frontiers | Frontiers in Robotics and AI

TYPE Original Research
PUBLISHED 13 August 2024
DOI 10.3389/frobt.2024.1375393

OPEN ACCESS

EDITED BY
Anders Lyhne Christensen,
University of Southern Denmark, Denmark

REVIEWED BY
Edmund R. Hunt,
University of Bristol, United Kingdom
Ayan Dutta,
University of North Florida, United States

*CORRESPONDENCE
Fatemeh Rekabi Bana,
✉ fatemeh.rekabi-bana@durham.ac.uk

RECEIVED 23 January 2024
ACCEPTED 17 July 2024
PUBLISHED 13 August 2024

CITATION
Rekabi Bana F, Krajník T and Arvin F (2024), Evolutionary optimization for risk-aware heterogeneous multi-agent path planning in uncertain environments.
Front. Robot. AI 11:1375393.
doi: 10.3389/frobt.2024.1375393

Evolutionary optimization for risk-aware heterogeneous multi-agent path planning in uncertain environments

Fatemeh Rekabi Bana[1]*, Tomáš Krajník[2] and Farshad Arvin[1]

[1]Swarm and Computational Intelligence Laboratory (SwaCIL), Department of Computer Science, Durham University, Durham, United Kingdom, [2]Artificial Intelligence Center, Faculty of Electrical Engineering, Czech Technical University in Prague, Prague, Czechia

Cooperative multi-agent systems make it possible to employ miniature robots in order to perform different experiments for data collection in wide open areas to physical interactions with test subjects in confined environments such as a hive. This paper proposes a new multi-agent path-planning approach to determine a set of trajectories where the agents do not collide with each other or any obstacle. The proposed algorithm leverages a risk-aware probabilistic roadmap algorithm to generate a map, employs node classification to delineate exploration regions, and incorporates a customized genetic framework to address the combinatorial optimization, with the ultimate goal of computing safe trajectories for the team. Furthermore, the proposed planning algorithm makes the agents explore all subdomains in the workspace together as a formation to allow the team to perform different tasks or collect multiple datasets for reliable localization or hazard detection. The objective function for minimization includes two major parts, the traveling distance of all the agents in the entire mission and the probability of collisions between the agents or agents with obstacles. A sampling method is used to determine the objective function considering the agents' dynamic behavior influenced by environmental disturbances and uncertainties. The algorithm's performance is evaluated for different group sizes by using a simulation environment, and two different benchmark scenarios are introduced to compare the exploration behavior. The proposed optimization method establishes stable and convergent properties regardless of the group size.

KEYWORDS

multi-agent, path planning, probabilistic roadmap, collision avoidance, genetic optimization, bio-hybrid systems

1 Introduction

Exploring animal–robot interactions involves the ambitious goal of comprehending how living species behave in the presence of robotic systems. Driven by a range of motivations, biologists and robotic researchers are compelled to investigate these hybrid dynamics, recognizing their potential to revolutionize biological–artificial integration (Romano et al., 2019). For instance, let us consider the concept of ecosystem hacking through a multi-robot system designed to influence honeybee colonies by interacting with the queen (Stefanec et al., 2022). Another example involves the utilization of a robotic hive to investigate collective behavior within this hybrid context concerning Western honeybees (Barmak et al., 2023). However, transition from the concept to real-world scenarios demands more sophisticated algorithm development, particularly for multi-agent systems that consider cooperation between the robots. For instance, the multi-arm robot presented in Rekabi-Bana et al. (2023), which is responsible for managing various tasks for the biological experiments described in Stefanec et al. (2022), requires an elaborate framework for decision-making, path planning, and motion control to achieve the expected goals. Although investigating the hybrid collaborative systems' behavior results in extensive challenges for algorithm design, the former applications that utilized multi-agent solution challenges (Zlot et al., 2002; Amigoni et al., 2017; Vinyals et al., 2019) can be inspiring to design new frameworks compatible with new problems. Those algorithms allow the formulation of diverse cooperative frameworks and evaluate various aspects of multi-agent systems to solve the complexities. In addition, utilization of a team of heterogeneous robots with different capabilities creates a more versatile system that manages different sensing and actuation tasks, although it will increase the control complications (Rizk et al., 2019). In similar multi-robot applications such as exploration missions in disaster relief or search and rescue, cooperating robots spread throughout the targeted area to obtain its map as quickly as possible, while satisfying various constraints, such as communication (Rouček et al., 2020) or physical couplings between robots (Rekabi-Bana et al., 2023). These scenarios often require the robots to operate in adverse conditions, such as rain and wind, and they also experience conditions of reduced visibility due to smoke, fire, or dust (Tranzatto et al., 2022). Furthermore, in cases such as pollution source localization (Štibinger et al., 2020) or underwater inspection (Xiang et al., 2010), the robotic team has to assume or keep a specific formation to exploit all agents' sensing capabilities. These scenarios require that the exploration methods take into account the environmental disturbances and uncertainties. Such scenarios have many features in common with the scenarios where robots should work in environments such as a hive, which is a confined and adverse environment. This paper presents a risk-aware path-planning algorithm that allows a group of robots to find safe trajectories, while considering the formation objectives for data collection in an adverse environment where actuation disturbances and sensor uncertainties affect the performance considerably.

1.1 Background

Various missions and objectives employ different approaches to describe the multi-agent system management problem (Dorri et al., 2018). For instance, Markov decision process (Liu et al., 2020); MAPF problem formulation, which is a layered optimization framework (Stern et al., 2019); and sampling-based approaches such as Monte Carlo simulation (Dalmasso et al., 2021) are some of the approaches that can describe the uncertainties in the environment properly and result in more robust solutions.

Although problem formulation is a key element to reach the main objectives, different methods might be applicable to solve those problems that become impossible to solve analytically in most cases according to the complexity order. Regardless of the problem formulation, two prevalent attitudes toward finding solutions are reinforcement learning (Qie et al., 2019; Semnani et al., 2020) and evolutionary optimization (Biswas et al., 2017; Li et al., 2021; Bahaidarah et al., 2024). Although the structures of both approaches are quite similar, various studies demonstrate differences between those methods to establish the advantages and weaknesses of both in different applications (Di Mario et al., 2013; Das et al., 2016). One of the best descriptions for the multi-robot path-planning problem is that it exploits combinatorial optimization to convert the discretized states into an efficient framework for optimization (Yu and LaValle, 2013; Okubo and Takahashi, 2023). Although the combinatorial problems allow considering multiple states and effective features regarding the mission objectives, the optimization search space becomes highly complicated and nonlinear. Some studies demonstrate that reinforcement learning is capable of solving such problems (Bengio et al., 2021). However, evolutionary optimization and particularly the genetic algorithm are a dominant methods for solving such problems according to their reliable heuristic behavior and convergence capability (Blum et al., 2011; Juan et al., 2015). Therefore, if a well-defined combinatorial optimization is used to describe the multi-agent path planning, the genetic algorithm will be a considerable approach for finding the solution. Although some research studies developed promising frameworks employing customized forms of the genetic algorithm to solve the pathfinding problem in 2D workspaces (Kala, 2012; Nazarahari et al., 2019) or independent coverage for large areas (Sun et al., 2019), many robotic applications in real-world scenarios demand considering other aspects such as uncertainties, robustness, and cooperative performance to fulfill their objectives. Therefore, the development of new optimization frameworks to investigate other practical objectives facilitates the applicability of employing reliable multi-agent robotic systems in real-world scenarios.

Path planning in an uncertain environment, which increases the risk of collision with the static or dynamic obstacle, is another challenging aspect that has been investigated in many recent studies (Rafai et al., 2022). One of the promising solutions to confront the uncertainties, particularly with the environment, is the risk-aware framework to avoid collisions with the obstacles (Pereira et al., 2013; Cai et al., 2021). However, such risk-aware frameworks can consider various types of risks in dynamic environments to address not only the collision but also other risk factors that cause mission failure (Barbosa et al., 2021).

On the other hand, the basic framework for pathfinding has a significant influence on the system's performance. Although many different approaches are studied for path planning in various applications (Galceran and Carreras, 2013), the probabilistic roadmap (PRM) algorithm (Kavraki et al., 1996) proposed a flexible foundation that is adaptable to a wide range of planning problem formulation and optimization techniques to make robust planning frameworks, particularly in uncertain environments (Agha-mohammadi et al., 2014; Ravankar et al., 2020).

1.2 Contributions

This paper proposes a novel form of genetic optimization to find a solution for a multi-agent robotic system employing a risk-aware PRM scheme that was presented in Rekabi-Bana et al. (2024) as a robust framework for one robot to explore an area efficiently with minimum failure risk. The proposed planning algorithm utilizes the genetic algorithm in conjunction with the single-agent exploration method to find a conflict-free solution that enforces the agents exploring all subdomains together. Each subdomain is considered a district in the exploration area that includes at least one exploration point. Moreover, it considers the risk of collision with the obstacles according to a risk assessment function obtained from *a priori* knowledge of the environment and the robot's dynamics. However, according to the problem definition, the risk assessment function can be defined to consider different sources of failure risk according to different scenarios. The proposed optimization framework allows incorporating the risk of failure according to the environmental conditions and, therefore, results in a reliable solution for exploration scenarios in workspaces with uncertainties. It allows the team to combine the data collected from each district to obtain as much information about the explored environment as possible. Therefore, the mission planning will include finding a proper formation shape for data collection, allocating the destinations to the agents, and collision-free pathfinding through the environment considering obstacle avoidance. The algorithm is capable of path planning in both 2D and 3D environments. Therefore, it will be applicable to use in the robotic system introduced in Rekabi-Bana et al. (2023) and small drones as well. Furthermore, the secondary objective is to develop a compatible path-planning algorithm with the distributed control framework designed for a multi-drone system (Rekabi-Bana et al., 2021) to make an entire autonomous robust framework for a group of flying robots to explore uncertain environments. The optimization scheme considers the traveling distance and the failure probability to determine the cost function. According to the problem definition, the exploration subdomain sequence, the agents' destination in each district, and the priority order to find a collision-free set of trajectories are considered effective parameters that should be optimized to reach the mission objective. The combinatorial nature of the problem implies a particular structure to efficiently find the optimum point. Therefore, a customized structure is introduced to generate the chromosomes, reproduce the population, and analyze the convergence in each generation. The algorithm was designed as a centralized motion planning framework and has no dependencies on the agents' communication. Therefore, this algorithm is applicable to cases where the robots are working close to the ground station and have a reliable network with it. The main contributions of this paper can be summarized as follows:

- Development of a multi-agent collision-free PRM-based path-planning algorithm for a group of robots working in an arbitrary 2D or 3D environment to explore the area cooperatively. The proposed algorithm allows the team to explore the area in a particular formation geometry for application in the multi-arm manipulator described in Rekabi-Bana et al., (2023).
- Proposing a new customized genetic framework to solve the path-planning problem as a combinatorial optimization. The proposed structure establishes a stable converging performance regardless of the number of agents in the multi-agent system.

The subsequent sections of the paper are structured as follows: Section 2 outlines the problem formulation, followed by Section 3 presenting the genetic algorithm framework developed to address the depicted problem. Section 4 explains the simulation environment and the evaluation procedure, and Section 5 demonstrates the results derived from the analysis. Finally, Section 6 provides the concluding remarks for the paper.

2 Problem formulation

The first step toward the algorithm design is describing the problem mathematically to formulate the objectives and constraints. This paper aims to develop a new framework for heterogeneous multi-agent path planning for a team of robots with different sensing capabilities exploring an uncertain environment. The robots should stay in a particular formation shape to maintain reliable communication, exploiting the measurement and actuation capabilities together, and comply with coupling constraints according to the robots' dependencies on each other. The collision-free multi-agent path planning for area exploration is described and formulated in this section. The exploration area, $\mathcal{A}_e \subset \mathbb{R}^3$, is described as an arbitrary three-dimensional space according to the following definition:

$$\begin{aligned}
&\mathcal{A}_e = \mathcal{F} \cup \mathcal{O}\\
&\mathcal{A}_e = \left\{ \begin{bmatrix} x_e & y_e & z_e \end{bmatrix}^T | x_e^{min} \leq x_e \leq x_e^{max},\right.\\
&\qquad \left. y_e^{min} \leq y_e \leq y_e^{max}, z_e^{min} \leq z_e \leq z_e^{max} \right\},
\end{aligned} \tag{1}$$

where $x_e^{min}, \ldots, z_e^{max}$ represents the workspace borders and $\mathcal{O}$ and $\mathcal{F}$ demonstrate the occupied and free space, respectively.

It is assumed that the workspace for the exploration is mapped using the risk-aware PRM method and according to the available data set from the environment, including the workspace borders and occupied areas' boundaries. Nevertheless, the proposed algorithm in Rekabi-Bana et al. (2024) needs few data, considers uncertainties of the environment data, and creates a pathfinding graph according to a predefined admissible risk to handle the collision risk with obstacles. Therefore, the environment graph is described as follows:

$$\begin{aligned}
&\mathcal{G}_e = \left(V_e, E_e\right)\\
&V_e = \left\{ v_k^e \in \mathcal{A}_e \right\}, \quad k = 1, \ldots, N_{prm}^{node}, \mathcal{R}\left(v_k^e\right) \leq R_{ad}\\
&E_e = \left\{ e_t^e = \left(v_i^e, v_j^e\right) \right\}, \quad t = 1, \ldots, N_{prm}^{edge}, \mathcal{R}\left(e_t^e\right) \leq R_{ad},
\end{aligned} \tag{2}$$

where $\mathcal{G}_e$ is the environment exploration graph; $v_k^e \in \mathbb{R}^3$ and $e_t^e \in \mathbb{R}^{3\times2}$ represent the graph nodes and graph edges, respectively; N_{prm}^{node} and N_{prm}^{edge} are the number of nodes and edges in the graph, respectively; R_{ad} is the admissible collision risk considered to calculate the PRM nodes and edges, and $\mathcal{R}(.)$ is the risk assessment function obtained according to Rekabi-Bana et al. (2024). The exploration objective is to cover a set of randomly distributed exploration points in the workspace to harvest information. The workspace is divided into districts according to the robots' sensors' capabilities for coverage. Each district includes some of the exploration points, and the robots should cover each of them cooperatively. According to the number of agents, the agents' sensing radius, and the exploration targets' distribution, the destination points form an equilateral polygon to place the agents in a formation shape that allows the agents to cover the exploration points as much as possible. Consequently, the formation shape at each district is defined according to the following equation:

$$\begin{aligned} &\mathcal{C}^d_{fr} = \bar{P}^d_{ex}\\ &\mathcal{R}^d_{fr} = \mathcal{F}^{\mathcal{R}}\left(\|\sigma\left(P^d_{ex}\right)\|, \bar{\mathcal{R}}^s\right)\\ &\alpha^d_{fr} = \angle\sigma\left(P^d_{ex}\right),\ d = 1, \ldots, N_{dist}, \end{aligned} \tag{3}$$

where $P^d_{ex} = \{p^d_{i_{ex}} \in \mathbb{R}^3\}$ is the set of exploration target points, $\mathcal{C}^d_{fr}$ is the center point according to P^d_{ex}, $\mathcal{R}^d_{fr}$ is the formation radius, α^d_{fr} is the formation polygon rotation angle, $R^s = \{r^s_i \in \mathbb{R}\}$ is the set of sensing radii according to the agents' measurement system capability, and N_{dist} is the number of districts for exploration. In addition, $\bar{(.)}$ is the mean value, $\|(.)\|$ is the norm, $\angle(.)$ is the vector angle, and $\sigma(.)$ is the variance operator. $\mathcal{F}^{\mathcal{R}}$ is a function used to determine the circumradius of the polygon and is defined as follows:

$$\mathcal{F}^{\mathcal{R}}\left(\|\sigma\left(P^d_{ex}\right)\|, \bar{\mathcal{R}}^s\right) = \begin{cases} \|\sigma\left(P^d_{ex}\right)\|, & \textbf{if}\ \ \|\sigma\left(P^d_{ex}\right)\| \le \bar{\mathcal{R}}^s\\ \dfrac{\bar{\mathcal{R}}^s + \|\sigma\left(P^d_{ex}\right)\|}{2}, & \textbf{if}\ \ \|\sigma\left(P^d_{ex}\right)\| \ge \bar{\mathcal{R}}^s. \end{cases} \tag{4}$$

The formation shape obtained from Equation 3 represents the robots' target positions, allowing them to access the exploration points as much as possible for data collection using this configuration. Figure 1 presents the graphical perspective of Equation 3. According to the aforementioned environment partitioning and target formations, the path-planning algorithm should find a set of collision-free paths in the environment graph to pass the agents through the safe area and prevent them from colliding. Additionally, the traveling distance for the group should be minimized, and the agents should go through each district once. Furthermore, placing the agents in the formation considering their sensing radius is important to maximize the number of exploration targets. Therefore, path planning can be defined as a constrained optimization problem described in the following equation:

$$\begin{aligned} &if\ \mathbf{\Pi} = \{\pi_i\}, \pi_i = \left\{\pi^d_i\right\} \subset \Phi\left(\mathcal{G}_e\right), \quad i = 1, \ldots, N_{ag}\\ &\mathbf{\Pi}^* = \mathbf{arg\,min}\left(\mathcal{J}_{mission}\right),\\ &\quad s.t.\ \pi_i \cap \pi_j = \emptyset\ \ \textbf{if}\ \ i \ne j,\ \pi_i \cap \mathcal{O} = \emptyset\\ &\quad d = 1, \ldots, N_{dist}, \end{aligned} \tag{5}$$

considering

$$\mathcal{J}_{mission} = \mathbb{L}\left(\mathbf{\Pi}\right) + \Sigma^{N_{dist}}_{d=1} \Sigma^{N^d_{ex}}_{i_{ex}=1} \lambda^d_{i_{ex}} \mathbb{S}\left(p^d_{i_{ex}}\right), \tag{6}$$

$$\mathbb{L}\left(\mathbf{\Pi}\right) = \gamma \mathbb{P}\left(\Pi\right) + \Sigma^{N_{dist}}_{d=1} \Sigma^{N_{ag}}_{i=1} D\left(\pi^d_i\right),$$

$$\mathbb{S}\left(p^d_{i_{ex}}\right) = \begin{cases} 0 & \textbf{if}\ \ \mathbf{min}\left(\|p^d_{i_{ex}} - q^d_i\|\right) \le r^s_i\\ 1 & \textbf{otherwise} \end{cases}, \quad i = 1, \ldots, N_{ag},$$

where π_i is the path for the i_{th} agent, π^d_i is part of π_i that allows the i_{th} agent to reach its destination in d_{th} districts, $\Phi(\mathcal{G}_e)$ is the set of all possible paths in the environment graph, $\mathcal{J}_{mission}$ is the mission overall cost, $D(\pi^d_i)$ is the distance traveled by the i_{th} agent through π^d_i, $\mathbb{P}(\pi_i)$ is the failure probability for Π implementation regarding the collision between the agents or with obstacles, γ is the importance factor for the failure probability, q^d_i is the destination point for the i_{th} agent in the d_{th} district, and $\lambda_{i^d_{ex}}$ is the importance weight for each $p^d_{i_{ex}}$ exploration point. According to Equations 5, 6, the solution of the problem includes a set of paths that belong to the possible paths in the environment graph which do not undergo any collision with each other. On the other hand, the agents should reach a target formation in each district, and therefore, they should use paths that might be close to each other. Consequently, making a set of collision-free paths needs to prevent use of the same nodes and edges for two agents simultaneously. Therefore, in this paper, the proposed algorithm for path planning considers three main parameters that affect the mission cost:

- The order of districts that the agents should pass through to explore the area.
- The destination point of each agent in the formation.
- The order of the path association in the exploration graph for the agents.

The procedure to solve the problem, defined in Equations 5, 6, using genetic optimization is described in the next section.

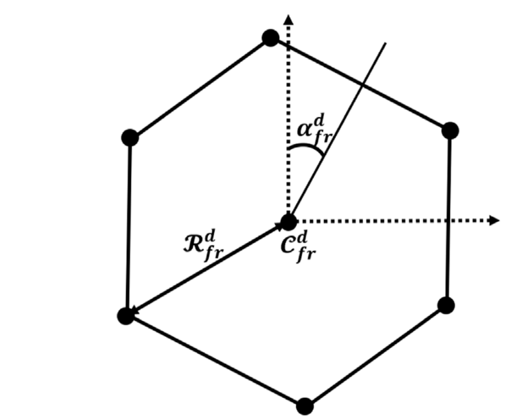


FIGURE 1
Formation shape for six agents according to the definitions in Equation 3.

3 GA multi-agent path planning

The path-planning algorithm is a combinatorial optimization problem. The final set of paths includes two main features: 1) the best combination of the districts that the agents should pass through and 2) the best mapping from the agents' IDs and the destination points in the target formation. Furthermore, the paths should have no similar nodes and edges so as to reduce the collision risk between the agents. In addition, it is necessary to check if the final trajectories through all the waypoints are collision-free. The proposed strategy to find collision-free paths in the graph is explained in Algorithm 1. Therefore, those features should be considered in the genetic algorithm to find the solution.

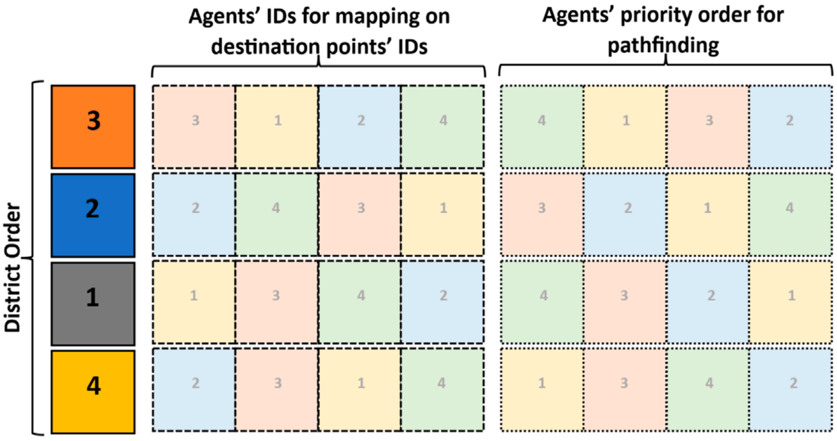
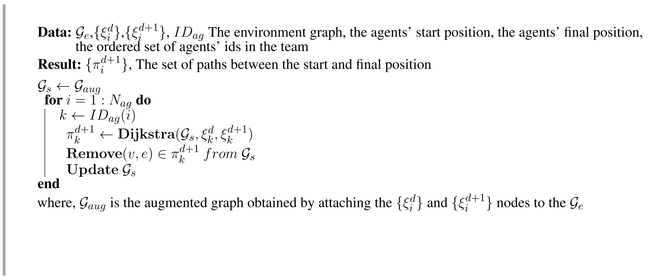

Agents' IDs for mapping on destination points' IDs
Agents' priority order for pathfinding
District Order
3
2
1
4
3 1 2 4 4 1 3 2
2 4 3 1 3 2 1 4
1 3 4 2 4 3 2 1
2 3 1 4 1 3 4 2

FIGURE 2
A sample chromosome (for four exploration subdomains and four agents) to demonstrate the structure defined to solve the collision-free multi-agent pathfinding.



```
Data: $\mathcal{G}_e$,$\{\xi_i^d\}$,$\{\xi_i^{d+1}\}$, $ID_{ag}$ The environment graph, the agents' start position, the agents' final position,
      the ordered set of agents' ids in the team
Result: $\{\pi_i^{d+1}\}$, The set of paths between the start and final position
$\mathcal{G}_s \leftarrow \mathcal{G}_{aug}$
for $i = 1 : N_{ag}$ do
    $k \leftarrow ID_{ag}(i)$
    $\pi_k^{d+1} \leftarrow$ Dijkstra$(\mathcal{G}_s, \xi_k^d, \xi_k^{d+1})$
    Remove$(v, e) \in \pi_k^{d+1}$ $from$ $\mathcal{G}_s$
    Update $\mathcal{G}_s$
end
where, $\mathcal{G}_{aug}$ is the augmented graph obtained by attaching the $\{\xi_i^d\}$ and $\{\xi_i^{d+1}\}$ nodes to the $\mathcal{G}_e$
```

Algorithm 1. Collision-free pathfinding.

3.1 Genetic population

According to the features defined for the final solution, the suggested chromosome structure is composed of the district identifiers, the agents' identifiers allocated to the destination points, and the order of agents' identifiers for collision-free pathfinding. Therefore, each chromosome can be described as a matrix that is structured as shown in Figure 2.

According to the proposed structure, the initial population is created by generating N_{pop} random permutations for the districts' sequence, agents' identifiers for mapping to the destination points, and the sequence for pathfinding in the graph. The population size is determined according to Ahn and Ramakrishna (2002) to comply with the nature of the problem.

3.2 Evolutionary regeneration

The first reproduction method is to transfer the N_{el} best chromosomes directly to the next generation. However, the proposed 2D chromosome structure and the combinatorial nature of the problem do not allow for using popular strategies for regeneration through crossover and mutation. Therefore, two new strategies are proposed and applied to create the next generation from the current parents. The first strategy is a one-parent crossover according to the following steps:

- Selection of N_{crs} random chromosomes from the current population as the crossover parents.
- For each selected chromosome, one or two random breakpoints for the first column and a permutation are determined to reorder the districts' sequence.
- For each selected chromosome, two or three random columns will be selected among the next N_{ag} columns (depending on the number of agents), and their position will be switched according to a random permutation to rearrange the agents' destination points in the target formations.
- For each selected chromosome, two random columns are selected among the last N_{ag} columns, and their position will be switched to reorder the agents' priority for pathfinding in the graph.

Considering the above procedure, each child will inherit some of the parents' characteristics and shows new behavior. Furthermore, the parents are selected randomly regardless of their achieved cost function in the current generation to prevent the algorithm from following the current best population members and preserve the algorithm's heuristic performance. In addition, a proposed mutation is used to generate $N_{mut} = N_{pop} - (N_{el} + N_{crs})$ remaining children to complete the next-generation chromosomes. The proposed mutation works as follows:

- For each selected chromosome, a random number between 1 and 3 is generated to determine which part of the chromosome should be mutated.
- If the first part is selected, then the current district sequence will be replaced by a random permutation for the districts' identifier to generate a completely different one.
- If the second part is chosen, then a random row will be designated and the mapping order will be replaced with a random permutation of the agents' identifiers.

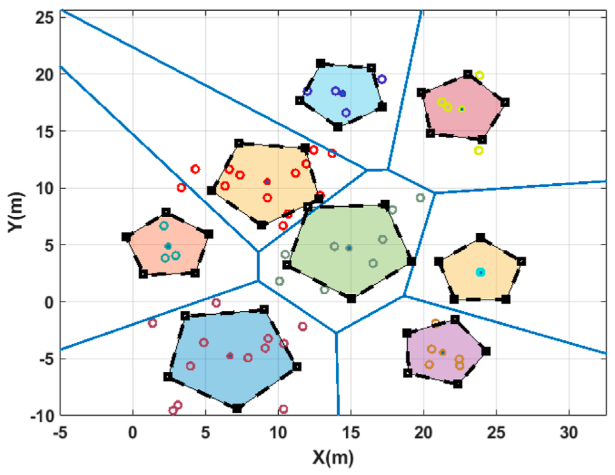


FIGURE 3
2D representation of a mission for a group of five robots in an arbitrary workspace. Blue lines show the borders for the subdomains, circular points demonstrate the exploration points, and square points and dashed lines depict the formation shape boundaries and the destination points in each district, respectively, which are represented by different colors.

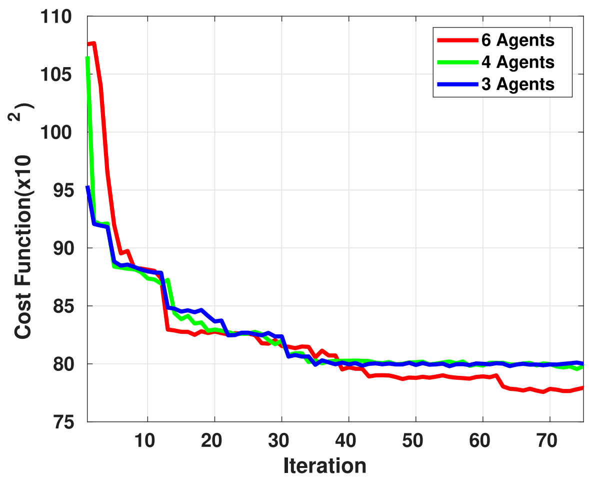


FIGURE 4
Best value of the cost function in the genetic optimization obtained for three, four, and six agents in the exploration team.

- If the third part is chosen, a random row will be selected and the pathfinding priority order will be replaced with a random permutation of the agents' identifiers.

The further important parameters significantly affecting the algorithm's behavior are the number of chromosomes that will be selected to pass through the elite, crossover, or mutation process for regeneration. In this paper, the following process is proposed to determine those parameters:

- N_{el} is a constant value, and it is equivalent to the top 5% of the population.
- N_{crs} is variable to cover between 80% and 90% of the population. Its value depends on the population density. The higher the population density, the lower the N_{crs} to allow mutation to reinforce the heuristic features.
- The number of chromosomes for the mutation will be obtained from $N_{mut} = N_{pop} - (N_{el} + N_{crs})$ to keep the population size over generations.

3.3 Convergence criteria

There is no theoretical method to establish the final convergence for the genetic algorithm in an arbitrary problem. Therefore, it is necessary to stop the algorithm while it satisfies a set of predefined conditions. Different types of convergence analysis are applicable to the genetic algorithms in different problems (Safe et al., 2004). In this paper, the following criteria are considered to determine the algorithm's convergence:

- Elite group sustainability: if the chromosomes that reached the top 5% remain in their positions for at least 20 generations, it means the search is converged, and it is more likely for the algorithm to stay on the same point as the optimum solution in the next generations.
- The cost function gradient: according to the combinatorial nature of the problem, it is possible to have more than one solution as the best combination of features to satisfy the minimum value for the cost function. Therefore, it is necessary to consider another condition rather than an elite set of chromosome sustainability. The cost function gradient is another condition that demonstrates the algorithm's convergence. In this paper, if the cost function gradient remains less than 2% for 50 generations, it means the algorithm has reached its final convergence.
- Population distribution: checking the population distribution is not very easy in combinatorial problems, particularly with the 2D chromosome structure. Accordingly, a new strategy is proposed to compare the current population with the previous generations to check how many new points in the search domain are considered in the current generation. For this reason, a random key is generated in the beginning to select particular genes from the chromosomes and devise a comparison feature to evaluate the heuristic performance level. Accordingly, if the number of new chromosomes remains less than 2% of the population size for more than 20 generations, it means the algorithm has reached its final convergence.
- The maximum iteration: a maximum number of iterations equivalent to 10 times the population size is considered to prevent computational overloading.

4 Simulation environment and evaluation

The performance evaluation for the proposed algorithm is accomplished using a simulation environment presented in our previous works (Rekabi-Bana et al., 2020; 2021). Furthermore, to guarantee cooperative stability and robustness against uncertainties

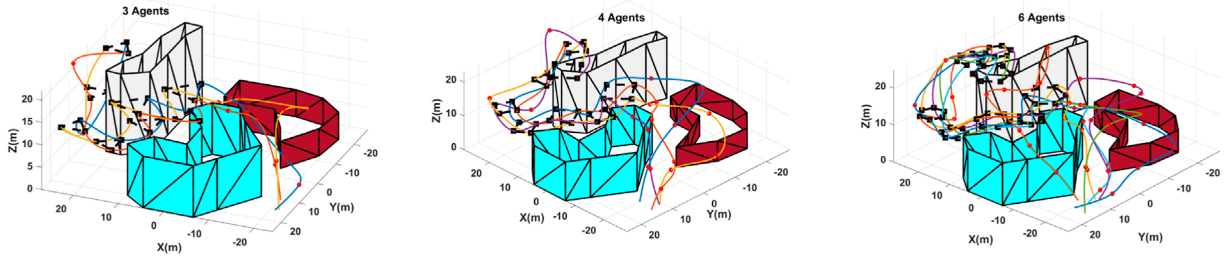


FIGURE 5
Path-planning results obtained from the proposed genetic optimization framework for three, four, and six agents in an arbitrary workspace. The obstacles' boundaries are shown by gray, brown, and blue surfaces; the target formations are depicted by square points and dashed lines; the path waypoints are presented by red dots; and the final trajectories are demonstrated by colored lines.

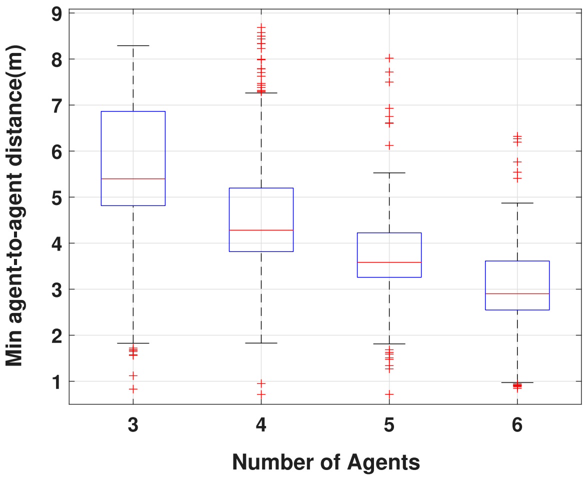


FIGURE 6
Collision-free behavior evaluation using the minimum agent-to-agent distance in four team configurations.

and disturbances, the distributed NLH_{∞} presented in Rekabi-Bana et al. (2021) is applied. The test scenario includes an arbitrary workspace with three obstacles. The procedure to map the environment and create a risk-aware exploration graph is similar to that mentioned in our previous work, which is discussed in Rekabi-Bana et al. (2024).

Furthermore, in order to compare the algorithm's behavior for different group sizes, four conditions are studied. The group size varies from three to six agents. The population size for the optimization is determined as 20 times the number of agents. That value is obtained by a trade-off between the computational burden and the heuristic performance of the algorithm. In each iteration, a set of 15 simulations is implemented considering the robots' dynamics and environmental disturbances to evaluate the traveling distance and failure probability due to collision between the agents or the obstacles. Furthermore, the number of exploration points covered according to the agents' sensing radius and their destination position in the formation is determined to calculate the second part of the cost function. Accordingly, the cost function will be calculated for each chromosome based on those values afterward.

In order to compare the proposed algorithm's performance with that of other planning schemes, two other methods are considered. It is assumed that there is no possibility of collision with the obstacles, and the agents can fly between different formations directly in both comparison algorithms. In the first scenario, an ideal exploration is considered, which includes the shortest path between all the destination points. In the second case, the random search is considered for the group, which is a classic method for exploration. The coverage score is defined to consider the number of exploration points in each district proportionally and the points scattering diversely. It means that the districts with more exploration points which are less sparse have the highest score. It is evident that if the agents cover all the districts, they will achieve the highest score.

Figure 3 demonstrates one of the sample missions for performance evaluation. It depicts how the exploration subdomains are formed and how the formation shape will look according to the exploration points' distribution in each subdomain.

5 Results and discussion

The results obtained from the proposed algorithm implementation in the simulation environment are discussed in this section. According to the problem's definition in Section 2, the preliminary stage before the optimization involves determining the target formations in each district to place the agents in the vicinity of exploration points for data collection. A sample for exploration points' distribution, workspace partitioning, and target formations is presented in Figure 3.

5.1 Exploration with aerial robots

As is presented in Figure 1, the formation shape is considered an equilateral polygon regarding the number of agents in the exploration team. The size and orientation of the polygons are determined according to the exploration points' distribution in each district and the data collection radius for the agents. Therefore, the more sporadic the exploration points, the larger polygon will be considered to place the agents in positions for a complete coverage.

The results obtained for the cost function's best value at each generation are presented in Figure 4. The results demonstrated

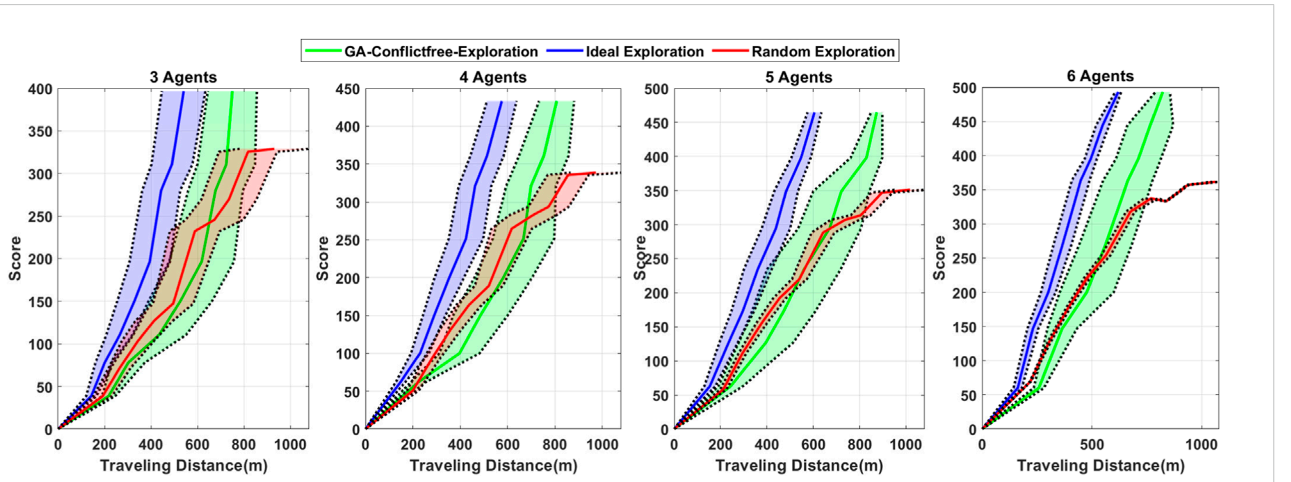


FIGURE 7
Comparison results for the proposed algorithm exploration performance in three group configurations. The solid lines represent the average performance, and the shaded area with dotted boundary lines demonstrates the 25% and 75% quantiles for the statistical results.

TABLE 1 Algorithm run-time in minutes for different scenarios in 3D and 2D environments with PRM graphs of 2,663 and 12,812 edges respectively.

	Number of districts			
	10	8	7	5
Three agents	15.34	12.22	10.68	7.63
Four agents	20.31	16.39	14.34	10.16
Five agents	25.53	20.36	17.83	15.14
Six agents	30.54	24.47	21.43	15.26
	Number of districts			
	10	7	6	5
Three agents	8.30	6.10	5.24	4.36
Four agents	12.72	7.95	6.86	5.72
Five agents	13.90	9.84	8.57	7.06
Six agents	16.62	11.76	10.09	8.45

3D: PRM graph with 2,663 edges.
2D: PRM graph with 12,812 edges.

in Figure 4 show the first 75 iterations of the optimization for all the configurations. The result indicates that the algorithm structure is designed properly and the heuristic and optimization stability is not affected by the group size. However, the more agents in the group, the larger the optimization search domain, and accordingly, the more iteration is required to satisfy the convergence criteria. Furthermore, the simulation results obtained using the proposed planning algorithm for three different group sizes are presented in Figure 5, demonstrating the trajectories that each agent should pass to reach the allocated destination points in target formations.

Although Figure 5 demonstrates the proposed algorithm's output for path planning, it is necessary to analyze the minimum agent-to-agent distance criterion to evaluate the collision-free feature of the proposed multi-agent path-planning algorithm. The results obtained for that criterion in four team configurations are demonstrated in Figure 6. The results depicted in Figure 6 show that in all the studied conditions, the minimum agent-to-agent distance is more than almost 0.5 m, and no collision has occurred between the agents. Furthermore, it is evident that the variation bound of the distance between the agents decreases with increase in the number of agents. Although the presented results establish the capability of the proposed optimization for multi-agent collision-free path planning in an arbitrary environment, it is necessary to evaluate the proposed algorithm's performance in comparison with other methods. The comparison results with two benchmark scenarios based on the statistical data obtained from multiple simulations for different configurations are presented in Figure 7. The first benchmark is the ideal exploration, which means that the agents explore the area by moving directly between each pair of districts regardless of the existing obstacles and collision risk according to the best combination of the districts obtained according to a TSP problem. The second benchmark is the random exploration without any risk of collision with the obstacle and directly between districts, as considered in the first benchmark. According to the results demonstrated in Figure 7, the behavior of the proposed algorithm is similar to the ideal benchmarks considered for each configuration, although it is supposed to avoid collision with obstacles and find the trajectory through the PRM graph. Furthermore, the proposed algorithm outperforms the random exploration by preventing the agents from repeatedly visiting the districts, which is a problem with the random exploration method.

Although the proposed algorithm provides exploration plans similar to an ideal case, the constrained optimization framework requires a considerable computation resource. Algorithm speed was analyzed using its implementation in MATLAB 2022b on a computer with a 12th Gen Intel(R) Core(TM) i9-1200k processing unit and 64 Gb of RAM. The results obtained for different numbers

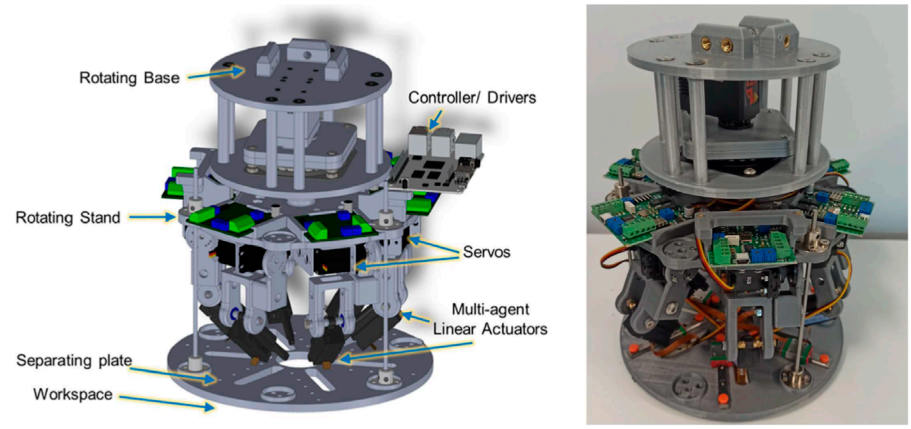


FIGURE 8
Multi-arm manipulator designed for data collection from the observation hive (Rekabi-Bana et al., 2023).

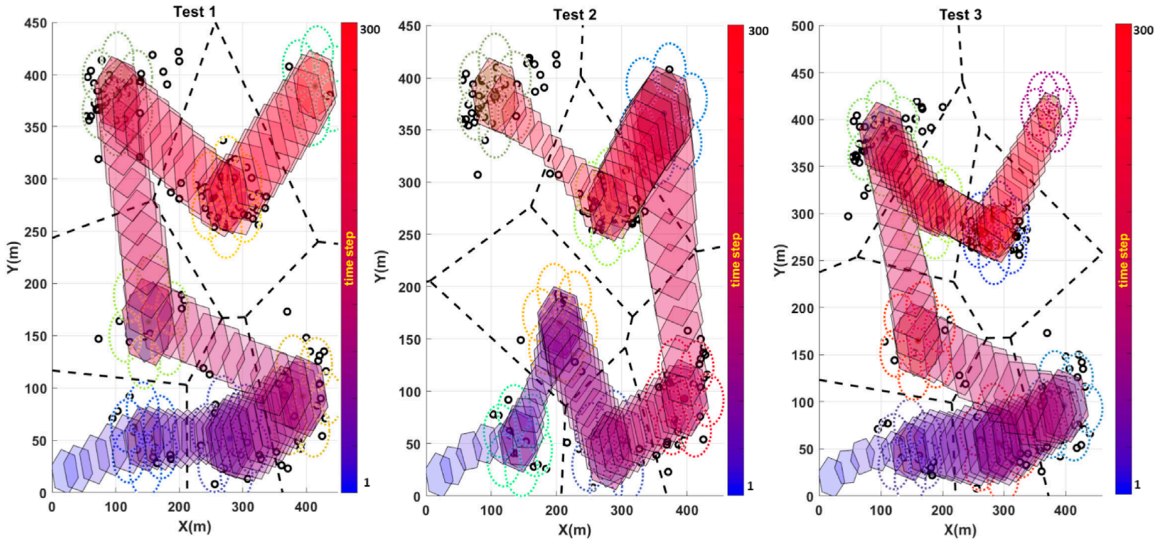


FIGURE 9
Results obtained for the multi-arm manipulator trajectory planning to cover egg-laying spots in three different sampling cases. The black circles represent the egg-laying spots, the black dashed lines represent the workspace partitioning according to the selected egg-laying samples, and the colored hexagon shows the footprint of the manipulator through time, starting from blue and shifting to red as time passes.

of agents, exploration districts, and environment dimensionality are presented in Table 1. It is clear that compared to the 3D environments, path planning in two-dimensional environments achieves relatively short run times, even with a significantly larger graph. The results also indicate that increasing the number of agents or districts increases the run time approximately linearly.

5.2 Exploration of a honeybee colony with a multi-arm manipulator

The proposed algorithm is also applied to determine a proper set of trajectories for the multi-arm robot designed for the RoboRoyale project (Rekabi-Bana et al., 2023). In this study, a set of robotic agents, attached to arms on the manipulator, will interact with the honeybee queen (Stefanec et al., 2022). Each agent will be equipped with a miniature sensor to retrieve information about the queen and the comb. The robot configuration is presented in Figure 8.

When the queen is active, the agents follow her and monitor her activity, and they detect where she lays the eggs. To assess the healthiness of the brood, the agents have to visit the locations of the previous egg-laying events and investigate the progress in the brood development. This requires that the agents leave the queen during the resting times and gather the observations of the brood as quickly as possible to minimize the risk of missing important queen activities. During the exploration, the agents have to avoid disturbing the colony and prevent collisions with any elements of the hive.

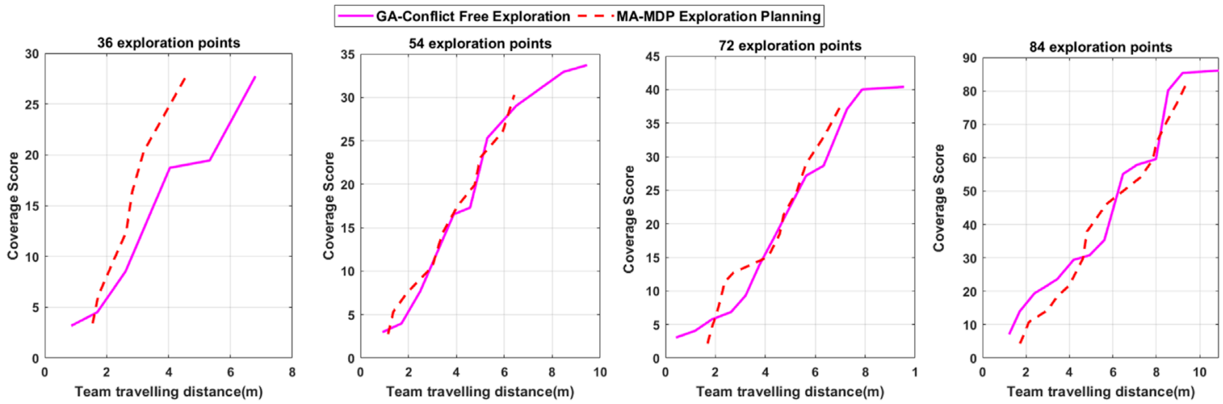


FIGURE 10
Comparison results in four scenarios with different numbers of exploration target points.

Our experiment is based on real egg-laying data collected with respect to a honeybee observation hive. There, an image processing system was used to determine a set of egg-laying events (Žampachů et al., 2022), and the events gathered over the 24-h period were considered target points for workspace partitioning and target formation determination. A geometrical constraint was also added to the optimization framework to comply with the robot's coupling and kinematic constraints. Accordingly, the obtained results for three sets of egg-laying spots are demonstrated in Figure 9.

The presented results in Figure 9 demonstrate the optimization results and workspace partitioning in three cases. In each case, 150 egg-laying samples are selected randomly to make a different distribution. Therefore, the workspace partitioning and each district value for exploration would be different according to a new sample distribution. Accordingly, the resulting trajectory becomes different in each case and complies with the new sample distribution. The robot's endpoint footprint presented in Figure 9 demonstrates how the algorithm attempts to find the best placement for the robot arms' endpoint according to the kinematic constraints. The robot arms' mechanical constraints make solid restrictions such as maximum and minimum diameter for expansion and retraction and the overall rotation, which do not allow for full coverage of target points in each domain. However, the proposed algorithm tries to attain the maximum exploration score by reaching the position to cover the maximum number of the target points at each district. The performance of our method is compared with that of an efficient algorithm recently proposed for a joint area coverage scenario with a team of robots (Nawaz and Ornik, 2023). The performance of both algorithms was evaluated in different conditions with different numbers of exploration target points sampled randomly from the egg-laying events. The results obtained from the comparison are shown in Figure 10. The results in Figure 10 show that the two algorithms achieve similar performance. However, our algorithm finds optimal efficient solutions considering additional criteria such as kinematic coupling and the conflict-free path between the agents, which are not considered in the MA-MDP framework proposed in Nawaz and Ornik (2023). Therefore, the comparison result establishes the efficiency and optimality of the algorithm as it complies with the practical requirements of the robots for the exploration.

5.3 Limitations and future works

According to the results presented in previous sections, it is evident that the proposed algorithm has some limitations, particularly regarding the run-time performance. The results demonstrated in Table 1 show that the current implementation framework is time-consuming and limited to applications that do not demand fast planning and re-planning. For instance, the studied case of a multi-arm manipulator for egg-laying coverage in an observation hive is one of the applications not needing prompt planning, and the proposed algorithm is a good solution for that application. However, implementing the algorithm with a more efficient programming system and employing parallel computation will enhance the performance and reduce the run-time to cover more practical robotic applications. Moreover, the centralized framework for the algorithm limited its applicability to those cases that rely on a ground station with a reliable communication link between the agents and the main processing unit. Therefore, developing a distributed form of the algorithm will be one of the future solutions that allow utilizing the proposed method for applications that have less reliance on the ground station.

6 Conclusion

This paper presents an innovative path-planning algorithm that leverages the risk-aware probabilistic roadmap (PRM) method, combined with a customized genetic optimization approach, which is compatible with the complexities of the path-planning problem. The algorithm prioritizes two key criteria: minimizing travel distance and reducing the probability of collision. These criteria collectively constitute the objective function guiding the optimization process. The proposed evolutionary framework employs a sampling-based method to compute the objective function. This method relies on a series of simulations that consider the dynamics of agents and environmental disturbances. Therefore, those simulation results can reflect the statistical characteristics of the objective function according to all random processes considered in the model. The optimization results demonstrate a reasonable convergence regardless of all the variations in the environment and team configuration. Furthermore, the algorithm's effectiveness in ensuring collision-free trajectories is validated by an analysis of the minimum agent-to-agent distances along their paths. This analysis serves to underscore the collision-free properties inherent in the algorithm. A comparison of exploration scores for various group configurations demonstrates the superior performance of the proposed algorithm, positioning it as a promising optimization solution for benchmark cases. It outperforms random exploration methods. Importantly, while originally designed for finding three-dimensional trajectories for cooperative exploration executed by a group of drones, the algorithm possesses the adaptability needed to address the path-planning requirements of multi-arm robots operating in higher dimensions, while maintaining collision avoidance capabilities. Looking ahead, the current research envisions further enhancements, including real-world testing with physical robots and integration with distributed control algorithms, ultimately presenting a unified framework for multi-robot autonomous exploration in uncertain environments with dynamic disturbances. These future endeavors will contribute to a more comprehensive and practical solution for a wide range of robotic applications.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation. These will be available through the funding project website https://roboroyale.eu.

Ethics statement

The manuscript presents research on animals that do not require ethical approval for their study. No direct physical interaction occurred with the animals. The animals were observed in a way that does not cause them any stress or discomfort.

Author contributions

FR: conceptualization, data curation, investigation, methodology, software, validation, visualization, writing–original draft, and writing–review and editing. TK: data curation, investigation, methodology, supervision, validation, and writing–review and editing. FA: funding acquisition, methodology, project administration, resources, supervision, and writing–review and editing.

Funding

The author(s) declare that financial support was received for the research, authorship, and/or publication of this article. This work was supported by the EU H2020-FET RoboRoyale project (964492).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

The author(s) declared that they were an editorial board member of Frontiers, at the time of submission. This had no impact on the peer review process and the final decision.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Agha-mohammadi, A., Chakravorty, S., and Amato, N. M. (2014). Firm: sampling-based feedback motion-planning under motion uncertainty and imperfect measurements. *Int. J. Robotics Res.* 33, 268–304. doi:10.1177/0278364913501564

Ahn, C. W., and Ramakrishna, R. (2002). A genetic algorithm for shortest path routing problem and the sizing of populations. *IEEE Trans. Evol. Comput.* 6, 566–579. doi:10.1109/tevc.2002.804323

Amigoni, F., Banfi, J., and Basilico, N. (2017). Multirobot exploration of communication-restricted environments: a survey. *IEEE Intell. Syst.* 32, 48–57. doi:10.1109/mis.2017.4531226

Bahaidarah, M., Rekabi-Bana, F., Marjanovic, O., and Arvin, F. (2024). Swarm flocking using optimisation for a self-organised collective motion. *Swarm Evol. Comput.* 86, 101491. doi:10.1016/j.swevo.2024.101491

Barbosa, F. S., Lacerda, B., Duckworth, P., Tumova, J., and Hawes, N. (2021). "Risk-aware motion planning in partially known environments," in 2021 60th IEEE Conference on Decision and Control (IEEE), 5220–5226.

Barmak, R., Stefanec, M., Hofstadler, D. N., Piotet, L., Schönwetter-Fuchs-Schistek, S., Mondada, F., et al. (2023). A robotic honeycomb for interaction with a honeybee colony. *Sci. Robotics* 8, eadd7385. doi:10.1126/scirobotics.add7385

Bengio, Y., Lodi, A., and Prouvost, A. (2021). Machine learning for combinatorial optimization: a methodological tour d'horizon. *Eur. J. Operational Res.* 290, 405–421. doi:10.1016/j.ejor.2020.07.063

Biswas, S., Anavatti, S. G., and Garratt, M. A. (2017). "Obstacle avoidance for multi-agent path planning based on vectorized particle swarm optimization," in *Intelligent and evolutionary systems*. Editors G. Leu, H. K. Singh, and S. Elsayed (Cham: Springer International Publishing), 61–74.

Blum, C., Puchinger, J., Raidl, G. R., and Roli, A. (2011). Hybrid metaheuristics in combinatorial optimization: a survey. *Appl. Soft Comput.* 11, 4135–4151. doi:10.1016/j.asoc.2011.02.032

Cai, K., Wang, C., Song, S., Chen, H., and Meng, M. Q.-H. (2021). Risk-aware path planning under uncertainty in dynamic environments. *J. Intelligent Robotic Syst.* 101, 47–15. doi:10.1007/s10846-021-01323-3

Dalmasso, M., Garrell, A., Domínguez, J. E., Jiménez, P., and Sanfeliu, A. (2021). "Human-robot collaborative multi-agent path planning using Monte Carlo tree search and social reward sources," in 2021 IEEE international conference on robotics and automation (ICRA), 10133–10138.

Das, P., Behera, H., and Panigrahi, B. (2016). Intelligent-based multi-robot path planning inspired by improved classical q-learning and improved particle swarm optimization with perturbed velocity. *Eng. Sci. Technol. Int. J.* 19, 651–669. doi:10.1016/j.jestch.2015.09.009

Di Mario, E., Talebpour, Z., and Martinoli, A. (2013). "A comparison of pso and reinforcement learning for multi-robot obstacle avoidance," in *2013 IEEE congress on evolutionary computation*, 149–156.

Dorri, A., Kanhere, S. S., and Jurdak, R. (2018). Multi-agent systems: a survey. *IEEE Access* 6, 28573–28593. doi:10.1109/access.2018.2831228

Galceran, E., and Carreras, M. (2013). A survey on coverage path planning for robotics. *Robotics Aut. Syst.* 61, 1258–1276. doi:10.1016/j.robot.2013.09.004

Juan, A. A., Faulin, J., Grasman, S. E., Rabe, M., and Figueira, G. (2015). A review of simheuristics: extending metaheuristics to deal with stochastic combinatorial optimization problems. *Operations Res. Perspect.* 2, 62–72. doi:10.1016/j.orp.2015.03.001

Kala, R. (2012). Multi-robot path planning using co-evolutionary genetic programming. *Expert Syst. Appl.* 39, 3817–3831. doi:10.1016/j.eswa.2011.09.090

Kavraki, L., Svestka, P., Latombe, J.-C., and Overmars, M. (1996). Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Trans. Robotics Automation* 12, 566–580. doi:10.1109/70.508439

Li, M., Richards, A., and Sooriyabandara, M. (2021). "Reliability-aware multi-uav coverage path planning using a genetic algorithm," in Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems, 1584–1586.

Liu, Z., Chen, B., Zhou, H., Koushik, G., Hebert, M., and Zhao, D. (2020). "Mapper: multi-agent path planning with evolutionary reinforcement learning in mixed dynamic environments," in IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, 11748–11754.

Nawaz, F., and Ornik, M. (2023). Multi-agent, multi-target path planning in markov decision processes. *IEEE Trans. Automatic Control* 68, 7560–7574. doi:10.1109/tac.2023.3286807

Nazarahari, M., Khanmirza, E., and Doostie, S. (2019). Multi-objective multi-robot path planning in continuous environment using an enhanced genetic algorithm. *Expert Syst. Appl.* 115, 106–120. doi:10.1016/j.eswa.2018.08.008

Okubo, T., and Takahashi, M. (2023). Multi-agent action graph based task allocation and path planning considering changes in environment. *IEEE Access* 11, 21160–21175. doi:10.1109/access.2023.3249757

Pereira, A. A., Binney, J., Hollinger, G. A., and Sukhatme, G. S. (2013). Risk-aware path planning for autonomous underwater vehicles using predictive ocean models. *J. Field Robotics* 30, 741–762. doi:10.1002/rob.21472

Qie, H., Shi, D., Shen, T., Xu, X., Li, Y., and Wang, L. (2019). Joint optimization of multi-uav target assignment and path planning based on multi-agent reinforcement learning. *IEEE Access* 7, 146264–146272. doi:10.1109/access.2019.2943253

Rafai, A. N. A., Adzhar, N., and Jaini, N. I. (2022). A review on path planning and obstacle avoidance algorithms for autonomous mobile robots. *J. Robotics* 2022, 1–14. doi:10.1155/2022/2538220

Ravankar, A. A., Ravankar, A., Emaru, T., and Kobayashi, Y. (2020). Hpprm: hybrid potential based probabilistic roadmap algorithm for improved dynamic path planning of mobile robots. *IEEE Access* 8, 221743–221766. doi:10.1109/access.2020.3043333

Rekabi-Bana, F., Hu, J., Krajník, T., and Arvin, F. (2024). Unified robust path planning and optimal trajectory generation for efficient 3d area coverage of quadrotor uavs. *IEEE Trans. Intelligent Transp. Syst.* 25, 2492–2507. doi:10.1109/tits.2023.3320049

Rekabi-Bana, F., Shirazi, F. A., and Sadigh, M. J. (2020). Distributed nonlinear h control algorithm for multi-agent quadrotor formation flying. *ISA Trans.* 96, 81–94. doi:10.1016/j.isatra.2019.04.036

Rekabi-Bana, F., Shirazi, F. A., Sadigh, M. J., and Saadat, M. (2021). Distributed output feedback nonlinear formation control algorithm for heterogeneous aerial robotic teams. *Robotics Aut. Syst.* 136, 103689. doi:10.1016/j.robot.2020.103689

Rekabi-Bana, F., Stefanec, M., Ulrich, J., Keyvan, E. E., Rouček, T., Broughton, G., et al. (2023). "Mechatronic design for multi robots-insect swarms interactions," in 2023 IEEE International Conference on Mechatronics (ICM), 1–6.

Rizk, Y., Awad, M., and Tunstel, E. W. (2019). Cooperative heterogeneous multi-robot systems: a survey. *ACM Comput. Surv.* 52, 1–31. doi:10.1145/3303848

Romano, D., Donati, E., Benelli, G., and Stefanini, C. (2019). A review on animal–robot interaction: from bio-hybrid organisms to mixed societies. *Biol. Cybern.* 113, 201–225. doi:10.1007/s00422-018-0787-5

Rouček, T., Pecka, M., Čížek, P., Petříček, T., Bayer, J., Šalanský, V., et al. (2020). "Darpa subterranean challenge: multi-robotic exploration of underground environments," in *Modelling and simulation for autonomous systems*. Editors J. Mazal, A. Fagiolini, and P. Vasik (Springer International Publishing), 274–290.

Safe, M., Carballido, J., Ponzoni, I., and Brignole, N. (2004). "On stopping criteria for genetic algorithms," in *Advances in artificial intelligence – sbia 2004*. Editors A. L. C. Bazzan, and S. Labidi (Springer Berlin Heidelberg), 405–413.

Semnani, S. H., Liu, H., Everett, M., de Ruiter, A., and How, J. P. (2020). Multi-agent motion planning for dense and dynamic environments via deep reinforcement learning. *IEEE Robotics Automation Lett.* 5, 3221–3226. doi:10.1109/lra.2020.2974695

Stefanec, M., Hofstadler, D. N., Krajník, T., Turgut, A. E., Alemdar, H., Lennox, B., et al. (2022). A minimally invasive approach towards "ecosystem hacking" with honeybees. *Front. Robotics AI* 9, 791921. doi:10.3389/frobt.2022.791921

Stern, R., Sturtevant, N., Felner, A., Koenig, S., Ma, H., Walker, T., et al. (2019). Multi-agent pathfinding: definitions, variants, and benchmarks. *Proc. Int. Symposium Comb. Search* 10, 151–158. doi:10.1609/socs.v10i1.18510

Štibinger, P., Báča, T., and Saska, M. (2020). Localization of ionizing radiation sources by cooperating micro aerial vehicles with pixel detectors in real-time. *IEEE Robotics Automation Lett.* 5, 3634–3641. doi:10.1109/lra.2020.2978456

Sun, R., Tang, C., Zheng, J., Zhou, Y., and Yu, S. (2019). "Multi-robot path planning for complete coverage with genetic algorithms," in Intelligent Robotics and Applications: 12th International Conference, ICIRA 2019, Shenyang, China, August 8–11, 2019 (Springer), 349–361283.

Tranzatto, M., Miki, T., Dharmadhikari, M., Bernreiter, L., Kulkarni, M., Mascarich, F., et al. (2022). Cerberus in the darpa subterranean challenge. *Sci. Robotics* 7, eabp9742. doi:10.1126/scirobotics.abp9742

Vinyals, O., Babuschkin, I., Czarnecki, W. M., Mathieu, M., Dudzik, A., Chung, J., et al. (2019). Grandmaster level in starcraft ii using multi-agent reinforcement learning. *nature* 575, 350–354. doi:10.1038/s41586-019-1724-z

Xiang, X., Jouvencel, B., and Parodi, O. (2010). Coordinated formation control of multiple autonomous underwater vehicles for pipeline inspection. *Int. J. Adv. Robotic Syst.* 7, 3. doi:10.5772/7242

Yu, J., and LaValle, S. M. (2013). "Multi-agent path planning and network flow," in *Algorithmic foundations of robotics X*. Editors E. Frazzoli, T. Lozano-Perez, N. Roy, and D. Rus (Springer Berlin Heidelberg), 157–173.

Žampachů, K., Ulrich, J., Rouček, T., Stefanec, M., Dvořáček, D., Fedotoff, L., et al. (2022). "A vision-based system for social insect tracking," in 2022 2nd International Conference on Robotics, Automation and Artificial Intelligence (RAAI) (IEEE), 277–283.

Zlot, R., Stentz, A., Dias, M., and Thayer, S. (2002). "Multi-robot exploration controlled by a market economy," in Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292), 3016–3023.3