Expert Systems With Applications 173 (2021) 114660

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Review

Trajectory planning for multi-robot systems: Methods and applications

Ángel Madridano, Abdulla Al-Kaff, David Martín *, Arturo de la Escalera

Intelligent Systems Lab, Universidad Carlos III de Madrid, Calle Butarque 15, Leganés (28911), Madrid, Spain

A R T I C L E I N F O

Keywords:
Autonomous vehicles
Path planning
Multi-robot systems
UAVs
UGVs

2010 MSC:
00–01
99–00

A B S T R A C T

In the multiple fields covered by Artificial Intelligence (AI), path planning is undoubtedly one of the issues that cover a wide range of research lines. To be able to find an optimal solution, which allows one or several vehicles to establish a safe and effective way to reach a final state from an initial state, is a challenge that continues to be studied today. The increasingly widespread use of autonomous vehicles, both aerial and ground-based, make path planning an essential aspect for incorporating these systems into an endless number of applications. Besides, in recent years, the use of Multi-Robot Systems (MRS) has spread, consisting of both Unmanned Aerial Vehicles (UAVs) and Unmanned Ground Vehicles (UGVs), gaining versatility and robustness in their operation. The possibility of using heterogeneous robotic teams allows tackling, autonomously, and simultaneously, a wide range of tasks with different characteristics in the same environment. For this purpose, path planning becomes a crucial aspect and, for this reason, this work aims to offer a general vision of trajectory planning, to establish a comparison between the methods and algorithms present in the literature for the resolution of this problem within MRS, and finally, to show the applicability of these methods in different areas, together with the importance of these methods for achieving autonomous and safe navigation of different types of vehicles.

1. Introduction

In the last decade, the advances in the field of electronics have improved the characteristics of both Unmanned Aerial Vehicles (UAVs) and Unmanned Ground Vehicles (UGVs), as crucial as their autonomy or weight, along with a considerable reduction in their cost. This, coupled with the wider variety of vehicles available in both classes, has led to an exponential increase in the use of these vehicles in a wide range of tasks.

In addition, the implementation of systems made up of a set of vehicles, whether all aerial, ground-based, or both, is now spreading. The possibility of using a Multi-Robot System (MRS) autonomously and simultaneously, improves the efficiency of these vehicles in missions; where response time and accessibility to different areas are established as crucial factors for success; such as search and rescue missions, emergencies, monitoring large areas or tracking multiple targets.

A key aspect of achieving safe and coordinated navigation of a system consists of different vehicles is path planning. Therefore, there is a need to develop algorithms that can generate a solution to the problem of reaching, a specific location by a vehicle that moves freely in a given environment without human intervention. In addition, the vehicle must be able to reach that objective without colliding with obstacles present in the environment, and avoiding the rest of the vehicles that are part of the MRS. Not only to reach the goal safely is necessary, but also, when working with systems and environments where the distance traveled and the time spent are determining factors, it becomes essential that the obtained final solution to be optimal. For this reason, many of the path planning methods are based on different types of algorithms that allow, on the one hand, to find a set of safe paths to the goal, and on the other hand, to establish which of these possibilities is the most optimal in terms of aspects; such as the distance traveled, dynamics of the system, or time spent to travel it. Although the planning of trajectories presents difficulties when generating a solution, its application to MRS implies a series of aspects linked to the type of vehicles used.

Thus, in the case of UAVs, the most crucial feature from the planning point of view is the possibility of modifying their altitude; to avoid obstacles or restricted areas, i.e., the possibility of establishing a 3D path planning. Also, aspects such as the possibility of covering large areas in a short time, or their ease of access to remote places should be considered. On the other hand, UAVs present a series of restrictions to be considered during the task planning phase, such as the influence of weather conditions, particularly wind, which may cause UAVs to be unable to follow a particular path precisely, or to reach a specific position; or the

* Corresponding author.
E-mail addresses: amadrida@ing.uc3m.es (Á. Madridano), akaff@ing.uc3m.es (A. Al-Kaff), dmgomez@ing.uc3m.es (D. Martín), escalera@ing.uc3m.es (A. de la Escalera).

https://doi.org/10.1016/j.eswa.2021.114660
Received 29 April 2020; Received in revised form 26 January 2021; Accepted 26 January 2021
Available online 6 February 2021

'Downwash' effect, i.e., the impossibility of UAVs to fly near another UAV; because of the effect of air disturbances generated by the rotors of one UAV to dynamics of the rest of the UAVs.

In the case of the UGVs, their main disadvantage in terms of path planning is the restriction to 2D, which makes it impossible to reach certain places or access remote locations because they cannot cross or overcome certain obstacles. On the contrary, they have the advantage that, in the most cases, UGVs have greater autonomy than UAVs, being able to reach locations further away from the starting point, provided that they are accessible.

Therefore, the main objective of this work is to analyze, and compare the main methods and algorithms collected in the current literature; to carry out the path planning in MRS. Moreover, the document includes a study on the fields that use these systems, trying in this way to generate a complete work that allows to know the current state of the art of the techniques of trajectory planning in MRS, its applicability in different areas, and to establish conclusions about which of the methods are more propitious for each case, and how to face future developments in this field.

The structure of the document is organized as follows: Section 2 provides an introduction to the essential concepts in the field of MRS, helping to understand certain aspects of the path planning algorithms in this field, which are covered in Section 3. Then, Section 4 analyses the applicability of MRS to different fields. Finally, Section 5 collects the conclusions of the work on the applicability of the different path planning algorithms in the field of MRS, and the possible future steps to be undertaken in this area are detailed.

2. Multi-robot systems

Multi-Robot Systems includes all those groups formed by two or more robots sharing the same work space. This general concept covers industrial robotic arms, humanoid robots, ground and aerial mobile systems, and autonomous vehicles.

Although this work is exclusively oriented to MRS formed by autonomous mobile systems (UGVs and UAVs) either homogeneous of heterogeneous, some concepts related to the classification of MRS and their terminology are established.

As stated in Zakiev, Tsoy, and Magid (2018), in the field of robotics research there are multiple terms such as *Multi-Robot Systems*, *Multi-Agent Systems*, *Robotic Swarms* or *Sensor Networks* that are usually used interchangeably; to refer to groups formed by more than one robots working in a coordinated manner in the same area. Although this general concept is maintained in these subgroups, each term presents particular characteristics:

1. **Multi-Robot Systems:** Includes all systems containing multiple robots. Therefore, a robotic swarm is included within the MRS, but not all MRS are swarms.
2. **Multi-Agent Systems:** This term does not belong only to robotics, and in general, refers to a system composed of multiple intelligent agents capable of interacting with each other. For this reason, areas such as computer science, biology, psychology, or economics also present research in this field. If their study is focused on robotics, they are usually used to talk about ideal models of robotic limbs and are considered necessary when establishing theories whose viability is in doubt.
3. **Robotic Swarms:** This term includes some essential aspects that distinguish swarms; such as scalability, inter-robot communications, and the advantage of the whole over individuality. Although other studies; such as (Şahin, 2004), include aspects like the autonomy to relate to the environment or the homogeneity of all the elements that form the swarm, the first three are common to most studies in this field.
4. **Sensor Networks:** This term is related to a set of mobile sensors that interact with each other. Therefore, any of the groups described above could be in itself a sensor network.

Another important aspect within the MRS to be considered from a path planning point of view, is the control and decision-making architecture of the MRS. Two main types of architectures are established:

1. **Centralized Architecture:** it is characterized by having within the MRS a single element or node in charge of collecting all the information, processing it, and establishing the set of actions or decisions to perform (Jose & Pratihar, 2016, Yan, Jouandeau, & Cherif, 2010, 2012, 2011). Therefore, the main characteristic of these systems is the capacity to have a global vision of the whole system in a single agent, and be able to establish optimal global plans. This facilitates the decision-making process as there is only one agent, which with all the information, communicates to the different elements the steps to be carried out, having control of all the movements and decisions of the system.

   On the contrary, it presents disadvantages such as the limitations of the communication systems. The centralized architecture is characterized by an essential communication system, so that, the central agent receives information from the rest of the elements of the system and communicates the decisions or actions to be taken. Therefore, the radius of the use of MRS with centralized architecture is limited to the distance allowed by the communication systems. Furthermore, the bandwidths of the communications can be established as a disadvantage in the scalability of the MRS. A centralized architecture reduces the robustness of the MRS, since the achievement of a mission is conditioned to the maintenance of this central agent, any loss of it, leads to terminate the tasks of the rest of the elements of the MRS.
2. **Decentralized Architecture:** Unlike the previous one, there is no single agent or node in charge of controlling the whole MRS (Amato, Konidaris, How, & Kaelbling, 2014, 2015, 2015, 2017, 2018). Within this type of architecture, they are distinguished:
   (a) *Distributed Architectures:* These are characterized by the fact that each member of the MRS has the power to make decisions for himself. Although it is necessary to establish a communication system that allows some exchange of information between the elements of the MRS; to cooperate and achieve common objectives, the loss of one or more elements does not lead to the failure of the mission.
   (b) *Hierarchical Architectures:* Characterized by establishing a local order between the different elements of the system. They are complex architectures but present good robustness in terms of failures and autonomy of operation (Cao, Fukunaga, & Kahng, 1997).

One of the essential characteristics within MRS is the type of robots that form it. Thus, homogeneous MRS consist of robots or vehicles with identical characteristics. Habibi, Kingston, Xie, Jellins, and McLurkin (2015), Habibi, Xie, Jellins, and McLurkin (2016), Wawerla and Vaughan (2010) (Fig. 1). On the contrary, when at least one of the agents that form the MRS presents different characteristics or capabilities from the rest of the MRS, it is said to be **heterogeneous MRS** (Gregory et al., 2016; Mathew, Smith, & Waslander, 2015; Roldán et al., 2016; Zhou, Yu, Sun, & Yu, 2015) (Fig. 2). As indicated, this paper focus on MRS formed by UAVs and UGVs, and it is considered that a system formed by different UAVs with different characteristics is also a heterogeneous MRS (Fig. 2a). From the point of view of path planing its importance lies in the possibility of carrying out this planning in 2D, 3D or mixing both depending on the possibilities of movement of the vehicles used.

Finally, another critical aspect within MRS is the interaction of the elements of the system with each other; that is, how they relate to each other for the simple reason of sharing the same environment. This

(a) MRS formed by UAVs. (b) MRS formed by UGVs.

Fig. 1. Examples of Homogeneous MRS.

(a) MRS formed by different UAVs. (b) MRS formed by UGVs-UAVs.

Fig. 2. Examples of Heterogeneous MRS.

generates the appearance of collective behavior. This behavior manifests itself in different ways:

- **Indifference:** In which, each mission of a robot is independent of the rest, and there may not be a relationship with the rest of the agents.
- **Cooperation:** The association of different agents whose objective is a common end or task. Associated with the cooperation appears the term "*Awareness*," defined as the property of the MRS robots of knowing of the existence of other members of the system.
- **Competition or antagonism:** where the objectives of the system's agents are entirely incompatible with each other.

In this section, a set of concepts and characteristics intrinsic to MRS have been synthesized, and which facilitate the analysis of path planning methods. In this way, the state of the art of the MRS has been summarized, to be able to understand better why to apply some algorithms or others at the time of solving the problem of the path planning in MRS.

3. Path Planning

Path planning is considered as one of the most critical challenges in Artificial Intelligence (AI), and is directly related to the autonomous movement of all types of intelligent systems. So much so, that works such as (Hedrick, Ohi, & Gu, 2020) include the development of an efficient path planning algorithm for the exploration of planets such as Mars.

Path planning algorithms attempt to establish paths and movements that allow mobile vehicles to reach a goal; navigating autonomously and safely in a working environment. Therefore, they generate an optimal solution for a system to change from an initial state to a final state by avoiding static and dynamic obstacles, present in the environment.

The importance of finding a solution to the problem of autonomous navigation and path planning have led to that the developments reached in this area have grown exponentially in the last years, and cover a wide range of methods based on different techniques and characteristics.

The expansion of autonomous vehicles and their frequent use within MRS has led to the adaptation of classic methods of path planning, initially designed for a single vehicle. Along with this adaptation, new techniques have appeared that consider multiple vehicles in the planning, and allow planning in real time, as the vehicles navigate through the environment.

In recent years, several studies have been generated that attempt to analyze the state of implementation and development of trajectory planning techniques aimed at different types of vehicles (Bormann, Jordan, Hampp, & Hägele, 2018; Gayathri & Uma, 2018; Injarapu & Gawre, 2017; Cai, Wang, Cheng, De Silva, & Meng, 2020; Dewangan, Shukla, & Godfrey, 2017; Costa & Silva, 2019). The great extension of different types of techniques and algorithms used within the field of path planning means that the set of studies carried out to date complement each other and, if necessary, continue to analyze and explore this area of research to add studies that complement the knowledge achieved to date and, in new publications, bring together advances made in this field.

Therefore, the main objective of this work is to study and analyze the state of the literature in the field of MRS-oriented path planning and, with this work, to complement previous studies related to this field. This survey collects and analyzes a set of techniques, methods and algorithms aimed at solving the problem of path planning in MRS. For this reason, a general description of the problem is first given, followed by an analysis of the different ways to reach a solution, and finally, the work is focused on the path planning in MRS.

Before analyzing MRS Path Planning-based methods, it is convenient to detail some essential aspects of path planning. There is a set of terms that can be used indifferently to talk about path planning, but that have aspects that allow us to differentiate between them. These terms are:

- **Path Planning:** Related to find a continuous curve, not necessarily smoothed, in the C-space that starts from an initial X_{init} point and reaches an end X_{goal} point. This curve is formed by a set of segments and includes stops at defined positions along the curve. This term focuses on providing a raw solution and, for this reason, sometimes complementary methods are needed to generate an optimal solution.
- **Optimal Path Planning:** This term introduces a cost function based on aspects, such as distance traveled or time; to try to find a set of paths, which optimizes this cost function.
- **Trajectory Planning:** This term is intrinsically linked to knowing the dynamic characteristics of vehicles. Therefore, trajectory planning methods are a further step in optimal path planning. They not only determine where vehicles move, but also establish how they should move along that path. Path planning is included in a Kinodynamic planning problem by considering speeds, accelerations, and kinodynamic constraints of vehicles (Cruz-Martin et al., 2004).

Recently in the literature, a new term has emerged related to path planning, which brings together those methods in which an optimal solution to the problem of path planning is generated by satisfying a specific risk and, called Chance-Constrained Path Planning (Ariu, Fang, da Silva Arantes, Toledo, & Williams, 2017). The developments made in this area are focused on using techniques and algorithms of road planning within this type of planning. Thus, within the analysis of techniques and algorithms made throughout this document may appear implementations aimed at generating an optimal solution within this field, which focus their research on the use of such algorithms.

In the definition of path planning, another relevant term appears as it is the configuration space. The configuration C-space is a mathematical tool developed to collect all the configurations and positions of a vehicle (Goerzen, Kong, & Mettler, 2010). This C-space is divided into two subsets: the free space, with the positions that the vehicle can reach, and the obstacle space, with those positions that are unreachable or susceptible to collision.

When analyzing and comparing the methods of path planning, it is necessary to establish objective criteria. From the terms used in path planning, two types of approaches can be obtained when comparing results (LaValle, 2006):

- **Feasibility:** finding a safe path from the initial setup to the target, regardless of efficiency.
- **Optimality:** to generate an access plan that allows optimally reaching the objective.

Also, path planning is a problem whose computational complexity is linked to the dimensions of the problem. Therefore, depending on the type of vehicle and the size of the MRS, the size of the problem varies considerably from the computational point of view. This aspect causes those methods that manage to reduce this computational cost are better positioned to generate optimal solutions to this problem, and that can be used in real-time dynamic planning.

Path planning methods are considered complete if, whenever there is a solution, they are able to find it and, if there is no possible path, they report the inability to establish a path.

In this paper, the path planning methods applied to MRS are divided into four categories (Fig. 3): each category is characterized with some specifications and has advantages and disadvantages according to the type of MRS, and working environment. Throughout this section, each category and its algorithms is analyzed based on recent work in the literature.

3.1. Decomposition graph-based methods

The idea behind these methods is to break up the environment into a grid. In this way, it is possible to obtain a representation of the environment in the form of cells, establishing a code that allows discriminating which cells correspond to obstacles, to free space, and to start or end nodes of the path. Once this decomposition is generated, nodes are established in the free cells, and joined to each other through edges, creating a structure known as graph (Fig. 4). In this way, maps of the real environment are modeled with a set of vertices (V) and edges (E).

Therefore, the solution to the path planning problem is to find a sequence of consecutive edges that join an initial node with a final one, i.e., look for a discrete optimization through the cellular decomposition of the environment or graph. To find the optimized solution, they calculate

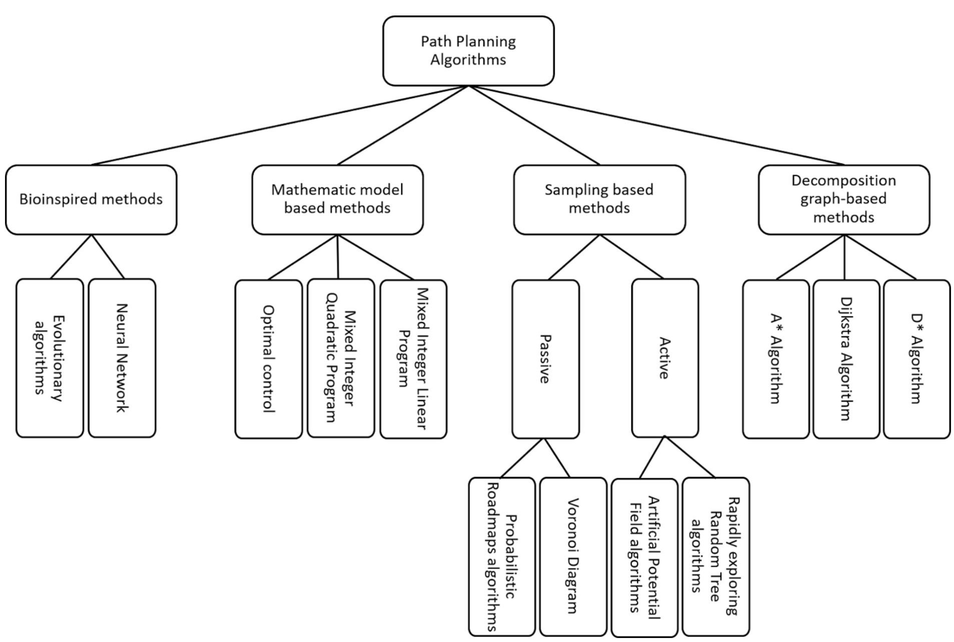


Fig. 3. Diagram of Path Planning Algorithms.

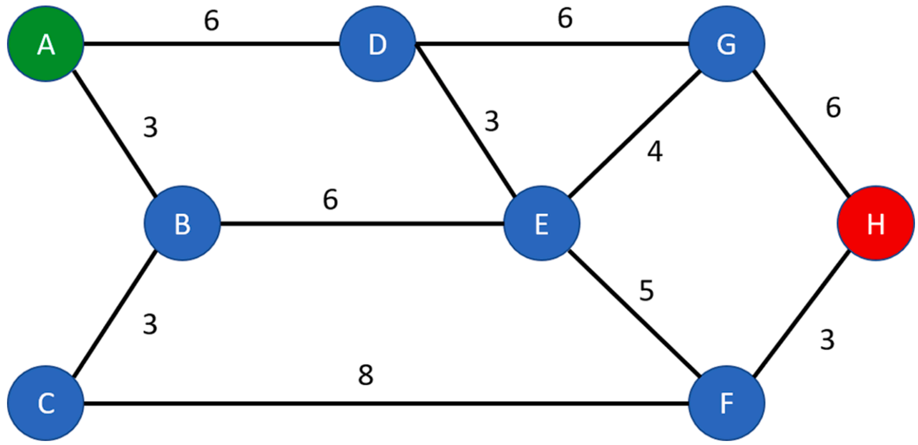


Fig. 4. Graph Example.

the cost by scanning through the nodes and considering the weights of each node and edge they pass through. Therefore, it is frequent to find developments based on other methods that employ these methods in later phases; to obtain an optimal solution.

This type of method includes an extensive set of algorithms, which in this work are simplified into three: Dijkstra, A*, and D*; because they are the most used in MRS-oriented path planning.

3.1.1. Dijkstra algorithm

The Dijkstra algorithm (Thomas, Charles, Ronald, & Clifford, 2001) tries to find the shortest path in a graph, where the weights of the edges are already known. It emerged as a solution to the problem of finding the shortest path between two cities (starting node and end node).

Its procedure starts from the initial node with a value of 0, going through all the adjacent nodes looking for the lowest edge weight. The procedure is repeated while there are adjacent nodes to cover, while comparing the weights and storing the one that optimizes the path. When the end node has been reached, the procedure is terminated.

The Dijkstra algorithm has the advantage of the capability of finding the shortest path between two locations, however, when performing a complete exploration of the environment presents problems of the computational cost, where the complexity of the problem increases. Thus, it is established that the computational cost grows quadratically with the number of nodes in the graph.

In Mac, Copot, Tran, and De Keyser (2017), the Dijkstra algorithm is used in the second level of development; to obtain collision-free paths from a graph obtained by triangular decomposition. Once this graph has been obtained, the algorithm finds the optimal solution for various objectives. In this work, a third phase based on genetic algorithms is also included; to achieve the smoothing of the paths generated by the Dijkstra algorithm. Furthermore, the work shows how the algorithm's execution time increases as the environment become more complex. Despite this setback, execution times allow this development to be considered as a solution to path planning for multiple vehicles; such as exploration missions, surveillance operations, or agricultural work.

In Bai, Yan, Cao, and Xue (2019), a complete planning algorithm has been proposed for a heterogeneous multi-vehicle system. In which, a task assignment phase is established, along with a Dijkstra-based path planning algorithm; to minimize the travel time between two given locations. This algorithm allows obtaining an optimal solution to the path planning problem from a scenario previously divided into cells. Although, it is used for an invariant scenario, it does not provide the option of dynamic planning while the vehicle navigates towards its destination.

In Chen, Zhang, Huang, Liu, and Dai (2019), the authors develop a Dijkstra-based path planning algorithm; to generate an optimal and coordinated multi-path solution for substation inspection robots. In this work, once a task is assigned to a robot, the Dijkstra algorithm plans the shortest path, adding to the solution the time of occupying each cell. In this way, when establishing paths for the set of robots, the algorithm can check, through a time window, if there is any conflict in the planning, and if so, recalculates the path. Therefore, through this work, the Dijkstra algorithm can be considered as a solution to dynamic path planning problems.

3.1.2. A* algorithm

Similar to Dijkstra algorithm, the A* algorithm is complete (LaValle, 2006), i.e., it finds an optimal solution as long as it exists. However, unlike Dijkstra algorithm, A* does not go through the entire graph in search of this solution, so it gets better results for problems with large environments.

The A* algorithm gathers a set of procedures called best-first search algorithms; that search for a set of possibilities, using an approximate cost heuristic function to order the different alternatives, and inspect the different options in order. Specifically, the Heuristic functions (H) are used to map the nodes in the graph to return a non-negative value, indicating the distance from the node to the target. The Heuristic Function Criteria are:

1. $H(Goal) = 0$
2. $H(x) <= H(y) + d(x, y)$ Where x and y two adjacent nodes and $d(x, y)$ are the length of the edge between these two nodes.

In the path planning problems, the following heuristic functions are usually used:

1. **Euclidean Distance:**

$$H(x_n, y_n) = \sqrt{((x_n - x_g)^2 + (y_n - y_g)^2)} \quad (1)$$

2. **Manhattan Distance:**

$$H(x_n, y_n) = |(x_n - x_g)| + |(y_n - y_g)| \quad (2)$$

The implementation of the A* algorithm is as follows: each node has the initial distance from the node, and the sum of this distance and the estimated distance to the target node. In each iteration, the algorithm tries to select the node that has more probabilities of being in the shortest path between the start and the target.

Considering the direction to the end node, it causes the A* algorithm to be substantially faster than other graph-based method algorithms, and in the worst case, its performance is the same as Dijkstra's. Therefore, its main advantage is to consider the location of the end node with respect to the initial one, and to work in directions that could be more fruitful.

The work Erokhin, Erokhin, Sotnikov, and Gogolevsky (2018) implemented an A*-based algorithm with modifications; to apply it to the resolution of the path planning problem in MRS. For this reason, in this work a dynamic calculation of the value of the costs associated with the network nodes is performed. As the path of a robot passes through a node, the value of that node changes, so that the rest of the robots consider that predefined path, and generate their path accordingly. The obtained results from simulations showed how the A* algorithm, with dynamic cost modification, can generate optimal solutions for navigating groups of robots in unknown maps.

Another modifications in the classic A* algorithm have been presented in Le, Prabakaran, Sivanantham, and Mohan (2018). Where, for a reconfigurable robot, path planning solutions are set up, so that paths are established for each part of the robot. This implementation allows the robot to cross narrow corridors thanks to the coordinated movement of its parts, which allows changes in its morphology. In this case, a Zig-Zag-based A* approach is used; to achieve full area coverage. Where, a set of waypoints with a Zig-Zag pattern is defined, and the A* is used to calculate the shortest path.

In Sun, Zhou, Di, Dong, and Wang (2019), the A* algorithm is part of significant development; to solve the cooperative path planning oriented to coverage problems of large areas. In this work, different types of algorithms are combined; to delimit the environment in areas, and to be able to realize a complete coverage of the land; using the minor number of robots. Once the areas are determined, and the waypoints to be reached by each robot are established, the A* algorithm is implemented to find the shortest path between two waypoints on the route. Therefore, in this work, the A* algorithm does not generate the paths; but optimizes the previously calculated ones, to minimize the distance traveled by the vehicles and, deliver an optimal solution to full MRS coverage problems.

In addition to mobile robots, the A* algorithm is usable in autonomous vehicles moving in structured road environments. In Boroujeni, Goehring, Ulbrich, Neumann, and Rojas (2017), the authors present a modification of the classic A* algorithm, called Flexible Unit A* (FU-A*). In which, the path planning is carried out in structured environments taking into account both static and dynamic obstacles. The main idea is to have a dynamic grid that adapts to the speed, and to the dynamic obstacles present in the road, making a prediction of the position of these obstacles on the map.

3.1.3. D algorithm*

Algorithm D* Stentz (1997) is derived from the abbreviation A* dynamic. D* tries to detect dynamic obstacles, and changes the weights of the edges in real-time; to create a temporary map. Then using that temporary map, it establishes safe navigation of the vehicles from the current location to the destination; by minimizing the time through the unblocked road.

Similar to A*, heuristic functions are used in D*. In the presence of obstacles, the heuristics function is updated and minimized, allowing a powerful and efficient search for paths.

Works like Peng, Li, Chien, Hsu, and Wang (2015), showed an example of how this algorithm is applied to path planning problems for multiple vehicles. In this work, an improved D* Lite algorithm is implemented; to generate solutions considering aspects like the robot size. Through a server, the robots provide information of their positioning, and the costs for the rest of the robots are updated. As the robots move, the costs are updated, and the paths are re-planned. The results of the work showed how the D* Lite algorithm can obtain a solution in less time than the genetic algorithms. Also, with parallel processing, as proposed in the work, the time taken to find a solution is less than in the case of the classic D* algorithm.

3.2. Sampling based methods

These methods are based on a random mapping of the environment; to try to achieve a path between two locations. For the exploration, it is necessary to have a mathematical representation that describes the work space.

This random sampling is usually carried out in the form of nodes or cells. In each iteration, a new point is accessed, and it is determined if it is free space, in this case, it is connected with other nearby samples, generating connections between nodes belonging to the free space. It is establishing a structure or graphic of the free space within the C-space.

An important classification of sampling-based methods is whether they are active or passive methods. Thus, an active method that provide the best path to the target by its procedure. While the passive methods, generate a network of paths from the beginning to the destination, and it is necessary to complement them with algorithms that determine which is the optimal path.

Another essential feature of random sampling methods is that, despite working well in large 3D environments, they are not complete methods. This means that there may be path connects the starting point with the endpoint, but the algorithm is unable to find it. This is because the sampling performed is not enough, and is what is known as the failed case of the Twisty Passageway.

This case consists of a narrow corridor that communicates two areas of the environment. If the random sampling is not able to generate nodes in this corridor, the algorithm does not find a solution, although, in reality, there is such a solution.

Although there are techniques to address this problem (increased sampling density, more dense sampling in areas near obstacles), none guarantees proper operation in all cases. Also, the most effective solutions to this problem are opposed to the main advantage of the methods; by sampling that is its capacity to control the computational cost despite increasing the dimensions of the environment.

Finally, another aspect of sampling methods is sometimes the generation of zigzagging paths, so sometimes path smoothing methods are required; to establish a more social dynamic behavior of the systems.

3.2.1. Probabilistic Road Map

The Probabilistic Road Maps (PRM) (LaValle, 2006) is the first of the passive algorithms included in the sampling-based methods. It is an algorithm that allows exploring large work areas with a lower computational cost compared to the methods detailed in the previous section.

The implementation of a PRM algorithm is as follows: from a C-space, locations are taken randomly, it is checked if this location is to free space or an obstacle, and if it is free, it is a matter of communicating this node with the nearest neighbors; that allows establishing a continuous and obstacle-free border between both. In this way, the environment is explored, and a network within the C-space is elaborated, on which other network-based methods are used to obtain an optimal solution between the starting point and the endpoint.

Although being a passive method can be a disadvantage, sometimes the combination of simple algorithms allows obtaining an optimal solution; with lower computational cost than if active methods are used. Also there is an advantage that makes it a competitive algorithm in MRS field, that is once the network has been generated with all the possible paths in the environment, it has the possibility to be reused for all vehicles in the system, which is different from other methods; such as RRT that requires creating a tree per each vehicle. In this way, a more dense network could be implemented, since the whole MRS would use this complete exploration of the environment; to find the optimal path to the destination.

One of the advantages of PRM algorithms is the ability to explore 3D spaces with a low computational cost, which is why there is more focus in the literature to path planning problems, in systems formed by multiple UAVs.

There is a line of research oriented to the path planning in MRS, which presented a set of works whose basis for the generation of paths is the use of PRM (Preiss, Hönig, Ayanian, & Sukhatme, 2017; Debord, Hönig, & Ayanian, 2018; Hönig, Preiss, Kumar, Sukhatme, & Ayanian, 2018). These works, apart from considering important aspects for UAVs ("Downwash" effect), include the aspects described above; such as the need to introduce optimizing and smoothing paths methods, or time restrictions to adapt the problem to MRS. Among the results of their work, it can be seen how the use of PRM is justified for MRS as it is a scalable algorithm in terms of computation time.

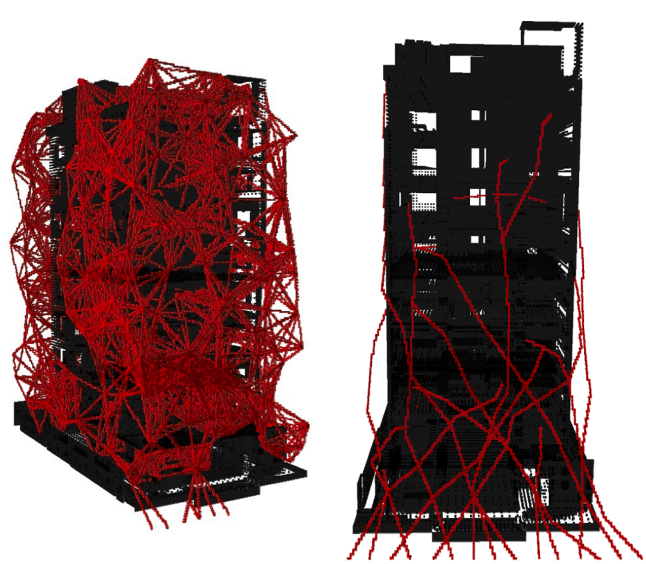
Work such as Madridano, Al-Kaff, and Martín (2020) oriented the path planning to be able to carry out emergency tasks in urban areas, with UAVs swarm. For this, PRM algorithms are used to establish, with a low computational expense, a set of possible paths in large environments such as a building (Fig. 5a). In the results of this work, it is observed how the possibility of reusing the network allows reducing the increase of the computational cost. In this way, through the use of PRM, scalable path planning methods can be created, with applicability in such crucial fields as emergencies. Moreover, although it is not a complete method, its development, together with classic optimization algorithms such as A*, allows the generation of multiple paths in large areas in a short time (Fig. 5b).

Although its main advantage makes it suitable for problems related to UAVs and 3D planning, works such as (Madridano, Al-Kaff, Gómez, & de la Escalera, 2019), showed that it also has applicability in 2D, with even better computation times than in the case of 3D. Although the work is also related to UAVs, and such 2D planning, a fixed flight height is included to adapt it to UAVs. This development is valid for both heterogeneous MRS and UGVs swarm.

3.2.2. Voronoi diagram

Voronoi diagrams (Yan, Jouandeau, & Cherif, 2013) are widely used in the field of path planning. Its concept based on generating a topological connection, which allows space to be divided into regions; taking into account the presence of obstacles. The diagram is constructed in such a way that the distances from the edges to the nearest obstacles are the same.

The procedure to generate a Voronoi diagram starts with the selection of a starting point, which has coordinates with the property that the minimum distance to nearby obstacles is the same. Then, the rest of the regions are calculated, which are determined by the obstacles present in the environment. In this way, the workspace is fragmented, creating regions until all the places are registered.



(a) PRM in Building Case. (b) Final Paths in Building Case.

Fig. 5. PRM Algorithm in Building Case (Madridano et al., 2020).

Therefore, Voronoi algorithm generates a global graph, as it happens with PRM, but as it is a passive method, it cannot generate an optimal solution at the same time. For that reason, algorithms like A* or D* are required to establish safe and optimal paths; using the edges of the regions of the Voronoi diagram.

In Chen, Li, and Chen (2017), the Voronoi diagram is used together with methods based on consistency theory; to establish the optimal paths for a set of UAVs. This work focuses on solving the planning of multiple UAV paths, that must attack multiple targets in a static environment. A Voronoi diagram is established to represent a threat graph, and is used as input for UAVs to cooperatively search, through the theory of consistency, for optimal paths. In this way, the combination of Voronoi diagrams with the combined theory of consistency makes it possible to generate a solution to the problem of path planning in MRS, and that multiple UAVs can simultaneously and cooperatively reach multiple objectives.

In Turanli and Temeltas (2017), a control system for mobile robot coordination based on Voronoi diagrams has been proposed. This algorithm is extended, implementing Voronoi diagrams of Ppower guarantee that allows enabling dynamic Voronoi partitioning. With this advance, the weights in the workspace areas that assigned to the robots are changing, allowing the paths to converge in a coordinated movement of the system.

Works such as Wei, Mao, Guan, and Li (2017), oriented the Voronoi diagrams to the construction of a Centroidal Voronoi Tessellation (CVT) that allows the movement and assembly of a robotic swarm autonomously. The CVT is characterized by the fact that the points corresponding to the cells in the Voronoi diagram are located in the centroids of the cells. The use of this method allows the planning of different paths for each of the swarm robots, allowing the self-assembly of the swarm in different configurations.

Finally, works like Kim and Son (2020), showed another application of Voronoi diagrams for path planning. This consists of using Voronoi diagrams to plan the paths in exploration and search tasks; since the generated graph allows maximizing the resulting paths as a function of the distance traveled. In this work, the Voronoi diagram allows an MRS, oriented to agricultural tasks, to divide the workspaces according to the number of robots. This partitioning through the Voronoi diagram allows the optimization of task assignment, and path planning processes.

3.2.3. Rapidly exploring random trees

The Rapid Random Exploring Tree (RRT) method (LaValle, 2006) aims to sample the environment, taking into account the initial and goal locations. The RRT has the ability to handle problems of multiple degrees of freedom, allowing the RRT to be suitable for work with Personal Robots and robotic arms, however, it has its applications in MRS systems.

The procedure implemented to create the graph (tree) is the following. The C-space is divided into free and obstacle spaces. From there, both the start node and the destination node are introduced in the free space. Next, a random node is entered, which must be in the free space, and be able to connect to the start node. The next step is to check whether the end node can be reached from this new node. If not, the process is repeated, but in this case, the new randomly created node is connected to the end node, and it is checked if it can reach the branch from the starting node.

In this way, in each iteration, the branches of the tree coming from the initial and end nodes grow outwards. It is an efficient procedure to explore the free space since when generating the two branches; the tree is growing and exploring the C-space in parallel.

In each iteration, the system generates a random sample in one of the branches and tries to make it grow until the sample if successful; the next step tries to generate a bridge between both trees. If it is positive, then the objective is met, and a path can be established between the nodes of interest; if it is negative, a new iteration is launched, but this time in the other tree.

RRT is a complete and efficient algorithm for path planning in a high dimensional C-space compared to the graph based algorithms seen before, as it allows a probabilistic exploration of the environment. In addition, it can be used together with dynamic constraints. The most recent works related to RRT algorithms are applied to path planning introducing improvements and modifications, but showing the essential features of classical RRT.

Cui, Li, and Yan (2016) presented a multidimensional RRT*; to carry out path planning within a system with multiple autonomous underwater vehicles. The idea of this work is to develop a method to improve the efficiency in the tasks of search, surveillance, and monitoring of the seabed. The purpose of the RRT* is to create a set of optimal, reachable and collision-free paths through which the different AUVs travel from an initial state to a final one.

Another work that relates the discrete-RRT (dRRT) in multi-robots systems is Solovey, Salzman, and Halperin (2015). It presents an adaptation of the RRT algorithm for the discrete case of a graphic. It provides a fast and high dimensional exploration of the C-space. The technique developed focuses on the search for paths within the roadmap in scenarios involving the coupling of various robots. To do this, a partial exploration of the roadmap is carried out, in such a way that it is only considered a neighbour of a vertex visited at each step, so that the dRRT can quickly explore the C-space represented by the implicit graphic and solve multi-robot problems by exploring only a small portion of that space.

The work presented in Aguilar and Morales (2016), detailed the research on path planning in 3D for a mobile robot; using the RRT algorithm with variants (RTT*). Kinect V2 camera is used to generate a pointcloud of the robot's environment, establishing the regions of free and occupied obstacles spaces. The RRT* method is based on the classic RRT, but introduces two optimisation phases; to minimise the cost of the path obtained. In addition, this work presented two more variations of the RRT* algorithm, with the idea of reducing the time spent on finding a viable solution, and accelerating the rate of convergence, and optimization. These modifications are based on the change in the probability of the random node generated, and are called RRT* Goal and RRT* Limits. The results obtained to establish that RRT* Goal reduces considerably the time to find a feasible solution, increasing the time and computational resources dedicated to optimization, while RRT* Limits generates a trajectory with lower costs than the standard RRT*.

Works like the one carried out in Wu, Li, Xie, and Chen (2020) propose the RRT* algorithm together with a chance constrained formulation to establish a safe solution to path planning within urban air mobility problems. In this way, the combined use of the RRT* algorithm together with formulation uncertainty allows establishing a method for the generation of collision-free trajectories for operations within an urban air area. In line with this work is also the work presented in Berning, Girard, Kolmanovsky, and D'Souza (2020). As in the previous case, the RRT* algorithm is used together with chance constrained to be able to establish safe trajectories within future airspace shared between traditional commercial aviation and UAVs. Both works present the advantage and the novelty of being able to generate a collision-free planning solution considering the uncertainty propagation and, being able to establish, in a computationally efficient way, safe paths with constrained chance for different UAV models.

Finally, in Solana, Furci, Cortés, and Franchi (2017), a new tool is presented to give a solution to the problem of the path planning in a MRS that sails by a disordered environment. The work combines the theory of Maintenance of the Generalized Connectivity with an extension of the RRT algorithm, to establish paths within the MRS that consider aspects like the obstacles or the maximum range of communication. With this combination, a global planner that generates adequate paths for the MRS is generated, respecting the connectivity requirements to guarantee the coordination between them. The RRT extension presented is called T-RRT, and is used for the management of problems related to path planning in disordered environments, and where it is necessary to consider additional cost criteria during the scanning process. With this expansion, the aim is to avoid local minimums thanks to a self-adaptation mechanism that efficiently explores the valleys. This work shows how improvements and expansions of the classic RRT allow obtaining optimal solutions to the path planning problem in MRS with this type of algorithm.

3.2.4. Artificial potential field methods

Artificial Potential Field (APF) algorithms are widely used due to their low computational complexity (Yang et al., 2016). They are based on establishing a potential function of C-space; to create a relation between free and obstacles spaces, in such a way that the function has high values when it approaches the obstacles, and low when it is far from them. Furthermore, a minimum is set in the function that coincides with the location of the goal.

The first part of the implementation consists of creating this potential function, which it is required to define a repulsive field related to the obstacles of the C-space, and an attractive field, with a minimum located in the goal, indicating the goal in the free space. Once the function is established, the gradient is used to guide the vehicle from the starting to the end points.

The advantage of these methods is that they are relatively simple, to operate in real-time at tens of hertz, and can work with large problems. However, their main disadvantage is that they are incomplete; they cannot always ensure success. This problem is because the potential function can be more than minimal; when combining attractive and repulsive forces. This means that vehicles may not reach their destination, and fall to a local minimum.

In practice, it is difficult to eliminate the local minima, and to know when the algorithm can converge and when it cannot. This problem has been avoided with developments such as navigation functions, or the calculation of potential with restrictions.

Works such as Ying and Xu (2015), studied the use of APF; for free movement of obstacles from Formations Leader–Followers of mobile robots. Two potential fields are generated, one attractive to the destination, and one repulsive of the obstacles working with both jointly to form a composite potential field, and making the MRS to reach the goal avoiding the obstacles, moving relative formations in order to performe the task safely and reliably.

The work presented in Sun, Tang, and Lao (2017), showed an optimized APF algorithm for operations of multi-UAV systems in 3D dynamic space. In which, an APF method is developed with a distance factor and a jump strategy; to solve common problems such as avoiding obstacles. Also, this algorithm takes into account the dynamic objects, which are the rest of the vehicles located in the environment; in order to generate safe paths in a collaborative system. The results showed the validation of this method in simulations with 6 UAVs and 30 obstacles present in different directions and configurations.

In Hassan, Elias, Shehata, and Morgan (2017), an APF algorithm for safe navigation of a UGV swarm is presented. For this purpose, a potential 3D map is created, considering the attractive field towards the goal, and two repulsive fields, related to the obstacles and to the robots themselves; to avoid collisions between them. Also, a virtual obstacles method is implemented to deal with the local minimums problem. The obtained results showed that the APF is a solution for the path planning in MRS, implemented together with the APF algorithms; to avoid local minima.

Finally, works such as Wu, Su, and Li (2019), introduced improved APF algorithms (IAPF); to obtain an optimal solution to the path planning problem, solving problems like local minima or smoothing of final paths. The proposed IAPF introduces a gain restriction to the potential repulsive field model. In addition, a random factor is added to avoid falling into local minima, and finally, a method based on B-spline curves; to optimize and smooth the obtained paths. The gain restriction allows controlling the repulsive forces of the obstacles; depending on the distance of the robots to the goal. If an obstacle is close to the goal, the robots are able to reach the target by reducing the repulsive field of that obstacle. The obtained results showed how the implemented methods allow the robots to reach the goal through smooth paths, avoiding both collisions and falling into a local minimum.

3.3. Mathematics model based methods

This section includes algorithms based on mathematical models; such as Mixed Integer Linear Program (MILP), Mixed Integer Quadratic Program (MIQP), and Optimal Control.

These methods are characterized by establishing kinematic and dynamic constraints, for modeling the environment and the system. In addition, a cost function is used to include the limits of constraints as equations or inequalities. The minimization of this cost function allows for an optimal solution.

The fact that this type of method considers aspects such as dynamic constraints; to achieve an optimal cost, causes that these methods are called as trajectory planning algorithms.

The methods based on mathematical models tend to have a complex formulation, and therefore, a high computational cost (Song, Kim, & Morrison, 2016). To face this problem, discrete decision processes are established or, sometimes, the algorithms are oriented to describe concrete parts of the problem, combining methods to cover the whole path planning problem.

3.3.1. Mixed integer linear program

The MILP algorithms are based on a cost function that takes into account aspects; such as kinodynamic constraints, minimum distance, energy, or threats in the environment. Also, the linear algorithms are characterized; by being able to model both the kinematics and the dynamics of the environment; to represent the workspace and the systems. In summary, the MILP presents a high capacity to model the essential aspects of the problem, and to describe almost all the information (Yang et al., 2016).

In Song et al. (2016), a MILP formulation is proposed to be able to carry out an escort service with UAVs; that work simultaneously with different clients. The MILP model allows the problems of this application to be formally represented, and also introduces the possibility of using a rolling horizon planner based on the initial locations, and status of the UAV batteries. The work presented a path planning formulation based on MILP, in which the objective function minimizes the sum of the weighted total distance traveled, and the number of jobs attended. Also, restrictions are introduced within the MILP model; to allow the coordination of UAV paths. These restrictions consider the initial location of UAVs, the guarantee that UAVs reach their destination, battery levels, or that tasks are performed by at least one UAV. With the MILP formulation, and the development of an efficient heuristic, it is possible to develop a tool for the planning of routes on the horizon, so that a set of UAVs can provide an escort service to a group of consumers.

Another work that presents a MILP algorithm development is Lal, Sharda, and Prabhakar (2017). The aim of this work is to generate an optimal path planning for an MRS for pesticide spraying in agricultural fields. For this purpose, extensive modeling of the workspace and the systems used was carried out, and through the formulation of MILP, restrictions are established to obtain a complete path for each robot, so that the robot is able to visit all the nodes previously set.

3.3.2. Mixed integer quadratic program

The MIQP are related to the MILP algorithms. In this case, the difference with the previous ones is that the resolution of an objective quadratic function must be carried out (Lazimy, 1982).

Mellinger, Kushleyev, and Kumar (2012) presented the use of MIQPs for generating 3D paths in environments with obstacles. The idea is to create optimal paths in systems with multiple quadcopters of different sizes, characteristics and capacities. It is important to emphasize that in this work, the use of piece-wise smooth polynomial functions; to synthesize trajectories in the flat exit space, appears as a key aspect. This allows reinforcing the continuity between the waypoints until arriving at any derivation of the desired position. The results showed a feasible solution for small teams moving in simple environments, i.e. with a low number of obstacles. In addition, it is necessary to know both the starting and end positions of the different agents in the team. This method can impose restrictions on the positions, speeds, accelerations, shocks, and inputs, allowing different sizes, capacities, and dynamic effects vary between different quadcopters. Although it is capable of reaching feasible solutions in milliseconds. This same research group presented another work (Kushleyev, Mellinger, Powers, & Kumar, 2013) that collects the use of the MIQPs method; to generate trajectories in swarms of micro quadcopters. In order to avoid the increase in computational expenditure suffered by MIQPs algorithms by increasing variables and restrictions, the set of UAVs is divided into small groups with a small number of robots in rigid formation, reducing planning time.

Furthermore, the MIQP formulation also extends to the field of autonomous cars (Burger & Lauer, 2018). In this work, a cooperative MIQP formulation is introduced for the trajectory planning of multiple vehicles. In particular, the problem of establishing a set of cooperative trajectories in autonomous vehicles, which are communicated between each other, and aimed at non–hazardous road scenarios, is raised. Using MIQP, a global optimal solution is established, which, together with the obtained results, showed the viability of this method to solve problems of cooperative path planning, in non–hazardous scenarios for autonomous vehicles.

3.3.3. Optimal control

The third algorithm included in the mathematical models is planning through Optimal Control (OC). Optimal control seeks to find the state and path based on control from a set of differential equations. It is considered an extension of the linear methods, but it works with an infinite number of variables. To solve the problems through optimal control, the Hamiltonian is used; to solve the optimization problem based on the maximum principle, and to continue with a standard optimal solution procedure; to generate a global optimal path (Yang et al., 2016).

The optimal control is not centered in a single type of algorithms, but under a similar initial plant, different procedures are established; that allow reaching the final objective that a MRS navigates in an autonomous way.

One of the most popular techniques within optimization-based methods is the Model Predictive Control (MPC). The main idea of the MPC is to find the optimal control actions, which must be carried out in the future, from the prediction of the system behavior. In the Spurny, Baca, and Saska (2016), an MPC algorithm is formulated to allow the planning of trajectories of a leader, and to control the navigation of the follower robots. The obtained results allow observing how the MPC method is capable of generating collision-free paths, with static and dynamic obstacles while maintaining a pre-set formation.

A Distributed MPC (DMPC) is presented in Luis, Vukosavljev, and Schoellig (2020); to generate paths in real-time for multiple robots. Through the parallelization of the method, the authors obtained high scalability in their work. The obtained results showed the possibility to send paths for a swarm of 20 UAVs at a frequency of 20 Hz. Furthermore, with the DMPC, it is possible to reduce up to 50% of the flight time required to complete the whole path, compared to the buffered Voronoi cell approach.

Another technique included in this section is the Covariant Hamiltonian Optimization for Motion Planning (CHOMP). CHOMP is a method that allows the optimization of paths and improves the quality of initial paths by optimizing a functional objective. This optimization is done by looking for path smoothing and obstacle avoidance. Its development is based on the use of superior notions of geometry, and allows the algorithm to be positioned with a powerful tool in the field of path planning (Ratliff, Zucker, Bagnell, & Srinivasa, 2009).

Works such as David, Valencia, and Iagnemma (2016), showed how CHOMP could be used to solve path planning problems in MRS. Specifically, in this work, the authors uses CHOMP for a path refinement phase, which allows a set of autonomous guided vehicles (AGVs) to move autonomously and safely, while performing work at loading terminals. CHOMP is used as a local planner to avoid dynamic obstacles, and to consider the curvature limitations of AGVs. Also, an extension of the method is included to be able to resolve potential path conflicts between multiple AGVs.

3.4. Bio-inspired methods

These methods seek to mimic biological behavior to try to obtain a solution to the planning problem. Unlike mathematical methods, bio-inspired methods do not focus on modeling the environment and present a development that avoids problems such as falling into local minima or solving complex objective functions.

The bio-inspired algorithms try to imitate how living organisms behave and act, to try to generate an optimal path. Among the main characteristics of these methods are: not being fully deterministic, presenting parallel structures, and being adaptive. This set of factors allows these methods to generate optimal solutions to path-finding problems without having to know, in an exhaustive manner, the environment in which the mission is being carried out. This fact makes them effective methods to solve multi-objective problems (Guzmán et al., 2013).

Within these methods, two differentiated groups are established: the Evolutionary algorithms, which analyze the behavior of a species, and the Neural Networks (NN), based on imitating the connections and functioning of neurons when processing information.

3.4.1. Neuronal network

Since the last decade of the last century, Neural Networks (NN) have been used for navigation and obstacle avoidance with applications of path planning, both ground and aerial vehicles (Glasius, Komoda, & Gielen, 1995).

The idea of NN is to generate a dynamic landscape, which shares some standards with APF methods; in particular, in unexplored areas, they try to attract vehicles along with the whole environment. Therefore, NNs are designed to establish the maximum neuronal activity between the different layers of neurons. Then, by introducing the dynamics of the vehicles into the NN, it is guaranteed that the neuronal activity spreads throughout the free space, being able to navigate through safe paths in an autonomous way.

Although in recent years there has been an expansion of these techniques, their main disadvantage is that being a bio-inspired method, they cannot be standardized, that is, they cannot form either rules or canonical models. Therefore, finding an optimal solution does not guarantee that it is applicable to a path planning problem in a different environment (Yang et al., 2016).

The NNs encompass a wide range of techniques, such as Deep Learning (DL), Reinforcement Learning (RL), or a combination of both, as shown below, where works in the recent literature show that there is a set of techniques within the NN that are applicable to MRS path planning problem.

The authors in Bae, Kim, Kim, Qian, and Lee (2019), proposed a path planning algorithm for MRS based on Deep Q-Learning combined with Convolutional Neural Networks (CNNs). The idea of using this combination is to generate an agile situation analysis. While the CNN analyzes the situation using the information captured by the robots, the Deep Q-Learning generates the actions for the robot to navigate. The obtained results showed how this technique eliminates problems of conventional methods, as the network learns through the information collected by all the robots, which have a mutual influence. Furthermore, the results showed how this kind of technique is applicable in static and dynamic environments, thanks to the robots sharing the memory used for learning.

In Qie et al. (2019), an NN-based algorithm is presented, which tries to give an optimal solution not only to the path planning problem, but also to the task assignment. The technique used in this work is based on a Multi-Agent Deep Deterministic Policy Gradient (MADDPG), which belongs to the field of multi-agent reinforcement learning. The procedure followed consists of using the MADDPG algorithm to train the system, simultaneously solving the assignment of objectives and the path planning, following the corresponding reward structure. To guarantee the application of this development in dynamic environments, and in real-time, a simple NN is used in the system. The results showed that the implemented algorithm is effective in several scenarios, and applicable to a scalable set of UAVs.

Other works such as Cruz and Yu (2017), looked for modifications in the RL algorithms; to solve problems like a slow learning speed, or the impossibility to learn in completely unknown environments. For this purpose, a multi-agent reinforcement learning algorithm called WoLF-PHC (Win or Learn Fast Policy Hill-Climbing) is proposed, which is modified to adapt it to unknown environments. The main modification is to ensure that any robot has its answers when the rest remains in a stationary state. In this way, it makes that the performance of the agent resembles a deterministic strategy, in which previous knowledge as the transition function, or the reward functions are not necessary. The results showed that the technique allows a set of mobile robots to generate paths in dynamic environments, where there are coordination or dimensionality problems.

Finally, in Afifi, Alhosainy, Elias, Shehata, and Morgan (2019), an RL-based algorithm is proposed to allow the navigation of an MRS in a given formation; to achieve the objective, a multi-layer architecture is implemented to be able to control the formation of several vehicles, through the planning of the movements. This multi-layer is divided into two subsets with different tasks. Firstly, a Q-Learning algorithm to determine the configuration of the vehicles. Secondly, a deep reinforcement policy-gradient algorithm is established to carry out the planning of collision-free paths. The idea of using a combination of these methods is that the Q-Learning algorithm has a low computational time and cost, while the deep reinforcement policy-gradient algorithm allows training the system in new environments and discover them by trial and error. The results showed how the agents are able to establish an optimal path; to deploy in the environment with a given formation.

3.4.2. Evolutionary algorithms

Evolutionary algorithms are implemented to solve traditional linear and dynamic programming problems that provide a large number of variables. Their implementation is based on a stochastic search; that imitates the evolution and social behavior of biological systems (Back, 1996).

The procedure followed by the evolutionary algorithms consists, first, in randomly selecting the possible solutions with a first-generation. In the next step, aspects are considered, such as the capacity of the robots, the objective to achieve, or existing limitations in the whole environment-systems. Then, a set of parents of the first generation are selected for the next generation, for in the last step; to perform a process of mutation and crossover, which is repeated until the goal is achieved. Finally, the best individuals are decoded and used as nodes to establish the optimal path.

The evolutionary algorithms comprise a set of techniques that follow this procedure, which are the Genetic Algorithm (GA), Memetic Algorithm (MA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), Differential Evolution (DE) and Shuffled Frog Leaping Algorithm (SFLA). The GA was the first proposal for an evolutionary algorithm, and later, the rest of the techniques appeared inspired by different processes in nature.

In Bai et al. (2019), a coevolutionary multi-population genetic algorithm (CMGA) is proposed; to minimize the total travel time of a set of vehicles. In addition to path planning, the algorithm is oriented to task assignment, where different vehicles must reach a set of destinations. The problem applies to a set of autonomous marine/airborne vehicles navigating in a drift field. The obtained results showed that the CMGA has an excellent performance in solving path planning problems with multiple objectives compared to classical GA, for a heterogeneous team of vehicles.

In works such as Han, Wang, Liu, and Zhao (2017), an improved genetic algorithm is presented for path planning in multiple automated guided vehicles (AGVs), the improvements implemented of which consists of: using three crossover heuristic operators, producing optimal offspring characterized by more information; and exercising double-track restrictions to minimize the total distance traveled by all AGVs, and the distance traveled by each AGV. The results guarantee that these improvements make it possible to reduce the distances traveled by each AGV, and failing that, by the whole set, compared with the use of classical genetic algorithms.

The use of Enhanced Genetic Algorithms (EGA) is also studied in Nazarahari, Khanmirza, and Doostie (2019). The authors presented a method that combines APF and EGA to be able to plan paths of multiple robots in continuous environments. The APF generates all possible paths between the initial and final points. While the EGA finds the optimal paths between the locations. In this work, the EGA employs five cross and mutation operators; to improve the initial paths. In addition, the EGA includes within the objective function a parameter; to avoid possible collisions between the paths. The results showed the efficiency of this method compared to other classical methods, or other evolutionary algorithms such as PSO, not only establishes collision-free paths, but the solution found for all robots is the optimal one.

Finally, although the GA are the most used in the field of path planning, there are evolutionary algorithms such as PSO that are also used in recent work to solve this problem. In Zhen, Enze, and Qingwei (2020), an improved PSO is presented for establishing UAV paths in known, rough, and static environments. The improvement introduced in the PSO consists of a vibration function which improves the collided solutions rather than forsaking them. The obtained results showed how the implemented method is able to set multiple paths for a set of UAVs in rough terrain.

Within the evolutionary methods, there is also differential evolution (DE), which is considered an optimization method applied to the resolution of complex problems. This feature has caused that within the literature appear a set of works that use this technique to carry out the planning of trajectories for different types of vehicles.

A field of application of this type of algorithm is the underwater vehicle gliders. In works such as Zamuda and Sosa (2019) a trajectory planner based on the Success History Adaptive Differential Evolution (SHADE) algorithm is presented. The proposed algorithm adds the linear reduction of the population size (L-SHADE) in order to achieve an optimal trajectory planner in underwater glider missions whose efficiency is compared with similar algorithms present in the literature obtaining a competitive result and, establishing this method as an effective alternative to the problem of trajectory planning for underwater glider vehicles. In addition, the introduction of the L-SHADE improvement in other classical evolutionary algorithms allows them to improve their performance in some cases.

In line with the previous work is MahmoudZadeh, Powers, Yazdani, Sammut, and Atyabi (2018). In this work, the differential evolution algorithm is used to establish a solution to the problem of trajectory planning for unmanned underwater vehicles. In this case, it is demonstrated the performance of this planner to generate time-efficient trajectories that allow the underwater vehicles to reach a location of interest in an adverse and dynamic environment such as the ocean floor. The results of this work demonstrate the efficiency of the algorithm to be able to extract feasible areas of a real environment and determine the spaces allowed for the navigation of the vehicle, considering aspects such as marine disturbances, desired currents, or collision avoidance.

Although this algorithm has not been exploited only in the field of UAVs, works such as Yu, Li, and Zhou (2020) use as a basis the differential evolution algorithm to establish a solution to trajectory planning in the field of UAVs. In this case, the planning is modeled as an optimization problem of proficiency functions that include aspects such as distance traveled or risk of UAVs and include constraints such as flight altitude or course of the UAVs. In this case, the differential algorithm presents an adaptive selection mutation, in which individuals are selected according to their aptitude values and the violation of the imposed restrictions. These individuals are used to make the mutation, and the algorithm is responsible for finding the best individual among those selected. Thus, the results show a competitive algorithm that allows planning optimal trajectories for UAVs in natural disaster environments.

This section, discussed the most recent used methods and implemented algorithms for path planning in MRS. In which, and as shown in Tables 1 and 2, the advantages and disadvantages of each method are highlighted.

4. Applications

The coordinated and autonomous work of several vehicles in the same working environment allows, among other aspects, to improve the response times, which is a crucial aspect in most of the applications. The cooperative work of the MRS has some advantages over individual use, which makes them the best solution in achieving a specific type of missions; by performing different tasks simultaneously, increasing the overall efficiency of the system. Moreover, by setting different configurations, this will provide the multi-robot teams with a series of heterogeneous skills, which leads to adapting each agent to the mission that best suits its specifications. Finally, presents higher levels of robustness, due to the ability to tolerate failures or breakdowns; by having different sources of information.

The field of MRS systems has undergone exponential development, presenting innovative solutions with a wide range of applications. Among them, this work throws light on the emergency missions (Gregory et al., 2016; Mouradian, Sahoo, Glitho, Morrow, & Polakos, 2017; Huang, Chiba, Arai, Ueyama, & Ota, 2015); such as fire-fighting, search and rescue, or surveillance.

Works such as Couceiro, Portugal, Ferreira, and Rocha (2019), presented a project called SEMFIRE, where a system formed by a UAV and a UGV destined; to carry out actions that help to prevent forest fires. The UGV’s main mission is to eliminate forest residues, while the UAV works in coordination with the UAV; to explore areas where the UGV’s action is necessary, and also to monitor the area.

In Marchant and Tosunoglu (2016), a tool based on a swarm of UAVs that can support emergency teams is proposed. The idea is to have a set of UAVs equipped with different sensors, so that, they can autonomously access remote areas, collecting information, and even, through portable fire-fighting technology, carry out fire mitigation.

Innocente and Grasso (2019) presented a work that shows that the use of robotic swarms is a viable and powerful solution; to be used in fire-fighting in an autonomous way. The idea of this work is to develop self-coordinating mechanisms that allow effective swarm behaviour in fire fighting. The authors presented a self-organizing algorithm for UAVs capable of adapting to physical models of fire propagation, whose results showed an effective, scalable and fault-tolerant solution.

In Gregory et al. (2016), a tool based on an MRS is presented, which supports human teams deployed in the field. The objective of the work is to provide the MRS with autonomous navigation, so that in case of communication losses, and therefore the possibility of teleoperation, it can continue its function. Therefore, the authors presented an MRS capable of navigating autonomously, with the mission of gathering information from the environment, in areas of natural disaster where communication losses have occurred.

In addition, search and rescue missions have been encountered a great focus within MRS; taking the advantage of having a set of vehicles

Table 1
Most common methods used to solve path planning problems in MRS (Part 1).

Method	Algorithm	Advantages	Disadvantages	References
Decomposition graph-based	Dijkstra	• Easy implementation • Optimal solution in discretized space • Multiple environments	–High computational cost –Static environments	Mac et al. (2017), Bai et al. (2019), Chen et al. (2019)
	A*	• Optimal Solution in discretized space • Guided and fast search • Online Implementation	–High computational cost in large environments –Static environments	Erokhin et al. (2018), Le et al. (2018), Sun et al. (2019), Boroujeni et al. (2017)
	D*	• Optimal Solution in discretized space • Guided and fast search • Dynamic environments	–Heuristic function uses unrealistic distances	Peng et al. (2015)
Sampling-based	PRM	• Reusable solution • Fast exploration of large environments	–Non-optimal solution –Passive method	Preiss et al. (2017), Debord et al. (2018) Hönig et al. (2018), Madridano et al. (2020), Madridano et al. (2019)
	Voronoi	• Easy implementation • Low computational cost	–Passive method –Nonconvergence –Static environments	Chen et al. (2017), Turanli and Temeltas (2017), Wei et al. (2017), Kim and Son (2020)
	RRT	• Low computational cost • Fast exploration	–Single Path Generation	Cui et al. (2016), Solovey et al. (2015), Aguilar and Morales (2016), Solana et al. (2017)
	APF	• Low time complexity • Online implementation	–Local minima	Ying and Xu (2015), Sun et al. (2017), Hassan et al. (2017), Wu et al. (2019)

Table 2
Most common methods used to solve path planning problems in MRS (Part 2).

Method	Algorithm	Advantages	Disadvantages	References
Mathematic model-based	MILP	• Complete system and environmental information	–High time complexity	Song et al. (2016), Lal et al. (2017)
	MIQP	• Time consideration for path planning	–Complex mathematical model solving	Mellinger et al. (2012), Kushleyev et al. (2013), Burger and Lauer (2018)
	OC	• Solution considers the control of the systems	–No analytical solutions	Spurny et al. (2016), Luis et al. (2020), Ratliff et al. (2009), David et al. (2016)
Bioinspired	NN	• Infinite environments and systems	–Non-generic solutions –Success conditional on correctly parameterized rules and organisms	Bae et al. (2019), Qie et al. (2019), Cruz and Yu (2017), Afifi et al. (2019)
	EA	• Suitable for NP-hard and multi-objective problems	–High time complexity	Bai et al. (2019), Han et al. (2017), Nazarahari et al. (2019), Zhen et al. (2020)

available to explore a wide range of terrain, in a coordinated manner, which in turn reduces the time required to search for survivors or missing persons.

In Fung et al. (2019), a search algorithm of a team of robots is presented to explore the environment, collecting information about the areas, and providing high priority points of interest. From this output, a division of the terrain into regions is established; according to the effort required to explore that region, and then the divisions are assigned to the different robots; to try to achieve a fast and coordinated exploration.

Work such as Bakhshipour, Ghadi, and Namdari (2017), focuses on the use of coordinated robotic swarms to search for victims at disaster sites. The authors proposed a heuristic algorithm that solves nonlinear continuous optimization problems. With this implementation, they achieves that the location of the victim is the best solution to the algorithm. Within the swarm, the a master robot supervises the swarm, and collaboratively, it tries to reach the victim's location. The proposed algorithm is compared with other methods and showed good results in terms of speed of convergence.

Finally, in Tardós, Aragues, Sagüés, and Rubio (2018), a swarm formed by UGV and UAV is presented. The idea of this work is to combine the information collected by both platforms, so that, a reconstruction of the environment is obtained with information from two different points of view. The authors proposed two strategies to solve the problem of coverage in communications, and simultaneous monitoring. With these proposals, the authors implemented two solutions to deploy in an area of interest UGV and UAV teams that can explore the region, establishing a communication link between both, and allowing the robots to have a mutual tracking of their navigation; in order not to lose the link in the communications.

5. Conclusion

In this work, trajectory planning methods have been reviewed as a whole methodology to cope with cutting-edge Multi-Robot System (MRS) technology, where motion planning has been studied as a complex and essential task for swarm of vehicles. This work gathers a study of the current state of the literature in the field of path planning oriented mainly to MRS. This survey aims to complement previous studies in this field, analyzing a set of techniques, methods, and algorithms, through recent and innovative work, aimed at solving the problem of path planning for MRS.

In this work also analyzes the progress in the field of the MRS and its increasing application in real civil applications in different situations, such as swarm of robots, formation, and type of robots. A field in which the MRS are presented as a technological solution of high applicability is that of emergencies, where a rapid response time is critical to the success of the operation. Thus, in this work, recent work in which the MRS are used in critical tasks and applications within the field of emergencies have been collected, and have highlighted the difficulties of application of the same, which are lines of research to be addressed.

Furthermore, the presented survey provides a full review of the methods and applications in the last decade; to provide full understanding to the importance of MRS in critical missions of emergency.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work was supported also by the Comunidad de Madrid Government through the Industrial Doctorates Grants (GRANT IND2017/TIC-7834).

References

Afifi, A. M., Alhosainy, O. H., Elias, C. M., Shehata, O. M., & Morgan, E. I. (2019). Deep policy-gradient based path planning and reinforcement cooperative q-learning behavior of multi-vehicle systems. In 2019 IEEE International Conference of Vehicular Electronics and Safety (ICVES) (pp. 1–7). IEEE.

Aguilar, W. G., & Morales, S. G. (2016). 3d environment mapping using the kinect v2 and path planning based on rrt algorithms. *Electronics, 5*, 70.

Amato, C., Konidaris, G., Cruz, G., Maynor, C. A., How, J. P., & Kaelbling, L. P. (2015). Planning for decentralized control of multiple robots under uncertainty. In 2015 IEEE International Conference on Robotics and Automation (ICRA) (pp. 1241–1248). IEEE.

Amato, C., Konidaris, G., How, J. P., & Kaelbling, L. P. (2014). Decentralized decision-making under uncertainty for multi-robot teams. In *In the Workshop on Future of Multiple Robot Research and its Multiple Identitie at IROS*. IEEE.

Amato, C., Konidaris, G., Omidshafiei, S., Agha-mohammadi, A.-A., How, J. P., & Kaelbling, L. P. (2015). Probabilistic planning for decentralized multi-robot systems. In *In 2015 AAAI Fall Symposium Series*.

Ariu, K., Fang, C., da Silva Arantes, M., Toledo, C., & Williams, B. C. (2017). Chance-constrained path planning with continuous time safety guarantees. In AAAI Workshops.

Back, T. (1996). *Evolutionary algorithms in theory and practice: Evolution strategies, evolutionary programming, genetic algorithms*. Oxford University Press.

Bae, H., Kim, G., Kim, J., Qian, D., & Lee, S. (2019). Multi-robot path planning method using reinforcement learning. Applied Sciences, 9, 3057.

Bai, X., Yan, W., Cao, M., & Xue, D. (2019). Distributed multi-vehicle task assignment in a time-invariant drift field with obstacles. *IET Control Theory & Applications, 13*, 2886–2893.

Bakhshipour, M., Ghadi, M. J., & Namdari, F. (2017). Swarm robotics search & rescue: A novel artificial intelligence-inspired optimization approach. *Applied Soft Computing, 57*, 708–726.

Berning, A. W., Girard, A., Kolmanovsky, I., & D'Souza, S. N. (2020). Rapid uncertainty propagation and chance-constrained path planning for small unmanned aerial vehicles. *Advanced Control for Applications: Engineering and Industrial Systems, 2*, Article e23.

Bormann, R., Jordan, F., Hampp, J., & Hägele, M. (2018). Indoor coverage path planning: Survey, implementation, analysis. In *In 2018 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 1718–1725). https://doi.org/10.1109/ICRA.2018.8460566

Boroujeni, Z., Goehring, D., Ulbrich, F., Neumann, D., & Rojas, R. (2017). Flexible unit a-star trajectory planning for autonomous vehicles on structured road maps. In Vehicular Electronics and Safety (ICVES), 2017 IEEE International Conference on (pp. 7–12). IEEE.

Burger, C., & Lauer, M. (2018). Cooperative multiple vehicle trajectory planning using miqp. In 2018 21st International Conference on Intelligent Transportation Systems (ITSC) (pp. 602–607). IEEE.

Cai, K., Wang, C., Cheng, J., De Silva, C. W., & Meng, M. Q.-H. (2020). Mobile robot path planning in dynamic environments: A survey. arXiv preprint arXiv:2006.14195.

Cao, Y. U., Fukunaga, A. S., & Kahng, A. (1997). Cooperative mobile robotics: Antecedents and directions. *Autonomous Robots, 4*, 7–27.

Chen, X., Li, G.-y., & Chen, X.-m. (2017). Path planning and cooperative control for multiple uavs based on consistency theory and voronoi diagram. In 2017 29th Chinese Control And Decision Conference (CCDC) (pp. 881–886). IEEE.

Chen, X., Zhang, X., Huang, W., Liu, S., & Dai, H. (2019). Coordinated optimal path planning of multiple substation inspection robots based on conflict detection. In 2019 Chinese Automation Congress (CAC) (pp. 5069–5074). IEEE.

Costa, M. M., & Silva, M. F. (2019). A survey on path planning algorithms for mobile robots. In *In 2019 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)* (pp. 1–7). https://doi.org/10.1109/ICARSC.2019.8733623

Couceiro, M. S., Portugal, D., Ferreira, J. F., & Rocha, R. P. (2019). Semfire: Towards a new generation of forestry maintenance multi-robot systems. In 2019 IEEE/SICE International Symposium on System Integration (SII) (pp. 270–276). IEEE.

Cruz, D. L., & Yu, W. (2017). Path planning of multi-agent systems in unknown environment with neural kernel smoothing and reinforcement learning. *Neurocomputing, 233*, 34–42.

Cruz-Martin, A. M., et al. (2004). *Planificación de trayectorias en sistemas multirrobot*.

Cui, R., Li, Y., & Yan, W. (2016). Mutual information-based multi-auv path planning for scalar field sampling using multidimensional rrt. *IEEE Transactions on Systems, Man, and Cybernetics: Systems, 46*, 993–1004.

David, J., Valencia, R., & Iagnemma, K. (2016). Task assignment and trajectory planning in dynamic environments for multiple vehicles. In *RSS 2016 Workshop on Task and Motion Planning.* Ann Arbor: Michigan, USA.

Debord, M., Hönig, W., & Ayanian, N. (2018). Trajectory planning for heterogeneous robot teams. In 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (pp. 7924–7931). IEEE.

Dewangan, R. K., Shukla, A., & Godfrey, W. W. (2017). Survey on prioritized multi robot path planning. In 2017 IEEE International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM) (pp. 423–428).https://doi.org/10.1109/ICSTM.2017.8089197.

Erokhin, A., Erokhin, V., Sotnikov, S., & Gogolevsky, A. (2018). Optimal multi-robot path finding algorithm based on a. In Proceedings of the Computational Methods in Systems and Software (pp. 172–182). Springer.

Fung, N., Rogers, J., Nieto, C., Christensen, H. I., Kemna, S., & Sukhatme, G. (2019). Coordinating multi-robot systems through environment partitioning for adaptive informative sampling. In 2019 International Conference on Robotics and Automation (ICRA) (pp. 3231–3237). IEEE.

Gayathri, R., & Uma, V. (2018). Ontology based knowledge representation technique, domain modeling languages and planners for robotic path planning: A survey. *ICT Express, 4*, 69–74.

Glasius, R., Komoda, A., & Gielen, S. C. (1995). Neural network dynamics for path planning and obstacle avoidance. *Neural Networks, 8*, 125–133.

Goerzen, C., Kong, Z., & Mettler, B. (2010). A survey of motion planning algorithms from the perspective of autonomous uav guidance. *Journal of Intelligent and Robotic Systems, 57*, 65.

Gregory, J., Fink, J., Stump, E., Twigg, J., Rogers, J., Baran, D., Fung, N., & Young, S. (2016). Application of multi-robot systems to disaster-relief scenarios with limited communication. In *In Field and Service Robotics* (pp. 639–653). Springer.

Guzmán, M. A., Peña, C., et al. (2013). Algoritmos bioinspirados en la planeación off-line de trayectorias de robots seriales. *Visión electrónica, 7*, 27–39.

Habibi, G., Kingston, Z., Xie, W., Jellins, M., & McLurkin, J. (2015). Distributed centroid estimation and motion controllers for collective transport by multi-robot systems. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on* (pp. 1282–1288). IEEE.

Habibi, G., Xie, W., Jellins, M., & McLurkin, J. (2016). Distributed path planning for collective transport using homogeneous multi-robot systems. In *Distributed Autonomous Robotic Systems* (pp. 151–164). Springer.

Han, Z., Wang, D., Liu, F., & Zhao, Z. (2017). Multi-agv path planning with double-path constraints by using an improved genetic algorithm. *PloS One, 12*.

Hassan, A. M., Elias, C. M., Shehata, O. M., & Morgan, E. I. (2017). A global integrated artificial potential field/virtual obstacles path planning algorithm for multi-robot system applications. *International Research Journal of Engineering and Technology, 4*, 1198–1204.

Hedrick, G., Ohi, N., & Gu, Y. (2020). Terrain-aware path planning and map update for mars sample return mission. *IEEE Robotics and Automation Letters, 5*, 5181–5188. https://doi.org/10.1109/LRA.2020.3005123

Hönig, W., Preiss, J. A., Kumar, T. S., Sukhatme, G. S., & Ayanian, N. (2018). Trajectory planning for quadrotor swarms. *IEEE Transactions on Robotics, 34*, 856–869.

Huang, Y., Chiba, R., Arai, T., Ueyama, T., & Ota, J. (2015). Robust multi-robot coordination in pick-and-place tasks based on part-dispatching rules. *Robotics and Autonomous Systems, 64*, 70–83.

Injarapu, A. S. H. H. V., & Gawre, S. K. (2017). A survey of autonomous mobile robot path planning approaches. In *In 2017 International Conference on Recent Innovations in Signal processing and Embedded Systems (RISE)* (pp. 624–628). https://doi.org/10.1109/RISE.2017.8378228

Innocente, M. S., & Grasso, P. (2019). Self-organising swarms of firefighting drones: Harnessing the power of collective intelligence in decentralised multi-robot systems. *Journal of Computational Science, 34*, 80–101.

Jose, K., & Pratihar, D. K. (2016). Task allocation and collision-free path planning of centralized multi-robots system for industrial plant inspection using heuristic methods. *Robotics and Autonomous Systems, 80*, 34–42.

Kim, J., & Son, H. I. (2020). A voronoi diagram-based workspace partition for weak cooperation of multi-robot system in orchard. *IEEE Access, 8*, 20676–20686.

Kushleyev, A., Mellinger, D., Powers, C., & Kumar, V. (2013). Towards a swarm of agile micro quadrotors. *Autonomous Robots, 35*, 287–300.

Lal, R., Sharda, A., & Prabhakar, P. (2017). Optimal multi-robot path planning for pesticide spraying in agricultural fields. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)* (pp. 5815–5820). IEEE.

LaValle, S. M. (2006). *Planning algorithms*. Cambridge University Press.

Lazimy, R. (1982). Mixed-integer quadratic programming. *Mathematical Programming, 22*, 332–349.

Le, A. V., Prabakaran, V., Sivanantham, V., & Mohan, R. E. (2018). Modified a-star algorithm for efficient coverage path planning in tetris inspired self-reconfigurable robot with integrated laser sensor. *Sensors, 18*, 2585.

Long, P., Fanl, T., Liao, X., Liu, W., Zhang, H., & Pan, J. (2018). Towards optimally decentralized multi-robot collision avoidance via deep reinforcement learning. In *In 2018 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 6252–6259). IEEE.

Luis, C. E., Vukosavljev, M., & Schoellig, A. P. (2020). Online trajectory generation with distributed model predictive control for multi-robot motion planning. *IEEE Robotics and Automation Letters, 5*, 604–611.

Luna, R., & Bekris, K. E. (2011). Efficient and complete centralized multi-robot path planning. In *In Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on* (pp. 3268–3275). IEEE.

Mac, T. T., Copot, C., Tran, D. T., & De Keyser, R. (2017). A hierarchical global path planning approach for mobile robots based on multi-objective particle swarm optimization. *Applied Soft Computing, 59*, 68–76.
Madridano, Á., Al-Kaff, A., Gómez, D. M., & de la Escalera, A. (2019). Multi-path planning method for uavs swarm purposes. In *2019 IEEE International Conference of Vehicular Electronics and Safety (ICVES)* (pp. 1–6). IEEE.
Madridano, Á., Al-Kaff, A., Martín, D., et al. (2020). 3d trajectory planning method for uavs swarm in building emergencies. *Sensors, 20*, 642.
MahmoudZadeh, S., Powers, D. W., Yazdani, A. M., Sammut, K., & Atyabi, A. (2018). Efficient auv path planning in time-variant underwater environment using differential evolution algorithm. *Journal of Marine Science and Application, 17*, 585–591.
Marchant, W., & Tosunoglu, S. (2016). Rethinking wildfire suppression with swarm robotics. In Proceedings of the 29th Florida Conference on Recent Advances in Robotics, FCRAR (pp. 12–13).
Mathew, N., Smith, S. L., & Waslander, S. L. (2015). Optimal path planning in cooperative heterogeneous multi-robot delivery systems. In *Algorithmic Foundations of Robotics XI* (pp. 407–423). Springer.
Mellinger, D., Kushleyev, A., & Kumar, V. (2012). Mixed-integer quadratic program trajectory generation for heterogeneous quadrotor teams. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on* (pp. 477–483). IEEE.
Mouradian, C., Sahoo, J., Glitho, R. H., Morrow, M. J., & Polakos, P. A. (2017). A coalition formation algorithm for multi-robot task allocation in large-scale natural disasters. In *In Wireless Communications and Mobile Computing Conference (IWCMC), 2017 13th International* (pp. 1909–1914). IEEE.
Nazarahari, M., Khanmirza, E., & Doostie, S. (2019). Multi-objective multi-robot path planning in continuous environment using an enhanced genetic algorithm. *Expert Systems with Applications, 115*, 106–120.
Omidshafiei, S., Agha-Mohammadi, A.-A., Amato, C., Liu, S.-Y., How, J. P., & Vian, J. (2017). Decentralized control of multi-robot partially observable markov decision processes using belief space macro-actions. *The International Journal of Robotics Research, 36*, 231–258.
Peng, J.-H., Li, I.-H., Chien, Y.-H., Hsu, C.-C., & Wang, W.-Y. (2015). Multi-robot path planning based on improved d* lite algorithm. In *2015 IEEE 12th International Conference on Networking, Sensing and Control* (pp. 350–353). IEEE.
Preiss, J. A., Hönig, W., Ayanian, N., & Sukhatme, G. S. (2017). Downwash-aware trajectory planning for large quadrotor teams. In *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on* (pp. 250–257). IEEE.
Qie, H., Shi, D., Shen, T., Xu, X., Li, Y., & Wang, L. (2019). Joint optimization of multi-uav target assignment and path planning based on multi-agent reinforcement learning. *IEEE Access, 7*, 146264–146272.
Ratliff, N., Zucker, M., Bagnell, J. A., & Srinivasa, S. (2009). Chomp: Gradient optimization techniques for efficient motion planning. In *2009 IEEE International Conference on Robotics and Automation* (pp. 489–494). IEEE.
Roldán, J. J., Garcia-Aunon, P., Garzón, M., de León, J., del Cerro, J., & Barrientos, A. (2016). Heterogeneous multi-robot system for mapping environmental variables of greenhouses. *Sensors, 16*, 1018.
Şahin, E. (2004). Swarm robotics: From sources of inspiration to domains of application. In *International workshop on swarm robotics* (pp. 10–20). Springer.
Solana, Y., Furci, M., Cortés, J., & Franchi, A. (2017). Multi-robot path planning with maintenance of generalized connectivity. In *2017 International Symposium on Multi-Robot and Multi-Agent Systems (MRS)* (pp. 63–70). IEEE.
Solovey, K., Salzman, O., & Halperin, D. (2015). *Finding a needle in an exponential haystack: Discrete rrt for exploration of implicit roadmaps in multi-robot motion planning. In Algorithmic Foundations of Robotics XI* (pp. 591–607). Springer.
Song, B. D., Kim, J., & Morrison, J. R. (2016). Rolling horizon path planning of an autonomous system of uavs for persistent cooperative service: Milp formulation and efficient heuristics. *Journal of Intelligent & Robotic Systems, 84*, 241–258.
Spurny, V., Baca, T., & Saska, M. (2016). Complex manoeuvres of heterogeneous mav-ugv formations using a model predictive control. In 2016 21st International Conference on Methods and Models in Automation and Robotics (MMAR) (pp. 998–1003). IEEE.
Stentz, A. (1997). Optimal and efficient path planning for partially known environments. In *Intelligent Unmanned Ground Vehicles* (pp. 203–220). Springer.
Sun, G., Zhou, R., Di, B., Dong, Z., & Wang, Y. (2019). A novel cooperative path planning for multi-robot persistent coverage with obstacles and coverage period constraints. *Sensors, 19*, 1994.
Sun, J., Tang, J., & Lao, S. (2017). Collision avoidance for cooperative uavs with optimized artificial potential field algorithm. *IEEE Access, 5*, 18382–18390.
Tardós, J., Aragues, R., Sagüés, C., & Rubio, C. (2018). Simultaneous deployment and tracking multi-robot strategies with connectivity maintenance. *Sensors, 18*, 927.
Thomas, H. C., Charles, E. L., Ronald, L. R., & Clifford, S. (2001). Section 24.3: Dijkstra's algorithm. In *Introduction to Algorithms* (pp. 595–601). MIT Press and McGraw-Hill.
Turanli, M., & Temeltas, H. (2017). Adaptive coverage control with guaranteed power voronoi diagrams. In *2017 4th International Conference on Systems and Informatics (ICSAI)* (pp. 7–13). IEEE.
Wawerla, J., & Vaughan, R. T. (2010). A fast and frugal method for team-task allocation in a multi-robot transportation system. In ICRA (pp. 1432–1437).
Wei, H.-X., Mao, Q., Guan, Y., & Li, Y.-D. (2017). A centroidal voronoi tessellation based intelligent control algorithm for the self-assembly path planning of swarm robots. *Expert Systems with Applications, 85*, 261–269.
Wu, P., Li, L., Xie, J., & Chen, J. (2020). Probabilistically guaranteed path planning for safe urban air mobility using chance constrained rrt. In *AIAA AVIATION 2020 FORUM* (p. 2914).
Wu, Z., Su, W., & Li, J. (2019). Multi-robot path planning based on improved artificial potential field and b-spline curve optimization. In 2019 Chinese Control Conference (CCC) (pp. 4691–4696). IEEE.
Yan, Z., Jouandeau, N., & Ali-Chérif, A. (2012). Multi-robot heuristic goods transportation. In Intelligent Systems (IS), 2012 6th IEEE International Conference (pp. 409–414). Citeseer.
Yan, Z., Jouandeau, N., & Cherif, A. A. (2010). Sampling-based multi-robot exploration. In ISR/ROBOTIK (pp. 1–6).
Yan, Z., Jouandeau, N., & Cherif, A. A. (2013). A survey and analysis of multi-robot coordination. *International Journal of Advanced Robotic Systems, 10*, 399.
Yang, L., Qi, J., Song, D., Xiao, J., Han, J., & Xia, Y. (2016). Survey of robot 3d path planning algorithms. Journal of Control Science and Engineering, 2016.
Ying, Z., & Xu, L. (2015). Leader-follower formation control and obstacle avoidance of multi-robot based on artificial potential field. In Control and Decision Conference (CCDC), 2015 27th Chinese (pp. 4355–4360). IEEE.
Yu, X., Li, C., & Zhou, J. (2020). A constrained differential evolution algorithm to solve uav path planning in disaster scenarios. *Knowledge-Based Systems, 204*, Article 106209.
Zakiev, A., Tsoy, T., & Magid, E. (2018). Swarm robotics: Remarks on terminology and classification. In International Conference on Interactive Collaborative Robotics (pp. 291–300). Springer.
Zamuda, A., & Sosa, J. D. H. (2019). Success history applied to expert system for underwater glider path planning using differential evolution. *Expert Systems with Applications, 119*, 155–170.
Zhen, X., Enze, Z., & Qingwei, C. (2020). Rotary unmanned aerial vehicles path planning in rough terrain based on multi-objective particle swarm optimization. *Journal of Systems Engineering and Electronics, 31*, 130–141.
Zhou, Y., Yu, X., Sun, C., & Yu, W. (2015). Higher order finite-time consensus protocol for heterogeneous multi-agent systems. *International Journal of Control, 88*, 285–294.