Swarm robotics in wireless distributed protocol design for coordinating robots involved in cooperative tasks

Floriano De Rango[1] · **Nunzia Palmieri**[1] · **Xin-She Yang**[2] · **Salvatore Marano**[1]

Abstract The mine detection in an unexplored area is an optimization problem where multiple mines, randomly distributed throughout an area, need to be discovered and disarmed in a minimum amount of time. We propose a strategy to explore an unknown area, using a stigmergy approach based on ants behavior, and a novel swarm based protocol to recruit and coordinate robots for disarming the mines cooperatively. Simulation tests are presented to show the effectiveness of our proposed Ant-based Task Robot Coordination (ATRC) with only the exploration task and with both exploration and recruiting strategies. Multiple minimization objectives have been considered: the robots' recruiting time and the overall area exploration time. We discuss, through simulation, different cases under different network and field conditions, performed by the robots. The results have shown that the proposed decentralized approaches enable the swarm of robots to perform cooperative tasks intelligently without any central control.

1 Floriano De Rango, Nunzia Palmieri, Salvatore Marano
Dept. Computer Engineering, Modeling, Electronics and System Science,
University of Calabria, Italy
E-mail: {n.palmieri,derango, marano}@dimes.unical.it

2 Xin-She Yang
School of Design Engineering and Mathematics,
Middlesex University London, UK
E-mail: x.yang@mdx.ac.uk

1 Introduction

Swarm Robotics (SR) is the study of robotic systems consisting of a large group of relatively small and simple robots that interact and cooperate with each other in order to jointly solve tasks that are beyond their own individual capabilities. SR is becoming an emerging research area in recent years and it, mostly, inherits the inspiration from decentralized self-organizing biological systems and from the collective behavior of social insects [1] [2]. The most important element of a multi-robot system is the ability of several individual robots to work cooperatively. By working together, the robots can complete tasks that a single robot is incapable of accomplishing. For these reasons, multi-robot systems are applied in many engineering problems such as rescue missions, mine detection, surveillance and problems in various domains.

In this paper, we study the mine detection problem in an unknown area. It is well known that landmines are one of the biggest problems that nowadays affect many countries in the world. Such mines can remain active for years after the end of a terrible conflict and thus pose a major problem causing serious restraint and delay on post-conflict reconstruction. Despite international efforts to ban the production and use of landmines, the situation continues to deteriorate with landmines being laid about twenty times faster than they are currently being cleared.

Current technology suggests that robots could be used instead of humans to perform the demining task as minefields are dangerous to humans; thus a robotic solution allows human operators to be physically removed from the hazardous area [3], [4]. In this work, we focus more on the coordination of the robots to accomplish the mission than the issue of disarming physically the

mines. For this purpose, we have proposed and applied some techniques inherit from Swarm Intelligent. Moreover, it is supposed that the robots have a number of attributes such as avoiding interference with each other, having sensorial capabilities, sharing the workload by providing information via different sensors or wireless networks, having systems that allow the identification and disarming of mines. More specifically, the task tackled in this paper involves three broad challenges:

1. Exploring unknown area: discovering of the unknown space in the minimum time, avoiding passing more times on the same previously traversed cells;
2. Self-organization: robots can perform this task in an efficient manner through appropriate sensors on board and able to perceive the environment;
3. Recruitment task: it is the cooperative work to coordinate the robots after the detection of one or more mines and disarm them cooperatively.

The main purpose of this paper is the presentation of an Ant based algorithm to jointly explore an unknown area and perform a recruitment/disarming task in order to analyze performance in terms of overall completion time and communication traffic to make the system highly efficient. The objective is to find and disarm all mines and to explore all area (this last condition assures that all mines could be correctly detected in the unknown area). Our approaches are inspired by pheromone-mediated navigation of ants and we use a direct and direct communication mechanism for the coordination of the swarm. Through simulation, we show how this system is able to explore unknown area in efficient manner helping the recruitment phase.

Basically, the mission is divided into two major phases: exploring and recruitment. In the exploration task, robots need to choose the direction where they will move, according to what they sense in the environment and according to the ACO based algorithm. We are interested in approaching the problem for a large group of robots following the swarm robotics principles, where the cooperation of the robots is performed, similarly in the insect world, by an indirect communication between agents through sensing of a chemical substance (pheromone) that attract other robots in particular directions [5]. In our proposal, the collaborative behavior of the robots is based on the repelling anti-pheromone that means the robots try to distribute them in different regions of the area, minimizing potentially the time.

When one or more robots detect a mine, the recruitment process can start. In this case we propose two mechanisms. The first tries inspiration, again, from Ant Colony and use, in this case, an attraction pheromone signaling in order to attract in the mines location the needed robots to perform the disarming process. The other approach uses WIFI model to communicate with the others. We propose a bio-inspired wireless distributed protocol to recruit the necessary robot in the mines location trying to reduce global communication traffic.

The paper is organized as follows: related work is presented in Section 2; problem statement and formulation are presented in Section 3. Anti-pheromone based algorithm for the exploration is described in Section 4; the attractive pheromone and the coordination protocol for the recruitment and disarming issue are presented in Section 5. Finally the performance evaluation and the conclusions are summarized in Section 6 and Section 7, respectively.

2 Related Work

2.1 Multi-robot exploration

Multi-robot exploration has received much attention in the research community. The unknown area exploration should not lead to an overlapping in robots movements and ideally, the robots should complete the exploration of the area with the minimum amount of the time. The overlapped area can occur when a location has been visited by one of the robots and it is visited again by the same or different robots of the team. Many approaches have been proposed for exploring unknown environments with a team of mobile robots.

Some exploration plans in the context of mapping are usually constructed without using environmental and/or boundary information. One of the well-known techniques is frontier-based exploration, which was proposed by Yamauchi [6]. In this approach, these robots act independently and make probabilistic judgements regarding frontiers areas of unexplored space in an environment. The environment is decomposed into cells with each cell being represented by a probability value, and can be classified as either free, occupied or unknown. Using this representation a robot can reach an unexplored zone by means of navigating to the frontier cells that separate the free cells from the unknown cells. However, other authors use different representations, and thus they identify the unexplored regions in different ways like expressed in [7], [8]. On the other hand, some researchers are focusing on the exploration by using knowledge about environmental boundary information, see [9], [10]. The authors assumed that they already had the information of all obstacles. Therefore, when the robot encountered an obstacle, it could immediately grasp the obstacle. However, this is not practical in real-world applications considering the unknown area. Other approaches [11], [12] coordinated the robots by means of dividing the environment into as many

disjoint regions as available robots and assigning a different region to each robot. Tree-cover algorithms, instead, used a pre-calculated spanning-tree to direct the exploration effort and distribute it among the agents. These algorithms required a priori knowledge of the environment. A typical example is the Multi-Robot Forest Coverage (MFC) algorithm, described in [13] and Multirobot Spanning Tree Coverage (MSTC) algorithm proposed by Hazon [14].

In real scenarios, we always have some uncertainty, so bio-inspired techniques have recently gained importance in computing due to the need for flexible, adaptable ways of solving engineering problems. Within the context of swarm robotics, most works on cooperative exploration are based on biologically behaviour and indirect stigmergic communication (rather than on local information, which can be applied to systems related to GPS, maps, wireless communications). This approach is typically inspired by the behaviour of certain types of animals and insects, like the ants, that use chemical substances known as pheromone to induce behavioral changes in other members of the same species. Previous work on pheromone signalling in robotics has been used for this issue proposed in [15], [16], [17] [18], [19].

2.2 Bio-inspired Self-Coordination of Multi-robot Systems

Coordination of multi-robot has been extensively studied in the scientific literature due to its real-world applications including aggregation, pattern formation, cooperative mapping, and foraging. All of these problems consist of multiple robots making decisions autonomously based on their local interactions with other robots and environments. For sharing information and accomplishing the assigned 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 neighbour. In [20], Tan discussed the influences of these three types of communications on the swarm performance and the impact in a behaviour of swarm. The self-organizing properties of animal swarms have been studied for better understanding the underlying concept of decentralized decision-making in nature, but it also gives a new approach in applications to multi-agent system engineering and robotics. Bio-inspired approaches have been proposed for multi-robot division of labour in applications such as exploration and path formation as described in [21], [22], [23]; cooperative transport or garbage

[24]; inspection [25] and cooperation [26]. Other approaches used a direct communication among the member of the swarm. For example, Ants based routing is gaining more popularity because of its adaptive and dynamic nature and these algorithms consist in the continual acquisition of routing information through path sampling and discovery using small control packets called artificial ants. Some examples are: AntHoc-Net proposed by Di Caro et al. in [27], Ant-Colony Based Routing Algorithm (ARA) described by Bouazizi in [28]. The probabilistic emergent routing algorithm (PERA) [29] has been proposed in which the routing table stores the probability distribution for the neighboring nodes. Singh [30] presented a detail analysis of protocols based on ant-like mobile agents. Moreover, authors proposed bio-inspired routing strategies able to minimize the number of hops, the energy wastage, see [31] or able to combine more bio-inspired techniques in the coordination actions [26].

2.3 Recruitment as aggregation strategy

Recruitment task is important in order to obtain a good exploitation of resources in tasks. Traditional approaches to recruitment in multi-robot systems mainly rely on centralised coordination and require global communication. These approaches are suitable for teams of a limited number of robots and they are not suitable for swarm robotic systems which usually consist of a large number of relatively simple robots. For swarm robotic systems, the control is completely distributed, while coordination is based on selforganisation through local interactions. Distributed coordination is suitable for multi-robot systems under a dynamic and unknown environment due to its robustness, flexibility, and reliability.

Recruitment plays a central role in social insects such as ants, bees, and termites. The recruitment task is a particular self-organization cooperative task in which robots need to aggregate in a point in order to accomplish a task as explained in [33], [34]. Other approches used chemical substances to recruit the robot for certain tasks, which was inspired by pheromone of some species of social insects, such as ants and termites [26], [36]. Pinciroli et al. [37] tried the inspiration, instead, from cockroaches. Meng et al. [38] used Particle Swarm Optimization to allocate reasonable robots to different target blocks. Other approaches use direct communication to coordinate and complete the tasks using wireless medium for communication (MANET) such as bio-inspired algorithms. An example was proposed in [32].

The main contributions of this work in comparison with the literature are listed as follows:

1. Mathematical formulation of a multi-objective optimization problem accounting multiple tasks in the robot coordination.
2. Design of swarm-based strategies where spatial and time pheromone dispersion is applied in order to carry out exploration and recruiting tasks (two joint tasks).
3. Design of a protocol where the data exchanges are balanced with stigmergy in order to assure scalability in the robots communication and in order to scale well in the problem complexity.

3 Description of the problem

In our collective task, there are many mines randomly distributed in an unknown area. The robots should find the mine first, and then remove them. But, treating a mine is to complex by one robot, so multiple robots need to work together. In this paper, swarm intelligence based algorithms have been proposed to search the mines and remove them. The completion time of the mission occurs when all area is explored and all mines are detected and disarmed. Though this is a potentially NP hard problem, the objective of this study is to develop a distributed technique for multi-robot systems in order that the robots can complete the mission as quickly as possible.

There are some assumptions for the problem that are divided into two parts: the geometry of the environment and the characteristics and capabilities of the mobile robots. Let A be the robots 2-D working field, in which are distributed a finite number of static obstacles. Obstacle cells are inaccessible to the robot and impenetrable to the sensors. Let A be discretized into a grid with $m \times n$ cells. Establish a Cartesian coordinate system which takes the upper left corner of A as the origin. Each cell has its own definite coordinate that can be represented by two coordinates (i,j), where i and j are two nonnegative integers. At each step, the robots state can be represented by its location (i,j). In the area there are T stationary targets that are mines. Each target is located in a cell with coordinates (i,j). For example, T = (0,0),(7,8),(20,6) indicates that there are 3 mines in the area with coordinates (0,0), (7,8) and (20,6).

All robots do not have any prior information about the location of the mines so they need to explore the whole environment. Once a mine is detected by a robot, the recruitment process is carried out. As far as the characterization of the robots is concerned, we assume that they live in a discrete-time domain and they can move on a cell by cell, that is, one cell at a time. They can visit all cells in the area except that the position is
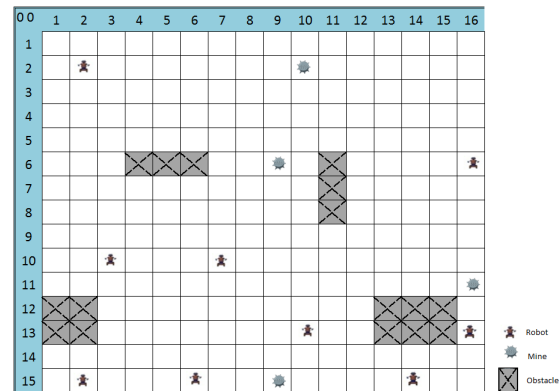

Fig. 1: A representation of the considered environment.

occupied by an obstacles or another robot. They have limited computing and memory capacities, but not limited to motion, sensing, communication and computation. They are capable of discovering and partially executing the tasks. However, for the sake of the simplicity, the robots have a simple set of common reactive behaviour that can enable them to avoid the obstacles and recognize the other robots in order to accomplish the mission together. The robots, at the beginning, can be placed on the same initial cell or can be randomly distributed on the grid area. We assume that a robot uses 45° as the unit for turning, since we only allow the robot to move from one cell to one of its eight neighbors, if all cells are free. The robot can have just local information about the others (neighbors robots) in order to provide a scalable strategy. It is assumed that each robot in a cell (i,j) can move just in the neighbor cells through discrete movements Fig. 1.

We assume that the robots are equipped with proper sensors to perceive, leave the pheromone and detect the mines. During the exploration task, they can leave the pheromone in the cell and it propagates until a certain distance. A mine is detected by a robot when the mine position represented by the (i,j) coordinates coincides with the robot's (i,j) location. The behaviour of the robots, in each state, has been described in the Fig. 2 on the basis of the events that can occur. We assume that the robots switch roles within a team to carry out the tasks encountered in the environment.

More specifically, at the beginning, when no mine is detected, each robot collects information from its immediate surrounding cells perceiving chemical substance (pheromone) by onboard sensors and uses this information to identify the direction where to move. Each robot calculates its best move in terms of next position locally according to an Ant Colony-based approach as explained below. The goal is that the robots should explore the undetected sub-areas as much as possible in
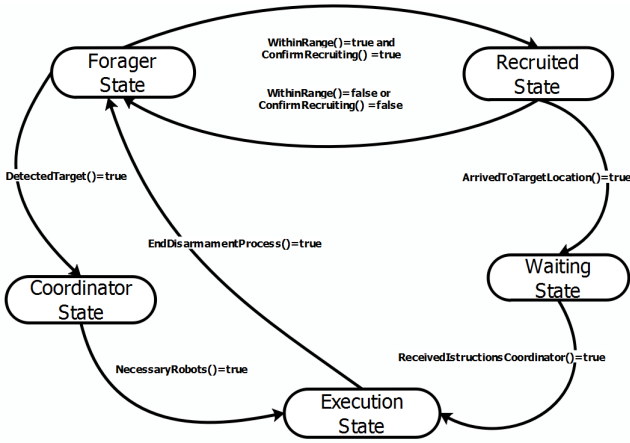
Fig. 2: State transition logic for a robot at each time step.

order to speed up the task. This state is named the Forager State and it is the initial state for each robot.

Once a robot discovers a target by itself, it will switch to a Coordinator State. Each coordinator robot is responsible for handling the disarmament process of the discovered target and for the recruitment of the others. The recruiting process ends when the predefined number of necessary robots ($R_{\min}$) have arrived at the targets location to form a coalition team. Then, the accumulated robots work together as a group, performing the disarmament task.

When a robot (say, k) receives one or more request by coordinator robots, it switches to the Recruited State. Then, the robot will make the decision about where to move and what target to perform. A key aspect of this state is that the robots react to events that occur. Unlike common approaches, they could change the decisions taken previously during the iterations. For example, for a certain type of mission, it is possible to meet a target or receive different requests, while reaching another target in response to a recruitment process, thus reconsidering the choice of the target to be handled. Moreover, the decision can be to restart to explore the area since the movements are too far from the targets location. When a recruited robot, once it reaches the targets location, it will wait until the other needed robots have arrived and thus enter into the waiting mode. This state is called the Waiting State. Finally, once the required robots reach the targets location, the group as a whole is involved in the disarming process and they will perform, for a fixed amount of time, some actions to deal with the targets properly. This state is the Execution State.

3.1 Mathematical Model

In order to describe the proposed system as proper mathematical models, it is useful to introduce the following notations and definitions:

- A: operational area, discretized as a grid map and $A \subset \mathbb{R}^2$
- R : set of robots
- N^R : number of robots $N^R = |\mathrm{R}|$
- $R_{\min}$ = number of robots needed to deal with a target
- T: set of mines
- N^T: number of mines, $N^T = |\mathrm{T}|$

Two main decisions have to be modelled properly. On the one hand, the position expressed by the coordinates where each robot $k \in R$ should be located at each step. On the other hand, given a robot k and a found mine z , it has to decide if it is to get involved in the manipulation process of the found target z.

The first decision is mathematically represented by the decision variables:

$$ v_{xy}^k = \begin{cases} 1 & \text{if the robot } k \text{ visits the cell } (x, y), \\ 0 & \text{otherwise.} \end{cases} \quad (1) $$

It is assumed that the time to visit a cell, denoted by T_e, is the same for all robots. Then the goal af an exploration task is to cover the whole area in the minimum amount of time, and thus the first objective becomes:

$$ \text{minimize} \sum_{k=1}^{N^R} \sum_{x=1}^{m} \sum_{y=1}^{n} T_e \; v_{xy}^k. \quad (2) $$

Similarly, the following decision variables allow to model if a robot k is involved in the recruitment process of the target z:

$$ u_z^k = \begin{cases} 1 & \text{if robot } k \text{ is involved with mine } z, \\ 0 & \text{otherwise.} \end{cases} \quad (3) $$

When a robot has eventually detected a target, it should act as an attractor, trying to recruit the required number of robots so as to disarm the discovered mine safely and properly.

Let $T_{Start,z}^k$ be the time step at which the robot k receive a help request for disarming the mine z and $T_{End,z}^k$ the time step at which the robot k has reached the mine z, then ($T_{End,z}^k$ - $T_{Start,z}^k$) is the coordination time. Thus, the objective is the minimization of the coordination time for each found mine, in order to speed up the disarming process and continue the mission effectively. Therefore, the second objective is

$$ \text{minimize} \sum_{k=1}^{N^R} \sum_{z=1}^{N^T} (T_{End,z}^k - T_{Start,z}^k) \; u_z^k. \quad (4) $$

3.1.1 The Bi-Objective Optimization Problem

The considered objective function is thus related to the minimization of the time needed to perform the overall mission. Since we have two objectives, it can be combined using the weighted sum method to convert into a single objective optimization problem. However, because both objectives are times, and we can put the same weighting for each objective. Thus, the optimization problem, accounting both the exploration time and the coordination time, can be mathematically stated as follows:

$$\min \sum_{k=1}^{N^R} \sum_{x=1}^{m} \sum_{y=1}^{n} T_e v_{xy}^k + \sum_{k=1}^{N^R} \sum_{z=1}^{N^T} (T_{End,z}^k - T_{Start,z}^k) u_z^k,$$

(5)

subject to

$$\sum_{k=1}^{N^R} v_{xy}^k \geq 1, \quad \forall (x,y) \in A,$$

(6)

$$\sum_{k=1}^{N^R} u_z^k = R_{min}, \quad \forall z \in T,$$

(7)

$$v_{xy}^k \in \{0,1\}, \quad \forall (x,y) \in A, \ k \in R,$$

(8)

$$u_z^k \in \{0,1\}, \quad \forall z \in T, \ k \in R.$$

(9)

$$T_e, \ T_{End,z}^k, \ T_{Start,z}^k \in \mathbb{R}, \quad \forall z \in T, \ k \in R.$$

(10)

The objective function in (5) to be minimized represents the total time consumed by the swarm of robots. It depends on the time for the exploration of the area and the time for coordinating the robots involved in the disarming process of the mines. Constraint (6) ensures that each cell is visited at least once. Constraint (7) defines that each mine z must be disarmed safely by $R_{\min}$ robots. The constraints (8)-(10) specify the domain of the decision variables. The optimization problem here is intrinsically multi-objective, but it have been formulated it as a combined single objective optimization problem. Future work will focus on the extension of the current approach to the analysis of multi-objective optimization.

4 Ant-Based Strategy for Area Exploration

Ant colonies provide some of the richest examples for the study of collective phenomena such as collective exploration. Exploration is a very important task in nature since it allows animals to discover resources, detect the presence of potential risks, forage for food and scout for new home. Ant colonies operate without central control, coordinating their behavior through local interactions with each other. Ants perceive only local, mostly chemical and tactile cues. For a colony to monitor its environment, to detect both resources and threats, ants must move around so that if something happens, or a food source appears, some ants are likely to be near enough to find it [38].

Ant colonies, despite the simplicity of single ants, demonstrate surprisingly good results in global problem solving. Consequently, ideas borrowed from insects and especially from ants behaviour are increasingly popular in robotics and distributed system. Ant Colony Optimization has been developed by Dorigo [39] inspired by the natural behaviour of trail laying and following by ants. They live in colonies and their behavior is governed by the goal of colony survival rather than being focused on the survival of individuals. During foraging, ants can often find shortest paths between food sources and their nest. When searching for food, ants initially explore the area surrounding their nest in a random manner. While moving, ants can leave and smell a chemical pheromone trail on the ground. When choosing their way, they tend to choose, in probability, paths marked by strong pheromone concentrations. As soon as an ant finds a food source, it evaluates the quantity and the quality of the food and carries some of it back to the nest. During the return trip, the quantity of pheromone that an ant leaves on the ground may depend on the quantity and quality of the food. The pheromone trails will guide other ants to the food source.

The central component of an ACO algorithm is a parametrized probabilistic model, which is called the pheromone model. Broadly speaking, the robots operate according to the following steps:

(a) The robots perceive the surrounding cells using onboard sensors.
(b) The robots compute the perceived information, in this case the concentration of pheromone, in neighbors cells.
(c) The robots decide where to go next.
(d) The robots move in their best local cell and start again from (a).

The basic intention behind the work described here is to design a motion policy which enables a group of robots, each equipped only with simple sensors, to efficiently explore environment eventually complex.

Broadly speaking, when the robots are exploring the area, they lay pheromone on the traversed cells and each robot uses the distribution of pheromone in its immediate vicinity to decide where to move. Like in nature,
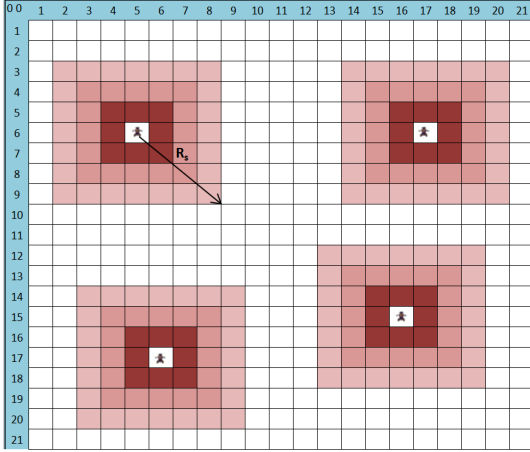
Fig. 3: Robots, during the exploration, spray a chemical substance (pheromone) that propagates in the neighbors cells. The quantity in each cell depends on the distance. The cell in which the robots are moving has a higher quantity and decrease with the distance. The robots try to avoid the cells in which perceive the pheromone, in order to explore different regions of the area.

the pheromone trails change in both space and time. The pheromone deposited by a robot on a cell diffuses outwards cell-by-cell until a certain distance R_s such that $R_s \subset A \subset \mathbb{R}^2$ and the amount of the pheromone decreases as the distance from the robot increases (see Fig. 3).

Mathematically, the pheromone diffusion is defined as follows: consider that robot k at iteration t is located in a cell of coordinates $(x_k^t, y_k^t) \in A$, then the amount of pheromone that the robot deposits at the cell c of coordinates (x, y) is given by:

$$\Delta\tau_c^{k,t} = \begin{cases} \Delta\tau_0 \ e^{\frac{-r_{kc}}{a_1}} - \frac{\varepsilon}{a_2} & \text{if } r_{kc} \leq R_s, \\ 0 & \text{otherwise,} \end{cases} \quad (11)$$

where r_{kc} is the distance between the robot k and the cell c and it is defined as:

$$r_{kc} = \sqrt{(x_k^t - x)^2 + (y_k^t - y)^2}. \quad (12)$$

This means that pheromone spreads up to a certain distance, as in the real world, after which it is no perceivable by other robots. In addition, $\Delta\tau_o$ is the quantity of pheromone sprayed in the cell where the robot k is placed and it is the maximum amount of pheromone, ε is a random value (noise) so that $\varepsilon \in (0, 1)$. Furthermore, a_1 and a_2 are two constants to reduce or increase the effect of the noise and pheromone (see Fig. 4 and

Fig. 5). It should be noted that multiple robots can deposit pheromone in the environment at same time, then the total amount of pheromone that can be sensed in a cell c depends on the contribution of many robots.

Furthermore, the deposited pheromone concentration is not fixed and evaporates with the time. The rate of evaporation of pheromone is given by ρ ($0 \leq \rho \leq 1$ (Fig. 6), and the total amount of pheromone evaporated in the cell c at step t is given by the following function:

$$\xi_c^t = \rho \ \tau_c^t, \quad (13)$$

where τ_c^t is the total amount of the pheromone on the cell c at iteration t.

Considering the evaporation of the pheromone and the diffusion according to the distance, the total amount of pheromone in the cell c at iteration t is given by

$$\tau_c^t = \tau_c^{(t-1)} - \xi_c^{(t-1)} + \sum_{k=1}^{N^R} \Delta\tau_c^{k,t}. \quad (14)$$

Each robot k, at each time step t, is placed on a particular cell c_k^t that is surrounded by a set of accessible neighbor cells $N(c_k^t)$. Essentially, each robot perceives the pheromone deposited into the nearby cells, and then it chooses which cell to move to at the next step. The probability at each step t for a robot k of moving from cell c_k^t to cell $c \in N(c_k^t)$ can be calculated by

$$p(c|c_k^t) = \frac{(\tau_c^t)^\varphi \ (\eta_c^t)^\lambda}{\sum_{b \in N(c_k^t)}(\tau_b^t)^\varphi \ (\eta_b^t)^\lambda}, \quad \forall \ c \in N(c_k^t), \quad (15)$$

where $(\tau_c^t)^\varphi$ is the quantity of pheromone in the cell c at iteration t, and $(\eta_c^t)^\lambda$ is the heuristic variable to avoid the robots being trapped in a local minimum. In addition, φ and λ are two constant parameters which balance the weight to be given to pheromone values and heuristic values, respectively. The robot k moves into the cell that satisfies the following condition:

$$c = \arg\min[p(c|c_k^t)]. \quad (16)$$

In this way, the robots will prefer less frequently visited regions and more likely they will direct towards unexplored regions.

5 Recruitment Strategies

5.1 Pheromone based Strategy for Recruitment Task

Once a robot detects a mine by itself or receives requests from the others, it should make the decision to search new area or go toward a mine location to cooperate with the others. In this case the robot that detects a mine becomes a coordinator and would like to attract
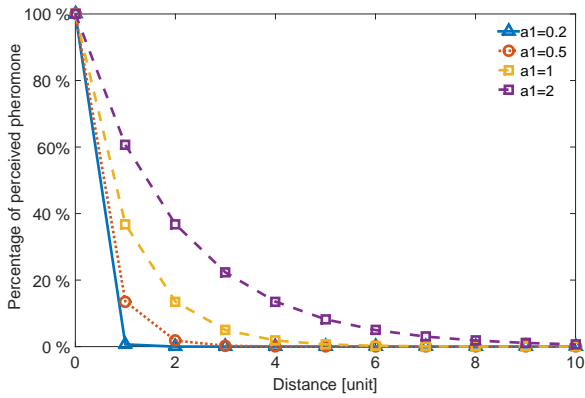
Fig. 4: Trend rate of the perceived pheromone according to the distance varying a_1 value.
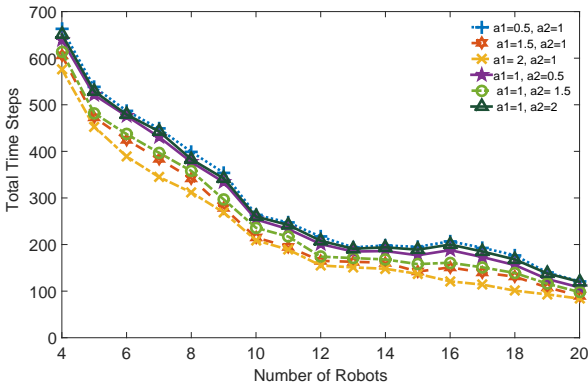

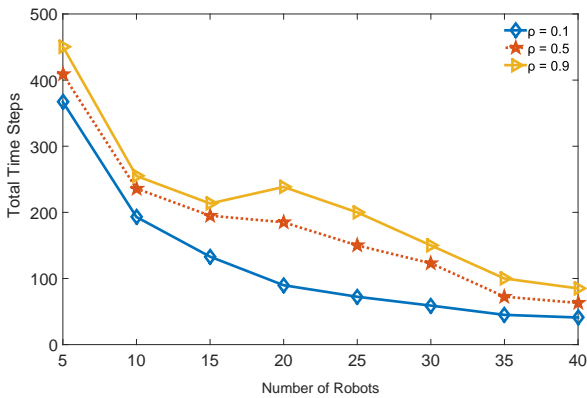Fig. 5: Impact of the a_1 and a_2 on the total time steps.

Fig. 6: Impact of ρ on the total time steps.

the necessary number of robots in the mines location for collaborative task completion. In our approach, the coordinator robots deposit the pheromone, different from the previous used for exploring; this kind of pheromone would attract other robots to guide them into the mines cell. The coordinator robots continue to spray until the necessary robots arrive into the cell (Fig. 7). However,
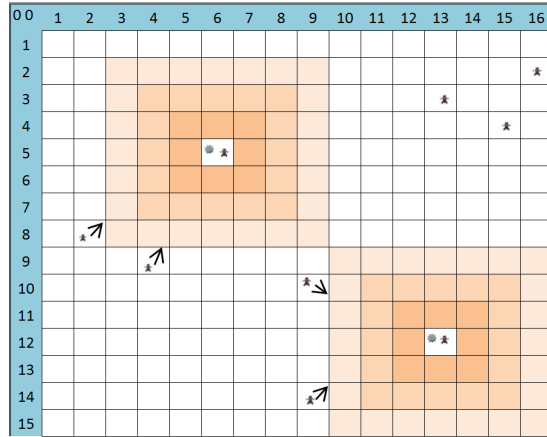

Fig. 7: When the robots detect the mines, try to recruit other robots of the swarm spraying a pheromone. The robots that perceives the pheromone try to reach the mine position, preferring the cell with a higher concentration of pheromone that means cells probably closer to mine's locations. The robots, outside the pheromone range, continue the exploration of the area.

this kind of pheromone follows the same evaporation rules explained in Section 5. More specifically, a robot k, in a cell c_k, that smells this kind of pheromone, chooses the next cell c on the basis of the following formula:

$$p(c|c_k^t) = \frac{(\theta_c^t)^\varphi \ (\eta_c^t)^\lambda}{\sum_{b \in N(c_k^t)} (\theta_b^t)^\varphi \ (\eta_b^t)^\lambda}, \quad \forall \ c \in N(c_k^t), \qquad (17)$$

where $(\theta_c^t)^\varphi$ is the quantity of pheromone (different form the previous pheromone that has a repellent characteristic) in the cell c at iteration t, and $(\eta_c^t)^\lambda$ is the heuristic variable to avoid the robots being trapped in a local minimum. In addition, φ and λ are two constant parameters which balance the weight to be given to pheromone values and heuristic values, respectively. The robot k moves into the cell that satisfies the following condition:

$$c = \arg\max[p(c|c_k^t)]. \qquad (18)$$

In this case the underlying idea was the Maximum Pheromone Following to allow to the robots to reach the mines location with a lower time. The mechanism that uses the pheromone in the exploration phase and in the recruiting phase is called ATRC-ERS (Exploration and Recruiting with only Stigmergy). It exploits just stigmergy, and the robots change behavior from Minimum Pheromone Follower to Maximum Pheromone Follower based on the roles that they assume during the mission.

5.2 Distributed Wireless Communication for Robots Coordination

In this section an on-demand mobile ad hoc network related to the problem to form coalitions in certain locations of the area is presented. The network architecture is created once a robot detects a target in the area and from this point that initiates communication with neighbor to neighbor. The idea is to use ad hoc routing protocol to report a detected target and the robots that wants to serve it over a MANET. Mobile ad-hoc networks (MANETs) consist of special kind of wireless mobile nodes which form a temporary network without using any infrastructure or centralized administration. In networks, all nodes are mobile and communicate with each other via wireless connections. Nodes can join or leave the network at any time. There is no fixed infrastructure. All nodes are equal and there is no centralized control or overview. There are no designated routers: all nodes can serve as routers for each other, and data packets are forwarded from node to node in a multi-hop fashion. Since in mobile ad-hoc networks there is no infrastructure support and nodes being out of range of a source node transmitting packets; a routing procedure is always needed to find a path so as to forward the packets appropriately between the source and the destination.

Moreover, due to limited resources such as power, bandwidth, processing capability, and storage space at the nodes as well as mobility, it is important to reduce routing overheads in MANETs, while ensuring a high rate of packet delivery. Due to the dynamic nature of MANETs, route maintenance is quite a difficult task. Basically, routing is the process of choosing paths in a network along which the source can send data packets towards destination. Routing is an important aspect of network communication because the characteristics like throughput, reliability and congestion depends upon the routing information. An ideal routing algorithm is one which is able to deliver the packet to its destination with minimum amount of delay and network overhead. The nodes update the routing tables by exchanging routing information between the other nodes in the network.

In the literature there exists a large family of ad hoc routing protocols. However, it has been found that bio-inspired approach such as ant colony optimization (ACO) algorithms can give better results as they are having characterization of Swarm Intelligence (SI) which is highly suitable for finding the adaptive routing for such type of volatile network. ACO routing algorithms use simple agents called artificial ants which establish optimum paths between source and destination that communicate indirectly with each other by means of stigmergy. The basic idea behind ACO algorithms for routing is the acquisition of routing information through sampling of paths using small control packets, which are called ants. The ants are generated concurrently and independently by the nodes, with the task to test a path to an assigned destination. An ant going from source node s to destination node d collects information about the quality of the path (e.g. end-to-end delay, number of hops, etc.), and uses this on its way back from d to s to update the routing information at the intermediate nodes.

The routing tables contain for each destination a vector of real-valued entries, one for each known neighbor node. These entries are a measure of the goodness of going over that neighbor on the way to the destination. They are termed pheromone variables, and are continually updated according to path quality values calculated by the ants. The repeated and concurrent generation of path-sampling ants results in the availability at each node of a bundle of paths, each with an estimated measure of quality. In turn, the ants use the routing tables to define which path to their destination they sample: at each node they stochastically choose a next hop, giving higher probability to links with higher pheromone values. For this reason, generally, the routing tables are also called pheromone tables. The routing table at each node is organized on a perdestination basis and is of the form (Destination, Next hop, Probability). It contains the goodness values for a particular neighbor to be selected as the next hop for a particular destination. Further, each node also maintains a table of statistics for each destination d to which a forward ant has been previously sent.

More specifically, the network of robots is created when one or more robots find a target. That is, the robot that has detected a target sends announcement messages that are forwarded by the other robots so that the information about the target can spread among the swarm.

The messages that a robot can send or receive are:

1. **HELLO**: Hello packets are used to notify the robot presence in its transmission range to other robots. A HELLO packet contains the ID of the sending robot. When a robot receives this packet becomes aware of the presence of another robot in its range and it writes the ID in a data structure (neighbors table) which takes into account all the robots in the direct communication range. If, after a time period, it does not receive HELLO packets from other robots listed in its neighbor table, it deletes the correspondent entry line. In this way, a robot will know the robots that can be reached directly (one-hop).

2. *Requiring Task Forward Ant* (**RT-FANT**): it is a packet sent by the robot that has detected a mine (that is the coordinator robot) to know how many robots are available to treat the mine.

3. *Requiring Task Backward Ant* (**RT-BANT**): it is a packet that a robot in Forager State sends as response to a RT-FANT.

4. *Recruitment Fant* (**R-FANT**): it is a packet sent by a coordinator, to the link from which came the higher number of RT-BANT responses; this link has a higher recruitment probability.

5. *Recruitment Bant* (**R-BANT**): it is a packet sent by a robot in response to a positive recruitment by a coordinator.

6. *Leaving position* (**LP**): if a R-BANT, generated by a robot in response to the R-FANT, does not arrive to coordinator within a certain time (it is a timer), and in target's location has arrived the needed robots, the coordinator sends this message informing these robots to continue to explore the area or serve eventually other requests.

In the following the actions, in terms of received packets are described, in order to deeply understand the functioning of the protocol and the difference of packets that are sent during the mission.

For the most time, a robot is in *Forager* executing the exploration task. Its operations are essentially the following:

I. Process packets content: when a robot receives a packet it forwards the packet to another destination.
II. Exploration phase according to exploration algorithm.

A coordinator robot performs these operations:

– FANT Generating and Forwarding: it creates and sends broadcast requests in the network; in this step the coordinator sends a **RT-FANT** to know how many robots are, eventually, available for disarming the found target. The RT-FANT, identified by the triple (ID-Coordinator, Task-ID, ID-FANT), is sent in broadcast to all robots in the transmission range.

– Set waiting timer: after sending the RT-FANT, the coordinator sets a timer to wait the RT-BANT packets sent by robots available to be recruited; after timing out it checks the number of received RT-BANT. If the coordinator does not receive enough replies, analyses the number of received replies: if it does not receive any replies it becomes a Forager, else it creates and sends a new Request Task FANT and forwards in broadcast on the network. If the coordinator has enough replies (RT-BANT) to perform the task, it creates and sends R-FANT on the link with a higher recruitment probability.
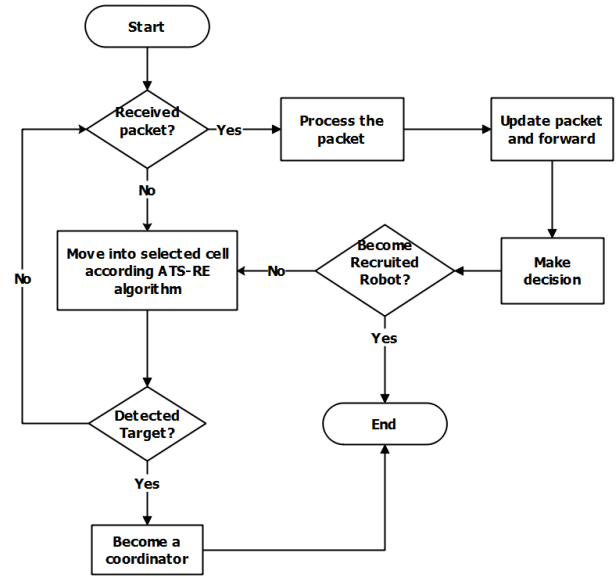

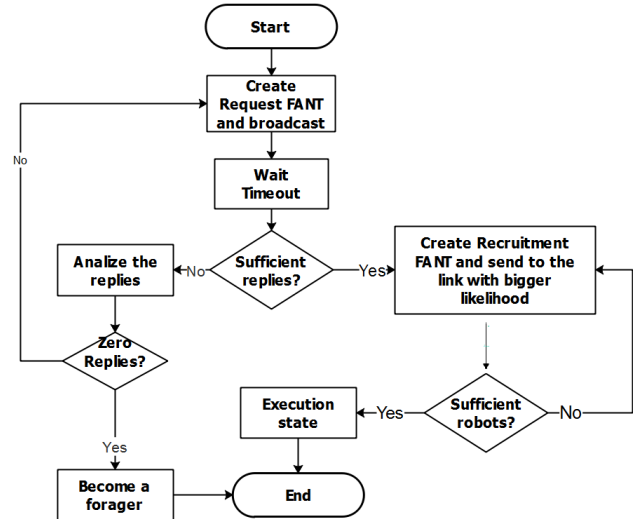Fig. 8: The Flow Chart of a Forager Robot.

Fig. 9: The flow Chart of a Coordinator Robot.

– Wait incoming robots: the coordinator waits for the incoming recruited robots.

– Submit disarming order: When all needed robots are recruited into the interested cell, the coordinator sends a message to announce the starting of the manipulation task of the target.

When a robot receives a RT-FANT packet and sends a RT-BANT to the coordinator, it becomes a *Recruited Robot*. Then, its task is to reach the destination cell. Essentially, the recruited robot moves into the area in order to reach the target's location.
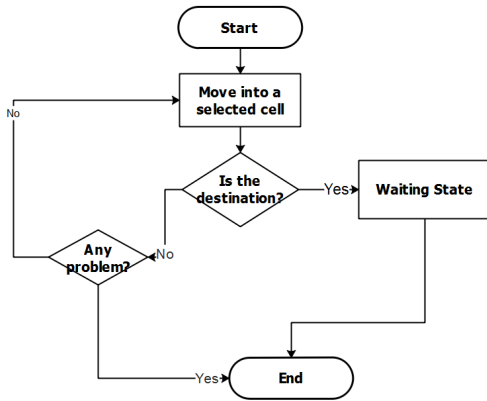
Fig. 10: The Flow Chart of a Recruited Robot.

5.3 Forwarding mechanism of FANT and BANT

In the considered problem, an Ant-based Team Robot Coordination (ATRC) protocol has been applied and it uses typically probabilistic routing tables to establish to which robots distribute the coordination tasks. This routing table is populated and updated on the basis of the packets sent from coordinators to recruited (Forward ANT: R-FANT and RT-FANT) and vice versa (Backward ANT: R-BANT and RT-BANT). To ensure that for every FANT sent on the path from the coordinator to the recruited sent back a BANT on the reverse-path forwarding to the coordinator, each node crossed by the FANT enters its ID in the packet. Once it reaches its destination a Backward ANT (BANT) response is created; in this packet the ID of crossed robots and additional information for updating the routing tables are copied. BANT follows the route tracked by FANT so it reaches the destination host (coordinator). For this behavior, the two considered packets are called Forward (FANT) and Backward ANT (BANT).

During this discovery procedure, BANT updates the entry in the routing table of the node. The law for updating the pheromone is usually based on the path length, that is the number of hops (in terms of robots) crossed by FANT to reach the destination. The routing table in this work are not deterministic, but probabilistic.

Essentially a packet has the following fields:

- ID Coordinator: ID of the coordinator robot and it is added in a RT-FANT;
- Task ID: it is the ID of the task requested by the coordinator. Each time the same coordinator runs different tasks this value is incremented.
- Task Type: in this case there are three tasks (recruiting, disarming and discovery), but this field can

be useful for future purpose and extensions to multiple and more complicated tasks.
- Path Degree p_D: it is a weight given to a path in order to understand which route can be the best according with some specific metrics; it can affect the link selection probability for each link between the current robot and its neighbors.

The ID Coordinator, Task ID and Task Type allow the unique identification of an entry. Initially, when a RT-FANT is sent on the network, each robot receives RT-FANT and creates an entry in the routing table and sets a balanced selection probability of the neighbors. These probabilities are then updated through the response RT-BANT. Each robot that receives an RT-BANT from a particular link, updates the probability associated to that link and decreases the other link probabilities through the use of two concepts:

1. Evaporation
2. Reinforcement.

The evaporation is applied to all links, while reinforcement learning is applied to the link receiving the RT-BANT. The quality of a link depends on the distance of the robot that creates the RT-BANT to the destination (cell where the mine needs to be deactivated). In this way the probability of the link that receives the highest number of RT-BANT increases. Having to submit the RT-FANT in a deterministic way, a robot is able to choose the link with the highest recruitment probability. Also, the received R-BANT contains a recruitment task during the travelling for each link, the robot only executes the process of evaporation. This is made to improve the link selection probability, indicating a high number of robots willing to perform the task requested.

5.4 Task Requesting BANT and FANT Management

When the coordinator sends RT-FANT, only foragers process this packet. If the packet is received by robots that are in other state they forward in broadcast the RT-FANT. The forager receiving RT-FANT performs the same operations below:

– Checking uniqueness of received FANTs: a forager, after receiving a packet containing RT-FANT, controls if it processed this packet previously. In this case the robot drops the packets and carries on its operations, otherwise it saves the ID FANT in a data structure and processes the packet content.
– Process requirements: If the received RT-FANT is not duplicated, the forager checks the required characteristics. If it is able to perform the task, it controls the percentage of BANTs already forwarded
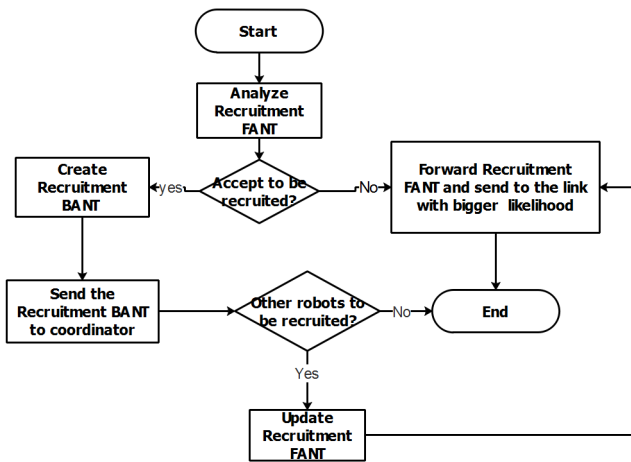
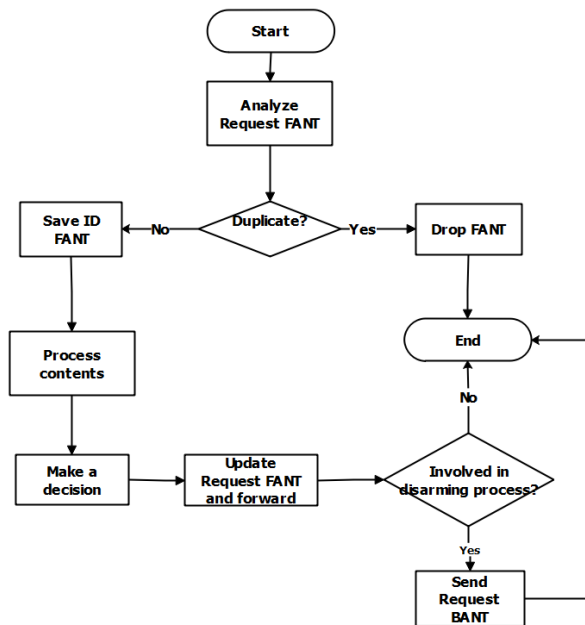Fig. 11: The Flow Chart of a Recruitment FANT and BANT.

Fig. 12: The Flow Chart of a Request FANT and BANT.

to the coordinator, according with previously forwarded FANTs, and decides, in a probabilistic manner, whether to forward or not its answer. Next it creates and sends an RT-BANT to the coordinator. The forager, finally, sends the received RT-FANT in broadcast also if it is not able to perform the task.

5.5 Recruitment FANT and BANT Management

A coordinator, after receiving enough responses by foragers, sends R-FANT on the link that has the highest success probability. The foragers receiving this FANT execute these operations:

– Processing R-FANT: Initially, the forager checks if the FANT has been previously processed; in this case it discards the packet. In other case it adds its identifier in the list of crossed robots by R-FANT and then processes the recruitment request.
– BANT Management: if the robot decides to participate in the disarmament of the target, it creates and sends a R-BANT to coordinator as a recruitment confirmation. The R-BANT updates the routing table of the crossed nodes.
– FANT Forwarding: independently by the response of R-BANT, a forager receiving a R-FANT creates and sends new R-FANT to other robots if there is the need to recruit other robots on the link with higher recruitment probability otherwise, if itself is the last robot, it does not forward any R-FANT.

6 Simulation Experiments

A set of experiments have been performed in order to show and analyze the effectiveness of the proposed approach. For such purpose, a hand-designed simulator have been implemented in Java. This simulator was built from the start as a multi-robot simulator. It is capable of modeling motion, targets, obstacles and local communication in a discrete world, and it can be easily extended to simulate other scenarios and domains since it is generalized. Screenshots of the simulators graphical output option could be seen in Fig. 13, in which the parameters, regarding both exploration and recruiting tasks are represented.

The simulations were executed varying different parameters of the problem. We started to evaluate the Antbased Team Robot Coordination (ATRC) with only exploration in comparison with IAS-SS proposed by Calvo [15] et al. in an area with obstacles and not. Later, we evaluated the performance of our algorithm with both exploration and recruiting strategy applying the wireless communication (ATRC-ERP) or using just stigmergy (ATRC-ERS).

The performance metrics considered for the simulation are:

– Average Task Execution Time: it is the total task execution time evaluated in terms of number of iterations. If more tasks are considered such as exploring, recruiting and disarming, this metric accounts for the total average time necessary to complete all tasks.
– Control Overhead: it accounts for the number of control packets such as R-FANT, R-BANT, RT-
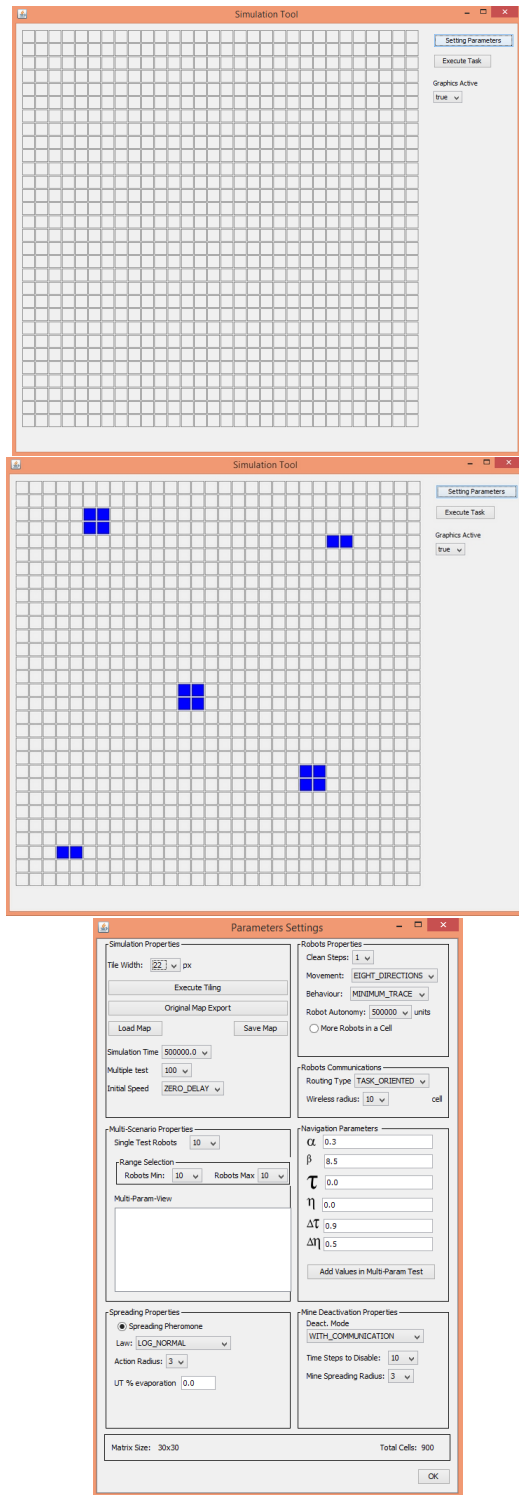
Fig. 13: Simulator Front-end (a) Environment without obstacles (b) Environment with obstacles (c) Parameters Setting.

FANT, RT-BANT sent on the network to perform the protocol operations.

In Table I the simulation parameters are shown. We have used a minimum of 4 robots to disarm the mine (Rmin=4) changing the number of robots in a mined region; the transmission range $R:t$=9; this value has been fixed just to reduce the number of simulations due to space limitations. However, the proposed approach is general and the RW value can be also changed without affecting the algorithm convergence and simulation trend. In addition, we considered a grid area without obstacles and with obstacles, varying, during the simulation tests, the number of grid cells.

6.1 Stigmergy aware Space Discovery vs Protocol aware Bio-inspired strategy

We evaluate, firstly, the performance of the proposed exploration algorithm (ATRC-OE) in comparison with IAS-SS [15]. This last strategy tries inspiration by the inverse ant-colony optimization and it can be considered as a special case of our proposal changing in opportune manner the a_1 and a_2 value. In Fig. 14 the performance of both strategies are depicted varying the value of the parameters in the problem such as a_1 and a_2. The figure considers the total time to explore, increasing the number of robots is shown. As we expected, a higher number of robots reduces the cells discovery time for both IAS-SS and ATRC-OE.

Our approach is able to obtain a lower discovery time through the swarm based solution. The trend is similar both in free environment and in environment with obstacles. Generally, a higher number of robots can assure a lower convergence time. However, we do not need to increase a lot the number of robots but we can stop to a minimum number after which no more gain is obtained

6.2 ATRC-ERP vs ATRC-ERS Performance

In this subsection, ATRC with exploration and recruiting tasks has been evaluated. Two versions of the ATRC with only the stigmergy to perform both tasks (ATRC-ERS) and with the addition and support of the bioinspired protocol (ATRC-ERP), such as explained in Section 5, has been tested under different parameters conditions in order to verify its robustness, convergence and scalability for increasing complexity.

In Fig. 15 and Fig. 16 are shown the convergence time under increasing number of mines and increasing grid size. It is possible to see as the number of mines

that can increase the recruiting time and indirectly affect the discovery time does not affect too much the overall convergence time. This means that the ATRC is able to dynamically adapt its strategy in recruiting and in the discovery in order to maintain low the difference if the complexity increases. Concerning Fig. 16, where the grid size increases, the ATRC-ERP increases the convergence time for larger area. This is expected because with the same number of robots it is necessary to take more time to explore all the un-known area. In this case it is the exploration time that affects the overall convergence time. However, if the number of robots increases the convergence time can be reduced and, after a certain amount, having more robots do not introduce more any benefits in the space discovery time.

In Fig. 17, we compare the two proposed recruiting strategies in a grid area 30x30 with 3 mines to disarm. It can be seen as for a lower number of robots the wireless communication (ATRC-ERP) performs better than the mechanism with only stigmergy (ATRC-ERS) in terms of number of iterations. This means that the communication among the robots allows to complete the tasks (exploring and recruiting/disarming) more quickly. Increasing the size of swarm, the results are comparable because the higher number of robots assures a natural distribution among exploring and disarming tasks leading to a reduced overall execution time. Regarding the number of packets in Fig. 18 it is shown that it mainly depends on the number of mines in the area. The number of robots does not affect the overhead because the proposed algorithm, such as designed, avoids an excessive increase of packets forwarding in the network. The number of packets in the network is nearly constant increasing the number of robots with a certain number of mines; instead increasing the number of mines with a certain number of robots the number of packets increases. This is due to the scalable approach of ATRC that adopts just local information to know where to send packets (highest link selection probability) and global information through the stigmergy avoiding to increase the control overhead to maintain the robot topology and distribute tasks.

In Fig. 19 it is shown the number of packets sent on the network varying the grid area size. In this case the number of packets increases proportionally to the size of area when there are few robots because the network is instable and all tracks cannot be completed and robots are not immediately released to complete the exploration. However, the network reaches the stability increasing the number of robots and with the possibility to distribute both tasks (recruiting and exploration) in the overall area.
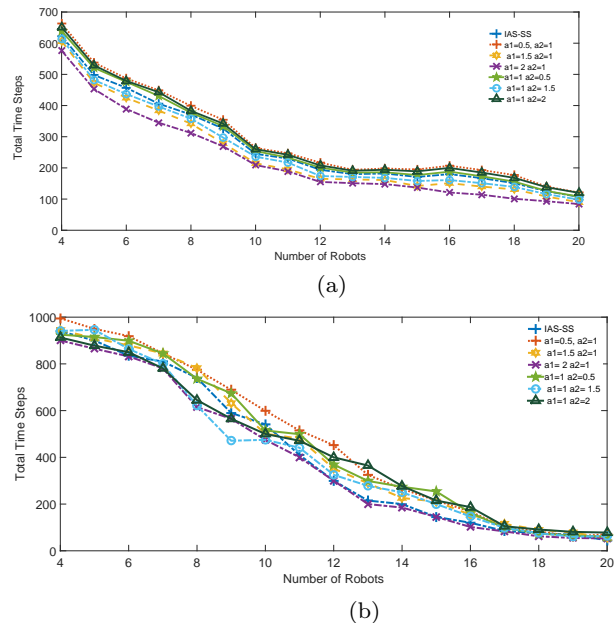

(a)

(b)

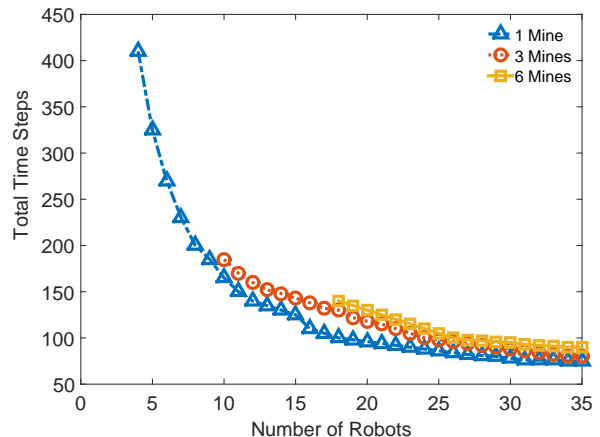Fig. 14: ATR-RE vs IAS-SS (a) Free Environment (b) Obstacle Environment.

Fig. 15: Evaluation of the performance of ATRC in 30x30 grid area in terms of total time steps.

Table 1: Parameters used in the exploration algorithm.

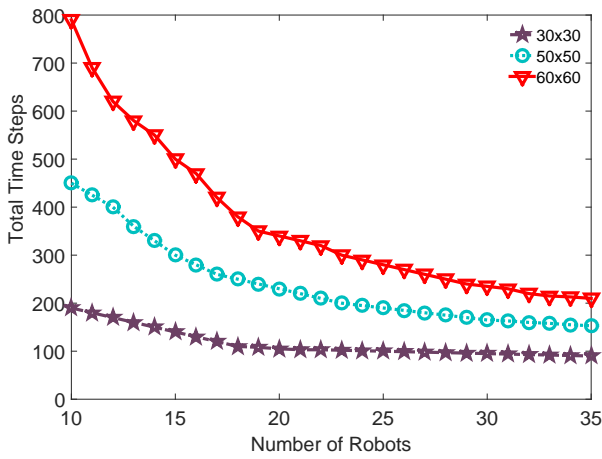| Parameters	Value
Sensing range R_s	4
ρ	0.2
$\Delta\tau_0$	2
φ	1
λ	1
η	0.9
a_1	0.5
a_2	0.5
ε	Uniform [0 1]

Fig. 16: Evaluation of the influence of the dimension of the area in terms of total Time Steps.
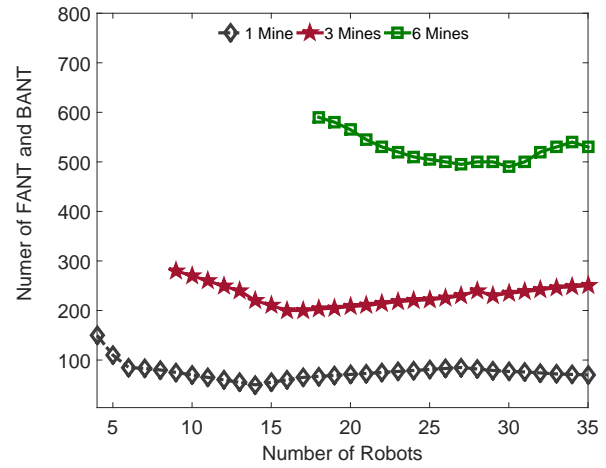

Fig. 18: Evaluation of the influence of the number of mines on the performance of ATRC in 30x30 grid area in terms of number of sent packets.
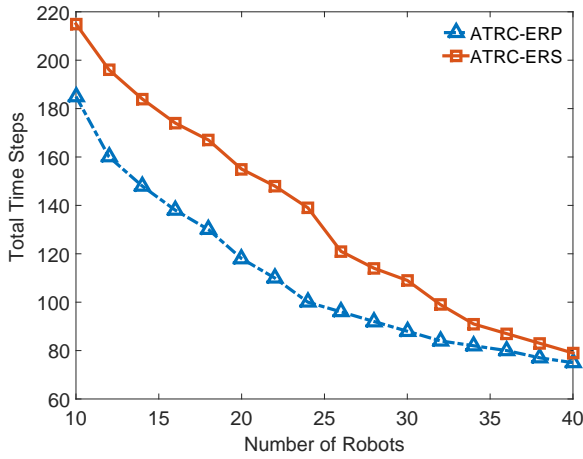

Fig. 17: Comparison between ATRC-ERP and ATRC-ERS in a grid area 30x30 evaluating the total time steps.

Fig. 19: Evaluation of the influence of the dimension of the area on the performance of ATRC in terms of number of sent packets.

7 Conclusions

In this paper, we have formulated a multiple task optimization problem for multiple mobile robots, and these main tasks are: the exploration of unknown area for detection mines and the recruitment for disarming them. We have developed biologically inspired coordination strategies for robot swarms under complex constraints. Based on the Ant Colony Algorithm, some modifications have been carried out to make these algorithms suitable for robot coordination and exploration tasks.

For the exploration task, we have used an indirect communication mechanism between the swarm based on the repelling anti pheromone that tried to spread the robots in different regions of the area. For the re-
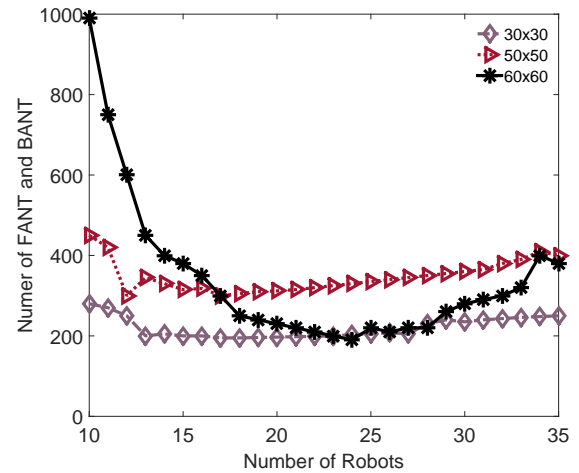
cruitment task, we have proposed two strategies. The first is based on an indirect approach and uses an attractive pheromone to guide the swarm, the second uses a direct communication between the robots. For this purpose, a new protocol able to disseminate recruiting requests and to recall the right number of robots to disarm mines in the minimal amount of time is presented. This protocol applies a probabilistic approach inherited by swarm-robotics in order to offer a scalable and distributed solution to the mine disarming field issue. Such as verified by simulation results, our algorithm reduces the convergence time in comparison with IAS-SS. Moreover, the increase of the number of mines in the field lightly increases the average convergence time while the

increase of the research area (cells) lightly affects the system performance. Self-organization of robots team with the addition of wireless communications to disseminate tasks and coordinate the robots (ATRC) reveal to be a good merging approach in the design of new kinds of protocols in this interesting research area.

Possible future works include the extension of methods to dynamically adjust the number of hops to send the packets during the mission so as to be adaptive to the resource of the robots or other constraints. In addition, the proposed method can be modified to potentially deal with the unknown but mobile targets in an unknown area. Furthermore, further research can also consider the uncertainty concerning unreliable communication than can cause packets loss and inaccurate information, and thus make the overall system more reliable and robust.

References

1. Ducatelle F, Di Caro GA, Gambardella LM (2010) Cooperative Self-Organization in a Heterogeneous Swarm Robotic System. In Proceedings of the genetic and evolutionary computation conference (GECCO), pp. 8794 Fujisawa R, Dobata S, Sugawara K, Matsuno F (2014).
2. Nouyan S, Gro R, Bonani M, Mondada F, Dorigo M (2009) Teamwork in Self-Organized Robot Colonies. IEEE Transaction On Evolutionary Computation, 13(4): 695–711
3. Cassinis R, Bianco G, Cavagnini A, Ransenigo P (1999) Strategies for navigation of robot swarms to be used in landmines detection. In: Eurobot99, pp 211–218.
4. Kumar V, Sahin F (2003) Cognitive Maps in Swarm Robots for the Mine Detection Application. In: IEEE Systems, Man and Cybernetics. 4: 364–3369.
5. Holland O, Melhuis C (1999) Stigmergy, self-organisation, and sorting in collective robotics. Artificial Life, 5(2):173–202
6. Yamauchi B (1998) Decentralized coordination for multirobot exploration. Robot Auton Syst 29(2): 111–118
7. Mobarhani A, Nazari S, Tamjidi AH, Taghirad HD (2011) Histogram based frontier exploration. In: IEEE International Conference on Intelligent Robots and Systems (IROS), pp 1128–1133
8. Prieto RA, Cuadra-Troncoso JM, Alvarez-Sanchez JR, NavarroSantosjuanes IN (2013) Reactive navigation and online SLAM in autonomous frontier based exploration. In Natural and Artificial Computation in Engineering and Medical Applications, 7931, pp 45–55
9. Choset H, Acar E, Rizzi AA, Luntz J (2000) Exact cellular decompositions in terms of critical points of Morse functions. In: IEEE International Conference on Robotics and Automation (ICRA), pp 2270–2277
10. Wattanavekin T, Ogata T, Hara T, Ota J (2013) Mobile Robot Exploration by Using Environmental Boundary Information. ISRN Robotics Article ID 954610.
11. Solanas A, Garcia M (2004) Coordinated multi-robot exploration through unsupervised clustering of unknown space. In: IEEE International conference on intelligent robots and systems (IROS), pp 717–721
12. Gifford CM, Webb R, Bley J, Leung D, Calnon M, Makarewicz J, Banz B, Agah A (2010) A novel low-cost, limited resource approach to autonomous multi-robot exploration and mapping. Robot Auton Syst 58(2):186202
13. Zheng X, Koenig S, Kempe D, Jain S (2010) Multi-robot forest coverage for weighted and un-weighted terrain. EEE Transactions on Robotics. 26(6):1018–1031
14. Hazon N, Kaminka G, (2005) Redundancy, efficiency, and robustness in multi-robot coverage. In: IEEE International Conference on Robotics and Automation (ICRA),pp 735–741
15. Calvo R, De Oliviera JR, Figueiredo M, Romero RAF (2012) A Bio-inspired coordination strategy for Controlling of Multiple Robots in Surveillance Taks. International Journal on Advances in Software 5(3-4):146–165.
16. Ranjbar-Sahraei B, Weiss G, Nakisaee A (2012) A multi-robot coverage approach based on stigmergic communication. In I. J. Timm & C. Guttmann (Eds.), Multiagent system technologies, lecture notes in computer science, (Vol. 7598, pp 126–138). Berlin: Springer.
17. Masr M (2013) A biologically inspired swarm robot coordination algorithm for exploration and surveillance. In: IEEE 17th International conference on intelligent engineering systems (INES), pp 271275
18. Kuyucu T, Tanev I, Shimohara K (2015) Superadditive effect of multi-robot coordination in the exploration of unknown environments via stigmergy. Neurocomputing, 148: 83–90
19. Ravankar A et al. (2016). On a bio inspired hybrid pheromone signalling for efficient map exploration of multiple mobile service robots. In Art Life Robotics 21:221–231
20. Tan Y, Zheng ZY (2013) Research Advance in Swarm Robotics. Defence Technology 9(1):1839 Yamauchi B (1998) Decentralized coordination for multirobot exploration. Robot Auton Syst 29(2): 111–118,
21. Chen X, Kong Y, Fang X (2013) A fast two-stage ACO algorithm for robotic path planning. Neural Comput & Applic, 22:313–319.
22. Hidalgo-Paniagua A, Vega-Rodrguez M, Ferruz J, Pavn N (2015) Solving the multi-objective path planning problem in mobile robotics with a firefly-based approach. Soft Comput 1–16
23. Liu J, Yang J, Liu H, Tian X, Gao M (2016) An improved ant colony algorithm for robot path planning. Soft Comput 1–11 doi:10.1007/s00500-016-2161-7
24. Pessin G et al (2013) Swarm intelligence and the quest to solve a garbage and recycling collection problem. Soft Comput 17:2311–2325
25. Liu C, Kroll A (2014) Memetic algorithms for optimal task allocation in multi-robot systems for inspection problems with cooperative tasks. J. Soft Comput. 19(3): 567–584.
26. De Rango F, Palmieri N, Yang XS, Marano S (2105) Bioinspired Recruiting and Exploring Tasks in a Team of Distributed Robots in a Mined Region. In International Symposium on Performance Evaluation of Computer & Telecommunication Systems, (SPECTS) DOI: 10.1109/SPECTS.2015.7285279
27. Di Caro G, Ducatelle F, Gambardella LM (2005) AntHocNet: An adaptive nature-inspired algorithm for routing in mobile ad hoc networks. Eur. Trans. Telecommun. 16: 443455 Dorigo M, Bonabeau E, Theraulaz G (2000) Ant algorithms and stigmergy. Future Generation Computer Systems, 16(8), 851–871.

28. Bouazizi I (2002) ARAThe Ant-Colony Based Routing Algorithm for MANETs. In Proc. International Conference on Parallel Processing Workshops, pp 79–85.

29. Baras JS, Mehta H (2003). A Probabilistic Emergent Routing Algorithm (PERA) For Mobile ad Hoc Networks. In proc. of the WiOpt '03: Modeling and Optimization in Mobile, ad Hoc Networks.

30. Singh G, Kumar N, Verma, A K (2012) Ant colony algorithms in MANETs. Journal of Network and Computer Applications. 35(6): 1964–1972.

31. De Rango F, Tropea M (2009) Energy saving and load balancing in wireless ad hoc networks through ant-based routing. In International Symposium on Performance Evaluation of Computer & Telecommunication Systems, (SPECTS), pp 77–84.

32. De Rango F, Palmieri N (2012) A swarm-based robot team coordination protocol for mine detection and unknown space discovery. In International Wireless Communications and Mobile Computing Conference (IWCMC), pp703–708.

33. Dias MB, Zinck MB, Zlot RM, Stentz A (2004) Robust multirobot coordination in dynamic environments. In: IEEE International Conference on Robotics and Automation (ICRA), pp 3435–3442.

34. Palmieri Nunzia, Yang Xin-She, De Rango Floriano and Marano Salvatore (2017) Comparison of bio-inspired algorithms applied to the coordination of mobile robots considering the energy consumption. Neural Computing and Applications. pp 1–24, First Online https://link.springer.com/article/10.1007/s00521-017-2998-4

35. Fujisawa R, Dobata S, Sugawara K, Matsuno F (2014) Designing pheromone communication in swarm robotics: Group foraging behaviour mediated by chemical substance. Swarm Intelligence 8(3):227–246.

36. Pinciroli C, OGrady R, Christensen A, Dorigo, M (2009) Selforganised recruitment in a heterogeneous swarm. In Proc. of the 14th international conference on advanced robotics (ICAR), pp 1–8

37. Meng Y, Gan J (2008) A distributed Swarm Intelligence based Algorithm for a Cooperative Multi-Robot Construction Task. In: IEEE Swarm Intelligence Symposium (SIS), pp. 1–6

38. Countryman, S. M., Stumpe, M. C., Crow, S. P., Adler, F. R., Greene, M. J., Vonshak, M., and Gordon, D. M. (2015). Collective search by ants in microgravity. Frontiers in Ecology and Evolution, 3:25.

39. Dorigo, M. and Stutzle, T. (2003). The ant colony optimization metaheuristic:algorithm, applications, and advance. In Handbook of Metaheuristic, pages 250–285. Springer US.