Proceedings of the 2011 IEEE International Symposium on
Safety,Security and Rescue Robotics
Kyoto,Japan,November 1-5 2011

Ensuring Ad Hoc Connectivity in Distributed Search with Robotic Darwinian Particle Swarms

Micael S. Couceiro, Rui P. Rocha
Institute of Systems and Robotics
University of Coimbra, Pólo II, 3030-290
Coimbra, Portugal
{micaelcouceiro, rprocha}@isr.uc.pt

Nuno M. F. Ferreira
RoboCorp, Electrical Engineering Department,
Engineering Institute of Coimbra, 3030-199
Coimbra, Portugal
nunomig@isec.pt

Abstract**—**This paper presents an enforcing multi-hop network connectivity algorithm experimentally validated using a modified version of the Darwinian Particle Swarm Optimization (*DPSO*), denoted as *RDPSO* (Robotic *DPSO*) on groups of simulated robots performing a distributed exploration task. This work aims to overcome limitations of multi-robot systems (*MRS*) in difficult scenarios (*e.g.*, search and rescue) concerning the need and the ability to actively maintain an available inter-robot communication channel, through the development of effective multi-robot cooperation without relying on a preexisting communication network. Although there is no linear relationship between the number of robots (*i.e.*, nodes) and the maximum communication range, experimental results show that the decreased performance by the developed algorithm under communication constraints can be overcome by slightly increasing the number of robots as the maximum communication range is decreased.

Keywords**:** *distributed search, swarms, MANET, deployment.*

I. INTRODUCTION

In spite of the potential advantages of multi-robot systems (*MRS*) related with space and time distribution, it is necessary that each robot maintains a sufficient and consistent level of awareness about the mission assigned to the team and about its teammates in order to attain effective cooperation. Cooperative architectures [1] usually assume a reliable pre-existent communication network to support teamwork. However, in many cases, robots have to move to complete their tasks while maintaining communication among themselves without the aid of a communication infrastructure. Just like in *MRS* where groups of robots interact to accomplish their goals [2], particle optimization algorithms such as the well-known Particle Swarm Optimization (*PSO*) [3] and the Darwinian Particle Swarm Optimization (*DPSO*) [4] use groups of interacting virtual agents, *a.k.a.* particles, in order to achieve their optimization. However, contrarily to virtual agents, robots are designed to act in the real world where obstacles need to be taken into account. Also, and since that in certain environments or applications, such as hostile environments, search and rescue, disaster recovery, battlefields, space and others, the communication infrastructure may be damaged or missing, the self-spreading of autonomous mobile nodes of a mobile ad hoc network (*MANET*) over a geographical area needs to be considered.

An algorithm to guarantee *MANET* connectivity is proposed, being used in an adapted version of the *DPSO*, denoted as *RDPSO* (Robotic *DPSO*). In order to establish the initial deployment of robots while preserving the *MANET* connectivity, this paper also proposes a novel approach inspired on the *Spiral of Theodorus*. The algorithm is demonstrated in multi-robot exploration tasks, wherein each robot is represented by a particle that needs to be evaluated at each iteration. After each set of evaluations, robots communicate to share the objective information (*e.g.*, cost or fitness) needed to progress to the next iteration of the algorithm while avoiding obstacles and fulfilling the *MANET* connectivity. Bearing these ideas in mind, this paper is organized as it follows. Section II presents some other works in the area. A brief review of the *RDPSO* algorithm proposed in [5] is given in section III and extended in section IV and V to maintain *MANET* connectivity. Experimental results validating the proposed algorithm are demonstrated in section VI. Finally, section VII outlines the main conclusions and future work.

II. RELATED WORK

Communication constitutes one of the most important resources for more effective cooperation among robots and improved robust collective performance. The development of robot teams for surveillance or rescue missions, require that robots have to be able to maintain communication among them without the aid of a communication infrastructure. Besides that, robots also need to be able to deploy and maintain a *MANET* in order to explicitly exchange information within multi-hop network paths, thus not restricting unnecessarily the team's range.

In [6], it is presented an experimental study of strategies for maintaining end-to-end communication links for tasks such as surveillance, reconnaissance, and search, where team connectivity is required. The authors show experimental results obtained using a multi-robot testbed in representative urban environments that do not capture all static and dynamic aspects of an environment's radio propagation characteristics.

The authors in [7] proposed a system to maintain multi-hop routes between nodes of sufficient quality, in order to avoid the network becoming disconnected. Thus, a measure of the communication link quality is used [8] instead of the commonly used communication range, such as [9]. Like in our approach the robots movements are restricted if necessary by using this measure. However, the use of Spring-

Damper Systems (*SDS*s) to force the connectivity between nodes introduces constraints that traditional and simpler allocation methods do not face.

An adapted version of the *PSO* to distributed unsupervised robotic learning in groups of robots with only local information is developed in [10]. The main difference between this algorithm and traditional *PSO* is that each robot only takes into consideration the information of the robots within a fixed radius r (omnidirectional communication). Nevertheless, contrarily to the experimental results shown in this work, the authors does not use multi-hop connectivity and does not apply any kind of algorithm to ensure communication between robots.

Another behavior-based strategy to maintain *MRS* connectivity has been studied in [11] and, successively, in [12] and [13]. The authors presented the extension of the Null-Space-based Behavioral (*NSB*) approach to control a group of marine vehicles to execute multi-robot missions. This is a promising approach since it would be possible to merge the behaviors of the *RDPSO* with different priorities, in order to define the final motion directives of the robots. However, the design choices concerning how to organize the behaviors in priority represents a higher complexity, since these choices derive from practical considerations related to both the mission objective and the hardware/software features of the robotic system.

A new graph structure called Separated Distance Graph (*SDG*) is proposed in [14]. *SDG* allows specifying and maintaining a large class of constraints, like ensuring that each individual robot never departs from the group by more than a certain distance, maintaining a formation in which every robot can be seen by its neighbors, and keeping up the wireless communication within a complete group of robots. However, experimental results show that the algorithm is not optimal with respect to the number of robots required to solve a task.

III. ROBOTIC DARWINIAN PSO

This section briefly presents the *RDPSO* algorithm proposed in [5]. The *DPSO* [4] is an evolutionary algorithm that extends the well-known *PSO* [3] using natural selection, or survival-of-the-fittest, to enhance the ability to escape from local optima. The *RDPSO* approach mainly differs from the common *DPSO* based on three general assumptions:

- *Social Exclusion and Social Inclusion* - The *RDPSO* is represented by *multiple swarms* (*i.e.*, group of robots) where each swarm individually performs just like an ordinary *PSO* in search for the solution and some rules governs the whole population of robots. Those socially excluded robots, instead of searching for the objective function's global optimum like the other robots in the active swarms do, they basically randomly wander in the scenario. However, they are always aware of their individual solution and the global solution of the socially excluded group.
- *Obstacle Avoidance* - A new cost or fitness function is defined in such a way that it would guide the robot to perform the main mission while avoiding obstacles. For this purpose it is assumed that each robot is equipped with sensors capable of sensing the environment for obstacle detection within a finite sensing radius r_s. A monotonic and positive *sensing function* $g(x_n[t])$ at each discrete time, or iteration, $t \in \mathbb{N}$, is defined. This function depends on the sensing information, *i.e.*, distance from the robot to obstacle.
- *Ensuring MANET Connectivity* - Robots' position need to be controlled in order to maintain the communication based on constraints such as maximum distance or minimum signal quality. The way network will be forced to preserve connectivity depends on the characteristics of the communication. Assuming that the network supports multi-hop connectivity, the communication between two end nodes (*i.e.*, robots) is carried out through a number of intermediate nodes whose function is to relay information from one point to another. Considering that nodes are mobile, it is necessary to guarantee the communication between all nodes. The robots' position are updated by means of the ensuring *MANET* connectivity algorithm that will be described in next section.

Furthermore, to model the swarm, each particle, denoted by n, moves in a multidimensional space according to position vector ($x_n[t]$) and velocity vector ($v_n[t]$), which are highly dependent on local best vector ($\breve{x}_n[t]$) and global best vector ($\breve{g}_n[t]$) information. The position of robot n that maximizes (or minimizes) the monotonically decreasing (or increasing) *sensing function* $g(x_n[t])$ is represented by $\breve{x}_n^g[t]$. The size of the vectors depends on the dimension of the multidimensional space.

$$\begin{aligned} v_n[t+1] = w v_n[t] &+ c_1 r_1 (\breve{g}_n[t] - x_n[t]) \\ &+ c_2 r_2 (\breve{x}_n[t] - x_n[t]) \\ &+ c_3 r_3 \left(\breve{x}_n^g[t] - x_n[t]\right) \end{aligned} \quad (1)$$

$$x_n[t+1] = x_n[t] + v_n[t+1] \quad (2)$$

wherein the coefficients w, c_1, c_2 and c_3 assign weights to the inertial influence, the global best ("social" component), the local best ("cognitive" component) and the obstacle avoidance when determining the new velocity, respectively. The value of c_3 depends on several conditions related with the main objective (*i.e.*, minimize a cost function or maximize a fitness function) and the sensing information (*i.e.*, monotonicity of the *sensing function* $g(x_n[t])$). Parameters r_1, r_2 and r_3 are random vectors wherein each component is generally a uniform random number between 0 and 1.

IV. ENSURING MANET CONNECTIVITY

It has generally been assumed in *MRS* that each robot has the ability to communicate with any other robot with small consideration for the quality and performance of the wireless communication network. Although being valid in particular situations, such an assumption does not generally hold. Since robots may move apart to further areas, it is important to have a pervasive networking environment for communications among robots. Furthermore, without a preexistent infrastructure, robots need to be able to act as intermediate nodes, *i.e.*, routers, in order to relay information from one point to another, thus supporting multi-hop communication in a *MANET* [15].

A. Problem Statement

Consider a population of N robots where each robot is both an exploring agent of the environment and a mobile node of a *MANET* that performs packet forwarding, according to a paradigm of *multi-hop communication*. The goal is to ensure that the robots explore an unknown environment, while ensuring that the *MANET* remains connected throughout the mission.

B. General Approach

The connectivity between robots can be described by means of a *link matrix* $L = \{l_{ij}\}$ for an N-node network, where each entry represents the link between node (*i.e.*, robot) i and j. The link is defined accordingly with the users' preferences. The most common approaches include:

1. Calculating the l_{ij} values as functions of the distance between pairs of nodes indicating the *link distance* between them [9];
2. Calculating the l_{ij} values as functions of the radio quality signal between pairs of nodes indicating the *link quality* between them [8].

Trying to maintain the network connectivity by only taking into account the communication range d_{max} (approach 1) does not match reality since the propagation model is more complex – the signal depends not only on the distance but also on the multiple paths from walls and other obstacles (approach 2). However, in simulation, the communication distance is a good approach and it is easier to implement. Depending on the chosen approach (1 or 2), an *adjacency matrix* $A = \{a_{ij}\}$ can be defined based on the maximum distance or minimum radio quality signal between nodes, respectively [15]. The adjacency matrix, *i.e.*, one-hop connectivity matrix, where a 1 entry at (i,j) indicates a connection between node i and j and a 0 entry at (i,j) indicates no connection between node i and j, represents the neighbors of each node, *i.e.*, direct connection between robots.

$$a_{ij} = \begin{cases} 1\,, connection\ between\ node\ i\ and\ j \\ 0\,, no\ connection\ between\ node\ i\ and\ j \end{cases} \tag{3}$$

Note that the diagonal elements (*i.e.*, when $i = j$) of the adjacency matrix are set equal to 0. If the communication system supports the relay of messages to distant nodes via intermediate nodes, then multi-hop connections can be made. Using the hop distances, *i.e.*, the smallest numbers of hops to connect non-adjacent robots, the zero-valued off-diagonal entries in the adjacency matrix can be manipulated in order to create a multi-hop *connectivity matrix* $C^{(k)} = \{c_{ij}^{(k)}\}$, for which the entry at (i,j) represents the minimum number of hops necessary to connect node i and j, and k represents the iteration which varies with the number of hops the network can handle. $N-1$ is the maximum number of possible hops. The calculation of the connectivity matrix can be defined as:

$$c_{ij}^{(k)} = \begin{cases} h\,, node\ i\ connected\ to\ j\ by\ h \le k\ hops \\ 0\,, otherwise \end{cases} \tag{4}$$

Note that the diagonal elements of the connectivity matrix $(i = j)$ are set equal to 0. Furthermore, the adjacency matrix is the first iteration in calculating the connectivity matrix $(C^{(1)} = A)$. When $k > 1$ (*i.e.*, for multi-hop connections) an auxiliary matrix $B^{(k)} = \{b_{ij}^{(k)}\}$ is then calculated based on the iteration (number of hops):

$$b_{ij}^{(k)} = \begin{cases} 0\,, c_{ij}^{(k-1)} > 0 \\ k\,, \sum_{t=1}^{N} c_{it}^{(k-1)}\, b_{tj}^{(k-1)} > 0\ and\ c_{ij}^{(k-1)} = 0 \end{cases} \tag{5}$$

Note that the diagonal elements $(i = j)$ of the auxiliary matrix are set equal to 0 and $(B^{(1)} = A)$. The connectivity matrix can now be calculated using the following equation:

$$C^{(k)} = C^{(k-1)} + B^{(k)} \tag{6}$$

After $N-1$ iterations, $C^{(N-1)}$ represents the multi-hop network connectivity. The existence of zero elements (except diagonal elements) indicates no connection between node i and j even using multi-hop. In this case it is necessary to implement an algorithm to ensure the complete connectivity of the network. One strategy is to define a *binary connectivity matrix* $C_B = \{c_{B_{ij}}\}$ wherein each non-zero element of the connectivity matrix matches the logic value 1.

$$c_{B_{ij}} = \begin{cases} 1\,, c_{ij}^{(N-1)} \neq 0 \\ 0\,, c_{ij}^{(N-1)} = 0 \end{cases} \tag{7}$$

Performing an element-by-element multiplication between the link matrix and the logical inverse (binary *NOT*) of the binary connectivity matrix yields a *break matrix* $C_{break} = \{c_{break_{ij}}\}$ containing the values that represent the break of connection between the nodes. In the case where each robot corresponds to a node, in order to overcome the non-connectivity between them, the desired position of each robot, *i.e.*, $x_n[t+1]$, must be controlled since it influences the link matrix. One way to ensure the full connectivity of the *MANET* is to "force" each robot to communicate with its nearest neighbor that has not chosen it as its nearest neighbor. Since the connectivity depends on the distance/signal quality, connectivity between nodes may be ensuring by computing the minimum/maximum value of each line of adjacency matrix A, after excluding zeros and (i,j) pairs previously chosen. Therefore, equation (1) can be rewritten as:

$$\begin{aligned} v_n[t+1] = {} & w v_n[t] + c_1 r_1(\breve{g}_n[t] - x_n[t]) \\ & + c_2 r_2(\breve{x}_n[t] - x_n[t]) \\ & + c_3 r_3\big(\breve{x}_n^{g}[t] - x_n[t]\big) \\ & + c_4 r_4\big(\breve{x}_n^{m}[t] - x_n[t]\big) \end{aligned} \tag{8}$$

where c_4 and r_4 are the communication ensuring weight and respective random vector, while $\breve{x}_n^{m}[t]$ is the position of the nearest neighbor of robot n increased by the maximum communication range d_{max} toward robot's current position. In this work, the multi-hop connectivity matrix $C^{(N-1)}$ and auxiliary matrices (C_B and C_{break}) will only be used as information about network topology (Algorithm 1).

Note that having multiple swarms, which is inherent to the proposed *RDPSO* algorithm, enables a distributed approach because the network that was previously defined by the whole population of robots is now divided into multiple smaller networks (one for each swarm,) thus decreasing the number of robots and the information exchanged between robots of the same network. In other words, robots interaction with other robots through communication is confined to local interactions inside the same group (swarm), thus making *RDPSO* scalable to large populations of robots. The exchanged data concerning to the signal quality or robot's position will allow the implicit processing of the enforcing network connectivity algorithm by the team in a distributed way. In other words, every robot needs to be aware of the position or signal quality of all other robots in the same swarm in order to compute the ensuring network connectivity algorithm. This is a limitation of the algorithm since all robots need to be equipped with localization systems (*e.g.*, *GPS*). An alternative to it would be extending the *GPS* capabilities of some robots to non-*GPS* robots [16] using strategies to find the teammates position under their visual range. For instance, if robots are equipped with laser range finders, retro-reflective markers can be used for recognition. Since the implementation of such strategies is out of scope of this paper they will not be taken into account. Also, one of the major concerns in this approach is that all robots should have an initial deployment that preserves the communication between robots in each swarm. Moreover, it is also known that in classical *PSO* algorithms particles need to be scattered throughout the scenario.

Next section presents a novel methodology to establish an initial planar deployment of the robots that preserves the connectivity of the *MANET*, while spreading out the robots as much as possible.

ALGORITHM I

ENSURING NETWORK CONNECTIVITY

1	Determines $\boldsymbol{N} \times \boldsymbol{N}$ link matrix $\boldsymbol{L}$
2	Calculates $\boldsymbol{N} \times \boldsymbol{N}$ adjacency matrix $\boldsymbol{A}$
3	Initialize $\boldsymbol{N} \times \boldsymbol{N}$ connectivity and auxiliary matrix $\boldsymbol{C}^{(1)} = \boldsymbol{A}$, $\boldsymbol{B}^{(1)} = \boldsymbol{A}$
4	for $\boldsymbol{k} = 2$ to the longest hop ($\boldsymbol{k} = \boldsymbol{N} - \boldsymbol{1}$)
5	for all node pairs $(\boldsymbol{i}, \boldsymbol{j}) = (\boldsymbol{1}, \boldsymbol{1})$ to $(\boldsymbol{N}, \boldsymbol{N})$
6	if $\boldsymbol{i} = \boldsymbol{j}$ or $\boldsymbol{c}_{\boldsymbol{ij}}^{(\boldsymbol{k})} > 0$, skip to the next node pair
7	if $\sum_{t=1}^{N} \boldsymbol{c}_{\boldsymbol{it}}^{(\boldsymbol{k}-\boldsymbol{1})} \boldsymbol{b}_{\boldsymbol{tj}}^{(\boldsymbol{k}-\boldsymbol{1})} > 0$ and $\boldsymbol{c}_{\boldsymbol{ij}}^{(\boldsymbol{k}-\boldsymbol{1})} = 0$, set $\boldsymbol{b}_{\boldsymbol{ij}}^{(\boldsymbol{k})} = \boldsymbol{k}$
8	End
9	Calculates $\boldsymbol{C}^{(\boldsymbol{k})} = \boldsymbol{C}^{(\boldsymbol{k}-\boldsymbol{1})} + \boldsymbol{B}^{(\boldsymbol{k})}$
10	End
11	Calculates $\boldsymbol{N} \times \boldsymbol{N}$ binary connectivity and break matrix $\boldsymbol{C}_{\boldsymbol{B}}$, $\boldsymbol{C}_{\boldsymbol{break}}$
12	If the connectivity depends on the distance/quality, find the minimum/maximum value of each line of adjacency matrix $\boldsymbol{A}$, excluding zeros and (i, j) pairs previously chosen
13	Computes equation 8

V. INITIAL DEPLOYMENT

One of the fundamental problems in *MRS* that has not been fully addressed is how to deploy a group of robots over an environment to carry out sensing, surveillance, data collection, or distributed servicing tasks [17]. The deployment problem considers the number of needed robots for a specific situation (*e.g.*, objective, scenario, constraints) and their initial locations. For instance, in a search and rescue (*SR*) mission, robots need to move in a catastrophic scenario in order to find survivors. When the robots are transported to the catastrophe site, they need to be deployed. The deployment problem is to decide how many robots and where they will be initially located before performing the *SR* mission using their control strategy (*e.g.*, coverage, herding, formation and others).

One of the common approaches in the initial deployment of mobile robots is using a random distribution along the scenario [18]. This methodology is the simplest way for deploying robots and in most situations (*e.g.*, *SR*), the distribution of the points of interest (*e.g.*, victims) is random. However, in real situations, it is necessary to ensure several constraints of the system. If the network supports multi-hop connectivity, this kind of constraints may significantly increase the complexity of the random distribution since it would depend not only on the communication constraints but also on the number of robots and their own position. Moreover, random deployment may cause unbalanced deployment therefore increasing the number of needed robots and energy depletion.

This work tries to get benefit of a random planar deployment of robots, while eliminating the disadvantages inherent to it and taking into account the communication constraints by using a deployment strategy based on the *Spiral of Theodorus*, *a.k.a.* square root spiral. This spiral is composed of contiguous right triangles, formerly called rectangled triangles, with each cathetus, *a.k.a.* leg, having a length equal to 1 [19]. Triangle's hypotenuses h_i is given by the square root to a consecutive natural number, with $h_1 = \sqrt{2}$. The use of the spiral of Theodorus to carry out the initial deployment of robots, requires two adjustments: *i*) the initial position of each robot is set at the further vertex of the centre of the spiral for each right triangle with a random orientation; and *ii*) the size of the cathetus is set as the maximum communication range d_{max} (instead of having the unit length 1) consequently changing the triangles' hypotenuses h_i to the product between the maximum communication range d_{max} and the square root of the consecutive natural number. These assumptions allow having an initial deployment of each swarm in the target area which depends on both the number of robots in the swarm and the communication constraints (Fig. 1). The growth of the angle φ_k of the next triangle (or spiral segment) k, can be calculated using the trigonometric properties of right triangles.

$$\varphi_k = arctan\left(\frac{1}{\sqrt{k}}\right) \tag{9}$$

The total angle φ_n for the n^{th} robot is calculated as the cumulative sum presented above.

$$\varphi_n = \sum_{k=1}^{n} \varphi_k \quad (10)$$

Once again, using the trigonometric functions, the initial planar position of each robot *n* can easily be calculated.

$$x_n[0] = x_0 + \begin{bmatrix} d_{max}\sqrt{n+1} \cdot cos(\varphi_n) \\ d_{max}\sqrt{n+1} \cdot sin(\varphi_n) \end{bmatrix} \quad (11)$$

where x_O is the centre of the spiral which can be randomly assigned at each trial ensuring the efficiency of the stochastic algorithms.

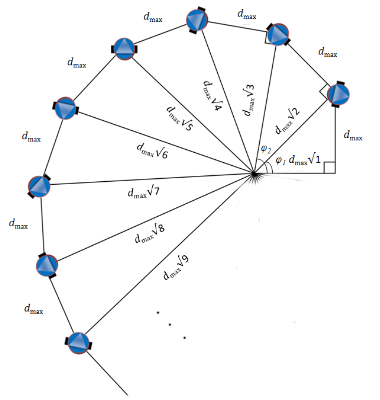

Fig. 1. Initial deployment based on the spiral of Theodorus of a swarm of robots using the *RDPSO* algorithm.

In order to understand the distribution of the robots over a scenario, the growth of the radius of the spiral at a certain robot *n* can be calculated using the following equation.

$$\Delta r = d_{max}\left(\sqrt{n+1} - \sqrt{n}\right) \quad (12)$$

VI. Experimental Results

In this section, it is explored the effectiveness of using a modified version of the *DPSO*, denoted as *RDPSO*, on a group of simulated robots while performing distributed unsupervised learning with local and global information under communication constraints. The number of robots *N* is set equal to the number of particles and the number of nodes in the network. Robots are deployed in the search space in a spiral manner (section V) where the radius depends on the maximum communication distance and the number of robots in the population. Since the *RDPSO* is a stochastic algorithm, every time it is executed it may lead to different trajectory convergence. Therefore, multiple test groups of 100 trials with 300 iterations each were considered. It is used a minimum, initial and maximum number of 1, 3 and 6 swarms, respectively, independently of the population of robots. The search space is represented by a Gaussian distribution consisting on a function of two variables of the search space, *x* and *y*-axis, which represents the position of the robot in meters. The particles will then move in an outdoor scenario of 300x300 meters with a regular density of obstacles randomly deployed at each trial where the *z*-axis represents the value of the objective function. In order to improve the interpretation of the algorithm performance, results were normalized in a way that the objective of robotic team is to find the optimal value of 1, while avoiding obstacles and ensuring the *MANET* connectivity. For the sake of simplicity and without lack of generality, a distance criterion d_{max} was used in simulations to model communication constraints. Moreover, the conclusion presented in this section can be extrapolated to real robots by replacing maximum distance by minimum signal quality since most part of wireless equipments benefit from the Received Signal Strength Indicator (*RSSI*) [20]. The maximum communication distance d_{max} will then vary depending on the chosen wireless protocol. Four conditions were described: *i*) Existence of a communication infrastructure (*i.e.*, without communication constraints $\equiv d_{max} \to \infty$); *ii*) *WiFi*; *iii*) *ZigBee*; *iv*) *Bluetooth*. Table I depicts the maximum communication distance adapted from a comparison between the key characteristics of each wireless protocol in [21]. The mean between the minimum and maximum range shown in [21] was considered as the maximum communication distance d_{max}.

TABLE I

TYPICAL MAXIMUM COMMUNICATION DISTANCES OF WIRELESS PROTOCOLS

	WiFi	ZibBee	Bluetooth
d_{max} [m]	100	55	10

In order to demonstrate the performance of the algorithm while constrained by the communication distance the number of robots will be increased until it achieves a mean value equal or superior to 90% of the optimal value 1 (Fig. 2).

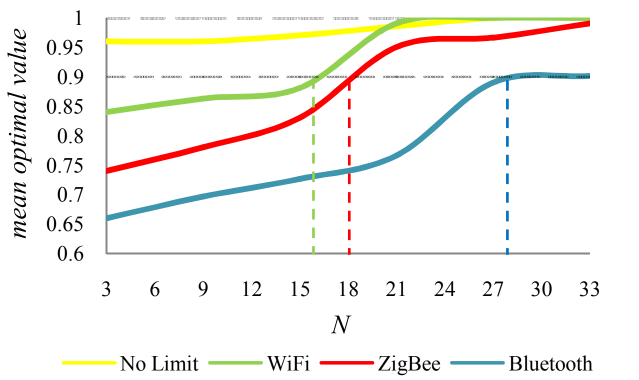

Fig. 2. Performance of the algorithm under limited communication distance.

It can be verified that in order to obtain a performance closer to the one obtained in a system with no communication constraints, the number of used robots must be higher. In a 300x300 meters scenario, by imposing a 100 meters range limitation (*i.e., WiFi*) 16 robots are needed. In the case of *ZigBee*, the range limitation is 55 meters, resulting in the need of 18 robots. The number of needed robots dramatically increases to 28 robots when the system has a range limitation of 10 meters (*i.e.*, *Bluetooth*). In order to allow a straightforward comparison of the performance of the algorithm under different communication constraints, while increasing the number of robots, Table II depicts the most relevant statistical measures of the simulation results. It should be noted that, although not having a significant influence on the performance of the wireless network while using *WiFi* (supports up to 2007 nodes) or *ZigBee* (supports up to 65000 nodes) the increase in the number of robots from a range of 3 to 33, may have a bigger influence when considering *Bluetooh* as the performance could be slightly decreased since each network (*i.e.*, piconet) can only be formed by 8 nodes resulting in the need to interconnect multiple piconets (*i.e.*, scatternet).

TABLE II

STATISTICAL MEASURES OF THE DATA

	N	3	9	15	21	27	33
No Limit	mean	0,9602	0,9604	0,9704	0,9859	1,0000	1,0000
	median	0,9997	0,9997	1,0000	1,0000	1,0000	1,0000
	std	0,1716	0,1663	0,1663	0,1562	0,0014	0,0014
WiFi	mean	0,8401	0,8632	0,8810	0,9903	1,0000	1,0000
	median	0,9997	0,9997	0,9998	1,0000	1,0000	1,0000
	std	0,3685	0,3637	0,3489	0,1000	0,1000	0,0992
ZigBee	mean	0,7403	0,7802	0,8303	0,9504	0,9664	0,9903
	median	0,9994	0,9997	0,9997	0,9998	1,0000	1,0000
	std	0,4411	0,4165	0,3777	0,2192	0,2190	0,0998
Bluetooth	mean	0,6601	0,6967	0,7269	0,7661	0,8903	0,9017
	median	0,9991	0,9992	0,9995	0,9997	1,0000	1,0000
	std	0,4915	0,4881	0,4795	0,4408	0,3840	0,3686

VII. CONCLUSION

An algorithm to guarantee *MANET* connectivity is proposed and validated within the *RDPSO* approach, which takes into account real-world multi-robot systems (*MRS*) characteristics, such as obstacle avoidance. Experimental results show that the influence inherent to communication's limitations can be attenuated as the number of robots or the communication range/quality increases. This is a promising result for communities of swarm robots with many individuals since they can develop efficient coordination techniques, just like natural swarm agents, allowing cooperative and competitive work in large and super-large societies. As future work, this multi-robot exploration strategy will be evaluated in real robots. Furthermore, one of the future approaches will be the analytical analysis of the *RDPSO* in order to find a relationship between parameters thus optimizing the algorithm with regard to the main objective, obstacles susceptibility and *MANET* connectivity in different optimization problems applied to *MRS*.

ACKNOWLEDGEMENT

This work was supported by a PhD scholarship (SFRH/BD /73382/2010) granted by the Portuguese Foundation for Science and Technology (FCT) and the Institute of Systems and Robotics (ISR) also under regular funding by FCT.

REFERENCES

[1] R. Rocha, "Building Volumetric Maps with Cooperative Mobile Robots and Useful Information Sharing: a Distributed Control Approach based on Entropy," PhD Thesis, Faculty of Engineering of University of Porto, 2006.

[2] R. Rocha, J. Dias, and A. Carvalho, "Cooperative Multi-Robot Systems: a study of Visionbased 3-D Mapping using Information Theory," *Robotics and Autonomous Systems*, vol. 53(3–4), pp. 282-311, 2005.

[3] J. Kennedy and R. Eberhart, "A new optimizer using particle swarm theory," in *Proceedings of the IEEE Sixth International Symposium on Micro Machine and Human Science*, 1995, pp. 39-43.

[4] J. Tillett, T. M. Rao, F. Sahin, R. Rao, and S. Brockport, "Darwinian Particle Swarm Optimization," *Proceedings of the 2nd Indian International Conference on Artificial Intelligence*, pp. 1474-1487, 2005.

[5] M. S. Couceiro, R. P. Rocha, and N. M. F. Ferreira, "A Novel Multi-Robot Exploration Approach based on Particle Swarm Optimization Algorithms," in *IEEE International Symposium on Safety, Security, and Rescue Robotics, SSRR2011*, Kyoto, Japan, 2011.

[6] M. A. Hsieh, R. V. Kumar, A. Cowley, and C. J. Taylor, "Maintaining network connectivity and performance in robot teams," *Special Issue on Search and Rescue Robots in the Journal of Field Robotics*, vol. 25, no. 1-2, pp. 111-131, 2008.

[7] D. Tardioli, A. R. Mosteo, L. Riazuelo, J. L. Villarroel, and L. Montano, "Enforcing Network Connectivity in Robot Team Missions," *The International Journal of Robotics Research*, 2010.

[8] D. Tardioli and J. L. Villarroel, "Real Time Communications over 802.11: RT-WMP," in *IEEE Internatonal Conference on Mobile Adhoc and Sensor Systems*, 2007.

[9] W. Sheng, Q. Yang, J. Tan, and N. Xi, "Distributed multi-robot coordination in area exploration," *Robotics and Autonomous Systems 54*, pp. 945-955, 2006.

[10] J. Pugh and A. Martinoli, "Multi-Robot Learning with Particle Swarm Optimization," in *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems*, 2006.

[11] F. Arrichiello, S. Chiaverini, and T. I. Fossen, "Formation control of marine surface vessels using the Null-Space-based Behavioral Control," *Group Coordination and Cooperative Control*, pp. 1-19, 2006.

[12] F. Arrichiello, et al., "Multi-robot collaboration with range-limited communication: Experiments with two underactuated ASVs," in *Proceedings 2009 International Conference on Field and Service Robots*, Cambridge, MA, USA, 2009.

[13] F. Arrichiello, H. Heidarsson, S. Chiaverini, and G. S. Sukhatme, "Cooperative caging using autonomous aquatic surface vehicles," in *2010 IEEE International Conference on Robotics*, 2010.

[14] B. Brüggemann and D. Schulz, "Coordinated Navigation of Multi-Robot Systems with Binary Constraints," in *The 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Taipei, Taiwan, 2010.

[15] L. E. Miller, "Multihop Connectivity of Arbitrary Networks," in *Wireless Communication Technologies Group, NIST*, 2001.

[16] R. V. Kulkarni and G. K. Venayagamoorthy, "Bio-inspired Algorithms for Autonomous Deployment and Localization of Sensor Nodes," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 40, no. 6, 2010.

[17] M. Schwager, D. Rus, and J.-J. Slotine, "Unifying Geometric, Probabilistic, and Potential Field Approaches to Multi-robot Deployment," *The International Journal of Robotics Research*, 2010.

[18] G. S. Tewolde, C. Wu, Y. Wang, and W. Sheng, "Distributed Multi-robot Work Load Partition In Manufacturing Automation," in *4th IEEE Conference on Automation Science and Engineering*, Key Bridge Marriott, Washington DC, USA, 2008.

[19] H. K. Hahn and K. Schoenberger, "The Ordered Distribution of Natural Numbers on the Square Root Spiral," *The Journal of Business*, 2007.

[20] J. Fink, N. Michael, A. Kushleyev, and V. Kumar, "Experimental characterization of radio signal propagation in indoor environments with application to estimation and control," in *IROS'09 Proceedings of the 2009 IEEE/RSJ international conference on Intelligent robots and systems*, St. Louis, MO, USA, 2009.

[21] J. S. Lee, Y. W. Su, and C. C. Shen, "A comparative study of wireless protocols: bluetooth, UWB, ZigBee, and Wi-Fi," *Proceedings of the 33rd Annual Conference of the IEEE Industrial Electronics Society (IECON '07)*, pp. 46-51, 2007.