Sheffield Hallam University

Optimisation Techniques for Multi-Robot Path Planning: A Review of Collision Avoidance and Performance Metrics in Connectivity, Efficiency and Safety

ALWAFI, Fatma AS and SAATCHI, Reza <http://orcid.org/0000-0002-2266-0187>

Available from Sheffield Hallam University Research Archive (SHURA) at:

https://shura.shu.ac.uk/37504/

This document is the Published Version [VoR]

Citation:

ALWAFI, Fatma AS and SAATCHI, Reza (2026). Optimisation Techniques for Multi-Robot Path Planning: A Review of Collision Avoidance and Performance Metrics in Connectivity, Efficiency and Safety. Technologies, 14 (6): 337. [Article]

Copyright and re-use policy

See http://shura.shu.ac.uk/information.html

Sheffield Hallam University Research Archive
http://shura.shu.ac.uk

technologies

MDPI

Review

Optimisation Techniques for Multi-Robot Path Planning: A Review of Collision Avoidance and Performance Metrics in Connectivity, Efficiency and Safety

Fatma A. S. Alwafi and Reza Saatchi *

School of Engineering and Built Environment, Sheffield Hallam University, City Campus, Sheffield S1 1WB, UK; fatmaalwafi@yahoo.com
* Correspondence: r.saatchi@shu.ac.uk

Abstract

Path planning is critical for multi-robot systems (MRS), directly affecting the operation efficiency, execution time, and operational cost. Despite extensive research and successful applications of multiple algorithms, achieving globally optimal solutions in cluttered or dynamic environments remains a significant challenge. Issues such as scalability with an increasing number of robots, computational efficiency, system robustness, and coordination complexity continue to drive the development of more reliable approaches. This study reviews modelling approaches, optimisation criteria, and solution algorithms based on the roadmap planning methods that are widely used for multi-robot path planning (MRPP). It focuses on three graph-based algorithms: MRPP algorithm, central algorithm (CA), and the optimisation central algorithm (OCA). These algorithms utilise visibility graphs (VG) for environment representation and Dijkstra's algorithm for shortest path computation, while incorporating algebraic connectivity to improve coordination, safety, and scalability. In addition, the technological context and implementation platforms, including simulation environments, cloud robotics, and AI-based frameworks, are conceptually examined. The potential applications of these methods in assistive robotics are highlighted, particularly in supporting a safe and reliable navigation in healthcare and human-centred environments. The article synthesises theoretical and practical insights, identifies current limitations and challenges, and outlines future research directions for efficient, scalable, and robust MRPP.

Keywords: multi-robot path planning; graph-based planning; visibility graphs; algebraic connectivity; collision avoidance; optimisation-based planning; connectivity preservation

Academic Editor: Tamás Haidegger

Received: 1 April 2026
Revised: 27 May 2026
Accepted: 29 May 2026
Published: 30 May 2026

Copyright: © 2026 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.

1. Introduction

The increasing deployment of autonomous robotic systems across industrial, medical, military, and environmental domains has significantly intensified the need for efficient multi-robot coordination strategies [1–3]. Multi-robot systems (MRS) offer distinct advantages over single-robot systems, e.g., improved task efficiency, robustness, and scalability [1,4,5]. These benefits are particularly important in complex, hazardous, or dynamic environments where a coordinated robotic team can perform tasks more effectively than a single robot [6–8]. Despite these advantages, multi-robot path planning (MRPP) remains a fundamental challenge. The objective in MRPP is to generate collision-free,

computationally efficient, and coordinated trajectories for multiple robots operating in a shared environment [2,9–13]. This problem becomes increasingly complex as the number of robots grows, introducing challenges related to path planning scalability, coordination, communication maintenance, and robustness. Furthermore, achieving globally optimal solutions in environments with dense obstacles and interaction constraints remains difficult despite extensive research efforts [6,14–16]. To address these challenges, a wide range of path planning approaches were developed that include grid-based, sampling-based, and roadmap-based. Among these, roadmap-based techniques are particularly attractive due to their ability to transform continuous motion planning problems into discrete graph search problems with provable optimality guarantees [17–25].

In this context, graph-based methods provide a structured and mathematically rigorous framework for modelling robot interactions and environmental constraints [17–20,26–29]. A prominent class of roadmap-based methods is the visibility graph (VG), which enables optimal path computation in static polygonal environments using classical shortest path algorithms [21,22,30–35]. VG methods can be further enhanced by incorporating graph-theoretic metrics, such as algebraic connectivity (λ_2), to support coordination, connectivity preservation, and efficient navigation in multi-robot systems [9,36–40]. These approaches enable the robots to navigate obstacle-rich environments efficiently [1,7,41–49]. Graph-based methods offer not only a mathematically rigorous foundation but also the flexibility to incorporate coordination and optimisation in modern MRS applications [24–27,50–60]. Furthermore, their practical implementation is supported by enabling technologies such as simulation platforms, cloud robotics, and artificial-intelligence-based frameworks, which facilitate evaluation and deployment in real-world scenarios [61–70]. In addition, these methods are increasingly relevant in emerging application domains such as assistive robotics, where safe, reliable, and human-aware navigation are essential [71–93].

Beyond traditional terrestrial applications, recent advances in mission planning for autonomous systems have extended these concepts to aerospace domains. Spacecraft navigation under orbital threats has motivated the development of autonomous collision avoidance strategies and constraint-based planning [94,95]. These approaches emphasise real-time adaptability, constraint satisfaction, and robustness under uncertainty—key requirements that closely align with the challenges of multi-robot path planning in dynamic and safety-critical environments. Therefore, insights from these emerging domains provide a promising direction for enhancing the robustness and safety guarantees of graph-based multi-robot planning frameworks. This article focuses on three representative graph-based algorithms: multi-robot path planning (MRPP), central algorithm (CA), and optimised central algorithm (OCA). These algorithms share a common foundation in VG construction and Dijkstra's algorithm for optimal path computation, while differing in planning architecture, optimisation criteria, and safety mechanisms [1,23–29]. While MRPP emphasises coordination and connectivity preservation, CA and OCA introduce improvements in computational efficiency and safety, respectively.

The main contribution of this study is to provide a structured and comparative review of these algorithms within a unified analytical framework. Specifically, this study:

- Provides a systematic review of graph-based motion planning approaches and their applications in multi-robotic systems.
- Introduces an analytical perspective that integrates VG planning with algebraic graph-theoretic metrics to evaluate coordination and connectivity.
- Compares MRPP, CA, and OCA in terms of their computational efficiency, path optimality, robustness, and safety performance.

The objective of this review is to analyse the modelling approaches, optimisation strategies, and performance trade-offs of these algorithms, while identifying key challenges and future research directions for scalable and efficient multi-robot systems.

The remainder of this article is organised as follows. Section 2 provides a description of the materials and methods. Section 3 provides an overview of multi-robot systems and relevant graph-theoretic foundations. Section 4 describes the methodology employed for this review. Section 5 presents a taxonomy of multi-robot path planning approaches. Section 6 analyses the MRPP, CA, and OCA algorithms in detail. Finally, Section 7 concludes the paper and outlines future research directions.

2. Materials and Methods

The study adopted a systematic mapping study (SMS) approach, utilizing the PRISMA (Preferred Reporting Items for Systematic reviews and Meta-Analyses) 2020 operational protocol to identify, select, map, and analyse relevant research on graph-based MRPP. The primary objective was to provide a structured, high-level evaluation and thematic categorisation of MRPP, CA, and OCA algorithms across a clear technical landscape.

While the PRISMA framework was originally developed for clinical evidence synthesis, its systematic pipeline for literature identification, screening, and eligibility assessment has been widely adopted as a rigorous tool within engineering and computer science domains to mitigate selection bias and ensure architectural traceability [2,9,31]. In a computationally and theoretically oriented context, this methodology functions effectively as a systematic mapping study classifying algorithmic taxonomies and mapping out structural boundaries rather than performing a statistical meta-analysis [66,80]. Therefore, this review is qualitative, structural, and analytical in nature, leveraging PRISMA strictly as a robust procedural pipeline well suited for managing and filtering large volumes of technical robotics literature.

2.1. Literature Search Strategy

A comprehensive literature search was undertaken across multiple academic databases, including IEEE Xplore, ScienceDirect, SpringerLink, and Google Scholar. The search covered publications from 2012 to 2026 to capture both foundational and recent advancements, such as decentralised coordination [6,15], communication-efficient task allocation [7], and adaptive formation tracking [8].

The search keywords included combinations of the following terms:

i. Multi-robot path planning (MRPP);
ii. Collision avoidance (CA) and optimisation central algorithm (OCA);
iii. Graph-based methods and visibility graphs (VG) [22,26,51];
iv. Algebraic connectivity (λ_2) and connectivity maintenance [19,24,28];
v. Decentralised communication and swarm control [3,12,18];
vi. Artificial intelligence-based frameworks, including reinforcement learning (RL) and graph neural networks (GNN) [10,81–83];
vii. Safety-critical and constraint-based mission planning approaches, including aerospace applications such as spacecraft autonomous navigation and orbital collision avoidance using satisfiability modulo theories (SMT) [94,95].

This expanded search strategy ensured that both the traditional robotic path planning methods and the emerging safety-critical autonomous planning paradigms were covered.

2.2. Study Selection and Eligibility Criteria

The study selection process, illustrated by the PRISMA flow diagram shown in Figure 1, followed predefined inclusion and exclusion criteria. Importantly, the study distinguished between:

(i) Core studies: identified through the PRISMA selection process and used for primary analysis.

(ii) Supporting references: included to provide theoretical background, contextual discussion, and technological insights.

This distinction ensured both methodological transparency and comprehensive coverage of the relevant field. The selections are further outlined in Table 1.

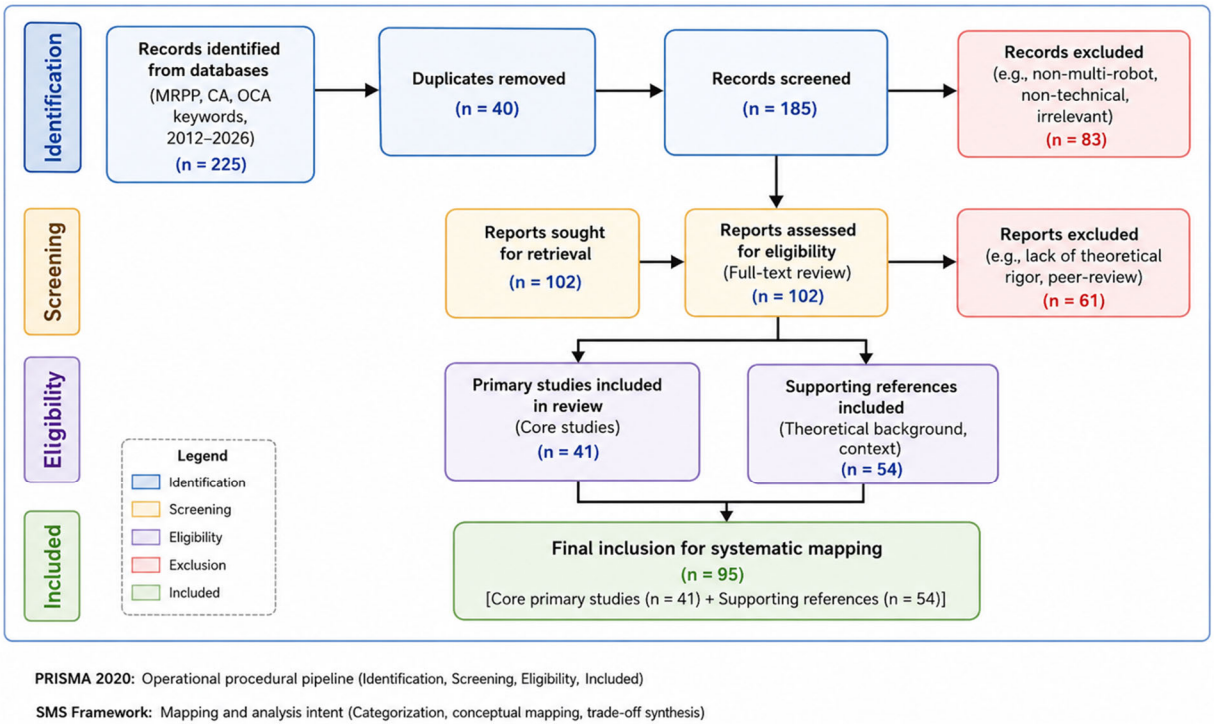

Figure 1. PRISMA-2020-guided systematic mapping and study selection process (n denote the number of studies/articles).

Table 1. Summary of literature search, selection process, and number of articles considered.

Phase	Action	Articles
Identification	Initial database search (2012–2026)	225
Deduplication	Removal of duplicate records	185
Screening	Title and abstract review (exclusion of irrelevant studies)	102
Eligibility	Full-text assessment of quality and methodology	41
Core Inclusion	Primary studies (MRPP, CA, and graph-based methods)	41
Supporting References	Foundational theory, technical surveys, and contextual literature	54
Final Inclusion	Total references cited in the bibliography	95

Studies were included if they met the following benchmarks:

i. Technical relevance: Original research or high-quality reviews focusing on MRPP, CA, or OCA algorithms [33,40,44].

ii. Methodological rigor: Studies contributing to multi-robot path planning, coordination, or collision avoidance using graph theory, Voronoi diagrams [14,50,68], or hybrid probabilistic roadmaps [13,49].

iii. Core techniques: Consideration of methods that provided graph-based, roadmap-based, or optimisation-based solutions contributing to scalability, connectivity, or robustness, such as shortest path algorithms (Dijkstra) [74], conflict-based search [29,54], and secure state estimation [20].
iv. Application diversity: Inclusion of representative applications such as planetary exploration [4,32,70], search and rescue [30], assistive robotics and healthcare systems [86,87,90,93], and safety-critical autonomous systems, including spacecraft mission planning and orbital collision avoidance [94,95].

Studies were excluded if they:

i. Focused exclusively on a single robot;
ii. Lacked methodological or mathematical clarity regarding path computation;
iii. Were not peer-reviewed or lacked sufficient technical details.

2.3. Data Extraction and Synthesis

The initial database search identified 225 publications, of which 185 remained after duplicate removal. Following title and abstract screening, 102 studies were selected for full-text reviews. A total of 41 core primary studies met the eligibility criteria and were included in the primary mapping analysis. In addition, 54 supporting references were incorporated to provide theoretical foundations (e.g., algebraic connectivity [19]), background concepts, and enabling technologies such as Robot Operating System (ROS), Gazebo, and Webots [75–77]. These references were not part of the sequential screening selection process but were included to ensure a comprehensive and contextualised review.

Unlike traditional systematic reviews in clinical domains, this study did not perform a statistical meta-analysis. Instead, it provides a comparative analytical synthesis of algorithmic approaches, mapping out performance trade-offs, design principles, and practical applicability across different environments. Furthermore, recent studies on safety-critical mission planning in aerospace domains, particularly those addressing autonomous spacecraft navigation and constraint-based planning under uncertainty [94,95], were analysed to provide cross-domain insights into robustness, real-time adaptability, and constraint satisfaction. These structural insights support a broader evaluation of terrestrial and orbital MRPP methods in dynamic and safety-critical environments.

3. Multi-Robot Systems

The robots in a multi-robot system work collaboratively within a shared environment to perform tasks that may be too complex, large-scale, or hazardous for a single robot [30–35]. The main reason for the growing interest in MRSs is their ability to withstand system failures by creating redundant processes and sharing responsibilities. Thus, the system reduces the failure rate [31–36]. Even when a robot fails, the remaining robots can seamlessly continue the tasks, thereby increasing the overall resilience and robustness of the system [3,4,34].

However, MRSs have limitations and disadvantages that must be carefully considered. Coordination complexity remains a major challenge, particularly in dynamic and uncertain environments where communication and collaboration algorithms must operate reliably under changing conditions [1,2,33,37–39]. Communication constraints can further degrade system performance, especially in obstacle-rich or interference-prone settings, thereby necessitating robust mechanisms for dependable information exchange. Ensuring system robustness and fault tolerance is critical, as the failure of individual robots may adversely affect the overall system performance [1,2,35,36,40]. This underscores the importance of continued research in this dynamic field to tackle emerging challenges such as coordination, path planning problems, communication, and autonomy under real-

world constraints [1,6,10,41–43]. Figure 2 shows an example of a multi-robot system consisting of five robots (R_1–R_5), represented as an undirected graph. In this representation, the edges indicate communication or visibility links between robots, meaning that two robots are connected only if they are within communication range or have direct line-of-sight interaction. For example, R_4 and R_5 are not connected because they are assumed to be outside each other's communication or visibility range in the given configuration. Therefore, no direct edge is formed between them in the graph.

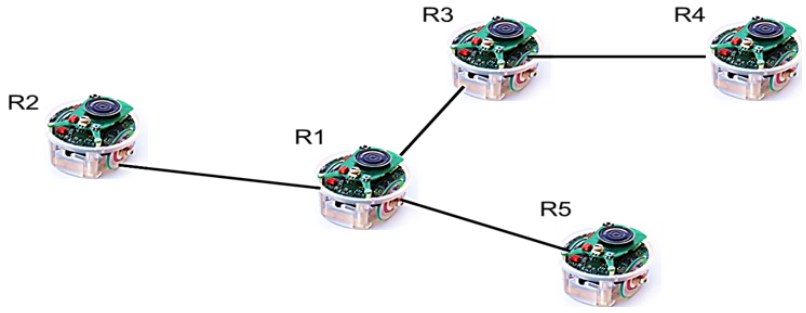

Figure 2. An **e**xample of a multi robot system consisting of robots R_1–R_5, represented as an undirected graph.

3.1. Classification of Multi-Robot Systems

Multi-robot systems can take many forms, varying by their configurations, missions, and operational domains. To contextualise this variety, it is essential to understand the various aspects that affect the development of these systems. The taxonomy proposed in this study focuses on three critical dimensions: composition, coordination, and communication [1,5,6]. These are explained in the following sections.

3.1.1. Composition (Homogeneous Versus Heterogeneous)

Composition refers to the physical and functional makeup of a robot team, particularly the distinction between homogeneous and heterogeneous multi-robot systems. This distinction is defined by the capabilities and functionalities of the robots comprising the system [1,2,4,5]. Homogeneous systems consist of identical robots equipped to perform similar tasks and are commonly employed in applications requiring parallel execution of restricted operations such as search and rescue, surveillance, and mapping. These systems are relatively easier to design, deploy, and control, and are often more cost-effective due to their uniform hardware and simplified coordination requirements. In contrast, heterogeneous systems are comprised of robots with diverse capabilities, such as combinations of aerial and ground vehicles, enabling them to address more complex tasks including construction, exploration, and transportation [32,43]. By leveraging their complementary strengths, heterogeneous teams offer greater flexibility and adaptability and can mitigate the limitations of homogeneous systems, particularly in dynamic or uncertain environments. However, this increased capability comes at the cost of a higher system complexity in terms of coordination, communication, and control [1,2,4,5,32].

3.1.2. Coordination (Centralised Versus Decentralised)

Coordination concerns how tasks are distributed and synchronised across a team of robots. Effective coordination is essential for achieving collective objectives, minimising conflicts, and ensuring temporal consistency in collaborative tasks [1,6,7,10,33,34]. Coordination strategies are commonly categorised by their decision-making structure. In a centralised coordination, a leader maintains global knowledge of the environment and the robots' states, enabling efficient global planning and simplified control. It is also possible

to have multiple leaders during missions. This approach offers strong performance guarantees but is sensitive to communication failures and is best suited to small teams operating in static, known environments [1,2,33,34]. In contrast, a decentralised coordination relies on autonomous decision making by individual robots, improving scalability and robustness to failures and environmental changes. However, decentralised methods often produce sub-optimal solutions and require sophisticated communication mechanisms to ensure collision avoidance and task completion [1,7,10,33,34].

3.1.3. Communication: Explicit, Implicit, and Networked in Multi-Robot Systems

Communication is the foundational backbone enabling coordinated action, information sharing, and avoidance in the MRS. The choice of communication technology directly affects system scalability, robustness, efficiency, and overall performance. Robots must frequently exchange information to synchronise tasks, particularly in dynamic or unknown environments [1,2,6,7,12,36]. Studies have emphasised communication-aware planning, where robots adapt their trajectories not only to avoid obstacles but also to maintain communication quality, as poor network reliability can significantly degrade team performance [6,12,15,20,27,35]. Addressing these challenges requires co-designing communication protocols and control strategies, with the current research focusing on scalable frameworks capable of supporting large robot teams and complex operations [6–8,10,24]. Communication modalities define how information is exchanged among robots. Communication can be explicit, such as wireless communication via Wi-Fi, or implicit, where robots infer information through sensing or interaction with the environment, which itself may act as a communication medium [1,7,8]. Accordingly, inter-agent interaction can occur through sensing, direct communication, or environmental cues [1,6]. The simplest form, sometimes referred to as cooperation without communication, relies solely on environmental interaction [1,35,36]. More recently, dynamic network models have been adopted to represent varying communication and sensing relationships among mobile robots, offering a flexible and effective framework for coordinating multi-robot operations [7,8,10,43]. These categories are illustrated in Figure 3, which provides a structured framework for comparing different MRS architectures.

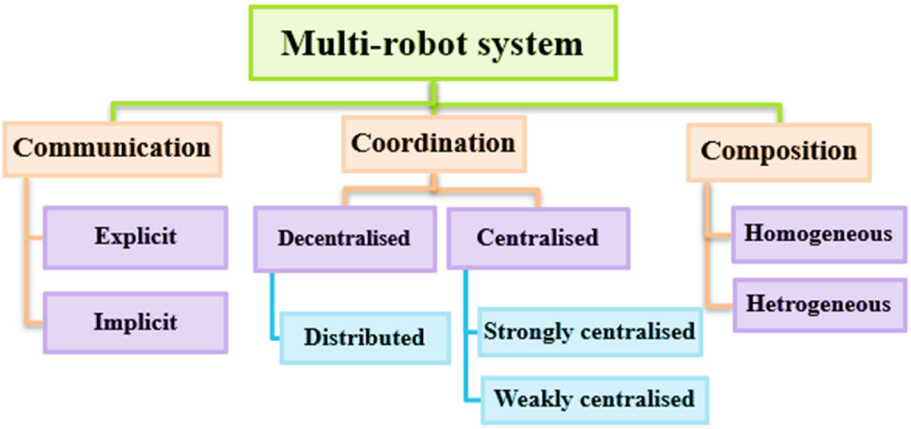

Figure 3. Classification of multi-robot systems.

3.2. Motivation and Challenges in Multi-Robot Path Planning

Path planning is a fundamental problem in robotics as it directly governs a robot's ability to navigate safely and efficiently in complex environments. The motivation for developing advanced path planning algorithms arises from the need to generate collision-free, optimal, and computationally efficient trajectories that can be reliably executed in real-world applications [1,25,40,41]. In an MRS, this motivation becomes more

pronounced due to the requirement for coordination among multiple robots operating within a shared workspace. Robots must not only avoid static obstacles but also prevent inter-robot collisions while accomplishing collective tasks [1,25,44]. These requirements introduce significant challenges, including scalability with respect to the number of robots, communication constraints for maintaining coordination, and robustness against individual robot failures. Addressing these challenges necessitates the development of planning algorithms capable of producing near-optimal solutions with low computational overhead, thereby bridging the gap between theoretical optimality and practical applicability [42,43].

Recent advances in mission planning for autonomous systems have also emerged in aerospace applications, particularly in spacecraft navigation under orbital threats. For example, mission planning strategies for autonomous avoidance of orbital debris and constraint-based planning approaches have been proposed to address safety critical decision making in highly dynamic environments [94,95]. These approaches emphasise real-time adaptability, constraint satisfaction, and robustness under uncertainty, offering valuable insights for multi-robot path planning in dynamic and safety-critical scenarios and providing a promising direction for extending multi-robot path planning toward more rigorous and safety-critical applications [94,95].

3.3. Graph-Theoretic Foundations for Multi-Robot Planning

Graph theory, a branch of discrete mathematics, has become a foundational tool in robotic motion planning, particularly within multi-robot systems [1,2,20,42]. Originating from Euler's solution to the Königsberg bridge problem in 1736 [1,19], graph theory has evolved into a powerful mathematical framework for modelling networks, connectivity, and pathfinding. In the context of multi-robot navigation, it provides an abstract yet structured representation of the workspace, enabling efficient modelling of spatial configurations, obstacles, communication links, and inter-robot interactions [25,26,42]. This abstraction is especially valuable for addressing key challenges in path optimisation, coordinated control, and connectivity preservation in complex environments [15,20,24]. A graph is formally defined as $G = (V, E)$, where $V \neq \emptyset$ is a finite set of vertices and $E \subseteq V \times V$ is a set of edges representing the connections between pairs of vertices. The vertex set can be expressed as $V = \{v_1, v_2, \ldots, v_n\}$, while the edge set is defined as $E = \{e_1, e_2, \ldots, e_m\}$, where m is not necessarily equal to n. Each edge $e_k \in E$ connects a pair of vertices, typically denoted as $e_k = (v_i, v_j)$. For undirected graphs, each edge is an unordered pair $(v_i, v_j) = (v_j, v_i)$, indicating a bidirectional connection between vertices.

In robotic motion planning, vertices may correspond to the robot positions, waypoints, or obstacle vertices, while the edges represent feasible paths, visibility relations, or communication links between entities [1,21,26,44]. Graphs can be directed or undirected and may be weighted or not weighted. Weighted graphs are particularly important in robotics, as the edge weights can encode distance, energy consumption, traversal cost, or collision risk, enabling more precise optimisation and decision making in complex environments [2,6,15,25,40].

3.3.1. Algebraic Graph Theory: Connectivity and Robustness

Algebraic graph theory extends the classical graph concepts by analysing the spectral properties of the matrices associated with graphs, most notably the adjacency and Laplacian matrices. In recent years, these tools have gained significant attention in multi-robot systems due to their ability to quantify and preserve communication connectivity and coordination robustness [15,18,19,24,45]. Connectivity-preserving control strategies based on Laplacian eigenvalues have been widely adopted to ensure reliable cooperation, even in partially connected or dynamically changing environments [15,24]. These methods not

only enhance robustness but also support the scalability of multi-robot systems in real-time applications [7,8,46,47].

For a weighted graph, the adjacency matrix $A \in \mathbb{R}^{n\times n}$ is defined as $A_{ij} = w_{ij}$, where $w_{ij} > 0$ represents the weight of the edge between nodes i and j, and $a_{ii} = 0$, for any i. The diagonal matrix (D) has entries corresponding to the sum of the weights of edges incident on each node [15,18,45]. The weighted Laplacian matrix is then defined as: $L = D - A$, where A is the adjacency matrix. The eigenvalues of the Laplacian matrix ($\lambda_1 < \lambda_2 \leq \cdots \leq \lambda_n$) provide a critical insight into network structure. For an undirected graph, the smallest eigenvalue $\lambda_1 = 0$ always exists and corresponds to a constant eigenvector. The multiplicity of these zero eigenvalues equals the number of connected components in the graph. The second smallest eigenvalue, known as the algebraic connectivity or Fiedler value λ_2 quantifies the robustness of the connected graph. A larger λ_2 indicates stronger connectivity, greater robustness to link failures, and improved resilience of communication networks [1,3,10,26–29,47].

In multi-robot networks, for a connected communication graph G, the zero eigenvalue $\lambda_1 = 0$ naturally arises from the Laplacian's structure and represents the baseline connectivity of the graph. The value of λ_2 ranges between 0 and the number of vertices (N), and the connectivity refers to the number of vertices in the graph, if the graph is completely connected. Thus, the maximum value of $\lambda_2 = N$, and it is obtained when the entries (i, j) of the adjacent matrix (A) are all equal to 1; this means all possible edges are present in it [10,28,29,47]. Figure 4 illustrates examples of robot interaction graphs with varying levels of algebraic connectivity, demonstrating how higher values of λ_2 correspond to stronger and more robust communication topologies [1,15,18,19,24].

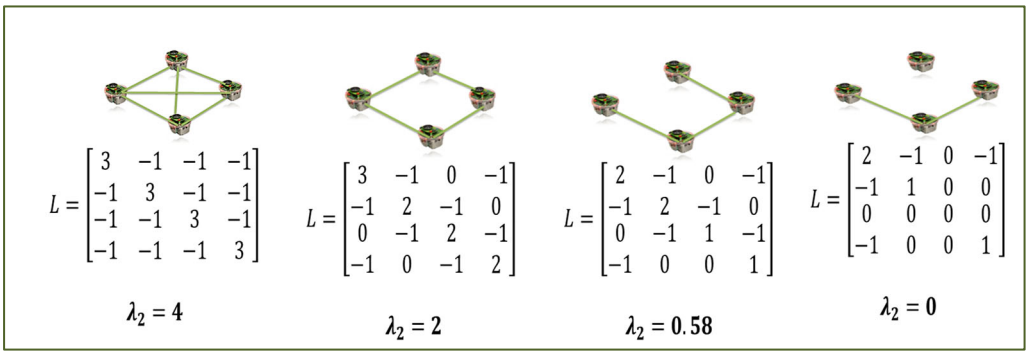

Figure 4. Examples of connected graphs with varying levels of algebraic connectivity.

3.3.2. Graph-Based Motion Planning Techniques

Graph-theoretic models form the backbone of many classical and modern motion planning techniques. Among the most prominent are cell decomposition (CD), roadmap methods (RM) (including visibility graphs (VG) and Voronoi diagrams (VD)), and potential fields (PF). Each technique offers distinct advantages, trade-offs, and suitable application contexts [9,11,13,14,21,26,48,49]. Our review focuses on the roadmap-based techniques. Figure 5 illustrates the structure of path planning techniques.

Roadmap Methods

Roadmap (RM) methods offer a balanced and scalable framework for multi-robot motion planning by representing the free configuration space as a graph, where the vertices correspond to valid configurations and the edges denote feasible transitions [21,25,26,49]. This abstraction enables efficient path computation using classical graph search algorithms [21,23,25]. Roadmap methods are scalable; are well suited to environments with sparse, known obstacles; and support efficient replanning in semi-dynamic

settings without requiring complete reconstruction of the graph [2,6,25,48–50]. Key roadmap-based methods include:

a. Visibility graph (VG): This is a subclass of roadmap methods that connects all pairs of mutually visible vertices (e.g., obstacle corners and robot or goal positions) with straight lines. It produces the shortest possible paths in a polygonal environment [1,21,25,50,51]. However, VG has the notable drawback of often generating paths that pass too close to the obstacles. This can be unsafe in practical applications, and it scales poorly as the number of obstacles increases. VG is most effective when path optimality is crucial, and the environment is static and fully understood [1,22,51].

b. Voronoi diagram (VD): This represents another roadmap technique that generates paths equidistant from the closest obstacles. This approach emphasises safety by maximising obstacle clearance, making it ideal in situations where collision risk must be minimised (e.g., high-speed navigation or uncertain sensing) [1,14]. Though VD paths may not be the shortest, the diagram is more scalable than VG and moderately complex to implement. Like VG, VD is not well-suited for rapidly changing environments that require pre-computation adjustments [14,27,50]. The operations of VG and VD are illustrated in Figure 6.

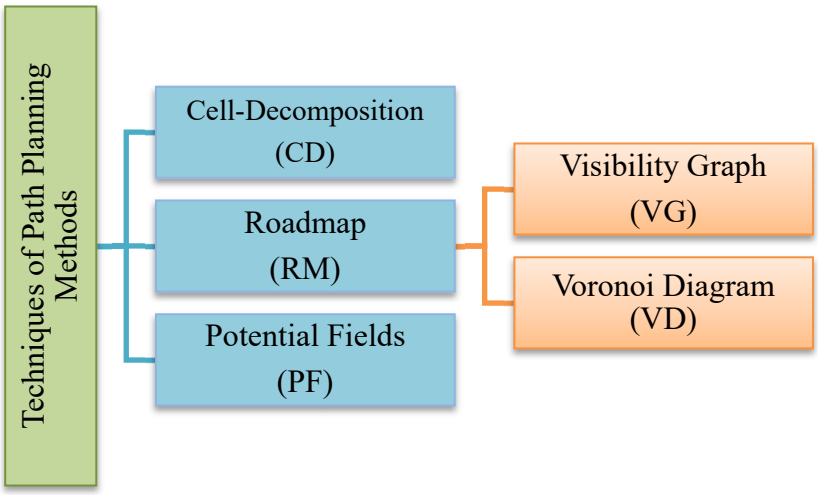

Figure 5. The structure of path planning techniques.

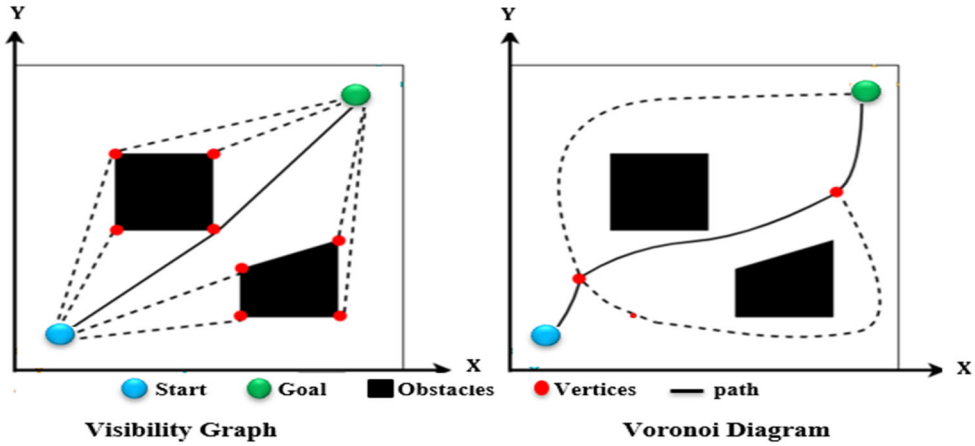

Figure 6. Operations of visibility graph (VG) versus Voronoi diagram (VD) for path planning (X and Y are the coordinates). In the VG, the dashed lines represent visibility edges connecting mutually visible vertices while the solid lines denote the computed shortest path between the start and goal locations. In the VD, the dashed lines represent the Voronoi roadmap (medial-axis network), and the solid lines indicate the selected path from the start to the goal.

VG and VD have long been foundational techniques in robot path planning for static, well-structured environments [22,26,49,50]. However, they often exhibit limitations when dealing with dynamic obstacles or complex, high-dimensional spaces due to their pre-computational nature and scalability constraints [14,22,26]. VG provides optimal shortest paths but often compromises safety by routing paths too close to obstacles, which is suboptimal for practical applications where robustness is critical [14,26]. In contrast, VD prioritises safe navigation by maximising obstacle clearance, making it more suitable for high-speed or uncertain environments, though at the cost of path optimality [14,22,50]. These techniques are fundamentally supported by graph-theoretic models, which facilitate planning through graph search algorithms (e.g., the Dijkstra's, A*, D*) [10,15,24,54–60].

Graph Search Algorithms

Graph search algorithms play a central role in robot path planning, particularly within roadmap-based and grid-based representations of the static environment. Among the most used algorithms are Dijkstra's and A*. Both algorithms compute shortest paths on weighted graphs, but they differ in efficiency and heuristic guidance.

Dijkstra's algorithm guarantees optimal solutions in graphs with non-negative edge weights and is deterministic, making it well suited to offline planning in fully known environments [1,21,25,57–59]. In contrast, A* improves computational efficiency through heuristic guidance but requires admissible and consistent heuristics, which can be difficult to define in complex polygonal environments. Incremental methods such as D* are designed for dynamic or partially known environments by updating the paths as conditions change, but they introduce additional computational overhead [1,10,15,16,24,41].

In multi-robot systems, Dijkstra's algorithm is therefore commonly preferred during offline planning stages due to its robustness, simplicity, and guaranteed optimality, whereas A* and D* are more suitable for real-time replanning in dynamic settings [1,2,4,12,25]. Since this review focuses on offline planning in static, fully known environments and employs VG representations, Dijkstra's algorithm is adopted to ensure deterministic behaviour and globally optimal path solutions, while heuristic-based and incremental methods are not considered [1,2,15,21,25,56].

3.4. Role of Graph Theory in Multi-Robot Planning Architectures

Graph-theoretic planning offers a structured and effective framework for modelling, analysing, and solving coordination and navigation problems in multi-robot systems. By abstracting robots, environments, and their interactions as vertices and edges, graph-based representations enable systematic analysis of connectivity, collision avoidance, and path optimality [1,43,52–54]. This abstraction is particularly effective for multi-robot path planning in structured and static environments. A key advantage of graph-theoretic planning lies in its modularity. The environment is represented as a roadmap graph, while robot states and interactions are incorporated as adaptable components, allowing obstacle geometry, start and goal positions, and inter-robot relationships to be modified independently [49,57]. This modular structure supports scalable and reusable planning architectures [1,2,5,53]. In addition, classical graph search algorithms, most notably Dijkstra's algorithm, guarantee optimal solutions in graphs with non-negative edge weights. In roadmap-based planners such as MRPP, CA, and OCA, this reduces continuous motion planning to a discrete shortest path problem, ensuring optimal trajectories with respect to the underlying graph [1–3,25].

Graph-theoretic frameworks also facilitate coordination and collision avoidance by encoding obstacle constraints and inter-robot distances within graph structures and edge weights [52,53]. When integrated with visibility graph representations and Laplacian-based connectivity analysis, these methods provide a mathematically grounded and computationally efficient basis for coordinated navigation [3,4,7,8,10,12,18]. Scalability and robustness are

further enhanced through spectral graph theory, where λ_2 quantifies network cohesion and supports connectivity-preserving planning strategies [52–54]. In multi-robot planning, λ_2 is used to preserve communication connectivity and guide robot prioritisation, a capability that is central to the MRPP framework [6,15,19,24]. Collectively, these properties establish graph theory as a central foundation for multi-robot path planning. The reviewed MRPP, CA, and OCA algorithms exploit visibility graph representations and spectral graph properties to enhance coordination, efficiency, and safety in structured and static environments.

4. Visibility-Graph-Based Planning and Algebraic Connectivity

VG-based planning provides a principled and mathematically rigorous framework for multi-robot path planning in static and structured environments. In a VG representation, the vertices correspond to obstacle vertices as well as the robots' start and goal positions. The edges represent direct line-of-sight connections that do not intersect any obstacles [1,51]. Each edge is typically weighted by its Euclidean length, transforming the continuous motion planning problem into a discrete shortest path search on an undirected weighted graph. When combined with exact graph search algorithms, this representation guarantees shortest path optimality with respect to the constructed roadmap [50–52].

In multi-robot path planning, the VG-based roadmaps are particularly effective in sequential planning architectures, where the paths generated for the earlier robots influence the planning space of the subsequent robots [1,52]. In this context, VG offers deterministic behaviour and predictable performance, which are essential for analysing coordination strategies and optimisation criteria. The MRPP, CA, and OCA algorithms all adopt VG-based representations as their foundational roadmaps. By integrating VG geometry with connectivity-aware coordination mechanisms, these algorithms achieve efficient, collision-free, and coordinated motion planning across multi-robot teams.

4.1. Visibility Graph Construction and Graph-Based Representation

VG is constructed from an undirected weighted graph $G = (V, E, w_{ij})$, where the vertex set V consists of the obstacle corners together with each robot's start and goal positions, and the edge set E represents the feasible straight-line connections between mutually visible vertices. An edge $(v_i, v_j) \in E$ exists, if and only if, the line segment connecting v_i and v_j does not intersect any obstacle boundary. For polygonal obstacles, vertices correspond to the obstacles' corners, and visibility edges are established between all pairs of mutually visible vertices. In this case, the edge set can be expressed as: $E = E_{VG} \cup E_{Obs}$, where E_{VG} denotes visibility edges and E_{Obs} represents the obstacle boundary edges. For non-polygonal obstacles, such as circular or curved shapes, the VG construction is adapted by approximating the obstacle boundaries using a finite set of sampled boundary points or tangent points. In the case of circular obstacles, edges are typically formed using tangent visibility lines that connect start and goal positions and obstacle boundaries without intersecting the obstacle interior. This ensures that the visibility graph remains valid while preserving collision-free path planning [1,25,52]. The weighted $w_{ij} = d_{ij}$ is the Euclidean distance associated with each edge, enabling shortest path calculations using Dijkstra's algorithm. This formulation reduces motion planning to a graph optimisation problem while preserving geometric optimality in polygonal environments [60–64]. In multi-robot scenarios, the robots operate within a shared two-dimensional workspace and are subject to communication constraints. The robot team can be modelled as an undirected weighted interaction graph, where each robot position $p_i \in \mathbb{R}^n$ represents a node and the edges indicate communication or sensing relationships within a limited range. Neighbourhood relationships are defined based on inter-robot distance, ensuring that robots can exchange information with nearby teammates. This graph-based representation provides a unified framework for integrating motion planning, communication awareness, and coordination constraints [1,25,52]. By combining VG roadmaps with inter-robot interaction graphs,

planning algorithms can account for both environmental geometry and robot-to-robot relationships. This abstraction forms the basis for coordinated path planning with inter-robot awareness, enabling scalable and structured multi-robot navigation [65–67].

4.2. Algebraic Connectivity and Collision Avoidance

Algebraic connectivity provides a critical measure for coordinating multiple robots by quantifying the robustness of communication links within a robot network. In multi-robot path planning, λ_2 is particularly valuable for guiding coordination and prioritisation strategies and for supporting reliable information exchange among the robots [15,19,24]. Collision avoidance in practical multi-robot planning can be addressed through several complementary mechanisms. First, the planning problem can be formulated in a higher-dimensional configuration space, where each dimension represents the state of an individual robot and the collisions correspond to forbidden regions [1,19,66–70]. Dijkstra's algorithm can then be employed to compute collision-free paths within this space. Second, prioritised planning is commonly adopted, in which robots plan sequentially, and paths generated for higher-priority robots are treated as dynamic obstacles for lower-priority robots, thereby reducing inter-robot conflicts [26,68,70]. Third, algebraic connectivity supports coordination and collision avoidance by guiding planning order and helping to prevent simultaneous conflicts in shared regions [3,14,15,19,24].

Among the reviewed algorithms, MRPP explicitly integrates algebraic connectivity into the planning process to sequence robots and preserve network cohesion during path generation. Although CA and OCA do not directly optimise λ_2, their reduced VG structures indirectly influence connectivity by simplifying interaction regions and improving safety margins. The combined use of VG and λ_2 therefore provides both geometric optimality and coordinated robustness, forming a strong foundation for scalable and reliable multi-robot path planning [14,25,64,65].

4.3. Multi-Robot Path Planning Algorithm

MRPP is a graph-based framework designed to address the coordination and navigation challenges in multi-robot systems operating within shared environments. The primary objective of MRPP is to generate collision-free and near-optimal paths for multiple robots while preserving robust inter-robot connectivity. MRPP integrates three fundamental components:

(i) VG for modelling the environment and capturing shortest line-of-sight connections between vertices.
(ii) Dijkstra's algorithm for computing optimal paths.
(iii) Algebraic connectivity (λ_2) to assess and preserve communication robustness among robots.

A distinguishing feature of MRPP is its sequential planning strategy where the robots' paths are generated individually rather than simultaneously. The planning order is determined using λ_2 such that robots whose trajectories are more critical to maintaining overall system connectivity are prioritised. This connectivity-aware sequencing reduces potential conflicts and enhances coordination robustness. Once a robot's path is computed, it is treated as a dynamic obstacle for subsequent robots, ensuring inter-robot collision avoidance without requiring full replanning.

To further minimise conflicts, edge weights within the graph are dynamically adjusted based on previously planned paths, discouraging excessive overlap and promoting spatial separation among robot trajectories. By applying Dijkstra's algorithm on a well-constructed VG with modified weights, MRPP guarantees the shortest feasible paths under the given constraints while maintaining safety. This combination enables effective coordination, collision

avoidance, and path optimality, making MRPP a suitable framework for multi-robot navigation in complex yet structured environments.

Despite its strengths, MRPP has certain limitations, e.g., paths may pass close to obstacle boundaries, thus potentially reducing safety margins in cluttered environments. The computational cost increases with environmental complexity and the number of robots, which may limit scalability in highly cluttered or large-scale workspaces. In addition, as MRPP primarily relies on offline planning, its performance may degrade in highly dynamic environments that require frequent replanning. These limitations motivate the development of reduced-graph approaches, such as CA and OCA, which aim to retain the strengths of MRPP while improving efficiency and safety [1,25,26].

4.4. Central Algorithm (CA)

The CA is a roadmap-based path planning approach proposed to mitigate the high computational complexity associated with traditional VG methods while preserving near-optimal path quality. Although VG-based planning guarantees shortest paths in static environments, its computational cost grows rapidly with the number of obstacle vertices, making it less efficient in densely cluttered workspaces. CA mitigates this issue by introducing a structured reduction of the VG through the concept of a central baseline (CB).

The CB is constructed between the start and goal positions of each robot. Only obstacles that intersect this baseline are considered during roadmap construction. For each relevant obstacle, a limited set of waypoints (W) is generated in the free configuration space surrounding the obstacle, typically four waypoints per obstacle at strategic locations that enable safe detours. This selective process results in a reduced visibility graph, significantly decreasing the number of vertices and edges compared to a full VG. The Dijkstra algorithm is applied to compute the shortest path. By operating on a streamlined graph, CA achieves substantial improvements in computational efficiency while maintaining path lengths comparable to those produced by full VG-based planning. This makes CA particularly suitable for dense environments with spatially distributed obstacles where full VG construction is prohibitive.

Overall, CA provides an effective trade-off between optimality and efficiency. Its structured roadmap reduction makes it well suited for rapid offline planning in static environments where computational efficiency is critical. However, because CA prioritises efficiency, the resulting paths may pass closer to obstacle boundaries, leading to reduced safety margins in highly cluttered environments. This limitation motivates the optimisation central algorithm (OCA), which extends CA by explicitly incorporating safety distance constraints while retaining its computational advantages [1,25,26].

4.5. Optimisation Central Algorithm (OCA)

The OCA algorithm is an extension of the CA algorithm that explicitly incorporates safety considerations into the roadmap construction and path planning process. While CA primarily focuses on reducing visibility graph complexity to improve computational efficiency, OCA enhances this framework by introducing a safety distance D_s to ensure robust and collision-free navigation in practical environments (D_s is defined as the average minimum allowable distance between the robot and surrounding obstacles). OCA applies an obstacle expansion strategy by a predefined safety margin δ. The expanded obstacles effectively shrink the free configuration space, ensuring that any planned path maintains sufficient clearance from real obstacles. Waypoints are then generated around the boundaries of these expanded obstacles in the configuration space, allowing the planner to produce general and feasible solutions across different map layouts. This contributes to smoother and more executable trajectories while preserving safety. Dijkstra's algorithm is employed to compute the shortest feasible path between the start and goal positions. Despite the introduction of expanded obstacles and

additional constraints, OCA retains the computational advantages of CA by considering only obstacles intersecting the central baseline, thereby maintaining tractability.

Overall, OCA achieves a balanced trade-off between safety, optimality, and efficiency. By integrating safety distance constraints and obstacle expansion, it provides a robust and practical solution for multi-robot path planning in static, fully known environments where reliable clearance and execution robustness are paramount [1,25,26].

5. Taxonomy of Multi-Robot Path Planning Methods

Multi-robot path planning methods can be systematically classified according to their environmental knowledge, planning architecture, environment representation, and optimisation criteria [1,66]. In addition, this taxonomy provides a structured framework for positioning MRPP, CA, and OCA within the broader landscape of multi-robot path planning methods, enabling clearer comparison with alternative approaches. These methods are illustrated in Figure 7.

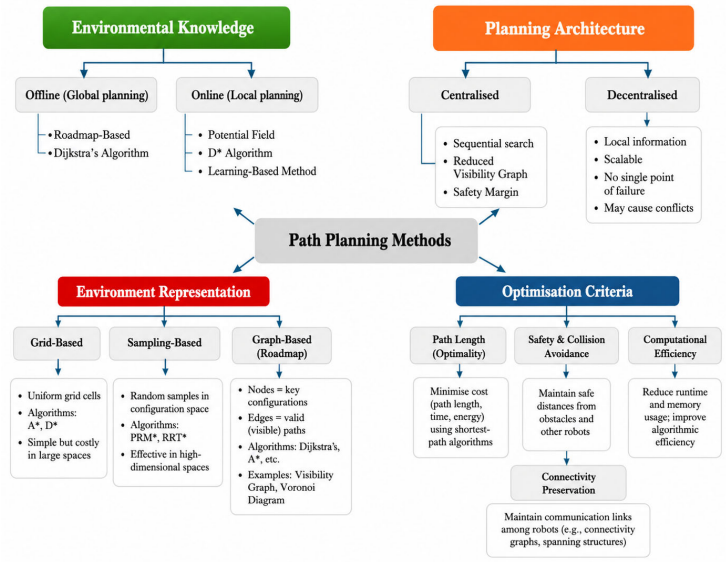

Figure 7. Taxonomy of multi-robot path planning methods: environmental knowledge, representations, architectures, and optimisation criteria.

5.1. Environmental Knowledge

Environmental knowledge refers to the extent of availability of the information about the workspace during the planning process. In multi-robot path planning, this knowledge can range from fully known static environments to partially known or dynamically changing settings. In practice, many multi-robot systems adopt hybrid approaches where an offline global plan is generated using roadmap-based techniques (e.g., VG, VD) and then refined online to accommodate dynamic obstacles or inter-robot interactions. This combination exploits the optimality and efficiency of offline graph search while retaining the flexibility of online replanning. Consequently, integrating offline graph construction with online adaptive control remains a promising direction for scalable and robust multi-robot path planning.

Offline and Online Planning

Path planning approaches can broadly be divided into offline (global) and online (local) methods, depending on whether the environment is fully known a priori or evolves during execution [2,38]. This distinction is particularly relevant for graph-based and roadmap-based planning frameworks. Offline planning assumes complete knowledge of

the environment, including the obstacle geometry and robot configurations [37]. Under this assumption, global roadmaps can be constructed and optimal paths computed prior to execution. Such approaches are well suited to static environments and allow the use of exact graph-based algorithms, such as Dijkstra's algorithm, to compute shortest paths with guaranteed optimality [38]. Roadmap-based methods, particularly VG, are therefore widely adopted for offline planning in polygonal environments [1,51,67].

In contrast, online planning addresses partially known or dynamic environments where the robots must update their paths during execution in response to environmental changes [1,15,70,73]. Common approaches include D*, potential field methods, and learning-based planners. Although these approaches enhance adaptability and real-time responsiveness, they typically forgo global optimality [27,43,59,65]. In graph-based frameworks, online planning is often realised through incremental roadmap updates or local replanning on a precomputed graph when new obstacles are detected [1,53,73].

5.2. Environment Representation

Environment representation determines how the workspace is modelled. It plays a critical role in determining the efficiency and optimality of robot path planning.

5.2.1. Grid-Based Representation

Grid-based representation divides the space into a regular grid of uniform cells. Common pathfinding algorithms include A* and D*. While straightforward and easy to implement, it can become computationally costly in large environments due to the high number of cells.

5.2.2. Sampling-Based Representation

Sampling-based representation configures the space using randomly generated samples. Common algorithms include probabilistic roadmap (PRM*) and rapidly exploring random tree (RRT*). This representation is particularly effective in high-dimensional spaces where grid-based methods become inefficient.

5.2.3. Graph-Based (Roadmap) Representation

Graph-based representation models the environment as a graph composed of nodes (key configurations) and edges (valid paths between them). It supports shortest path planning algorithms. Typical examples include Voronoi diagrams and visibility graphs. Roadmap construction is a fundamental stage in graph-based path planning where the continuous workspace is abstracted into a discrete graph that captures feasible robot motions. In an MRS operating in static environments, an effective roadmap must balance geometric accuracy with computational efficiency [1,30–43].

One of the common representations is the VG-based roadmap method [1,25]. It is particularly effective at providing an exact geometric representation of free space in static polygonal environments by connecting mutually visible obstacle vertices, start points, and goal locations to form a weighted graph [21,51]. This representation enables the computation of globally shortest paths using graph search algorithms such as Dijkstra's algorithm [1,21,25,40,49,57].

5.3. Planning Architecture

The planning architecture defines how decision making and coordination are organised in multi-robot systems. Planning strategies are generally classified as centralised or decentralised and can also be implemented using simultaneous or sequential approaches [68,72].

5.3.1. Centralised Planning

In central planning, a single global planner computes the paths for all robots within the system. This approach ensures coordinated behaviour and can produce globally optimal solutions. However, it suffers from high computational cost and poor scalability as the number of robots increases.

5.3.2. Decentralised Planning

In decentralised planning, each robot independently generates its own plan using local information. This approach improves scalability and robustness since there is no single point of failure. However, this may lead to conflicts between robots and generally produces suboptimal solutions compared to centralised methods.

5.4. Optimisation Criteria

Optimisation criteria define the objectives guiding planning and distinguish the functional priorities of different algorithms [65]. The most common criterion is path length minimisation, which is pursued by all three algorithms reviewed. Beyond geometric optimality, additional criteria are incorporated to address coordination and safety requirements in multi-robot systems [1,25,65,70,73].

- Path Optimality: Focuses on minimising cost metrics such as path length, travel time, or energy consumption, typically achieved using shortest path algorithms.
- Computational Efficiency: Aims to reduce runtime and memory usage by simplifying graph structures and improving algorithmic efficiency during planning.
- Safety and Collision Avoidance: Ensures robots maintain safe distances from obstacles and from each other, preventing collisions during navigation and execution.
- Connectivity Preservation: Maintains communication links among robots by ensuring network connectivity, often evaluated using graph-theoretic measures such as connectivity graphs or spanning structures.

5.5. Positioning of MRPP, CA, and OCA Within the Taxonomy

To contextualise the contributions of this study, the three reviewed algorithms are positioned within the proposed taxonomy as outlined in following sections.

5.5.1. The Offline Environmental Knowledge

The MRPP, CA, and OCA algorithms operate within an offline planning framework assuming complete and accurate prior knowledge of the environment. This allows detailed VG construction and the computation of globally optimal paths before execution. In multi-robot settings, offline planning also enables the use of λ_2 to sequence robots and reduce conflicts, as employed in MRPP. While offline methods provide high-quality solutions in static environments, they are limited in their ability to cope with unforeseen changes. The reliance on offline planning reflects a deliberate design choice that prioritises optimality and coordination over responsiveness to dynamic environments [1,25].

5.5.2. Centralised Planning Architecture

In this review, all considered algorithms adopt a centralised planning framework, where the global knowledge of the environment and robot states are available during the planning stage. Such architectures are particularly suitable for offline planning in static environments, as they enable the construction of complete roadmap representations and ensure globally consistent solutions [44,57]. Centralised planning also facilitates coordination among robots, reducing the likelihood of conflicts during execution [1,33,66]. Within this framework, the algorithms differ in their architectural strategies. The MRPP

algorithm follows a sequential planning strategy in which robots are prioritised based on their influence on system connectivity to reduce inter-robot conflicts [1,25]. In contrast, CA introduces a reduced VG to limit the search space and improve computational efficiency, while the OCA extends CA by incorporating a safety margin to enhance collision avoidance and robustness. These distinctions highlight how architectural choices directly influence efficiency, safety, and scalability while retaining centralised coordination [1,26].

5.5.3. Graph-Based (VG) Representation

VG algorithms utilise VG-based roadmaps to discretise the workspace [1,25,26]. These algorithms establish an undirected weighted graph as the search space, by connecting mutually visible vertices of polygonal obstacles, along with robot's start and goal positions [1,25,26]. However, the algorithms differ substantially in their implementation of roadmap reduction techniques and the specific criteria used to refine the resulting search space for multi-robot coordination [1,25,43]. MRPP utilises a full visibility graph, ensuring global optimality but at a higher computational cost in complex environments. CA reduces roadmap complexity by considering only obstacles intersecting a central baseline between start and goal regions, producing a reduced visibility graph that preserves path quality while improving efficiency. OCA further modifies the roadmap by expanding obstacles in configuration space, enforcing safety distances and generating safer trajectories. These progressive refinements demonstrate how roadmap construction directly influences efficiency, safety, and scalability in multi-robot path planning [1,25,26].

Although the algorithms differ in how the VG is constructed or reduced, they all maintain VG as the foundational representation, ensuring consistency in path optimality while allowing algorithm-specific optimisations [1,25,26,50–54]. This shared foundation enables fair comparison between the algorithms as differences in performance arise from roadmap refinement, coordination strategy, and optimisation objectives rather than from fundamentally different environment models [1,25].

5.5.4. Dijkstra's Algorithm: Path Search and Optimality

The MRPP, CA, and OCA algorithms utilise Dijkstra's algorithm as their primary pathfinding mechanism due to its deterministic behaviour and guaranteed optimality. Once the workspace is modelled as a graph-based roadmap, the motion planning problem is transformed into a shortest path search problem over the constructed graph. This choice ensures that the resulting paths are mathematically optimal with respect to the underlying graph construction and the specific design of each algorithm [1,36,60–68].

Dijkstra's algorithm is selected for several key reasons. First, it guarantees global optimality in VG-based roadmaps [74]. Second, it provides a deterministic and heuristic-free baseline, which is essential for fair evaluation of the graph construction and reduction strategies employed by CA and OCA [1,54]. Third, it enables a clear separation between optimal path computation and higher-level coordination mechanisms [18,25] (for example, λ_2-based sequencing in MRPP, where the emphasis is placed on inter-robot coordination rather than heuristic guidance of the search process, and preservation of network connectivity throughout the planning process [1,25,40]).

Although Dijkstra's algorithm is computationally demanding for large graphs, in the context of offline planning, its cost is effectively mitigated using reduced graph representations in CA and OCA. As a result, efficient computation is achieved while preserving the optimality guarantees of the underlying path planning process.

5.5.5. Optimisation Objectives

Path length minimisation is a common objective shared by MRPP, CA, and OCA. It is achieved through a shortest path search on the visibility graph [1,25,26]. This ensures

that all algorithms generate efficient paths with respect to the Euclidean distance. Beyond path length, each algorithm incorporates a distinct optimisation criterion aligned with its design goals [40,57,74]. To ensure a consistent and objective evaluation of the algorithms, the following performance metrics are employed.

- Path Length (PL): This represents the total distance travelled by a robot from its start position to its goal position. It is computed as the sum of the weights of all edges along the planned path.

$$PL = \sum_{(i,j)\in P_{\min}} w_{ij} \tag{1}$$

where P denotes the set of consecutive edges forming the path, and w_{ij} is the weight of the edge connecting vertices v_i and v_j, computed as the Euclidean distance between them, i.e., $w_{ij} = \| v_i - v_j \|$.

- Arrival Time (AT_i): Time to reach the goal, assuming constant robot speed (S) = 1 unit/second (can be changed if needed). The arrival time for robot i is calculated as

$$AT_i = \frac{PL}{S} \tag{2}$$

- Connectivity (λ_2): Algebraic connectivity value that posts each planning step (for MRPP).
- Central Baseline (CB): Reduce obstacles and generate waypoints (for CA). The efficiency of CA is evaluated by measuring the reduction in obstacles and vertices considered during planning. This reduction is expressed as the central baseline reduction (R_{CB}), $R_{\mathrm{CB}} = \frac{\hat{O}_{CB}}{O_{\mathrm{VG}}}$, where $\hat{O}_{CB}$ is the number of obstacles intersecting CB and O_{VG} is the total number of obstacles in the full VG. A lower value for R_{CB} indicates a greater computational saving.
- Safety Distance (D_s): This metric represents the minimum clearance between the robot trajectory and surrounding obstacles in the OCA framework. It is defined as

$$D_s = \min_{i\in P, j\in O} w_{ij} \tag{3}$$

where P denotes the set of vertices along the robot path, O represents the set of obstacle boundary points (or vertices), and $w_{ij} = \| v_i - o_j \|$ is the Euclidean distance between a path vertex v_i and an obstacle point o_j, computed as

$$\| v_i - o_j \| = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{4}$$

This formulation evaluates the distance between all path points and all obstacle boundary points, and selects the minimum value, representing the closest approach between a robot and obstacles. In the OCA framework, a safety constraint is enforced by ensuring that D_s is at least equal to a minimum distance. This ensures that the planned path maintains a predefined safety margin from all obstacles.

- Computation Time: This is the time required to compute the complete set of paths.

Figure 8 presents the taxonomy adopted in this review, positioning the MRPP, CA, and OCA algorithms within a unified conceptual framework. The figure illustrates how these methods diverge based on environment representation, planning architecture, and optimisation criteria, highlighting their shared foundation in offline visibility graph construction and shortest path search.

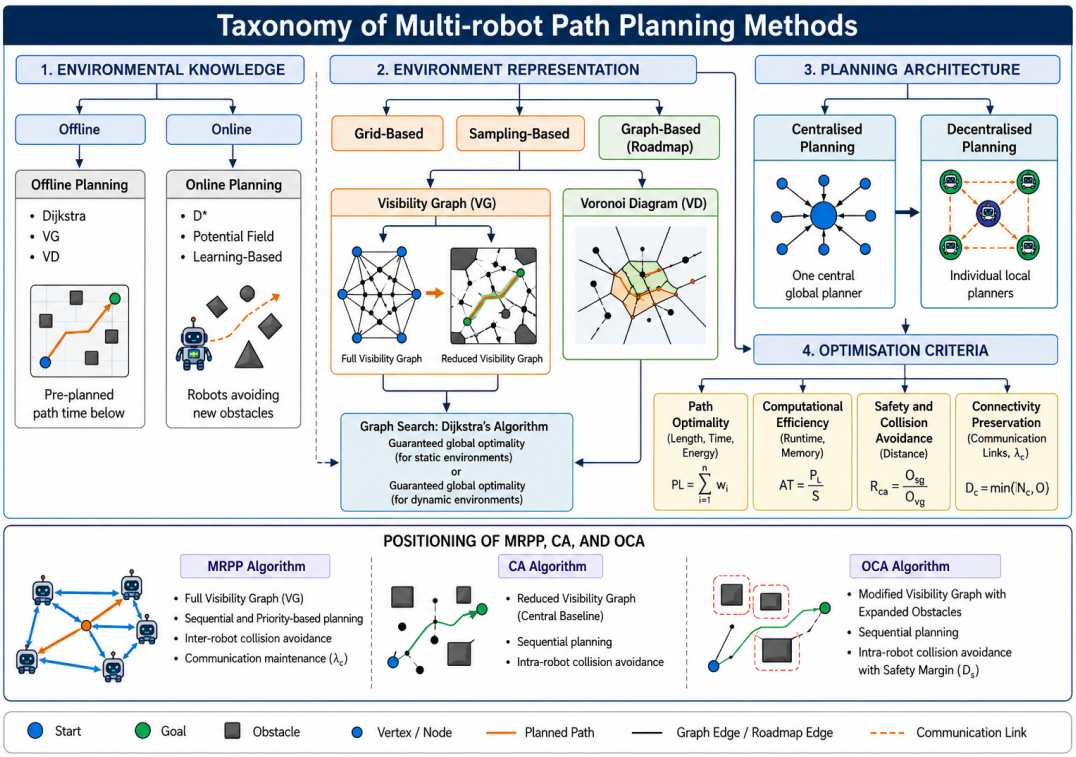

Figure 8. Multi-robot path planning taxonomy: a comparative framework for MRPP, CA, and OCA.

To demonstrate the above-defined optimisation metrics and provide objective comparison, a controlled experimental evaluation was conducted. The following subsection presents the quantitative results obtained under consistent simulation conditions.

Quantitative Performance Evaluation

To evaluate the practical performance of the MRPP, CA, and OCA algorithms, a common simulation scenario was considered, as illustrated in Figures 9–11. The environment consists of five robots (R_1-R_5) navigating within a shared obstacle-rich workspace. Red circles represent the robots' initial positions, while green stars indicate the corresponding goal locations (G_1-G_5). Static obstacles are represented by grey blocks. All algorithms were evaluated under identical initial and goal configurations, and robot velocity was assumed constant ($S = 1$ unit/s). The simulations provide representative examples of how each algorithm generates coordinated collision-free paths while balancing path efficiency, safety, and computational overhead.

Path Length Assessment: The path lengths generated by each algorithm are summarised in Table 2.

Table 2. Quantitative comparison of path length (m) across algorithms.

Robot	MRPP	CA	OCA
R_1	9.70	9.05	9.04
R_2	10.45	8.76	7.75
R_3	9.53	9.35	9.32
R_4	8.95	6.94	6.85
R_5	10.19	8.61	8.50

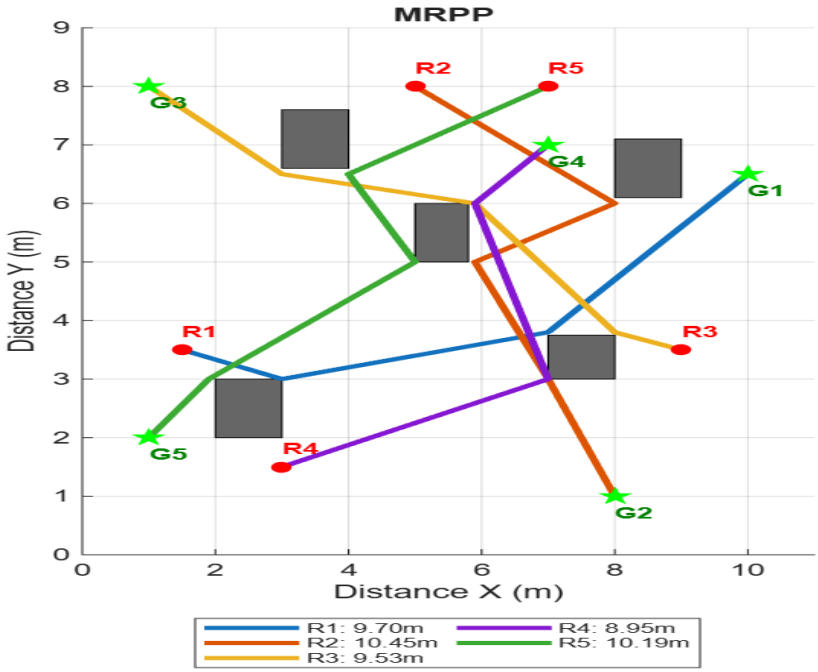

Figure 9. Illustrative simulation results of the application the MRPP for five robots (R_1–R_5 shown as red dots) with five goals (G_1–G_5 shown as green stars). The coloured connections are the paths the corresponding robots followed. The grey squares are the obstacles.

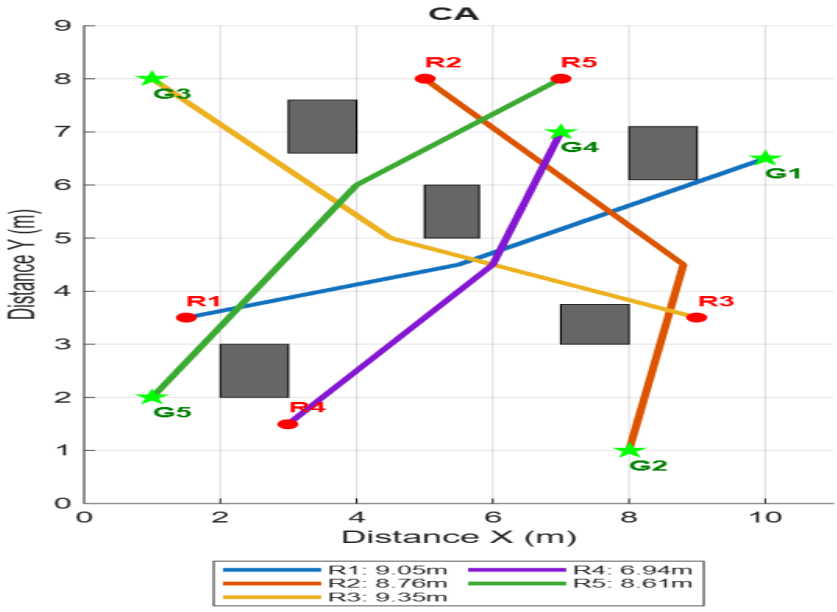

Figure 10. Illustrative simulation results of the application the CA for five robots with five goals (the symbols are as described in Figure 9).

Average Performance Across Robots: For N robots and individual path length PL_i, the average path length (APL) for each algorithm was computed as indicated below and summarised in Table 3.

Table 3. Comparative arrival time analysis for MRPP, CA, and OCA algorithms.

Algorithm	Average Arrival Time (s)	Relative Improvement
MRPP	9.76	Baseline
CA	8.54	12.5% faster than MRPP
OCA	8.29	15.1% faster than MRPP

OCA vs. CA	—	2.9% faster than CA

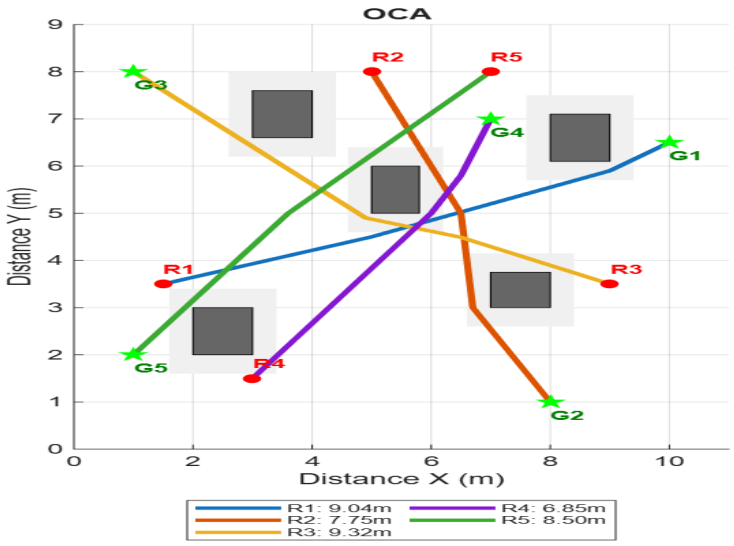

Figure 11. Illustrative simulation results of the application the OCA for five robots with five goals (the symbols are as described in Figure 9).

$$APL = \frac{1}{N}\sum_{i=1}^{N} P\,L_i \tag{5}$$

$$APL_{MRPP} = \frac{9.70 + 10.45 + 9.53 + 8.95 + 10.19}{5} = \frac{48.82}{5} = 9.76$$

$$APL_{CA} = \frac{9.05 + 8.76 + 9.35 + 6.94 + 8.61}{5} = \frac{42.71}{5} = 8.54$$

$$APL_{OCA} = \frac{9.04 + 7.75 + 9.32 + 6.85 + 8.50}{5} = \frac{41.46}{5} = 8.29$$

The percentage reduction ($PR\%$) for two methods with path length PL_A and PL_B is calculated as

$$PR(\%) = \frac{APL_A - APL_B}{APL_A} \times 100 \tag{6}$$

$$\text{Comparing CA and MRPP: } PR = \frac{9.76-8.54}{9.76} \times 100 = \frac{1.22}{9.76} \times 100 = 12.50\%$$

$$\text{Comparing OCA and MRPP: } PR = \frac{9.76-8.29}{9.76} \times 100 = \frac{1.47}{9.76} \times 100 = 15.06\%$$

$$\text{Comparing OCA and CA: } PR = \frac{8.54-8.29}{8.54} \times 100 = \frac{0.25}{8.54} \times 100 = 2.93\%$$

Arrival Time Analysis: To further evaluate coordination efficiency, the average arrival time of the robots was analysed under the same simulation conditions. Since all robots were assumed to move with a constant velocity of $S = 1$ unit/s, the arrival time is directly proportional to the travelled path length. Consequently, shorter paths correspond to earlier task completion and improved mission efficiency. The average arrival time (AAT) for each algorithm is defined as:

$$AAT = \frac{1}{N}\sum_{i=1}^{N} AT_i \tag{7}$$

where AT_i denotes the arrival time of robot i, and N is the number of robots. Based on the simulation results: $AAT_{MRPP} = 9.76$ s, $AAT_{CA} = 8.54$ s and $AAT_{OCA} = 8.29$ s.

Computation Time: To evaluate computational efficiency, the relative computation time (CT) required for collision-free path generation was estimated for each algorithm. These are summarised in Table 4. Computation time represents the processing overhead prior to robot motion execution. The estimated computational characteristics of the algorithms are:

- MRPP: highest computation time (connectivity maintenance + multi-robot coordination overhead).
- CA: lower computation time (centralised shortest path optimisation).
- OCA: slightly lower than CA (obstacle-aware optimisation with reduced redundant exploration).

Table 4. A normalised comparison for computational time.

Algorithm	Relative Computation Time (s)	Interpretation
MRPP	2.48	Highest computational overhead
CA	1.73	Faster due to simplified central planning
OCA	1.52	Most computationally efficient

Compute Reductions: The percentage reduction in computation time ($PR_{CT}\%$) between two comparative architectures was evaluated using the following framework:

$$PR_{CT} = \frac{CT_A - CT_B}{CT_A} \times 100$$

where CT denotes the computation time required by each algorithm for path generation, and PR represents the percentage reduction between two compared methods.

$$\text{CA vs. MRPP: } PR_{CA/MRPP} = \frac{CT_{MRPP} - CT_{CA}}{CT_{MRPP}} \times 100$$

$$PR = \frac{2.48 - 1.73}{2.48} \times 100 = 30.24\%$$

$$\text{OCA vs. MRPP: } PR_{OCA/MRPP} = \frac{CT_{MRPP} - CT_{OCA}}{CT_{MRPP}} \times 100$$

$$PR = \frac{2.48 - 1.52}{2.48} \times 100 = 38.71\%$$

$$\text{OCA vs. CA } PR_{OCA/CA} = \frac{CT_{CA} - CT_{OCA}}{CT_{CA}} \times 100$$

$$PR = \frac{1.73 - 1.52}{1.73} \times 100 = 12.14\%$$

The estimated reductions indicate that CA significantly decreases processing overhead relative to MRPP, while OCA achieves the highest computational efficiency among the evaluated methods. Figure 12 summarises the comparative performance of MRPP, CA, and OCA in terms of robot trajectories, path length, arrival time, computation time, and relative optimisation improvements under identical environmental conditions.

Comparative Synthesis and Algorithmic Trade-Offs

Building upon the quantitative results presented earlier, this section provides a comparative synthesis of the algorithmic trade-offs between MRPP, CA, and OCA in terms of efficiency, arrival time, and computational complexity.

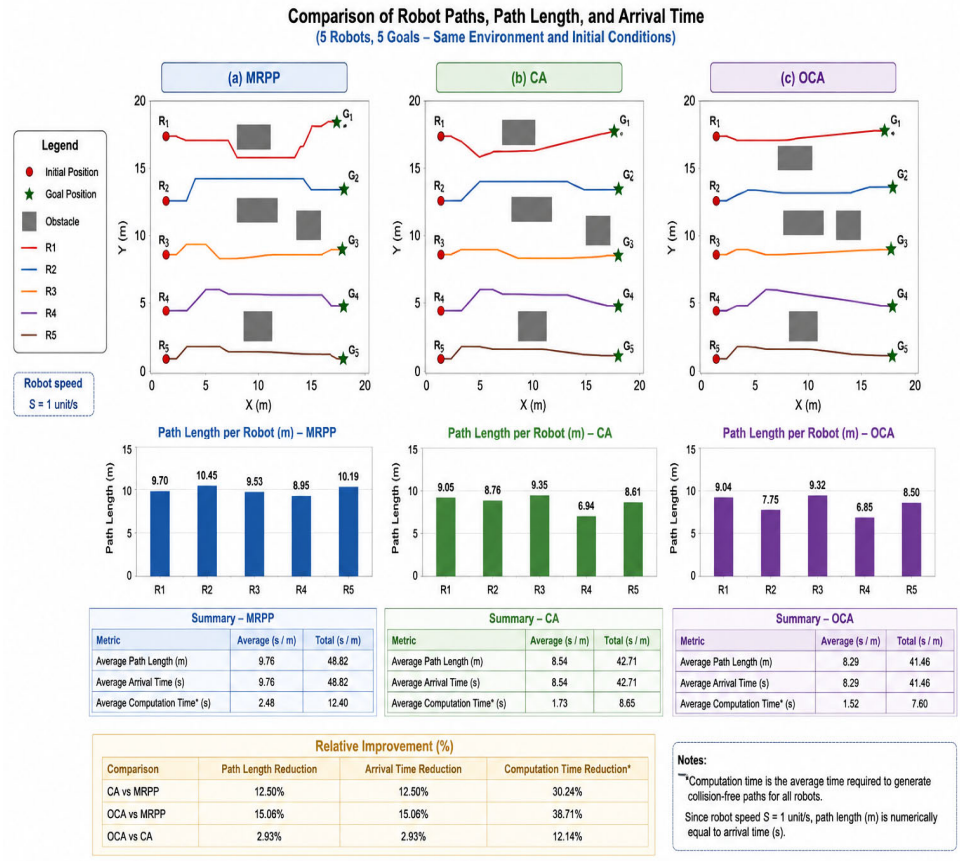

Figure 12. Comparative performance evaluation of the MRPP, CA, and OCA algorithms for five robots under identical environmental and initial conditions.

Path Efficiency and Connectivity Preservation: The comparative path length analysis showed that CA reduced the average path length by approximately 12.5% compared with MRPP, while OCA achieved a greater reduction of approximately 15.1% relative to MRPP. Furthermore, OCA improved upon CA by approximately 2.9%, indicating a more effective balance between path optimisation and obstacle-aware safety constraints. Among the evaluated methods, OCA consistently generated the shortest paths for most robots, demonstrating improved navigation efficiency in obstacle-rich environments. The CA approach also significantly outperformed MRPP due to its centralised optimisation strategy and reduced graph-search complexity. In contrast, MRPP produced longer paths because its primary objective emphasises connectivity preservation, coordinated navigation, and deadlock avoidance rather than strict path minimisation.

The most significant improvement was observed for robot R_2, where OCA reduced the path length by approximately 26% compared with MRPP. This highlights the effectiveness of obstacle-aware optimisation in reducing unnecessary detours and improving local navigation decisions.

Arrival Time and Coordination Efficiency: Since all robots were assumed to move at a constant velocity of $S = 1$ unit/s, the arrival time was directly proportional to the travelled path length. Consequently, shorter trajectories resulted in faster task completion and improved coordination efficiency. The results indicated that both CA and OCA achieved lower average arrival times than MRPP. Specifically, CA reduced the average arrival time by approximately 12.5%, while OCA achieved a reduction of approximately 15.1% relative to MRPP. In addition, OCA improved the arrival time by approximately 2.9% compared with CA. These findings confirm that the optimisation mechanisms integrated within CA and OCA enhance mission responsiveness and coordination efficiency. Although MRPP

provides a stronger emphasis on maintaining connectivity and coordinated motion, OCA achieves a more balanced trade-off between safety, coordination, and rapid task completion in dense environments.

Computational Complexity and Processing Overhead: Computation time represents the processing overhead required to generate collision-free paths before robot motion begins. The comparative analysis indicated that MRPP exhibited the highest computation time because of its continuous connectivity maintenance and coordination constraints across multiple robots. By simplifying graph exploration through centralised optimisation, CA reduced computational overhead by approximately 30.2% compared with MRPP. OCA achieved the lowest computation time overall, reducing computation time by approximately 38.7% relative to MRPP and improving upon CA by approximately 12.1%. The improved computational efficiency of OCA was primarily attributed to its obstacle-aware optimisation strategy and reduced redundant graph exploration, enabling more efficient path generation while preserving coordination and safety constraints. Overall, the results demonstrate that MRPP prioritises coordination robustness and connectivity preservation, whereas CA improves computational efficiency through centralised optimisation. OCA provides the best overall balance between path optimality, arrival efficiency, safety awareness, and computational performance, making it the most suitable approach among the evaluated methods for complex and obstacle-rich multi-robot environments.

Building upon these quantitative and structural insights, the succeeding sections provide a high-level systemic analysis evaluating how MRPP, CA, and OCA scale as fleet dimensions expand, how they guarantee safety bounds under uncertainty, and how they perform when translated into real-world industrial environments.

6. Review and Analysis of Graph-Based Multi-Robot Path Planning Algorithms: MRPP, CA, and OCA

The MRPP, CA, and OCA algorithms represent three progressively refined graph-based approaches for multi-robot path planning in structured environments. All three methods are fundamentally based on visibility graph (VG) representations and utilise Dijkstra's shortest path algorithm to compute collision-free trajectories. However, they differ in planning architecture, optimisation strategy, computational complexity, connectivity preservation, and safety performance [1,5,25,26]. MRPP adopts a centralised sequential planning architecture in which robots are prioritised and planned individually. A distinguishing characteristic of MRPP is the incorporation of algebraic connectivity (λ_2) as a coordination-aware metric to preserve communication and maintain network connectivity during navigation [1,7,25]. This enables robust cooperative behaviour and coordinated movement in multi-robot environments. Nevertheless, because MRPP relies on a complete visibility graph representation, the algorithm incurs relatively high computational overhead, particularly in dense obstacle environments or large-scale robot teams.

CA improves computational efficiency by introducing a reduced visibility graph constructed around a central baseline trajectory [1,26]. By reducing the search space, CA decreases processing complexity while maintaining near-optimal path quality. Compared with MRPP, CA achieves shorter trajectories and faster execution times; however, this improvement is obtained at the expense of reduced obstacle clearance and lower connectivity awareness.

OCA further extends CA by integrating obstacle-aware optimisation through configuration-space obstacle expansion. This mechanism introduces explicit safety distance constraints that maintain predefined clearance margins between robots and surrounding obstacles. Consequently, OCA improves navigation robustness and enhances operational safety in cluttered environments while retaining high computational efficiency [1,26].

The quantitative evaluation presented in Section Quantitative Performance Evaluation provides numerical evidence supporting these observations. The path length analysis demonstrated that CA reduced the average path length by approximately 12.5% compared with MRPP, whereas OCA achieved a greater reduction of approximately 15.1%. In addition, OCA improved path efficiency by approximately 2.9% relative to CA. Similar trends were observed in the arrival time analysis because robot velocity was assumed constant ($S = 1$ unit/s). Consequently, both CA and OCA achieved faster task completion compared with MRPP.

The computation time analysis further highlighted the structural differences between the algorithms. MRPP exhibited the highest computational overhead because of its connectivity-preserving coordination mechanisms and extensive graph exploration. CA reduced computation time by approximately 30.2% compared with MRPP through centralised optimisation and reduced graph complexity. OCA achieved the best computational performance overall, reducing computation time by approximately 38.7% relative to MRPP and approximately 12.1% relative to CA. These improvements were achieved through obstacle-aware optimisation and reduced redundant graph exploration.

The comparative synthesis presented earlier further demonstrates that the three algorithms prioritise different operational objectives. MRPP emphasises connectivity preservation, coordinated navigation, and robustness, making it suitable for communication-critical applications. CA provides a strong balance between path efficiency and scalability through simplified centralised planning. OCA achieves the best overall balance between path optimality, computational efficiency, safety awareness, and mission responsiveness, making it particularly suitable for complex and obstacle-rich operational environments.

Overall, the reviewed algorithms illustrate the fundamental trade-offs between coordination robustness, computational overhead, scalability, and safety in graph-based multi-robot path planning. Although no single approach can be considered universally optimal, each algorithm offers complementary strengths depending on application requirements and environmental constraints. The structured graph-theoretic framework adopted by these methods provides mathematically interpretable and scalable solutions for coordinated multi-robot navigation in static environments.

Despite these advantages, important challenges remain. The reviewed approaches primarily rely on offline or semi-static planning assumptions, which limit adaptability in highly dynamic and uncertain environments. Future research should therefore focus on integrating graph-based global planning with real-time reactive navigation, decentralised coordination, artificial intelligence techniques, and adaptive optimisation frameworks capable of handling dynamic obstacles, communication uncertainty, and large-scale heterogeneous robot teams.

Table 5 summarises the principal differences between MRPP, CA, and OCA in terms of planning structure, connectivity awareness, efficiency, scalability, and safety performance.

Table 5. Overview of key differences between MRPP, CA, and OCA algorithms.

Criteria	MRPP	CA	OCA
Base Method	VG + Dijkstra + λ_2	CB + reduced VG + Dijkstra	VG+ obstacle expansion + Dijkstra
Collision Avoidance	Good (λ_2 sequence)	Good (waypoints sequence)	Excellent
Connectivity Awareness	Yes	No	No
Efficiency	Moderate	High	High

Path Optimality	High (geometric)	High (coordinated))	Maximum (shortest path)
Safety Margin	Low	Medium	High
Scalability	Medium	High	High

6.1. Adaptability to Dynamic and Uncertain Environments

Despite their strengths, MRPP, CA, and OCA are primarily offline planners. Their effectiveness is predicated on the assumption of complete environmental transparency.

Roadmap rigidity: The reliance on precomputed VGs makes these algorithms sensitive to moving obstacles. Any change in the environment requires a partial or full reconstruction of the graph, leading to high computational overhead and lagged responsiveness.

Coordination complexity: While λ_2 is effective for sequencing in static settings, maintaining connectivity in dynamic scenarios requires real-time re-evaluation of the communication graph, which is mathematically intensive.

Mechanism invalidation: In rapidly changing environments, the central baseline of CA and the obstacle expansion of OCA may become invalid. Frequent updates to these structures often negate the efficiency gains they were designed to provide [1,6,12,15].

Comparative Analysis of Path Planning Methodologies

Beyond graph-based roadmaps, alternative paradigms such as conflict-based search (CBS) and priority-based planning (PBP) provide optimal or scalable solutions for multi-agent path finding (MAPF) problems [29,41]. For high-dimensional or continuous spaces, sampling-based methods such as probabilistic roadmaps (PRM*) and rapidly exploring random trees (RRT*) offer asymptotic optimality and flexibility in complex environments [49,57]. More recently, learning-based approaches, including multi-agent reinforcement learning (MARL) and graph neural networks (GNNs), have demonstrated strong performance in dynamic and uncertain environments due to their adaptability and data-driven decision-making capabilities [10,82,83].

Despite their versatility, these methods often involve trade-offs between optimality, computational efficiency, and theoretical guarantees. For instance, learning-based and heuristic-driven approaches may sacrifice global optimality and lack formal guarantees on safety, completeness, or coordination consistency [18,31]. Additionally, many of these methods introduce increased computational complexity or reduced interpretability, particularly in large-scale multi-robot systems [33].

In contrast, graph-based roadmap approaches provide a structured and deterministic framework that enables globally optimal path computation and clear coordination mechanisms. Recent research has increasingly focused on coordination, communication efficiency, and safety-aware planning in multi-robot systems, reinforcing the importance of connectivity-aware metrics such as λ_2 [2,6,7,24,33,36]. These developments align with the design principles of MRPP, CA, and OCA. This study focuses on such methods because they prioritise optimality, computational tractability, and coordination accuracy over reactive adaptability [1,25]. While the above analysis focuses on algorithmic design and performance, it is equally important to understand how these methods can be supported by modern technological frameworks and implemented in practical systems. These issues are summarised in Table 6.

Table 6. Comparative analysis of multi-robot path planning method classes and their key characteristics. RRT* and PRM* denote the asymptotically optimal extensions of the original RRT (rapidly exploring random tree) and PRM probabilistic roadmap) algorithms, respectively. The asterisk (*)

indicates asymptotic optimality, meaning that the solution converges toward the optimal path as the number of samples increases.

Method Class	Example Algorithms	Strengths	Limitations
Graph-based	MRPP, CA, OCA	Structured, optimal paths	Limited in dynamic environments
MAPF	CBS, PBP	Optimal, conflict resolution	High computational cost
Sampling-based	RRT*, PRM*	Good for continuous spaces	Not ideal for multi-agent coordination
Learning-based	MARL, GNN	Adaptive, handles uncertainty	Training complexity, lack of guarantees

6.2. Technological Context and Implementation Considerations

In addition to algorithmic analysis, understanding the technological context and implementation platforms is essential for evaluating the practical applicability of multi-robot path planning methods. It should be noted that this section provides a conceptual discussion of implementation considerations, rather than reporting real-world experimental deployment [75–77]. MRPP is typically associated with simulation platforms such as ROS, Gazebo, and Webots, which enable controlled testing and validation prior to potential real-world deployment. CA is conceptually linked to cloud robotics or digital twin platforms that can support applications such as smart factories, logistics systems, and coordinated warehouse operations [78–80]. OCA is considered suitable for large-scale autonomous fleets operating in dynamic or cluttered environments, where enhanced robustness, safety, and operational efficiency are required [81–83].

Figure 13 provides a structured visualisation of the technological context and implementation considerations for these algorithms across multiple layers, including algorithms, key technologies and methods, implementation platforms, and applications.

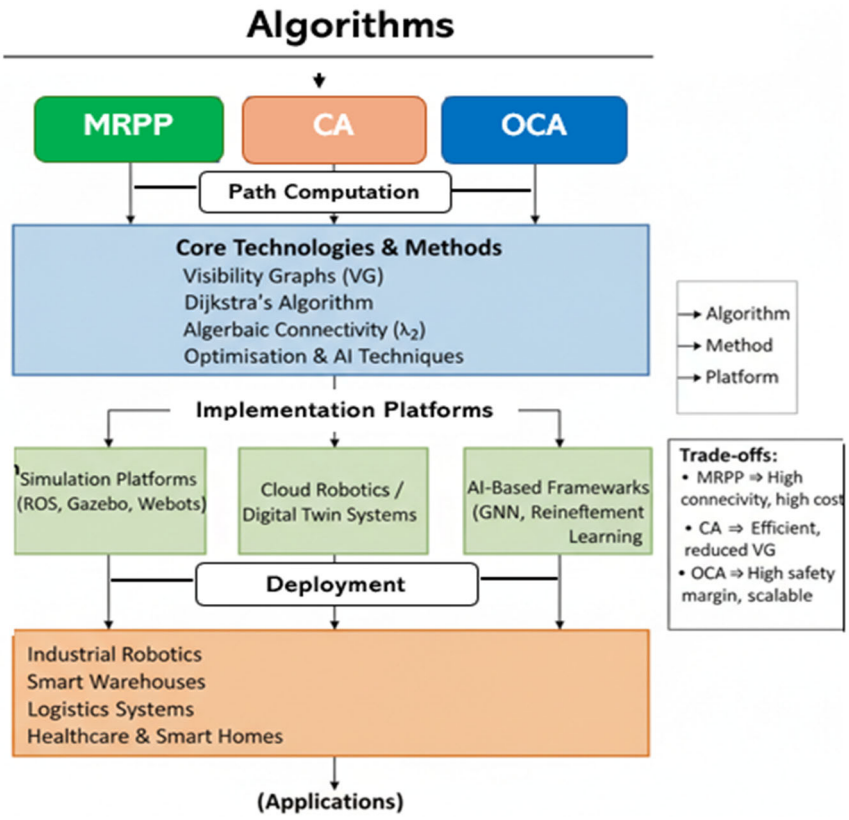

Figure 13. Structured framework illustrating the relationship between MRPP, CA, and OCA algorithms, their underlying graph-based methods, implementation platforms, and application domains.

Figure 13 highlights the interaction between the algorithm's design, enabling technologies, and deployment environments, including emerging applications such as assistive robotics. As illustrated, MRPP, CA, and OCA are connected to core technologies such as visibility graphs, Dijkstra's algorithm, algebraic connectivity (λ_2), and optimisation and artificial intelligence (AI) techniques that form the foundation for path computation. These methods are conceptually mapped to implementation platforms, including simulation environments, cloud robotics and digital twin systems, and AI-based frameworks [75,78,81]. In this context, MRPP emphasises connectivity maintenance within simulation-based environments, CA balances computational efficiency with centralised coordination via cloud or digital twin platforms, and OCA leverages AI and optimisation frameworks to support scalable and robust deployment [75,78,81,82]. This structured representation underscores the importance of aligning algorithm selection with application requirements, operational constraints, and deployment scenarios in multi-robot path planning. By situating MRPP, CA, and OCA within their technological and implementation contexts, researchers and practitioners can more effectively evaluate deployment strategies, identify computational bottlenecks, and guide future developments toward practical applications [81–83].

These considerations highlight the importance of selecting appropriate technological frameworks to ensure that path planning algorithms can effectively meet real-world requirements in terms of scalability, safety, and coordination. Furthermore, Table 7 summarises the key characteristics of MRPP, CA, and OCA, including their core technologies, implementation platforms, and primary advantages in multi-robot path planning.

Table 7. Comparisons of MRPP, CA, and OCA in terms of core technologies, implementation platforms, and key advantages.

Algorithm	Core Technology/Method	Implementation Platform (Conceptual)	Key Role/Advantage
MRPP	Visibility graphs (VG); Dijkstra's algorithm; graph-based path planning	ROS (Robot Operating System); Gazebo and Webots simulation platforms; industrial autonomous mobile robot environments	Computes optimal paths for multiple robots; enables basic coordination in structured environments; supports collision avoidance through path reservation
CA	Centralised coordination; shortest path algorithms (Dijkstra); algebraic connectivity (λ_2) for sequencing and ordering	Cloud robotics frameworks; digital twin platforms; warehouse robot fleet management systems	Ensures global coordination; improves system-wide efficiency; suitable for moderate-scale robot teams with centralised control
OCA	Centralised optimisation; multi-objective optimisation (e.g., task efficiency and safety); λ_2-based eigenvalue sequencing;	ROS integrated with cloud/offboard computation; graph neural network (GNN) or reinforcement learning (RL)-based simulation; smart	Enhances scalability and safety compared to CA; reduces collisions and congestion; optimises performance metrics such as time, energy, and task completion

	integration with AI/heuristics	factory and logistics platforms

In summary, the applicability of MRPP, CA, and OCA is closely associated with the availability of appropriate technological frameworks, including simulation environments, cloud robotics, digital twin systems, and AI-based platforms. These technologies play a critical role in enabling the transition from theoretical models to practical deployment. Future research should therefore focus not only on algorithmic improvements but also on the integration of these methods within suitable implementation infrastructures to enhance real-time performance, scalability, and system reliability [10,16,18,40,85].

6.3. Robotics and Assistive Technology Applications

Building upon the technological context discussed in Section 6.1, these implementation frameworks enable the deployment of multi-robot systems in various real-world domains. An important and emerging application area is assistive robotics, where safe and reliable navigation is critical. In addition to industrial and logistics domains, multi-robot path planning plays an increasingly important role in assistive technology domains. Assistive robotics focus on systems that support individuals with disabilities, elderly users, or patients in healthcare environments, improving independence and quality of life. Mobile assistive robots are widely used for tasks such as navigation assistance, object delivery, and daily activity support within indoor environments. For example, service robots and intelligent wheelchairs rely on autonomous navigation and obstacle avoidance to safely interact with users and operate in human-centred spaces. These systems require robust path planning to ensure safety, reliability, and efficiency in dynamic and uncertain environments.

Assistive robots can operate in healthcare and domestic settings to perform tasks such as medication delivery, patient monitoring, and mobility assistance. In such applications, path planning algorithms must account for human presence, unpredictable obstacles, and strict safety constraints. Techniques based on graph representations and shortest path computation provide structured and interpretable solutions, making them suitable for safety-critical environments. Although most current assistive systems rely on single-robot architectures, the integration of multi-robot systems presents promising opportunities. Examples include coordinated robotic assistants in hospitals, distributed service robots in smart homes, and collaborative rehabilitation systems. In this context, the algorithms reviewed in this paper (MRPP, CA, and OCA) can be conceptually extended to support coordinated navigation, task allocation, and collision avoidance among multiple assistive agents.

Overall, the applications of multi-robot path planning in assistive technologies highlights the importance of safety, adaptability, and human-aware navigation, reinforcing the need for scalable and robust planning strategies in real-world environments.

6.4. Limitations of the Study

This study has some limitations.

- The study focused only on graph-based visibility graph methods (MRPP, CA, OCA), so other approaches like RRT* and learning-based methods were not quantitatively evaluated.
- The results were based on controlled simulations and did not fully reflect real-world uncertainties such as noise, delays, or dynamic obstacles.
- All methods assumed offline planning with full environment knowledge, limiting real-time adaptability.

- No formal statistical testing was performed, so results showed trends but not statistical significance.
- The implementation remained theoretical, without real-world or hardware validation.
- The comparison framework was partly shaped by the selected algorithms, which may introduce bias toward graph-based centralised methods.
- While these limitations exist, the proposed algorithms remain promising, demonstrating consistent performance improvements and strong potential for efficient multi-robot path planning within the visibility graph framework.

7. Conclusions and Future Research Directions

This review presented a structured analysis of graph-based multi-robot path planning methods, with a particular focus on MRPP, CA, and OCA. By examining their planning architectures, optimisation criteria, and performance trade-offs, the study demonstrated how VG representations, combined with classical graph search techniques and connectivity-aware metrics, enable effective coordination, computational efficiency, and collision avoidance in structured environments [1,25]. While MRPP prioritises connectivity preservation through λ_2-based coordination, CA and OCA progressively improve computational efficiency and safety, reflecting an evolution toward more scalable and robust planning strategies.

Despite the strong theoretical foundation provided by visibility graphs, their practical effectiveness remains dependent on specific optimisation objectives and system constraints. Key challenges persist in scalability, particularly in large-scale multi-robot systems, and in adaptability to dynamic and uncertain environments. Addressing these limitations requires the development of planning frameworks capable of producing near-optimal solutions with reduced computational overhead while maintaining coordination and safety guarantees [2,9,33].

Recent research trends indicate a growing shift toward hybrid and intelligent navigation frameworks that combine global path planning with local reactive control, enabling improved responsiveness to dynamic obstacles and environmental uncertainty [10,16,36]. In addition, emerging approaches based on learning and distributed optimisation are increasingly being explored to enhance adaptability, scalability, and communication efficiency in multi-robot systems [7,18,24].

Future research should therefore focus on integrating graph-based planning with adaptive and data-driven methods, improving real-time performance, and ensuring robustness under communication constraints. Furthermore, the incorporation of modern technological frameworks, such as cloud robotics, digital twins, and advanced simulation platforms, offers promising opportunities for validating and deploying these algorithms in real-world applications.

Overall, achieving reliable and scalable multi-robot path planning requires a holistic approach that aligns algorithmic design, enabling technologies, and application-specific requirements. This integrated perspective is essential for advancing the practical deployment of multi-robot systems across diverse and complex operational domains.

Author Contributions: Conceptualisation, F.A.S.A. and R.S.; methodology, F.A.S.A. and R.S.; investigation, F.A.S.A. and R.S.; resources, F.A.S.A. and R.S.; writing—original draft preparation, F.A.S.A. and R.S.; writing—review and editing, F.A.S.A. and R.S.; visualisation, F.A.S.A. and R.S.; supervision, R.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new data were created.

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

Abbreviations

The following abbreviations are used in this manuscript:

CA	Central algorithm
CB	Central baseline
MRPP	Multi-robot path planning algorithm
MRS	Multi-robotic systems
OCA	Optimisation central algorithm
PL	Path length
PRM	Probabilistic road map
RM	Roadmap
RRT	Rapidly exploring random tree
UGI	User graphic interface
VD	Voronoi diagram
VG	Visibility graph

References

1. Alwafi, F.A.S. Social Graphs and Their Applications to Robotics. Ph.D. Thesis, Sheffield Hallam University, Sheffield, UK, 2022. Available online: https://shura.shu.ac.uk/31906/ (accessed on 30 March 2026).
2. 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.
3. Zhao, L.; Chen, B.; Hu, F. Research on Swarm Control Based on Complementary Collaboration of Unmanned Aerial Vehicle Swarms Under Complex Conditions. *Drones* **2025**, *9*, 119. https://doi.org/10.3390/drones9020119.
4. Suresh, A.; Laux, M.; Brinkmann, W.; Danter, L.C.; Kirchner, F. Enhancing Heterogeneous Multi-Robot Teaming for Planetary Exploration. *Eng. Proc.* **2025**, *90*, 112. https://doi.org/10.3390/engproc2025090112
5. Li, W.; Wang, Z.; Mai, R.; Ren, P.; Zhang, Q.; Zhou, Y.; Xu, N.; Zhuang, J.; Xin, B.; Gao, L.; et al. Modular Design Automation of the Morphologies, Controllers, and Vision Systems for Intelligent Robots: A Survey. *Vis. Intell.* **2023**, *1*, 1–23. Available online: https://link.springer.com/content/pdf/10.1007/s44267-023-00006-x.pdf (accessed on 28 May 2026).
6. Tang, W.; Li, C.; Wu, J.; Zhu, Q. Decentralized Communication-Maintained Coordination for Multi-Robot Exploration: Achieving Connectivity and Adaptability. In proceedings of the 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 25 2024, Abu Dhabi, United Arab Emirates, 14–18 October 2024; pp. 3417–3424. Available online: https://ieeexplore.ieee.org/abstract/document/10802832 (accessed on 28 May 2026).
7. Wang, S.; Li, S.; Huangfu, Y.; Qiu, Y.; Liu, Y. Communication-Efficient Decentralized Task Allocation for Large-Scale Multi-Agent Systems. *IEEE Robot. Autom. Lett.* **2025**, *10*, 10074–10081. https://doi.org/10.1109/LRA.2025.3597900.
8. Zhu, L.; Huang, Y.; Li, Y.; Cai, H.; Zhao, W.; Liu, X.; Guo, Y. Lipschitz-Nonlinear Heterogeneous Multi-Agent Adaptive Distributed Time-Varying Formation-Tracking Control with Jointly Connected Topology. *Entropy* **2025**, *27*, 648. https://doi.org/10.3390/e27060648.
9. Banik, S.; Banik, S.C.; Mahmud, S.S. Path Planning Approaches in Multi-Robot System: A Review. *Eng. Rep.* **2025**, *7*, e13035. Available online: https://onlinelibrary.wiley.com/doi/pdf/10.1002/eng2.13035 (accessed on 28 May 2026).
10. Peng, S. Multi-robot path planning combining heuristics and multi-agent reinforcement learning. *arXiv* **2025,** arXiv:2306.01270v2. https://arxiv.org/pdf/2306.01270.
11. Tafrishi, S.A.; Svinin, M.; Tahara, K. A Survey on Path Planning Problem of Rolling Contacts: Approaches, Applications and Future Challenges. *arXiv*, **2025**, arXiv:2501.04442. https://arxiv.org/abs/2501.04442.
12. Wen, G.; Duan, Z.; Yu, W.; Chen, G. Consensus in Multi-Agent Systems with Communication Constraints. *Int. J. Robust Nonlinear Control.* **2012**, *22*, 170–182. Available online: https://www.academia.edu/download/45150275/Consensus_in_multiagent_systems_with_com20160427-360-metxht.pdf (accessed on 28 May 2026).

13. Jathunga, T.; Rajapaksha, S. Improved Path Planning for Multi-Robot Systems Using a Hybrid Probabilistic Roadmap and Genetic Algorithm Approach. *J. Robot. Control.* **2025**, *6*, 715–733.
14. Zhou, M.; Li, J.; Wang, C.; Wang, J.; Wang, L. Applications of Voronoi Diagrams in Multi-Robot Coverage: A review. *J. Mar. Sci. Eng.* **2024**, *12*, 1022. https://doi.org/10.3390/jmse12061022.
15. Chaabi, H. Distributed Multi-Robot Exploration with Connectivity Maintenance Under QoS Constraints. Ph.D. Thesis, Université de Lille, Lille, France, 2025. Available online: https://hal.science/tel-05379346/document (accessed on 28 May 2026).
16. Shaoul, Y.; Mishani, I.; Vats, S.; Li, J.; Likhachev, M. Multi-Robot Motion Planning with Diffusion Models. *arXiv* **2025**, arXiv:2410.03072v2. https://arxiv.org/pdf/2410.03072.
17. Lee, S.H.; Jo, W.; Robert, L.P.; Tilbury, D.M. Local Minima Prediction using Dynamic Bayesian Filtering for UGV Navigation in Unstructured Environments. *arXiv* **2025**, arXiv:2505.14337v1. https://arxiv.org/pdf/2505.14337.
18. Shorinwa, O.; Halsted, T.; Yu, J.; Schwager, M. Distributed Optimization Methods for Multi-Robot Systems: Part i—A Tutorial. *arXiv* **2024**, arXiv:2301.11313v3. https://arxiv.org/pdf/2301.11313.
19. Fiedler, M. Algebraic Connectivity of Graphs. *Czechoslov. Math. J.* **1973**, *23*, 298–305. https://eudml.org/doc/12723.
20. Pasham, S.D. Using Graph Theory to Improve Communication Protocols in AI-Powered IoT Networks. *Metascience (Int. Multidiscip. Res. J.)* **2024**, *2*, 17–48. Available online: http://yuktabpublisher.com/index.php/TMS/article/download/129/117 (accessed on 28 May 2026).
21. Lozano-Pérez, T.; Wesley, M.A. An Algorithm for Planning Collision-Free Paths Among Polyhedral Obstacles. *Commun. ACM* **1979**, *22*, 560–570. https://dl.acm.org/doi/10.1145/359156.359164.
22. Takeno, J. Visibility graphs. In *Knowledge and References*; Taylor & Francis Group (Informa Business): Abingdon, UK, 2022. Available online: https://taylorandfrancis.com/knowledge/Engineering_and_technology/Systems_%26_control_engineering/Visibility_graphs/ (accessed 31 March 2026).
23. Jensen, J.H.; Sørensen, K.P.B.; le Fevre Sejersen, J.; Sarabakha, A. Multi-Agent Path Planning in Complex Environments using Gaussian Belief Propagation with Global Path Finding. *arXiv* **2025**, arXiv:2502.20369v1. https://arxiv.org/pdf/2502.20369.
24. Li, J.; Wang, C.; Li, B.; Wei, Q.; Song, L.; Huang, D. Maintaining Connectivity in Coverage Control: A Distributed Algebraic Connectivity Estimation Approach Using Control Barrier Functions. *Aerosp. Syst.* **2025**, *8*, 1–11. Available online: https://link.springer.com/article/10.1007/s42401-025-00424-3 (accessed on 28 May 2026).
25. Alwafi, F.A.S.; Xu, X.; Saatchi, R.; Alboul, L. Development and Evaluation of a Multi-Robot Path Planning Graph Algorithm. *Information* **2025**, *16*, 431. https://doi.org/10.3390/info16060431.
26. Alwafi, F.; Xu, X.; Alboul, L.; Penders, J. Visibility Graph (VG)-Based Path Planning Method for Multi-Robot Systems. In UKRAS22 Conference Robotics for Unconstrained Environment, University of the West of England (UWE Bristol), Bristol, UK, 11 November 2022; pp. 60–61. Available online: https://api.semanticscholar.org/CorpusID:253560451 (accessed on 28 May 2026).
27. Bai, R.; Yuan, S.; Li, K.; Guo, H.; Yau, W.Y.; Xie, L. Realm: Real-Time Line-Of-Sight Maintenance in Multi-Robot Navigation with Unknown Obstacles. *arXiv*, **2025**, arXiv:2502.15162. https://arxiv.org/abs/2502.15162.
28. Griparić, K. Algebraic Connectivity Control in Distributed Networks Using Multiple Communication Channels. *Sensors* **2021**, *21*, 5014. https://doi.org/10.3390/s21155014.
29. Sharon, G.; Stern, R.; Felner, A.; Sturtevant, N.R. Conflict-Based Search for Optimal Multi-Agent Path Finding. *Artif. Intell.* **2015**, *219*, 40–66. https://doi.org/10.1016/j.artint.2014.11.006.
30. Chitikena, H.; Sanfilippo, F.; Ma, S. Robotics in Search and Rescue (SAR). Applied. *Sciences* **2023**, *13*, 1800. https://doi.org/10.3390/app13031800.
31. Bui, H.D. A Survey of Multi-Robot Motion Planning. *arXiv* **2023**, arXiv:2310.08599. https://arxiv.org/abs/2310.08599.
32. Barth, A.; Ma, O. Cooperative Behavior of a Heterogeneous Robot Team for Planetary Exploration Using Deep Reinforcement Learning. *Acta Astronaut.* **2024**, *214*, 689–700. https://doi.org/10.1016/j.actaastro.2023.11.014.
33. Matos, D.M.; Costa, P.; Sobreira, H.; Valente, A.; Lima, J. Efficient Multi-Robot Path Planning in Real Environments: A Centralized Coordination System *Int. J. Intell. Robot. Appl.* **2025**, *9*, 217–244. https://doi.org/10.1007/s41315-024-00378-3.
34. Capelli, B.; Fouad, H.; Beltrame, G.; Sabattini, L. Decentralised Connectivity Maintenance with Time Delays Using Control Barrier Functions. 2021 IEEE International Conference on Robotics and Automation (ICRA), Conference Location: Xi'an, China (30 May 2021–5 June 2021). Available online: https://ieeexplore.ieee.org/abstract/document/9561066 (accessed on 28 May 2026).
35. Woosley, B.; Dasgupta, P.; Rogers, J.G.; Twigg, J. Multi-Robot Information Driven Path Planning Under Communication Constraints. *Auton. Robot.* **2019**, *44*, 721–737.
36. Zhou, C.; Li, J.; Shi, M.; Wu, T. Multi-Robot Path Planning Algorithm for Collaborative Mapping Under Communication Constraints. *Drones* **2024**, *8*, 493. https://doi.org/10.3390/drones8090493.

37. Ma, H. Graph-Based Multi-Robot Path Finding and Planning. *Spring Nat. Curr. Robot. Rep.* **2022**, *3*, 77–84. https://doi.org/10.1007/s43154-022-00083-8.
38. Al-Kamil, S.J.; Szabolcsi, R. Optimizing Path Planning in Mobile Robot Systems Using Motion Capture Technology. *Results Eng.* **2024**, *22*, 102043. https://doi.org/10.1016/j.rineng.2024.102043.
39. Hvězda, J. Comparison of Path Planning Methods for a Multi-Robot Team; Engineering Computer Science. Master's Thesis, Czech Technical University in Prague, Prague, Czech Republic, 2017. Available online: https://www.semanticscholar.org/paper/Comparison-of-path-planning-methods-for-a-team-Hv%C4%9Bzda/38fba26a68e234c40df6f9148abb9c1e450195b1 (accessed on 28 May 2026).
40. Zhang, H.Y.; Lin, W.M.; Chen, A.X. Path Planning for the Mobile Robot: A Review. *Symmetry* **2018**, *10*, 450. https://doi.org/10.3390/sym10100450.
41. Yu, J.; LaValle, S.M. Optimal Multi-Robot Path Planning on Graphs: Complete Algorithms and Effective Heuristics. *IEEE Trans. Robot.* **2016**, *32*, 1163–1177.
42. Yu, J. Intractability of Optimal Multi-Robot Path Planning on Planar Graphs. *IEEE Robot. Autom. Lett.* **2015**, *1*, 33–40. Available online: https://ieeexplore.ieee.org/abstract/document/7342901 (accessed on 28 May 2026).
43. Sabattini, L.; Chopra, N.; Secchi, C. Decentralized Connectivity Maintenance for Cooperative Control of Mobile Robotic Systems. *Int. J. Robot. Res.* **2013**, *32*, 1411–1423. https://doi.org/10.1177/0278364913499085.
44. Campbell, S.; O'Mahony, N.; Carvalho, A.; Krpalkova, L.; Riordan, D.; Walsh, J. Path Planning Techniques for Mobile Robots A Review. In Proceedings 2020 6th International Conference on Mechatronics and Robotics Engineering (ICMRE), Barcelona, Spain, 12–15 February 2020. Available online: https://ieeexplore.ieee.org/abstract/document/9065187 (accessed on 28 May 2026).
45. Malli, I.; Bechlioulis, C.P.; Kyriakopoulos, K.J. Robust Distributed Estimation of the Algebraic Connectivity for Networked Multi-Robot Systems. In Proceedings 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May 2021–5 June 2021; IEEE: Piscataway, NJ, USA, 2021. Available online: https://ieeexplore.ieee.org/document/9561201 (accessed on 28 May 2026).
46. Murayama, T. Dynamical Improvement of Network Connectivity for Multi-Robot Robustness Using Perturbed Algebraic Connectivity. In Proceedings 2021 IEEE/SICE International Symposium on System Integration (SII), Fukushima, Japan, 11–14 January 2021; IEEE: Piscataway, NJ, USA, 2021. Available online: https://ieeexplore.ieee.org/document/9382705 (accessed on 28 May 2026).
47. Murayama, T. Distributed Control for Bi-Connectivity of Multi-Robot Network. *SICE J. Control. Meas. Syst. Integration*. **2023**, *16*, 1–10. Available online: https://www.tandfonline.com/doi/full/10.1080/18824889.2022.2157194 (accessed on 28 May 2026).
48. Defoort, M.; Veluvolu, K.C. A Motion Planning Framework with Connectivity Management for Multiple Cooperative Robots. *J. Intell. Robot. Syst.* **2013**, *75*, 343–357. https://doi.org/10.1007/s10846-013-9872-0.
49. Ichter, B.; Schmerling, E.; Lee, T.W.E.; Faust, A Learned Critical Probabilistic Roadmaps for Robotic Motion Planning. In Proceedings 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May 2020–31 August 2020; IEEE: Piscataway, NJ, USA, 2020. Available online: https://ieeexplore.ieee.org/abstract/document/9197106 (accessed on 28 May 2026).
50. Niu, H.; Savvaris, A.; Tsourdos, A.; Ji, Z. Voronoi-Visibility Roadmap-Based Path Planning Algorithm for Unmanned Surface Vehicles. *J. Navig.* **2019**, *72*, 850–874. https://doi.org/10.1017/S0373463318001005.
51. Lee, W.; Choi, G.H.; Kim, T.W. Visibility Graph-Based Path-Planning Algorithm with Quadtree Representation. *Appl. Ocean. Res.* **2021**, *117*, 102887. Available online: https://www.sciencedirect.com/science/article/pii/S0141118721003588?via%3Dihub (accessed on 28 May 2026).
52. Ma, Y.; Mao, Z.; Wang, T.; Qin, J.; Ding, W.; Meng, X. Obstacle Avoidance Path Planning of Unmanned Submarine Vehicle Using Improved Firework–Ant Colony Algorithm. *Comput. Electrical. Eng.* **2020**, *87*, 106773. https://doi.org/10.1016/j.compeleceng.2020.106773.
53. Xu, K.; Xing, J.; Sun, W.; Xu, P.; Yang, R. Multi-Robot Collision Avoidance Method in Sweet Potato Fields. *Front. Plant Sci.* **2024**, *15*, 1393541. https://doi.org/10.3389/fpls.2024.1393541.
54. Solis, I.; Motes, J.; Sandström, R.; Amato, N.M. Representation-Optimal Multi-Robot Motion Planning Using Conflict-Based Search. *IEEE Robot. Autom. Lett.* **2021**, *6*, 4608–4615. Available online: https://ieeexplore.ieee.org/document/9387143 (accessed on 28 May 2026).
55. Luo, W.; Chakraborty, N.; Sycara, K. Minimally Disruptive Connectivity Enhancement for Resilient Multi-Robot Teams. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2020), Las Vegas, NV, USA,

24 October 2020–24 January 2021; pp. 11809–11816. Available online: https://www.cs.uic.edu/~wenhao/publications/IROS19_k_connectivity.pdf (accessed on 28 May 2026).
56. Liang, D.; Liu, Z.; Bhamra, R. Collaborative Multi-Robot Formation Control and Global Path Optimization. *Appl. Sci.* **2022**, *12*, 7046. https://doi.org/10.3390/app12147046.
57. Taghavifar, H.; Rakheja, S.; Reina, G. A Novel Optimal Path-Planning and Following Algorithm for Wheeled Robots on Deformable Terrains. *J. Terramechanics* **2020**, *96*, 147–157. https://doi.org/10.1016/j.jterra.2020.12.001.
58. 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. https://doi.org/10.25046/aj050330.
59. 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. https://doi.org/10.1016/j.dt.2019.04.011.
60. 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), 18–20 October 2015, Langkawi, Malaysia, 2015. Available online: https://ieeexplore.ieee.org/document/7451628 (accessed on 28 May 2026).
61. 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. https://doi.org/10.1080/02564602.2021.1894250.
62. Aggarwal, S.; Kumar, N. Path planning techniques for unmanned aerial vehicles: A review, solutions, and challenges. *Comput. Commun.* **2020**, *149*, 270–299. https://doi.org/10.1016/j.comcom.2019.10.014.
63. Liu, Z.; Zhang, Y.; Yu, X.; Yuan, C. Unmanned Surface Vehicles: An Overview of Developments and Challenges. *Annu. Rev. Control.* **2016**, *41*, 71–93. https://doi.org/10.1016/j.arcontrol.2016.04.018.
64. 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. https://doi.org/10.1007/s10846-021-01330-4.
65. Zafar, M.N.; Mohanta, J. Methodology for path planning and optimization of mobile robots: A review. *Procedia Comput. Sci.* **2018**, *133*, 141–152. https://doi.org/10.1016/j.procs.2018.07.018.
66. Verma, J.K.; Ranga, V. Multi-Robot Coordination Analysis, Taxonomy, Challenges and Future Scope. *J. Intell. Robot. Syst.* **2021**, *102*, 1–36. Available online: https://pmc.ncbi.nlm.nih.gov/articles/PMC8051283/ (accessed on 28 May 2026).
67. 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**, *102*, 17 Available online: https://link.springer.com/article/10.1007/s10846-021-01378-2 (accessed on 28 May 2026).
68. 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. https://doi.org/10.3390/app11209650.
69. 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**, *11*, 1–17. https://doi.org/10.1109/ACCESS.2023.3322230.
70. 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. https://doi.org/10.1109/LRA.2023.3254444.
71. 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. https://doi.org/10.1142/S0217984920501377.
72. 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. https://doi.org/10.3934/mbe.2022012.
73. 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. https://doi.org/10.1080/0952813X.2020.1764631.
74. Dijkstra, E.W. A Note on Two Problems in Connexion with Graphs. *Numer. Math.* **1959**, *1*, 269–271. https://doi.org/10.1007/BF01386390.
75. Quigley, M.; Gerkey, B.; Conley, K.; Faust, J.; Foote, T.; Leibs, J.; Berger, E.; Wheeler, R.; Ng, A. ROS: An Open-Source Robot Operating System. In Proceedings of the ICRA Workshop on Open Source Software, Kobe, Japan, 12–17 May 2009; p. 5. Available online: https://ai.stanford.edu/~mquigley/papers/icra2009-ros.pdf?utm_source=chatgpt.com (accessed on 28 May 2026).
76. Koenig, N.; Howard, A. Design and Use Paradigms for Gazebo, an Open-Source Multi-Robot Simulator. In Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566), Sendai, Japan, 28 September–2 October 2004; IEEE: Piscataway, NJ, USA, 2004. Available online: https://ieeexplore.ieee.org/document/1389727 (accessed on 28 May 2026).

77. Michel, O. Webots. Professional Mobile Robot Simulation. *Int. J. Adv. Robot. Syst.* **2004**, *1*, 39–42. Available online: https://www.researchgate.net/publication/221702459_WebotsTM_Professional_Mobile_Robot_Simulation (accessed on 28 May 2026).
78. Kehoe, B.; Patil, S.; Abbeel, P.; Goldberg, K. A Survey of Research on Cloud Robotics and Automation. In Proceedings of the 2014 International Conference on Audio, Language and Image Processing, 7–9 July 2014, Shanghai, China, 2014. Available online: https://ieeexplore.ieee.org/document/7009926 (accessed on 28 May 2026).
79. Hoebert, T.; Lepuschitz, W.; List, E.; Merdan, M. Cloud-based Digital Twin for Industrial Robotics. In *International Conference on Industrial Applications of Holonic and Multi-Agent Systems*; Industrial Applications of Holonic and Multi-Agent Systems; Springer: Cham, Switzerland, 2019, pp. 105–116. Available online:. https://www.researchgate.net/publication/335233874_Cloud-Based_Digital_Twin_for_Industrial_Robotics (accessed on 28 May 2026).
80. Kaur, H.; Bhatia, M. Scientometric Analysis of Digital Twin in Industry 4.0. *IEEE Internet Things J.* **2024**, *12*, 1200–1221. Available online: https://ieeexplore.ieee.org/abstract/document/10680079/ (accessed on 28 May 2026).
81. Kober, J.; Bagnell, J.A.; Peters, J. Reinforcement Learning in Robotics: A Survey. *Int. J. Robot. Res.* **2013**, *32*, 1238–1274. Available online: https://journals.sagepub.com/doi/10.1177/0278364913495721 (accessed on 28 May 2026).
82. Zhou, J.; Cui, G.; Zhang, Z.; Yang, C.; Liu, Z.; Wang, L.; Li, C.; Sun, M. Graph Neural Networks: A Review of Methods and Applications. *AI Open* **2020**, *1*, 57–81. Available online: https://www.sciencedirect.com/science/article/pii/S2666651021000012 (accessed on 28 May 2026).
83. Nguyen, N.D.; Nguyen, T.T.; Nguyen, H.; Creighton, D.; Nahavandi, S.; Review, Analysis and Design of a Comprehensive Deep Reinforcement Learning Framework. *arXiv* **2021**, arXiv:2002.11883v2. https://arxiv.org/pdf/2002.11883.
84. Zhao, K.; Ning, L. Hybrid Navigation Method for Multiple Robots Facing Dynamic Obstacles. *Tsinghua Sci. Technol.* **2022**, *27*, 894–901. https://doi.org/10.26599/TST.2021.9010073.
85. 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. Available online: https://link.springer.com/article/10.1007/s11370-023-00460-y (accessed on 28 May 2026).
86. Eirale, A.; Martini, M.; Tagliavini, L.; Gandini, D.; Chiaberge, M.; Quaglia, G. Marvin: Marvin: An Innovative Omni-Directional Robotic Assistant for Domestic Environments. *Sensors* **2022**, *22*, 5261. https://doi.org/10.3390/s22145261.
87. Holland, J.; Kingston, L.; McCarthy, C.; Armstrong, E.; O'dwyer, P.; Merz, F.; McConnell, M. Service Robots in the Healthcare Sector. *Robotics* **2021**, *10*, 47. https://doi.org/10.3390/robotics10010047.
88. Zhang, Z.; Xu, P.; Wu, C.; Yu, H. Smart Nursing Wheelchairs: A New Trend in Assisted Care and the Future of Multifunctional Integration. *Biomimetics* **2024**, *9*, 492. https://doi.org/10.3390/biomimetics9080492.
89. Levine, S.; Finn, C.; Darrell, T.; Abbeel, P. End-to-End Training of Deep Visuomotor Policies. *J. Mach. Learn. Res.* **2016**, *17*, 1–40. Available online: https://www.jmlr.org/papers/volume17/15-522/15-522.pdf (accessed on 28 May 2026).
90. ownsend, D.; MajidiRad, A. Trust in Human-Robot Interaction Within Healthcare Services: A Review Study. In *Proceedings of the International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC/CIE 2022)*, St. Louis, MO, USA, 14 August 2022; American Society of Mechanical Engineers: New York, NY, USA, 2022; Volume 86281, p. V007T07A030. Available online: https://digitalcommons.odu.edu/cgi/viewcontent.cgi?article=1125&context=mae_fac_pubs (accessed on 28 May 2026).
91. Sodhi, G.K.; Kaur, S.; Gaba, G.S.; Kansal, L.; Sharma, A., Dhiman, G. COVID-19: Role of Robotics, Artificial Intelligence and Machine Learning During the Pandemic. *Curr. Med. Imaging* **2022**, *18*, 124–134. https://doi.org/10.2174/1573405617666210224115722.
92. Das, G.P., McGinnity, T.M., Coleman, S.A., Behera. L. A Distributed Task Allocation Algorithm for a Multi-Robot System in Healthcare Facilities. *J. Intell. Robot. Syst.* **2015**, *80*, 33–58. Available online: http://www.gauthampdas.com/wp-content/uploads/downloads/2015/01/2014_Springer_JIRS_Das_et_al.pdf (accessed on 28 May 2026).
93. Zhang, T.; Zhao, D.; Yang, J.; Wang, S.; Liu, H. A Smart Home Based on Multi-heterogeneous Robots and Sensor Networks for Elderly Care. In *Proceedings of the 15th International Conference on Intelligent Robotics and Applications (ICIRA 2022)*, Harbin, China, 1–3 August 2022; Part I; Springer: Cham, Switzerland, 2022; pp. 1–20. Available online: https://link.springer.com/chapter/10.1007/978-3-031-13844-7_10 (accessed on 28 May 2026).
94. Chen, X.; Wang, T.; Qiu, J.; Feng, J. Mission Planning on Autonomous Avoidance for Spacecraft Confronting Orbital Debris. *IEEE Trans. Aerosp. Electron. Syst.* **2024**, *61*, 4115–4126. Available online: https://ieeexplore.ieee.org/abstract/document/10750382/ (accessed on 28 May 2026).

95. Chen, X.; Qiu, J.; Wang, T.; Li, M. Onboard Mission Planning for Autonomous Avoidance of Spacecraft Subject to Various Orbital Threats: An SMT-Based Approach. *IEEE Trans. Aerosp. Electron. Syst.* **2024**, *61*, 4866–4878. Available online: https://ieeexplore.ieee.org/abstract/document/10771678/ (accessed on 28 May 2026).

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.