Discrete Firefly Algorithm for Recruiting Task in a Swarm of Robots

Nunzia Palmieri and Salvatore Marano

Abstract In this chapter, we propose a Discrete Firefly Algorithm (DFA) for mine disarming tasks in an unknown area. Basically, a pheromone trail is used as indirect communication among the robots, and helps the swarm of robots to move in a grid area and explore different regions. Since a mine may need multiple robots to disarm, a coordination mechanism is necessary. In the proposed scenario, decision-making mechanism is distributed and the robots make the decision to move, balancing the exploration and exploitation, which help to allocate necessary robots to different regions in the area. The experiments were performed in a simulator, testing the scalability of the proposed DFA algorithm in terms of number of robots, number of mines and the dimension of grid. Control parameters inherent to DFA were tuned to test how they affect the solution of the problem.

Keywords Swarm intelligence · Swarm robotics · Firefly Algorithm · Nature-inspired algorithms

1 Introduction

In applications that do not allow human beings to access, multi-robot systems can play an important role to accomplish some tasks. Possible applications are exploration, search and rescue, monitoring, surveillance, cleaning. In order to successfully

N. Palmieri (✉) · S. Marano
Department of Computer Engineering, Modeling, Electronics, and Systems Science, University of Calabria, Rende, CS, Italy
e-mail: n.palmieri@dimes.unical.it; n.palmieri@mdx.ac.uk

S. Marano
e-mail: marano@dimes.unical.it

N. Palmieri
School of Science and Technology, Middlesex University, The Burroughs, London NW4 4BT, UK

X.-S. Yang (ed.), *Nature-Inspired Computation in Engineering*, Studies in Computational Intelligence 637, DOI 10.1007/978-3-319-30235-5_7

perform, distributed coordination and cooperation mechanisms are desirable for multi-robot systems, especially, under a dynamic environment, to ensure robustness, flexibility, and reliability to the system [1].

Swarm Robotics is a new approach to the coordination of multi-robot systems that consists in a large number of simple robots, that are autonomous, not controlled centrally, capable of local communication, that behave like a team and not merely as single entities and operates based on some sense of biological inspiration. Swarm robotics and the related concept of swarm intelligence, is inspired by an understanding of the decentralized mechanisms that underlie the organization of natural swarms such as ants, bees, birds, fish, wolfs and even humans. Social insects provide one of the best-known examples of biological self organized behavior. By means of local and limited communication, they are able to accomplish impressive behavioral feats. The analysis of the social characteristics of insects and animals is essential in order to apply these findings to the design of multi-robot systems and development of similar behaviours in a coordination tasks [2].

A key component of this system is the communication between the agents in the group which is normally local, and guarantees the system to be scalable and robust. A plain set of rules at individual level can produce a large set of complex behaviors at the swarm level. The rules of controlling the individuals are abstracted from the cooperative behaviors in the nature swarm. The swarm is distributed and decentralized, and the system shows high efficiency, parallelism, scalability and robustness [3].

Various communication mechanisms are used for the communication of the swarm in order to accomplish the coordination and cooperation tasks. Many of them try inspiration from the nature; nature-inspired metaheuristics represent a class of heuristic methods inspired by natural behaviors that have ensured the survivability of animals, birds, insects by enabling them to find perfect solutions to almost all their problems (food search, breading, mating). A new trend in developing nature-inspired metaheuristics is to combine the algorithmic components from various metaheuristic aiming to improve the performance of the original techniques in solving NP hard problems. In our case we used a modified version of Ant Colony Optimization [4] for the exploration task and a discrete version of Firefly Algorithm [5] to disarming task.

Basically, in this work, we consider autonomous exploration and search in an unknown environment. The goal is to explore in a minimum amount of time the environment, while searching for the mines that are disseminated in the area.

The environment is not known beforehand and it has to be fully explored; at the same time, the environment must be analyzed in order to detect and disarm some mines disseminated in the area.

More specifically, we focus on multi-objective exploration and search. On one hand the robots explore in independent manner the environment, distributing in the area trying to minimize the overall exploring time avoiding passing in the same region more time. On the other hand, when a mine is detected, multiple robots are needed to work together to accomplish the task so it is necessary a coordinated strategy among involved robots.

To address the social issues in the exploration, the algorithm is based on our recently published heuristic model, inspired by the foraging behavior of colonies of ant in nature [7]. During the exploration, the robots deposit in the terrain a quantity of pheromone that is perceived by other robots that make a decision based on the minimum amount of pheromone in the neighborhood. When they find a mine, a robot becomes a firefly and trying to recruit (attract) other robots according to a novel bio inspired algorithm called Firefly Algorithm [5, 6]. The information about the location of the mines is passed using broadcast communication in order to inform other robots, in the wireless range, about the mine. If a robot receives more requests of recruitment, it evaluates the quality of the target based on the minimum distance [8]. The strategy is able to adapt the current system dynamics if the number of robots or the environment structure or both change.

The algorithm has been previously validated, and this paper presents the analysis of the value of the proposed Firefly Algorithm parameters in a simulated environment. The simulations were performed for various sets of variables, including number of robots; number of mines and dimension of search area. Control parameters inherent to the DFA were tuned to test how they affect the time to complete the overall task (exploring and disarming).

The rest of this paper is organized as follows. In Sect. 2 we briefly revisit approaches related to the problem; in the Sect. 3, we describe the Firefly Algorithm; in Sect. 4 we analyze the problem and the proposed method. In Sect. 5, we report the simulation results tuning the control parameters of the algorithm, including a statistical considerations and finally, we provide concluding remarks in Sect. 6.

2 Related Work

Nature inspired metaheuristics, relying on concepts and search strategies inspired from nature, have emerged as a promising type of stochastic algorithms used for efficiently solving optimization problems.

For sharing information and accomplishing the tasks there are, basically, three ways of information sharing in the swarm: direct communication (wireless, GPS), communication through environment (stigmergy) and sensing. More than one type of interaction can be used in one swarm, for instance, each robot senses the environment and communicates with their neighbor. Balch [9] discussed the influences of three types of communications on the swarm performance and Tan and Zheng [2] presented an accurate analysis for the different types of communication and the impact in a behavior of swarm.

Within the context of swarm robotics, most works on cooperative tasks are based on biologically behavior and indirect stigmergic communication (rather than on local information, which can be applied to systems related to GPS, maps, and wireless communications).

The self-organizing properties of animal swarms such as insects have been studied for better understanding of the underlying concept of decentralized decision-making in nature, but they also gave a new approach in applications to multi-agent systems engineering and robotics. Bio-inspired approaches have been proposed for multi-robot division of labor in applications such as exploration and path formation, or cooperative transport and prey retrieval.

Within the context of swarm robotics, most works on cooperative tasks are based on social behavior like Ant Colony Optimization. The principle is simple: ants deposit a pheromone trail on the path they take during travel. Using this trail, they are able to navigate toward their nest or food and communicate with their peers. More specifically, ants employ an indirect recruitment strategy by accumulating pheromone trails. When a trail gets strong enough, other ants are attracted to it and will follow this trail toward a food destination. The more ants follow a trail, the more pheromone is accumulated and in turn, the trail becomes more attractive for being followed. Pheromone in swarm robotics can be viewed as a mechanism for inter-robot communication that can help reduce the complexity of individual agents [4, 10].

More strategies are based on these principles: Labella et al. [11] have analyzed the behavior of a group of robots involved in an object retrieval task, trying inspiration by a model of ants' foraging.

Other authors experiment with chemical pheromone traces, e.g. using alcohol like in Fujisawa et al. [12] or using a special phosphorescent glowing paint such as has presented by Mayet et al. [13].

An improved form of pheromone communication method called virtual pheromone was used by Payton et al. [14] to employ simple communication and coordination to achieve large scale results in the areas of surveillance, reconnaissance, hazard detection, and path finding. De Rango and Palmieri [7] used an Ant-based strategies incorporating both attraction and repulsion features.

Particle Swarm Optimization is another biologically-inspired algorithm [15] motivated by a social analogy, such as flocking, herding, and schooling behavior in animal populations and have received much attention in recent year.

Pugh and Martinoli [16] proposed an adapted version of Particle Swarm Optimization learning algorithm to distribute unsupervisioned robotic learning in groups of robots with only local information. More applications in swarm robotic use this strategy to coordinate the swarm in a collective search [17] or task allocation like in Meng and Gan [18].

Another very good example of natural swarm intelligence is honeybee foraging, because the group of foraging bees is not controlled by a central decision-making unit and because the collective decisions of colonies in varying environments were found to act in an intelligent way. In honeybees, foraging decisions are based on such local cues, which are exploited by individual forager and receiver bees. Based on these cues and on communication transferred through bee dances [19], one worker can affect the behavior of other nearby workers. This way behavioral feedback loops emerge, which lead to a self-organized regulation of workforce and foraging decisions. Jevetic et al. [20] proposed a distributed bees algorithm (DBA) for task allocation in a 2-D arena.

Explicit communication is a type of communication in which the robots directly pass messages to each other using for example wireless communication. In this case, a communication protocol and an efficient routing algorithm are important for a coordination of the swarm of mobile robots and for a robustness of the total system. In this case ACO routing algorithms can be used to solve routing problem and load balancing in the network. De Rango et al. [21–24] proposed novel routing algorithms based on Swarm Intelligence able to satisfy multiple metrics for a multi-objective optimization like end to end delay, load balancing and energy savings.

Our approach combines an implicit communication among the robots during the exploration task and kind of explicit communication among the robots during the recruiting task. During the exploration, the robots mark the crossed cell through the scent that can be detected by the other robots; the robots choose the cell that has the lowest quantity of substance to allow the exploration of the unvisited cells in order to cover the overall area in less time. After a while, the concentration of pheromone decreases due to the evaporation and diffusion associated with the distance and with the time; in this way we can allow continuous coverage of an area via implicit coordination. The other robots, through proper sensors, smell the scent in the environment and move in the direction with a minimum amount of pheromone that corresponds to an area less occupied and probably an unexplored area [7]. On the other hand, in order to deactivate the mines, the first robot that detects a mine (firefly) in a cell, tries to attract the other robots according to a novel bio-inspired approach called Firefly algorithm (FA) summarized in the next section.

3 The Firefly Algorithm

Firefly Algorithm (FA) is a nature-inspired stochastic global optimization method that was developed by Yang [5, 6]. The FA tries to mimic the flashing behavior of swarms of fireflies. In the FA algorithm, the two important issues are the variation of light intensity and the formulation of attractiveness. The brightness of a firefly is determined by the landscape of the object function. Attractiveness is proportional to brightness and, thus, for any two flashing fireflies, the less bright one move towards the brighter one. The light intensity decays with the square of the distance, the fireflies have limited visibility to other fireflies. This plays an important role in the communication of the fireflies and the attractiveness, which may be impaired by the distance. Some simplifications are assumed such as:

- it is assumed that all fireflies are unisex so they will be attracted to each other regardless of their sex.
- The attractiveness is proportional to their brightness and they both decrease as the distance increases.
- In the case of no existence of no brighter firefly on then, the fireflies will move randomly.
- The brightness of firefly is affected by its fitness.

The distance between any two fireflies i and j, at positions x_i and x_j, respectively, can be defined as the Euclidean distance as follows:

$$\mathrm{r_{ij}} = \|\mathrm{x_i} - \mathrm{x_j}\| = \sqrt{\sum_{\mathrm{k}=1}^{\mathrm{D}} (\mathrm{x_{i,k}} - \mathrm{x_{j,k}})^2} \tag{1}$$

As a firefly's attractiveness is proportional to the light intensity seen by adjacent fireflies, the variation of attractiveness β with the distance r is defined as followed:

$$\beta = \beta_0 e^{-\gamma r_{ij}^2} \tag{2}$$

The movement of a firefly i is attracted to another more attractive firefly j is determined by:

$$x_i^{t+1} = x_i^t + \beta_o e^{-\gamma r_{ij}^2} \left(x_j - x_i \right) + \alpha(\sigma - \frac{1}{2}) \tag{3}$$

The Firefly Algorithm has been proved very efficient and it has three key advantages [25]:

- Automatic subdivision of the whole population into subgroups so that each subgroup can swarm around a local mode. Among all the local modes, there exists the global optimality. Therefore, FA can deal with multimodal optimization naturally.
- FA has the novel attraction mechanism among its multiple agents and this attraction can speed up the convergence. The attractiveness term is nonlinear, and thus may be richer in terms of dynamical characteristics.
- FA can include PSO, DE, and SA as its special cases. Therefore, FA can efficiently deal with a diverse range of optimization problems.

4 Problem Statement

We consider an environment assuming that it is discretized into equally spaced cells that contains a certain number of mines. Robots can move among cells and they can have just local information about robots (neighbors) or regions to explore (neighbor cells) in order to provide a scalable strategy. They are assumed to be able to move in the grid; there are N robots ($R_1; :::; R_N$) placed initially in random positions on a unit grid. The considered scenario is presented under this assumption.

1. The robots are equipped with proper sensors that are able to deposit and smell the chemical substances (pheromone) leaved by the other robots in order to guide the swarm in the exploration task.
2. They make a probabilistic decision based on amount of pheromone in the cells.
3. They are equipped with proper sensor to detect the mines.

4. The robots are equipped with wireless module, indeed when a robot detects a mine, it becomes a firefly and tries to attract other robots sending messages via broadcast communication to robots in the wireless range.
5. The robots, that receive messages by different robots (fireflies), evaluate the light of fireflies and choose the best firefly (in our case the fireflies at minimum distance) and move toward firefly according to the Firefly Algorithm.

It is assumed that they are able to disarm a mine cooperatively. In this paper we investigated the recruiting strategy in order to lead the robots in the positions of the mines in the area and we did not consider properly the disarming task.

In our work, the map of the environment will be generated as a set of grid cells. The robots know only the adjacent cells and during the exploration they spray a scent (pheromone) into the cells to support the navigation of the others. In the algorithm, the robots decide the direction of the movement relying on a probabilistic law inherited by swarm intelligence and swarm robotics techniques. The scent evaporates not only due to diffusion effects in the time, but also in the space according to the distance this allows a higher concentration of scent in the cell where the robot is moving and a lower concentration depending on the distance [7].

Each robot (r), according to uniform distribution, stores in a grid-based map that is represented by a matrix M (dimension $m \times n$). It is assumed that each robot in a cell $(i, j) \epsilon M$ can move just in the neighbor cells through discrete movements. Let z be the number of mines on a set *MS* to distribute on the grid area according to an uniform distribution. The MS set is characterized by the coordinates of the mines. The objective is to detect and disarm all mines and discovery all cells in the grid (this last condition assures that all mines could be correctly detected in an unknown area). Let t_e be the time necessary for a robot to visit a cell, and let t_d be the time necessary to disarm a mine once it has been detected. It is assumed that a fixed number of robots ($rd_{\min}$) are necessary to disarm a mine stored in a cell $(i, j) \epsilon MS$. This means that for the exploration task, the robots can be distributed among the area because each robot independently explores all cells, whereas for the mine detection, more robots need to be recruited in the same cell in order to perform the task. *M(i, j)_a* is a variable representing the number of robots (accesses) that passed through the cell (i, j). We define a bi-objective function as both the time to detect and the disarming the mine through the exploration on the overall grid.

$$\min \sum t_e \quad \text{and} \quad \min \sum_{i=1}^{z} t_{d,i} \tag{4a}$$

subject to

$$M(i, j)\_a \geq 1 \quad i = 1 \ldots m; \quad j = 1 \ldots n \text{ with } (i, j) \epsilon M$$
$$M(i, j)\_a \geq rd_{\min} \text{ with } (i, j) \epsilon MS$$

This is a bi-objective optimization problem and its solutions will result in a Pareto front. However, in order to solve this problem more effectively, for simplicity, we will combine these two objectives to form a single objective optimization problem so as to minimize the overall total time as follows:

$$\min T_{tot} = \min\left(\sum t_e + \sum_{i=1}^{z} t_{d,i}\right) \tag{4b}$$

subject to

$$M(i, j)\_a \geq 1 \quad i = 1 \ldots m; \quad j = 1 \ldots n \text{ with } (i, j) \epsilon M$$
$$M(i, j)\_a \geq rd_{\min} \text{ with } (i, j) \epsilon MS$$

4.1 Exploration Task

The overall task can be divided into two sub tasks: exploration and disarming task. For the exploration we use a strategy inherit swarm intelligent trying inspiration by a modified version of Ant Colony Optimization. The law used by the robots to choose the cells during the movement, according with De Rango and Palmieri [7], is presented below. We consider a robot in a cell s and it will attribute to the set of next cells v_i following a probability as:

$$p\left(v_i|s\right) = \frac{\left[\tau_{v_i,t}\right]^{\varphi} \cdot \left[\eta_{v_i,t}\right]^{\vartheta}}{\sum_{i \in N(s)} \left[\tau_{v_i,t}\right]^{\varphi} \cdot \left[\eta_{v_i,t}\right]^{\vartheta}}, \quad \forall\, v_i \in N(s) \tag{5}$$

where $p(v_i|s)$ represents the probability that the robot, that is in the cell s, chooses the cell $\mathrm{v_i}$; $N(s)$ is the set of neighbors to the cell v_i; $\tau_{\mathrm{vi,t}}$ is the amount of pheromone in the cell vi; η_{vi}, t is the heuristic parameter introduced to make the model more realistic. In addition, φ and θ are two parameters which affect respectively the pheromone and heuristic values (Table 1). Taking into account the spatial dispersion of the pheromone and the temporal dispersion the amount of pheromone in the cell v where the robot will move during the exploration is:

$$\tau_{v,t+1}(d) = t_{v,t} + t_v(d) \tag{6}$$

In order to explore different areas of the environment, the robots choose the cell with a minimum amount of pheromone, corresponding to cell that probably is less frequented and therefore not explored cell. The chosen cell will be selected according with Eq. (5).

$$v_{next} = \min\left[p(v_i|s)\right] \quad \forall\, v_i \in N(s) \tag{7}$$

Table 1 Symbols adopted in the problem

Symbols	Meaning
r_{ij}	Euclidean distance
β_0	Attractiveness coefficient
γ	Absorption coefficient
σ	Random number
α	Randomization parameter
t_e	Time to visit a cell
$t_{d,i}$	Time to disarm the mine i
τ_{vi}, t	Pheromone in cell v_i at the time t
$\eta_{vi,t}$	Heuristic parameter in the cell v_i at time t
φ	Pheromone parameter
θ	Heuristic parameter
$p(v_i\|s)$	Probability that a robot r, that stores in the cell s, chooses the cell v_i
z	Number of mines
T_{tot}	Total time to complete the task (exploration and disarming)
M(i, j)_a	Number of accesses in the cell (i, j)

4.2 Proposed Discrete Firefly Algorithm for Disarming Task

When a robot finds a mine, during the exploration task, it becomes a recruiter (firefly) of the other robots in order to disarm the mine and it tries to attract the other robots based on the mine position.

In this case, the robots are assumed to have transmitters and receivers, using which they can communicate messages to each other. The messages are mostly coordinate positions of the detected mines. However, the robots are assumed to be able to broadcast messages in their wireless range; in this way, a robot can transmits its position only to its neighbors directly and there is not propagation of the messages (one hop communication).

The original version of FA is applied in the continuous space [5], and cannot be applied directly to tackle discrete problem, so we modified the algorithm in order to fit with our problem. In our case, a robot can move in a discrete space because it can go just in the contiguous cells step-by-step. This means that when a robot perceives, in its wireless range, the presence of a firefly (the recruiter robot) and it is in a cell with coordinates x_i and y_i, it can move according with the FA attraction rules such as expressed below:

$$\left\{ \begin{array}{l} x_i^{t+1} = x_i^t + \beta_o e^{-\gamma r_{ij}^2} \left(x_j - x_i\right) + \alpha \left(\sigma - \frac{1}{2}\right) \\ y_i^{t+1} = y_i^t + \beta_o e^{-\gamma r_{ij}^2} \left(y_j - y_i\right) + \alpha(\sigma - \frac{1}{2}) \end{array} \right\} \tag{8}$$

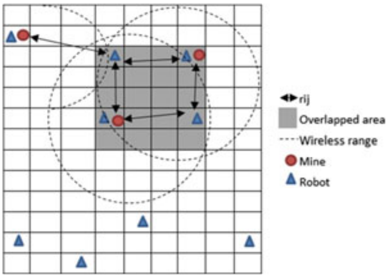

Fig. 1 Two robots during the exploration receive more than one recruiting calls because they are in an overlapped area. The DFA tries to coordinate the robots in the disarming task avoiding redundancy

where x_j and y_j represent the coordinates of detected mine translated in terms of row and column of the matrix area, r_{ij} is the Euclidean distance between mine (or firefly) according to the Eq. (1) and the robot that moves towards the mine. The robots movements are conditioned by mine (firefly) position, in the second term of the formula, and it depends on attractiveness of the firefly such as expressed in Eq. (2) and by a random component in the third term of Eq. (8).The coefficient α is a randomization parameter determined by a problem of interest. The σ coefficient is a random number generator uniformly distributed in the space [0, 1] and it is useful to avoid that more robots go towards the same mine if more robots are recruited by the same firefly and enabling to the algorithm to jump out of any local optimum (Fig. 1). In order to modify the FA to a discrete version, the robots movements have been considered through three possible value updates for each coordinates: $\{-1, 0, 1\}$ according to the following condition:

$$\left\{\begin{array}{ll} x_i^{t+1} = x_i^t + 1 & if\left[\beta_o e^{-\gamma r_{ij}^2}\left(x_j - x_i\right) + \alpha\left(\sigma - \frac{1}{2}\right) > 0\right] \\ x_i^{t+1} = x_i^t - 1 & if\left[\beta_o e^{-\gamma r_{ij}^2}\left(x_j - x_i\right) + \alpha\left(\sigma - \frac{1}{2}\right) < 0\right] \\ x_i^{t+1} = x_i^t + 0 & if\left[\beta_o e^{-\gamma r_{ij}^2}\left(x_j - x_i\right) + \alpha\left(\sigma - \frac{1}{2}\right) = 0\right] \end{array}\right\} \tag{9}$$

$$\left\{\begin{array}{ll} y_i^{t+1} = y_i^t + 1 & if\left[\beta_o e^{-\gamma r_{ij}^2}\left(y_j - y_i\right) + \alpha\left(\sigma - \frac{1}{2}\right) > 0\right] \\ y_i^{t+1} = y_i^t - 1 & if\left[\beta_o e^{-\gamma r_{ij}^2}\left(y_j - y_i\right) + \alpha\left(\sigma - \frac{1}{2}\right) < 0\right] \\ y_i^{t+1} = y_i^t + 0 & if\left[\beta_o e^{-\gamma r_{ij}^2}\left(y_j - y_i\right) + \alpha\left(\sigma - \frac{1}{2}\right) = 0\right] \end{array}\right\} \tag{10}$$

A robot r, which stores in the cell (x_i, y_i) as depicted in Fig. 2, can move in eight possible cells according with the three possible values attributed to x_i and y_i.

For example if the result of the Eqs. (9–10) is $(-1, 1)$ the robot will move in the cell $(x_i - 1, y_i + 1)$.

Fig. 2 Possible movement for a robot r stores in a cell (x_i, y_i)

		y_i	
	x_i-1, y_i-1	x_i-1,y_i	x_i-1,y_i+1
x_i	x_i, y_i-1	r	x_i, y_i+1
	x_i+1, y_i-1	x_i+1, y_i	x_i+1, y_i+1

In the described problem, the firefly algorithm is executed as follows:

- Step 1: Get the list of the detected mines (fireflies) and initialize algorithm's parameters: z number of fireflies, the attractiveness coefficient β_0, the light absorption coefficient γ, randomization parameter α.
- Step 2: Get the list of the robots in the wireless range of the fireflies.
- Step 3: For each robot calculate the distance r_{ij} from the fireflies in its range using the Euclidian distance.
- Step 4: For each robot find the firefly at minimum distance (the best firefly) and try to move the robots from their locations to the location of the best firefly according to the Eqs. (9)–(10).
- Step 6: Terminate if all detected mines are disarmed.

These steps are executed when the robots are recruited by others, indeed when no fireflies are detected or if the new location of the robots is outside of the wireless range of the fireflies, the robots explore in independent manner the area according the Eq. (5). This happens because the nature of the problem is bi-objective and the robots have to balance the two tasks.

5 Parameter Settings

A suitable setting for the DFA parameters values is needed for the proposed mechanism in order to work effectively. We have conducted experiments to find out the most suitable parameter values.

In this section, we evaluate the parameters, focusing on the overall time that is defined as the time needed for the robots to cover the area and disarm all mines disseminated in the area. We considered the overall time because the strategy of recruitment affects significantly the exploration strategy and finally the overall time.

To highlight the performance benefits, we used random positions of the mines and the robots in the area, varying the number of robots operating in the 2-D arena, the size of grid map and the number of robots needed to disarm a mine, in order to study the performance and scalability of the proposed DFA algorithm.

We considered three scenarios: the first scenario is represented by a grid map 20×20 with 3 mines to discover and disarm; the second scenario is represented by a grid map 30×30 with 5 mines to discover and disarm. The third scenario considered a grid map 40×40 with 10 mines to discover and disarm. In the first scenario are required 2 robots to disarm a mine; in the second and third scenario we considered 4 robots to disarm a mine, increasing the complexity of the task.

Experiment setup was created to check the influence of the control parameters in solutions, evaluating the time to complete all task (exploring and disarming task) measured in term of number of iterations. Each of the numerical experiments was repeated 50 times considering the following values: the absorption coefficient ($\gamma = \frac{1}{\mathrm{L}}; \frac{0.5}{\mathrm{L}}; \frac{2}{\mathrm{L}}; \frac{1}{\sqrt{\mathrm{L}}}$), where L is max $\{m, n\}$ and m and n are the number of rows and columns of the matrix M that represents the size of the grid area; randomization parameter ($\alpha = 0.1;\ 0.2;\ 0.5$) and the attraction coefficient ($\beta_0 = 1;\ 0.2;\ 0.5$).

The value of this parameters is important, expecially, when the number of robots to coordinate is low. Figure 3a–c show the performance relative to the attractiveness. It is possible to see that there is no significant difference using different value of the β_0 when the task in not complicated (Fig. 3a). When the complexity of the task increases (Fig. 3b) an high coefficient of attraction $\beta_0(\beta_0 = 1)$ influences negatively the performances when the number of robots is low and tend to be comparable when the number of robots increases. This is happens because an high value of the attractiveness means that the weight of the attraction in the Eq. (8) increases and it is possible that more robots, in an overlapped region, go towards the same mine, creating a not necessary redundancy increasing the time to complete the task. When the number of robot increases, the results are comparable.

Figure 4a–c show the trend of the randomization parameter. For the low number of the robots it is necessary balancing the attractiveness and randomization parameter in order to minimize the overall time, so the value of this parameter is important especially when the complexity of task increases in term of number of mines and number of robots needed to disarm a mine. It is reasonable to expect that by increasing the number of robots the efficiency of swarm improves and the values of parameter do not influence al lot the total performance.

In Fig. 5a–c are plotted the performance of the algorithm considering the absorption coefficient γ; its value is important when the complexity of the task increase as shown in Fig. 5c. In this case a high value of $\gamma(\gamma = 1/\sqrt{\mathrm{L}})$ influences negatively the attractiveness of the firefly making the firefly less bright to recruit the other robots increasing the time to complete all tasks.

5.1 Solution Quality Analysis

In order to determine which of the related parameters are statistically better for the problem we have also considered the p-values of two-samples Student t-tests. The t-tests is used to analyze the relationships between the results obtained from different simulation groups. The experiments are referred to the first scenario (grid 20×20, 3 mines to disarm) second scenario (grid 30×30, 5 mines to disarm) and third scenario (grid 40×40, 10 mines to disarm). The traditionally accepted p-value for something

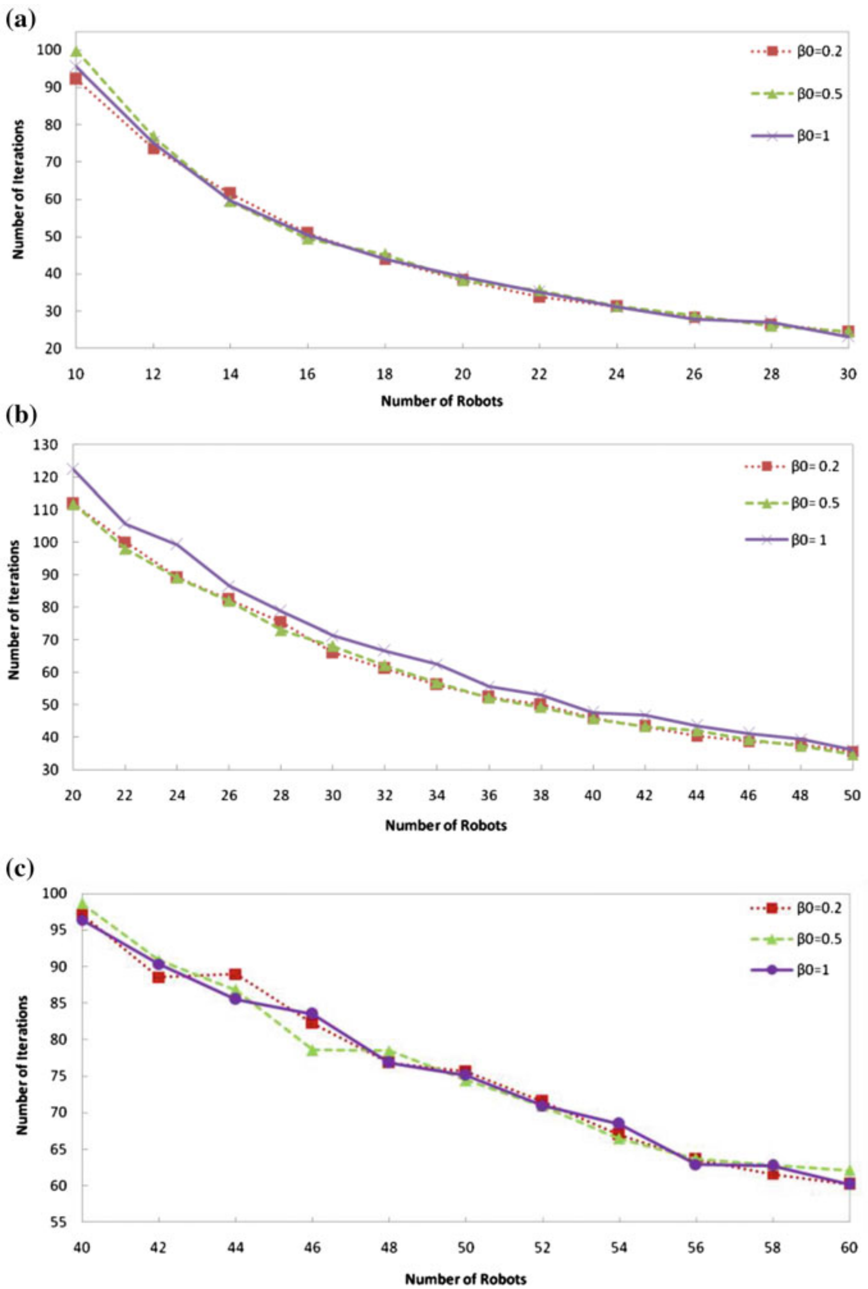

Fig. 3 Effect of the control parameter β_0 on the total time to complete the task measured by the number of iterations, **a** grid 20×20, 3 mines to disarme; **b** grid 30×30, 5 mines to disarme; **c** grid 40×40, 10 mines to disarme

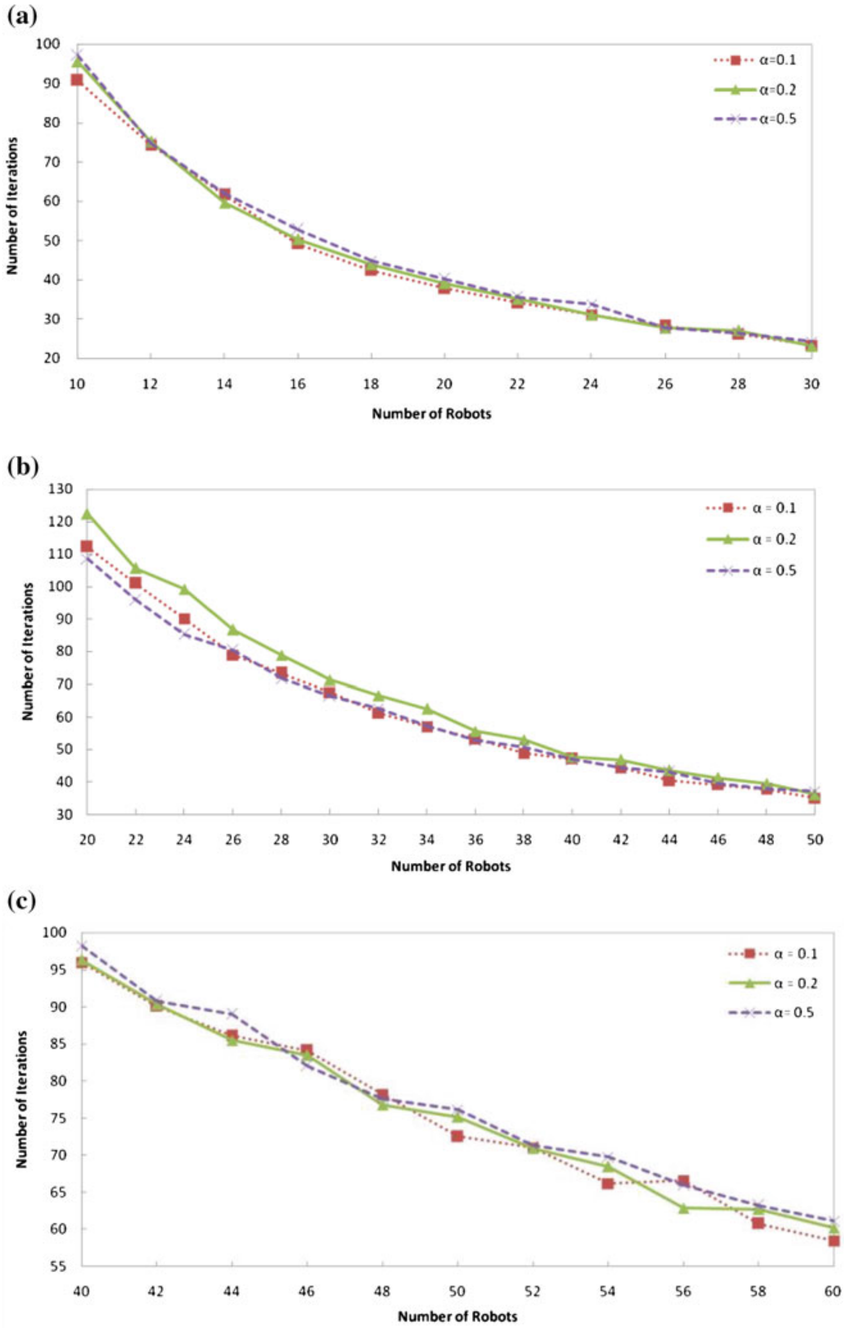

Fig. 4 Effect of the control parameter α on the total time to complete the task measured by the number of iterations, **a** grid 20×20, 3 mines to disarm; **b** grid 30×30, 5 mines to disarme; **c** grid 40×40, 10 mines to disarme

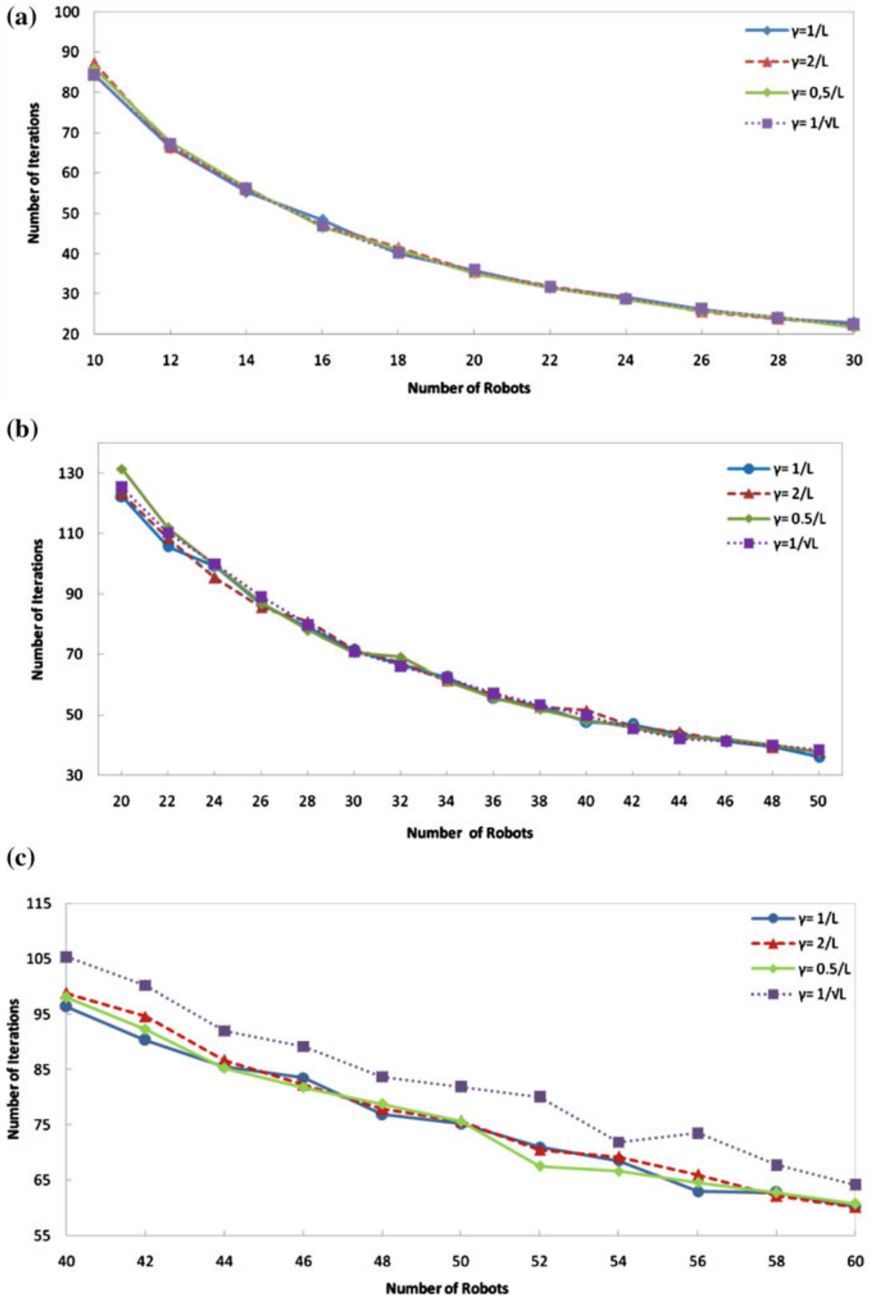

Fig. 5 Effect of the control parameter γ on the total time to complete the task measured by the number of iterations, **a** grid 20×20, 3 mines to disarme; **b** grid 30×30, 5 mines to disarme; **c** grid 40×40, 10 mines to disarme

Table 2 Results of p-values in the t-test for the DFA

Parameters	Scenario 1	Scenario 2	Scenario 3
$\beta_0 = 0.2$ and $\beta_0 = 0.5$	0.2529	0.6847	0.9335
$\beta_0 = 0.2$ and $\beta_0 = 1$	0.5150	**1.55E-05**	0.9524
$\beta_0 = 0.5$ and $\beta_0 = 1$	0.1882	**1.68E-05**	0.9060
$\alpha = 0.1$ and $\alpha = 0.2$	0.1427	**166E-05**	0.6385
$\alpha = 0.1$ and $\alpha = 0.5$	**0.0091**	0.5338	**0.0358**
$\alpha = 0.2$ and $\alpha = 0.5$	**0.0103**	**0.0008**	**0.0204**
$\gamma = \frac{1}{L}$ and $\gamma = \frac{2}{L}$	0.5348	0.3323	**0.0839**
$\gamma = \frac{1}{L}$ and $\gamma = \frac{0.5}{L}$	0.9325	0.1790	0.8886
$\gamma = \frac{1}{L}$ and $\gamma = \frac{1}{\sqrt{L}}$	0.7520	**0.0433**	**2.42E-06**
$\gamma = \frac{2}{L}$ and $\gamma = \frac{1}{\sqrt{L}}$	0.4058	0.3216	**6.95E-07**
$\gamma = \frac{0.5}{L}$ and $\gamma = \frac{1}{\sqrt{L}}$	0.6979	0.8691	**3.14E-04**
$\gamma = \frac{2}{L}$ and $\gamma = \frac{0.5}{L}$	0.3522	0.4459	0.0518

Table 3 Best parameters setting

Parameters	Scenario 1	Scenario 2	Scenario 3
α	0.1	0.5	0.2
β	0.2	0.5	0.5
γ	$\frac{1}{\mathrm{L}}$	$\frac{1}{\mathrm{L}}$	$\frac{1}{\mathrm{L}}$

to be significant is $p < 0.05$. In this case, there is an evidence that the means are significantly different at the significance level reported by the p-value.

Table 2 shows the p-value obtained from the t-tests using all above simulation results by considering each parameter for all considered scenario. In the table are highlighted in bold the p-value more interesting. In this case, there is a difference about the value assumed to the parameter of the problem, so in this case it is better choose the value that influence positively the time. In the other cases, although there is a difference about the time, there is not statistically significant, so it is possible to choose any value of the considered parameter.

To summarize the observations of the parameter settings we described in Table 3 the final best parameters considering the different Scenarios.

6 Conclusions

Swarm intelligence based algorithms are very efficient in solving a wide range of optimization problems in diverse applications in science and engineering.

Firefly Algorithm (FA) is a new swarm intelligence metaheuristic and it has been proven to be effective methods for solving some hard optimization problems. In this chapter, its application for a recruiting task in a swarm of mobile robots is investigated.

One key issue is that all metaheuristics have algorithm-dependent parameters, and the values of these parameters will largely influence the performance of an algorithm. Our experiments, through simulation, showed that the control of parameters provide a mechanism to adjust the robot behavior depending on the dimension of robots (swarm) and the complexity of task.

Results, from the tests, show that the values of the parameters are important when the complexity of tasks increases in terms of dimension of the area and the number of mines disseminated in the area. In particular it is better in this case to balance the attraction of the fireflies (mines) and the random movement in order to distribute better the robots in the area and avoid any redundancy in any region that involve in an increase of the time to complete all tasks.

Future work will include the analysis of the effect of these parameters considering other constraints like the battery that can pose a trade-off between efficiency of the motion and energy utilized. We also will consider the continuous movement of the robots in the area of interest. In addition, we will consider the impact of these parameters introducing obstacles in the area and dropping wireless connection.

References

1. Bellingham, J.G., Godin, M.: Robotics in remote and hostile environments. Science **318**, 1098–1102 (2007)
2. Tan, Y., Zheng, Z.-Y.: Research advance in swarm robotics. Defence Technol. **9**(1), 18–39 (2013)
3. Mohan, Y., Ponnambalam, S.: An extensive review of research in swarm robotics. In: Nature and Biologically Inspired Computing. NaBIC, Word Congress (2009)
4. Dorigo, M., Stutzle, T.: Ant Colony Optimization, MIT Press. ISBN 0-262-04219-3 (2004)
5. Yang, X.S.: Firefly algorithms for multimodal optimization. Lect. Notes Comput. Sci. **5792**, 169–178 (2009)
6. Yang, X.S.: Firefly algorithm, stochastic test functions and designoptimisation. Int. J. Bio-Inspir. Comput. **2**(2), 78–84 (2010)
7. De Rango, F., Palmieri, N.: A swarm-based robot team coordination protocol for mine detection and unknown space discovery. In: Proceedings of the 8th International Conference on Wireless Communications and Mobile Computing, IWCMC, Limassol (2012)
8. De Rango, F., Palmieri, N., Yang, X.S., Marano, S.: Bio-inspired exploring and recruiting tasks in a team of distributed robots over mined regions. In: Proceedings of the International Symposium on Performance Evaluation of Computer and Telecommunication System, Chicago (2015)
9. Balch, T.: Communication, diversity and learning: cornerstones of swarm behaviour. In: Swarm Robotics, Lecture Notes in Computer Science, vol. 3342, p. 21e30. Springer (2005)
10. Ducatelle, F., Di Caro, G.A., Pinciroli, C., Gambardella, L.M.: Selforganized cooperation between robotic swarms. Swarm Intell. **5**(2), 73–96 (2011)
11. Labella, T.H., Dorigo, M., Deneubourg, J.L.: Division of labour in a group inspired by ants' foraging behaviour. ACM Transactionds Auton. Adapt. Syst. **l**(1), 4–25 (2006)
12. Fujisawa, R., Dobata, S., Kubota, D., Imamura, H., Matsuno, F.: Dependency by concentration of pheromone trail for multiple robots. In: Proceedings of ANTS 2008, 6th International

Workshop on Ant Algorithms and Swarm Intelligence, volume 4217 of LNCS, pp. 283–290. Springer (2008)
13. Mayet, R., Roberz, J., Schmickl, T., Crailsheim, K.: Antbots: a feasible visual emulation of pheromone trails for swarm robots. In: Proceedings of the 7th International Conference on Swarm Intelligence (ANTS), pp. 84–94 (2010)
14. Payton, D., Daily, M., Estowski, R., Howard, M., Lee, C.: Pheromone robotics. Auton. Robots **11**(3), 319–324 (2001)
15. Kennedy, J., Eberhart, R.: Particle swarm optimization. Process. IEEE Int. Conf. Neural Netw. **4**, 1942–1948 (1995)
16. Pugh, J., Martinoli A.: Multi-robot learning with particle swarm optimization. Proceedings of Fifth International JointConference on Autonomous Robots and Multirobot Systems, pp. 441–448. Japan (2006)
17. Hereford, J.M., Siebold, M., Nichols, S.: Using the particle swarm optimization algorithm for robotic search applications. IEEE Swarm Intell. Symp. (2007)
18. Meng, Y., Gan, J.: A distributed swarm intelligence based algorithm for a cooperative multirobot construction task. IEEE Swarm Intell. Symp. (2008)
19. Seeley, T.D., Camazine, S., Sneyd, J.: Collective decisionmaking in honey bees: how colonies choose among nectar sources. Behav. Ecol. Sociobiol. **28**(4), 277–290 (1991)
20. Jevtic, A., Gutiérrez, A., Andina, D., Jamshidi, M.: Distributed bees algortithm for task allocation in swarm of robots. IEEE Syst. J. **6**, 2 (2012)
21. De Rango, F. Tropea, M.: Swarm intelligence based energy saving and load balancing in wireless ad hoc networks. In: Proceedings of the 2009 workshop on Bio-inspired algorithms for distributes systems, pp. 77–84. NY (2009)
22. De Rango, F. Tropea, M., Provato, A., Santamaria, A.F., Marano, S.: Minimum hop count and load balancing metrics based on ant behaviour over hap mesh. In: Proceedings of the Global Teleccomunications Conference, New Orleans (2008)
23. De Rango, F., Tropea, M.: Energy saving and load balancing in wireless ad hoc networks trough ant-based routing. In: Proceedings of the Internation Symposium on Performance Evaluation of Computer and Telecommunication Systems (2009)
24. De Rango, F., Tropea, M., Provato, A., Santamaria, A.F., Marano, S.: Multi-constraints routing algorithm based on swarm intelligence over high altitude platforms. In: Proceedings of the International Workshop on Nature Inspired Cooperative Strategies for Optimization, pp. 409–418. Acireale (2007)
25. Fister, I., Yang, X.S., Fister, D., Fister Jr., I.: Firefly algorithm: a brief review of the expanding literature. In: Cuckoo Search and Firefly Algorithm, pp. 347–360. Springer, NY (2014)