisions in real time. By transitioning from centralized control to decentralized coordination, multi-robot systems can achieve scalability and adaptability, as each robot can respond autonomously to changes in the environment without the need for explicit coordination from a central authority. This approach enables robots to navigate complex environments collaboratively while mitigating the computational and communication overheads associated with centralized control;
2. Cooperative path planning [7]: In the transition from single to multi-robot path planning through cooperative path planning, the focus shifts from individual robot autonomy to collaborative decision-making among multiple robots. Single-robot path planning typically involves generating a path for a single agent based on predefined objectives and environmental constraints. In contrast, cooperative path planning emphasizes communication and collaboration among multiple robots to optimize overall system performance. This transition is enabled by establishing communication channels among robots and developing protocols for sharing information and coordinating actions. Instead of treating each robot as an independent entity, cooperative path planning encourages robots to exchange information about their intended paths, perceived obstacles, and other relevant factors. By coordinating their movements and sharing the workload, robots can collectively optimize their paths to minimize conflicts and maximize efficiency. By transitioning from individual autonomy to collaborative decision-making, multi-robot systems can leverage synergies among robots to achieve common objectives more effectively. Cooperative path planning enables robots to exploit complementary capabilities, avoid redundant actions, and adapt to dynamic changes in the environment collaboratively. This approach fosters

teamwork and coordination among robots, leading to improved resource utilization and overall system throughput;

3. Consensus-based decision-making [91]: Transitioning from single to multi-robot path planning through consensus-based decision-making involves reaching an agreement among multiple robots on a common path plan through iterative negotiation and compromise. In single-robot path planning, the focus is on generating a path for a single agent based on predefined objectives and environmental constraints. In contrast, consensus-based decision-making emphasizes reaching consensus among multiple robots on a shared path plan.This transition is facilitated by establishing communication channels among robots and developing algorithms for reaching an agreement through iterative negotiation. Instead of each robot independently generating its path, consensus-based decision-making requires robots to share their proposed paths and adjust them based on feedback from other robots. Through iterative rounds of negotiation and compromise, robots converge on a common path plan that balances individual objectives and system-wide constraints. By transitioning from individual decision-making to consensus-based decision-making, multi-robot systems can achieve fairness and cooperation among robots. Consensus-based decision-making ensures that all robots' interests are considered in the path planning process, leading to equitable outcomes and fostering collaboration among robots. This approach promotes teamwork and cohesion, enabling robots to collectively navigate complex environments and achieve common objectives.

7. Conclusions

This paper presents a systematic review of the latest path planning techniques proposed for multi-robot systems published between 2018 and 2023. First, a general introduction to path planning techniques is presented. Then, a general outline of the current research directions is offered and gaps in the workflow of research in this area are identified. After that, the latest path planning techniques are classified based on various factors, such as obstacles, destination, communication, the shortest time, and shortest path. This research highlights the lack of proposed heterogeneous systems and dynamic target path planning approaches for robotic systems in the literature. Addressing such research gaps in the field of path planning for multi-robot systems is critical for future developments and enhancements.

Author Contributions: Conceptualization, I.K., T.R. and M.B.; methodology, N.A. and M.B.; validation, M.B.; writing—original draft preparation, N.A. and M.B.; writing—review and editing, M.B. and K.A.; supervision, I.K., T.R. and M.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data is contained within the article.

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

References

1. Schuster, A.; Kupke, M.; Larsen, L. Autonomous manufacturing of composite parts by a multi-robot system. *Procedia Manuf.* **2017**, *11*, 249–255. [CrossRef]
2. Amanatiadis, A.; Henschel, C.; Birkicht, B.; Andel, B.; Charalampous, K.; Kostavelis, I.; May, R.; Gasteratos, A. Avert: An autonomous multi-robot system for vehicle extraction and transportation. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 1662–1669.
3. Burgner-Kahrs, J.; Rucker, D.C.; Choset, H. Continuum robots for medical applications: A survey. *IEEE Trans. Robot.* **2015**, *31*, 1261–1280. [CrossRef]
4. Pascher, M.; Baumeister, A.; Klein, B.; Schneegass, S.; Gerken, J. Little Helper: A Multi-Robot System in Home Health Care Environments. In Proceedings of the 1st International Workshop on Human-Drone Interaction, Glasgow, UK, 5 May 2019.
5. Hameed, I.A. A coverage planner for multi-robot systems in agriculture. In Proceedings of the 2018 IEEE International Conference on Real-Time Computing and Robotics (RCAR), Kandima, Maldives, 1–5 August 2018; pp. 698–704.

6. Haghani, N.; Li, J.; Koenig, S.; Kunapuli, G.; Contardo, C.; Regan, A.; Yarkony, J. Multi-robot routing with time windows: A column generation approach. *arXiv* **2021**, arXiv:2103.08835.
7. Krizmancic, M.; Arbanas, B.; Petrovic, T.; Petric, F.; Bogdan, S. Cooperative aerial-ground multi-robot system for automated construction tasks. *IEEE Robot. Autom. Lett.* **2020**, *5*, 798–805. [CrossRef]
8. Pellegrinelli, S.; Pedrocchi, N.; Tosatti, L.M.; Fischer, A.; Tolio, T. Multi-robot spot-welding cells for car-body assembly: Design and motion planning. *Robot. Comput.-Integr. Manuf.* **2017**, *44*, 97–116. [CrossRef]
9. Ju, M.S.; Lin, C.C.; Lin, D.H.; Hwang, I.S.; Chen, S.M. A rehabilitation robot with force-position hybrid fuzzy controller: Hybrid fuzzy control of rehabilitation robot. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2005**, *13*, 349–358.
10. Chen, Y.Q.; Wang, Z. Formation control: A review and a new consideration. In Proceedings of the 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, Edmonton, AB, Canada, 2–6 August 2005; pp. 3181–3186.
11. Grayson, S. *Search & Rescue Using Multi-Robot Systems*; School of Computer Science and Informatics, University College Dublin: Dublin, Ireland, 2014; pp. 1–14.
12. Yan, Z.; Jouandeau, N.; Cherif, A.A. A survey and analysis of multi-robot coordination. *Int. J. Adv. Robot. Syst.* **2013**, *10*, 399. [CrossRef]
13. Fan, Z.; Yang, H.; Han, Y.; Ning, X. A Path Planning Method for Multi-robot Formation Based on Improved Q-Learning. In Proceedings of the 2021 Chinese Intelligent Systems Conference, Fuzhou, China, 16–17 October 2021; pp. 842–849.
14. Pan, Z.; Zhang, C.; Xia, Y.; Xiong, H.; Shao, X. An Improved Artificial Potential Field Method for Path Planning and Formation Control of the Multi-UAV Systems. *IEEE Trans. Circuits Syst. II Express Briefs* **2021**, *69*, 1129–1133. [CrossRef]
15. Kim, D.; Kim, H.; Kim, H.; Shin, J.U.; Myung, H.; Kim, Y.G. Path Planning for Multi-Agent Jellyfish Removal Robot System JEROS and Experimental Tests. In *Distributed Autonomous Robotic Systems*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 299–310.
16. Chen, M.; Chen, Y.; Chen, Z.; Yang, Y. Path planning of UAV-UGV heterogeneous robot system in road network. In Proceedings of the International Conference on Intelligent Robotics and Applications, Shenyang, China, 8–11 August 2019; pp. 497–507.
17. Ni, J.; Wang, X.; Tang, M.; Cao, W.; Shi, P.; Yang, S.X. An Improved Real-Time Path Planning Method Based on Dragonfly Algorithm for Heterogeneous Multi-Robot System. *IEEE Access* **2020**, *8*, 140558–140568. [CrossRef]
18. Ivanovic, A.; Markovic, L.; Car, M.; Duvnjak, I.; Orsag, M. Towards Autonomous Bridge Inspection: Sensor Mounting Using Aerial Manipulators. *Appl. Sci.* **2021**, *11*, 8279. [CrossRef]
19. Alonso-Mora, J.; Baker, S.; Rus, D. Multi-robot formation control and object transport in dynamic environments via constrained optimization. *Int. J. Robot. Res.* **2017**, *36*, 1000–1021. [CrossRef]
20. Gasparetto, A.; Boscariol, P.; Lanzutti, A.; Vidoni, R. Path planning and trajectory planning algorithms: A general overview. In *Motion and Operation Planning of Robotic Systems: Background and Practical Approaches*; Springer: Cham, Switzerland, 2015; pp. 3–27.
21. Sánchez-Ibáñez, J.R.; Pérez-del Pulgar, C.J.; García-Cerezo, A. Path planning for autonomous mobile robots: A review. *Sensors* **2021**, *21*, 7898. [CrossRef]
22. Campbell, S.; O'Mahony, N.; Carvalho, A.; Krpalkova, L.; Riordan, D.; Walsh, J. Path planning techniques for mobile robots a review. In Proceedings of the 2020 6th International Conference on Mechatronics and Robotics Engineering (ICMRE), Barcelona, Spain, 12–15 February 2020; pp. 12–16.
23. Adzhar, N.; Yusof, Y.; Ahmad, M.A. A review on autonomous mobile robot path planning algorithms. *Adv. Sci. Technol. Eng. Syst. J.* **2020**, *5*, 236–240. [CrossRef]
24. Patle, B.K.; Pandey, A.; Parhi, D.R.K.; Jagadeesh, A.J.D.T. A review: On path planning strategies for navigation of mobile robot. *Def. Technol.* **2019**, *15*, 582–606. [CrossRef]
25. Khaksar, W.; Vivekananthen, S.; Saharia, K.S.M.; Yousefi, M.; Ismail, F.B. A review on mobile robots motion path planning in unknown environments. In Proceedings of the 2015 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS), Langkawi, Malaysia, 18–20 October 2015; pp. 295–300.
26. Lin, S.; Liu, A.; Wang, J.; Kong, X. A Review of Path-Planning Approaches for Multiple Mobile Robots. *Machines* **2022**, *10*, 773. [CrossRef]
27. Sharma, A.; Shoval, S.; Sharma, A.; Pandey, J.K. Path planning for multiple targets interception by the swarm of UAVs based on swarm intelligence algorithms: A review. *IETE Tech. Rev.* **2022**, *39*, 675–697. [CrossRef]
28. Aggarwal, S.; Kumar, N. Path planning techniques for unmanned aerial vehicles: A review, solutions, and challenges. *Comput. Commun.* **2020**, *149*, 270–299. [CrossRef]
29. Liu, Z.; Zhang, Y.; Yu, X.; Yuan, C. Unmanned surface vehicles: An overview of developments and challenges. *Annu. Rev. Control* **2016**, *41*, 71–93. [CrossRef]
30. Hadi, B.; Khosravi, A.; Sarhadi, P. A review of the path planning and formation control for multiple autonomous underwater vehicles. *J. Intell. Robot. Syst.* **2021**, *101*, 1–26. [CrossRef]
31. Zafar, M.N.; Mohanta, J. Methodology for path planning and optimization of mobile robots: A review. *Procedia Comput. Sci.* **2018**, *133*, 141–152. [CrossRef]
32. Verma, J.K.; Ranga, V. Multi-robot coordination analysis, taxonomy, challenges and future scope. *J. Intell. Robot. Syst.* **2021**, *102*, 1–36.
33. Madridano, Á.; Al-Kaff, A.; Martín, D.; de la Escalera, A. Trajectory planning for multi-robot systems: Methods and applications. *Expert Syst. Appl.* **2021**, *173*, 114660. [CrossRef]
34. Kitchenham, B.; Charters, S. Guidelines for performing systematic literature reviews in software engineering version 2.3. *Engineering* **2007**, *45*, 1051.

35. Ravankar, A.; Ravankar, A.A.; Watanabe, M.; Hoshino, Y.; Rawankar, A. Multi-robot path planning for smart access of distributed charging points in map. *Artif. Life Robot.* **2021**, *26*, 52–60. [CrossRef]
36. Kiadi, M.; Villar, J.R.; Tan, Q. Synthesized a* multi-robot path planning in an indoor smart lab using distributed cloud computing. In Proceedings of the 15th International Conference on Soft Computing Models in Industrial and Environmental Applications (SOCO 2020), Burgos, Spain, 16–18 September 2021; pp. 580–589.
37. Ravikumar, V.; Shreedharan, S.; Kumaran, M.S. Priority Scheduling based Dynamic Path Planning System with Motion Feedback. In Proceedings of the 2020 First IEEE International Conference on Measurement, Instrumentation, Control and Automation (ICMICA), Kurukshetra, India, 24–26 June 2020; pp. 1–5.
38. Yang, Z.; Li, J.; Yang, L.; Wang, Q.; Li, P.; Xia, G. Path planning and collision avoidance methods for distributed multi-robot systems in complex dynamic environments. *Math. Biosci. Eng.* **2023**, *20*, 145–178. [CrossRef]
39. Trinh, L.A.; Ekström, M.; Cürüklü, B. Dependable Navigation for Multiple Autonomous Robots with Petri Nets Based Congestion Control and Dynamic Obstacle Avoidance. *J. Intell. Robot. Syst.* **2022**, *104*, 69. [CrossRef]
40. Dutta, A.; Bhattacharya, A.; Kreidl, O.P.; Ghosh, A.; Dasgupta, P. Multi-robot informative path planning in unknown environments through continuous region partitioning. *Int. J. Adv. Robot. Syst.* **2020**, *17*, 1729881420970461. [CrossRef]
41. Huang, S.K.; Wang, W.J.; Sun, C.H. A path planning strategy for multi-robot moving with path-priority order based on a generalized Voronoi diagram. *Appl. Sci.* **2021**, *11*, 9650. [CrossRef]
42. Govindaraju, M.; Fontanelli, D.; Kumar, S.S.; Pillai, A.S. Optimized Offline-Coverage Path Planning Algorithm for Multi-Robot for Weeding in Paddy Fields. *IEEE Access* **2023**. [CrossRef]
43. Nayak, S.; Paton, M.; Otte, M.W. A Heuristic-Guided Dynamical Multi-Rover Motion Planning Framework for Planetary Surface Missions. *IEEE Robot. Autom. Lett.* **2023**, *8*, 2542–2549. [CrossRef]
44. Neto, A.A.; Macharet, D.G.; Campos, M.F.M. Multi-agent rapidly-exploring pseudo-random tree. *J. Intell. Robot. Syst.* **2018**, *89*, 69–85. [CrossRef]
45. Yu, Z.; Xiang, L. NPQ-RRT*: An Improved RRT* Approach to Hybrid Path Planning. *Complexity* **2021**, *2021*, 1–10. [CrossRef]
46. Yang, H.a.; Li, Y.; Duan, X.; Shen, G.; Zhang, S. A parallel shape formation method for swarm robotics. *Robot. Auton. Syst.* **2022**, *151*, 104043. [CrossRef]
47. Das, M.S.; Sanyal, S.; Mandal, S. Navigation of multiple robots in formative manner in an unknown environment using artificial potential field based path planning algorithm. *Ain Shams Eng. J.* **2022**, *13*, 101675. [CrossRef]
48. Matoui, F.; Boussaid, B.; Abdelkrim, M.N. Distributed path planning of a multi-robot system based on the neighborhood artificial potential field approach. *Simulation* **2019**, *95*, 637–657. [CrossRef]
49. Zhu, H.; Claramunt, F.M.; Brito, B.; Alonso-Mora, J. Learning interaction-aware trajectory predictions for decentralized multi-robot motion planning in dynamic environments. *IEEE Robot. Autom. Lett.* **2021**, *6*, 2256–2263. [CrossRef]
50. Agrawal, P.; Agrawal, H. Adaptive algorithm design for cooperative hunting in multi-robots. *Int. J. Intell. Syst. Appl.* **2018**, *11*, 47. [CrossRef]
51. Wang, B.; Liu, Z.; Li, Q.; Prorok, A. Mobile robot path planning in dynamic environments through globally guided reinforcement learning. *IEEE Robot. Autom. Lett.* **2020**, *5*, 6932–6939. [CrossRef]
52. Schperberg, A.; Tsuei, S.; Soatto, S.; Hong, D. SABER: Data-driven motion planner for autonomously navigating heterogeneous robots. *IEEE Robot. Autom. Lett.* **2021**, *6*, 8086–8093. [CrossRef]
53. Nazarahari, M.; Khanmirza, E.; Doostie, S. Multi-objective multi-robot path planning in continuous environment using an enhanced genetic algorithm. *Expert Syst. Appl.* **2019**, *115*, 106–120. [CrossRef]
54. Li, B.; Patankar, S.; Moridian, B.; Mahmoudian, N. Planning large-scale search and rescue using team of uavs and charging stations. In Proceedings of the 2018 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), Philadelphia, PA, USA, 6–8 August 2018; pp. 1–8.
55. de Almeida, J.P.L.S.; Nakashima, R.T.; Neves-Jr, F.; de Arruda, L.V.R. A global/local path planner for multi-robot systems with uncertain robot localization. *J. Intell. Robot. Syst.* **2020**, *100*, 311–333. [CrossRef]
56. Abdi, M.I.I.; Khan, M.U.; Güneş, A.; Mishra, D. Escaping Local Minima in Path Planning Using a Robust Bacterial Foraging Algorithm. *Appl. Sci.* **2020**, *10*, 7905. [CrossRef]
57. Tian, S.; Li, Y.; Kang, Y.; Xia, J. Multi-robot path planning in wireless sensor networks based on jump mechanism PSO and safety gap obstacle avoidance. *Future Gener. Comput. Syst.* **2021**, *118*, 37–47. [CrossRef]
58. Dewangan, R.K.; Shukla, A.; Godfrey, W.W. A solution for priority-based multi-robot path planning problem with obstacles using ant lion optimization. *Mod. Phys. Lett. B* **2020**, *34*, 2050137. [CrossRef]
59. Yang, L.; Fu, L.; Li, P.; Mao, J.; Guo, N.; Du, L. LF-ACO: An effective formation path planning for multi-mobile robot. *Math. Biosci. Eng* **2022**, *19*, 225–252. [CrossRef]
60. Elmi, Z.; Efe, M.Ö. Online path planning of mobile robot using grasshopper algorithm in a dynamic and unknown environment. *J. Exp. Theor. Artif. Intell.* **2021**, *33*, 467–485. [CrossRef]
61. Shi, W.; He, Z.; Tang, W.; Liu, W.; Ma, Z. Path planning of multi-robot systems with boolean specifications based on simulated annealing. *IEEE Robot. Autom. Lett.* **2022**, *7*, 6091–6098. [CrossRef]
62. Hamed, O.; Hamlich, M.; Ennaji, M. Hunting strategy for multi-robot based on wolf swarm algorithm and artificial potential field. *Indones. J. Electr. Eng. Comput. Sci.* **2022**, *25*, 159–171. [CrossRef]

63. Zhao, K.; Ning, L. Hybrid Navigation Method for Multiple Robots Facing Dynamic Obstacles. *Tsinghua Sci. Technol.* **2022**, *27*, 894–901. [CrossRef]
64. Trinh, L.A.; Ekström, M.; Cürüklü, B. Petri net based navigation planning with dipole field and dynamic window approach for collision avoidance. In Proceedings of the 2019 6th International Conference on Control, Decision and Information Technologies (CoDIT), Paris, France, 23–26 April 2019; pp. 1013–1018.
65. Schack, M.A.; Rogers, J.G.; Han, Q.; Dantam, N.T. Optimizing Non-Markovian Information Gain under Physics-based Communication Constraints. *IEEE Robot. Autom. Lett.* **2021**, *6*, 4813–4819. [CrossRef]
66. Feng, L.; Zhou, M.; Hu, B. A hybrid motion planning algorithm for multi-robot formation in a dynamic environment. In Proceedings of the 2020 IEEE 16th International Conference on Automation Science and Engineering (CASE), Hong Kong, China, 20–21 August 2020; pp. 343–348.
67. Apoorva; Gautam, R.; Kala, R. Motion planning for a chain of mobile robots using a* and potential field. *Robotics* **2018**, *7*, 20. [CrossRef]
68. Kashyap, A.K.; Parhi, D.R.; Muni, M.K.; Pandey, K.K. A hybrid technique for path planning of humanoid robot NAO in static and dynamic terrains. *Appl. Soft Comput.* **2020**, *96*, 106581. [CrossRef]
69. Zhou, M.; Wang, Z.; Wang, J.; Dong, Z. A Hybrid Path Planning and Formation Control Strategy of Multi-Robots in a Dynamic Environment. *J. Adv. Comput. Intell. Intell. Inform.* **2022**, *26*, 342–354. [CrossRef]
70. Das, P.K. Hybridization of Kidney-Inspired and sine–cosine algorithm for multi-robot path planning. *Arab. J. Sci. Eng.* **2020**, *45*, 2883–2900. [CrossRef]
71. Faridi, A.Q.; Sharma, S.; Shukla, A.; Tiwari, R.; Dhar, J. Multi-robot multi-target dynamic path planning using artificial bee colony and evolutionary programming in unknown environment. *Intell. Serv. Robot.* **2018**, *11*, 171–186. [CrossRef]
72. Vikas; Parhi, D.R. Multi-robot path planning using a hybrid dynamic window approach and modified chaotic neural oscillator-based hyperbolic gravitational search algorithm in a complex terrain. *Intell. Serv. Robot.* **2023**, *16*, 213–230.
73. Muni, M.K.; Pradhan, P.K.; Dhal, P.R.; Kumar, S.; Sethi, R.; Patra, S.K. Improving navigational parameters and control of autonomous robot using hybrid SOMA–PSO technique. *Evol. Intell.* **2023**, 1–17. [CrossRef]
74. Huang, S.K.; Wang, W.J.; Sun, C.H. A New Multirobot Path Planning with Priority Order Based on the Generalized Voronoi Diagram. *IEEE Access* **2022**, *10*, 56564–56577. [CrossRef]
75. Chen, Z.; Alonso-Mora, J.; Bai, X.; Harabor, D.D.; Stuckey, P.J. Integrated task assignment and path planning for capacitated multi-agent pickup and delivery. *IEEE Robot. Autom. Lett.* **2021**, *6*, 5816–5823. [CrossRef]
76. Sharma, K.; Doriya, R. Coordination of multi-robot path planning for warehouse application using smart approach for identifying destinations. *Intell. Serv. Robot.* **2021**, *14*, 313–325. [CrossRef]
77. Choutri, K.; Lagha, M.; Meshoul, S.; Fadloun, S. Path Planning and Formation Control for UAV-Enabled Mobile Edge Computing Network. *Sensors* **2022**, *22*, 7243. [CrossRef]
78. Song, K.T.; Sun, Y.X. Coordinating multiple mobile robots for obstacle avoidance using cloud computing. *Asian J. Control* **2021**, *23*, 1225–1236. [CrossRef]
79. Prasanna Lakshmi, K. Motion of multiple robot in a curved boundary & obstacles. *Int. J. Eng. Adv. Technol.* **2019**, *9*, 6011–6014. [CrossRef]
80. Amoolya, G.; Lavanya, K.L.; Sudheer, A. Wi-fi rssi based optimal path planning and collision avoidance in a multi robot environment. In Proceedings of the 2019 IEEE International Conference on Intelligent Techniques in Control, Optimization and Signal Processing (INCOS), Tamilnadu, India, 11–13 April 2019; pp. 1–5.
81. Yorozu, A.; Hailu, H.; Ohya, A. Experimental Investigation of Mutual Collision Avoidance Behavior for Multiple Mobile Robots. In Proceedings of the 2021 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), Delft, The Netherlands, 12–16 July 2021; pp. 1164–1169.
82. Galceran, E.; Carreras, M. A survey on coverage path planning for robotics. *Robot. Auton. Syst.* **2013**, *61*, 1258–1276. [CrossRef]
83. Montiel, O.; Orozco-Rosas, U.; Sepúlveda, R. Path planning for mobile robots using bacterial potential field for avoiding static and dynamic obstacles. *Expert Syst. Appl.* **2015**, *42*, 5177–5191. [CrossRef]
84. Noreen, I.; Khan, A.; Habib, Z. A comparison of RRT, RRT* and RRT*-smart path planning algorithms. *Int. J. Comput. Sci. Netw. Secur. (IJCSNS)* **2016**, *16*, 20.
85. Low, E.S.; Ong, P.; Cheah, K.C. Solving the optimal path planning of a mobile robot using improved Q-learning. *Robot. Auton. Syst.* **2019**, *115*, 143–161. [CrossRef]
86. Wang, N.; Xu, H.; Li, C.; Yin, J. Hierarchical path planning of unmanned surface vehicles: A fuzzy artificial potential field approach. *Int. J. Fuzzy Syst.* **2021**, *23*, 1797–1808. [CrossRef]
87. Hou, Z.; Wang, W.; Zhang, G.; Han, C. A survey on the formation control of multiple quadrotors. In Proceedings of the 2017 14th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), Jeju, Republic of Korea, 28 June–1 July 2017; pp. 219–225.
88. Tanner, H.G.; Jadbabaie, A.; Pappas, G.J. Flocking in fixed and switching networks. *IEEE Trans. Autom. Control* **2007**, *52*, 863–868. [CrossRef]
89. Shao, X.X.; Gong, Y.J.; Zhan, Z.H.; Zhang, J. Bipartite cooperative coevolution for energy-aware coverage path planning of uavs. *IEEE Trans. Artif. Intell.* **2021**, *3*, 29–42. [CrossRef]

90. Lin, S.; Liu, A.; Wang, J.; Kong, X. An improved fault-tolerant cultural-PSO with probability for multi-AGV path planning. *Expert Syst. Appl.* **2024**, *237*, 121510. [CrossRef]
91. Rai, A.; Kannan, R.J. Consensus-based decision making in non-linearly multi-coupled IoT networked SLAM operations. *Int. J. Electr. Eng. Educ.* **2024**, *61*, 86–99. [CrossRef]

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