Optimal Relay Placement in Multi-hop Wireless Networks

Roberto Magán-Carrión[a],[*], Rafael A. Rodríguez-Gómez[a], José Camacho[a],
Pedro García-Teodoro[a]

[a]*Network Engineering & Security Group (NESG, http://nesg.ugr.es)*
Dpt. of Signal Theory, Telematics and Communications - CITIC
University of Granada - 18071 Granada (Spain)

Abstract

Relay node placement in wireless environments is a research topic recurrently studied in the specialized literature. A variety of network performance goals, such as coverage, data rate and network lifetime, are considered as criteria to lead the placement of the nodes. In this work, a new relay placement approach to maximize network connectivity in a multi-hop wireless network is presented. Here, connectivity is defined as a combination of inter-node reachability and network throughput. The nodes are placed following a two-step procedure: *(i)* initial distribution, and *(ii)* solution selection. Additionally, a third stage for placement optimization is optionally proposed to maximize throughput. This tries to be a general approach for placement, and several initialization, selection and optimization algorithms can be used in each of the steps. For experimentation purposes, a leave-one-out selection procedure and a PSO related optimization algorithm are employed and evaluated for second and third stages, respectively. Other node placement solutions available in the literature are compared with the proposed one in realistic simulated scenarios. The results obtained through the properly devised experiments show the improvements achieved by the proposed approach.

Keywords: Node placement, Multi-hop route, Optimization process, Relay node, Wireless network

---

[*]Corresponding author. Ph.: +34 958241717
  Email address: `rmagan@ugr.es` (Roberto Magán-Carrión)

1. Introduction

The topic of node placement in network environments has been recurrently addressed in the literature mainly aimed at defining the best communication infrastructure. Additionally, node placement is being increasingly used in fault-tolerant strategies for which the state of the network is monitored over the time and, in case of failure or misbehavior, the nodes are repositioned to recover the performance [1, 2]. The repositioning process may be restricted to a subset of the nodes of the network that operate as relaying devices (*i.e.*, relay nodes or RNs) to allow multi-hop origin-destination routes.

A number of proposals of node placement are based on the optimization of different performance goals [3]: coverage, data rate, load balancing, connectivity, cost, etc. This task becomes even more critical for wireless communications, since additional aspects such as propagation effects and interferences should be considered in deployments [4]. Besides, in wireless sensor networks (WSNs) new requirements appear [5, 6], such as latency and lifetime, among others.

To find the best node positions some optimization algorithm has to be executed. There is a great variety of such algorithms, from graph or polynomial time-based techniques [7, 8] to bio-inspired algorithms such as genetic algorithms [9], or those based on the *swarm intelligence* paradigm [10]. To the best of our knowledge, only a few works exist where network throughput and connectivity are jointly considered to optimize the placement of RNs [11].

Almost all relay node placement solutions can be grouped in two categories [12]: constrained- and unconstrained-based strategies. On the one hand, constrained-based strategies offer a solution restricted to certain parameters. As aforementioned, some examples of this are network lifetime preservation, coverage, delay or simply solutions restricting the location of the RNs to certain specific regions. On the other hand, unconstrained-based strategies are mainly intended to meet more general purposes like fault-tolerant and/or network connectivity without any precise constraint. The specific method should be previously studied, since both types of strategies have advantages and disadvantages.

The present paper introduces a new placement approach to maximize throughput and inter-node reachability in a multi-hop wireless network by providing and hybrid optimization approach that combines the advantages of constrained and unconstrained based approaches. The proposal relies on

splitting the overall problem into a number of steps: *(i)* initial distribution of the RNs, and *(ii)* a solution selection. Additionally, a third step aimed at subsequently estimating the optimal locations, can be optionally launched. Leave-one-out (LOO) and particle swarm optimization (PSO) algorithms are implemented here at the solution selection and placement optimization stages, respectively. This multi-stage placement procedure provides more versatile and scalable solutions, as it will be shown through experimentation.

In summary, the main contributions of this work are:

- An efficient RN placement procedure for multi-hop wireless networks based on the joint optimization of inter-node reachability and network throughput.

- A novel optimization formulation to maximize this global two-criteria performance goal. This will result in the improvement of the overall connectivity of the network.

- A multi-stage procedure to divide the overall problem into more elemental subsequent steps.

The rest of the paper is organized as follows. Section 2 presents several principal works in the field of node placement in wireless networks. Section 3 outlines the general assumptions under which the RN placement approach is developed, while its details are presented in Section 4. After that, Section 5 is devoted to describe the experimental framework used to evaluate the proposal, from which the results obtained are compared with other solutions in the literature, and carefully analyzed afterwards in Section 6. The proposal is also briefly discussed from the perspective of some open challenges to be addressed in further work in Section 7. Finally, Section 8 presents the main conclusions of the work. We also include an Appendix to introduce the *swarm intelligence* fundamentals and describe the PSO algorithm.

2. Related work

Strategies developed to position nodes in a wireless network can be grouped in two categories: constrained- and unconstrained-based strategies. On the one hand, constrained-based strategies offer a solution restricted to certain

parameters. Some examples of them are network lifetime preservation, coverage, delay or simply solutions restricting the location of the RNs to some specific regions. On the other hand, unconstrained-based strategies are mainly intended to meet more general purposes like fault-tolerant and/or network connectivity without any precise constraint.

Most of the unconstrained-based RN placement strategies present a common objective: to provide a k-connected network by using the minimum number of RNs. To accomplish with fault tolerance goals, the k parameter must be greater than or equal to 2. Otherwise, if $k = 1$ the network connectivity from the perspective of the inter-node reachability is contemplated. In this category a variety of works ([13], [14], [15], [16], [17], [18], [19], among others) can be found. For example, in [13] the authors address single- and two-tiered 1-connected solutions aimed at minimizing the number of RNs deployed. They provide 7-approximation and $(5+\epsilon)$-approximation algorithms for facing the previous purposes, respectively. The same authors in [14] extend this work by designing a 2-connected approximation algorithm also for single- and two-tiered approaches. K-connected networks are proposed in [15] for any desired k as well. The authors demonstrate that whatever the selected k parameter is, they can reach a polynomial based approximation solution. A similar approach is introduced in [17] where the authors are able to locate the minimum number of RNs getting a k-connected network. The proposed heuristic improves the previous one by applying a simple selection step to remove some RNs while the k-connectedness is preserved. In [16], the objective is to provide fault tolerance in heterogeneous WSNs, where sensors posse different transmission radii. They develop a $k \geq 1$ connected approximation solution taking into account the desired degree of fault tolerance (full- or partial-fault resiliency) and the communication flows (one- or two-ways).

Some of the most relevant placement solutions relying on constrained strategies are those in [20], [21], [22], [23], [24], [25], [12], [26], [27], [28], [29] and [30]. Some of them try to find (i) the number of relay nodes to minimize the network cost ([20], [21], [22], and [23]), (ii) the minimum number of relay nodes to achieve a fully connected network or a two-tiered network ([24], [25], [12], [28], [29] and [30]), or (iii) the minimum number of relay nodes to reduce the delay in the communications ([26] and [27]).

Authors in [20] determine the number and deployment of heterogeneous devices, so that the total network cost is minimized while constraints regarding lifetime, coverage and connectivity are satisfied. The work has been

4

further extended in [21], [22] and [23] to address the problem of deploying a second-tier of RNs to balance the traffic by using the least number of additional RNs.

A number of works propose placement approaches to maximize connectivity. In [24], the authors try to place the lowest number of RNs in the playing field of a sensor network in such a way that each sensor node can communicate with at least one RN, and the network of RNs is completely connected. This objective is achieved by means of two optimization problems: a connected relay node single cover (CRNSC) problem, and a 2-connected relay node double cover (2CRNDC) problem, and in both of them only certain available locations instead of the whole XY plane are considered. Authors in [28] try to place the minimum number of relay sensors to maintain global connectivity restricting the locations of the relay nodes using the minimum Steiner tree with minimum number of Steiner points. Similar idea is implemented in [29] for restoring the connectivity lost. They iteratively select those Steiner points connecting at least three network partitions until the Steiner tree has less than three points. After that, they locate the RNs on the line from each partitions to its closest Steiner point. A connectivity and energy constrained efficient approach for relay placement in WSNs is proposed in [30], where the relay nodes can only be placed at some pre-specified candidate locations. To provide survivability, the goal is to form a 2-connected network. References [25, 12] discuss similar approaches.

Authors in [26] (an improved version of [27]) design a multi-hop wireless mesh network with minimum number of additional RNs to facilitate wireless-communication between each of the sensor nodes and the base station. The placement of the relays should ensure that the delay on the paths between the base station and the sensors meet a pre-specified delay bound. Authors study the structure of the projection polyhedron of the problem and develop node-cut inequalities. A branch-and-cut algorithm is defined based upon the projection formulation to solve delay constrained RN placement problem (DCRNPP) optimally.

Beyond the specific metrics chosen to be optimized, the optimization algorithm itself considered to determine the best position of nodes can vary. Some examples come from those based on graph- or polynomial time techniques [7, 8] to others bio-inspired, among several others. A bi-objective (user coverage and network connectivity) genetic-based optimization algorithm is considered in [9] for node placement in wireless mesh networks. In [31], a PSO algorithm is proposed to determine the best placement of nodes in industrial

environments in terms of network reliability, load uniformity, total cost and convergence speed. For sensing coverage purposes in WSNs, the authors in [32] propose a PSO-based solution for minimizing the existing coverage holes through the use of a fitness function based on the computation of Voronoi regions. A modified PSO-based proposal is also used in [10] to address the sink placement problem in WSNs by minimizing the worst case delay path in the network. A minimax scheme is developed in [33] to optimize coverage ratio and uniformity. On the other hand, authors in [34] propose an algorithm that emulates the attractive force (as in a stretched spring) and the repulsive force (as the electrostatic force) in nature, such that a robotic node simply follows the resultant virtual force to move to locations that maximize coverage and minimize moving distance. Wang *et al.* address in [35] the problem of node placement in wireless networks for ensuring complete coverage in a long belt scenario. They also propose an inter-node distance-based approach to minimize the number of nodes needed. It is shown that the node density achieved can be lower than that required by the well-known regular triangular-lattice placement.

Optimization algorithms have a crucial role when addressing optimal RN placement problems in wireless multi-hop networks. Although there exist a lot of proposals, only a few works consider jointly network throughput and connectivity maximization as optimization goals. One of them is that in [11], where the authors make use of a PSO procedure enhanced with a model predictive control (MPC) for MANETs. Authors in [36] address system performance and connectivity issues. They maximize the packet delivery ratio providing a realistic modeling taking into account network aspects like channel interferences or congestion areas. Nevertheless, although throughput can be seen as an indirect measurement of the connectedness, it is not taken into account explicitly.

In the previous works three main drawbacks can be remarked. Firstly, in many cases the number of relay nodes cannot be a parameter of choice for a relay node placement strategy, since there is a specific number of available relay nodes. Practical problems appear, as that of how to optimally relocate a fixed number of RNs to maximize network performance. This is the case, for instance, of dynamic environments, where the optimum number of RNs may change in time, and thus this cannot be adequately tackled with physical agents. Secondly, the constraints applied to the possible locations of the relay nodes may help to simplify the optimization problem at the cost of leading to a less optimal solution. Thirdly, in an unconstrained approach it

6

is possible that the optimal position for the relay nodes cannot be physically implemented.

In this context, our work proposes a customizable approach with a multi-stage procedure considering a fixed number of relay nodes. This multi-stage approach can be seen as a hybrid solution since: (i) we use a constrained approach based on the distribution and subsequent selection of the locations of the RNs, and (ii) this allows running an additional unconstrained optimization stage to consider some other possible locations, not taken into account in previous stages, to place the nodes.

3. Problem formulation

The main goal of the present work is to maximize connectivity in a multi-hop wireless network using relay nodes. Connectivity can be defined in several ways, and for the present paper it is defined as a combination of inter-node reachability and network throughput. From this definition, an optimization problem should be properly formulated to find the optimal placement for the relay nodes.

Two types of nodes are considered in the network: user nodes (UNs) and relay nodes (RNs). UNs make use of top-layer services in the network and RNs only perform a relaying function to enable or improve reachability and throughput in the network. UNs may be personal devices in an ad-hoc local area network, sensors in a wireless sensor network (WSN), mesh nodes in a wireless mesh network (WMN), or base stations in wireless wide area networks (WWAN). Regarding RNs, they can be either static or mobile nodes. Some examples of mobile relays are: unmanned aerial vehicles (UAVs) [37] or mobile robots [38], among others.

For this optimization problem the following primary assumptions are done, which are found elsewhere as in [11]:

- *The optimization procedure is centralized.*

  This implies that there exists a node capable, through the adequate communication and processing means: (i) to retrieve the necessary network information, (ii) to run the optimization algorithm from this information, and (iii) to send control data, (*i.e.*, the placement) in accordance with the result of the optimization.

- *The position of UNs in the network is not controlled.*

7

Only RNs can be optimally positioned. This is a reasonable assumption that complicates the optimization problem. Moreover this, problem definition generalizes the simpler problem in which the location of UNs can also be optimized.

- *The network is partitioned.*

  That is, some nodes of the network are unaccessible due to the absence of physical links. This assumption is often met in scenarios like disaster areas or battlefields.

Additional secondary assumptions adopted in the paper are the following:

- *The optimization is limited to a 2D space.*

- *UNs are static in the environment.*

- *RNs are static in their optimum locations. The procedure to drive the RNs to those locations lies in the automatic control domain and is out of the scope of this paper.*

- *The network is single-tiered [12], so that both UNs and RNs relay information from other UNs.*

- *The communication range of both UNs and RNs is the same: r meters*

While primary assumptions constitute a basement for the proposal, secondary assumptions are stated to limit the research presented in this work. Thus, extensions of the present approach for mobile, 3D set-ups, two-tiered and for different UNs and RNs communications ranges can be defined.

Beyond the above, some mathematical formulations representing our environment are exposed in the following. A multi-hop wireless network with UNs and RNs can be specified as follows:

$$G = (N, E) \tag{1}$$

with

$$N = U \cup R \tag{2}$$

where U, R, N and E stand for the set of UNs, the set of RNs, the complete set of wireless nodes (UNs plus RNs) and the wireless links (edges), respectively. In a wireless network, the edges satisfy:

$$E := \{\mathbf{e}_{ij} \mid \|\mathbf{e}_{ij}\| \leq r, \forall n_i, n_j \in N\} \tag{3}$$

with r the range of a single link and n_i and n_j the i-th and j-th network nodes.

The optimal placement of RNs can be represented by the graph composed of the complete set of wireless nodes in the network:

$$\mathbf{G}^* := \arg\max_{\mathbf{G}} \{f(G) | G = (N, E) \ and \ U = U_0\} \tag{4}$$

where $f(G)$ is the function to maximize and U_0 represents the actual location of the UNs. According to the goal of the paper, $f(G)$ should be a direct or indirect measure of the connectivity in the network.

In [11], the following definition for network connectivity is proposed:

$$f_c(G) = \frac{2}{u \times (u-1)} \times \sum_{\forall u_i, u_j \in U, j > i} z(G, i, j) \tag{5}$$

where u is the number of UNs and u_i and u_j the i-th and j-th UNs. $z(G, i, j) = 1$ if there exists an available (either single or multi-hop) path connecting u_i and u_j in G, and $z(G, i, j) = 0$ otherwise. Function $z()$ is in our case the previously mentioned inter-node reachability, since two given nodes may be either accessible or not.

One of the main drawbacks of using Eq. (5) as the performance function in the optimization problem formulation is that f_c is a discrete function and thus it leads to a discrete optimization problem. In general, continuous optimization problems with smooth performance functions are easier to solve [39]. In these cases, the optimizer can be driven using the differential or derivative information of the performance function. Conversely, in a discrete optimization the search space is divided into flat regions, wherein differential information remains null. In flat regions, several solutions to the optimization problem obtain the same value of the performance function, while they may not be equally adequate.

This problem is addressed in [11] by combining (switching) f_c with two additional performance functions in a PSO solver: O_2 (network throughput) and O_3 (inter agents-attraction points distance). From here onwards, this approach will be called the DKS method (DKS stands for the initials of the authors). In a static scenario like the one considered in this paper, only one of these additional performance functions is of interest: throughput. Thus,

the DKS method searches for the best solution in terms of connectivity (f_c), and when several solutions with equal value of this measure are found, the one with the highest throughput is selected. The second additional function O_3, is only used as a decision criterion when the network is disconnected. If two or more solutions with equal overall connectivity are obtained, O_3 will be computed since O_2 cannot be applied in partitioned networks. The direct consequence when using several optimization function to drive the global process, is that the resulting optimization surface presents discontinuities in regions of the search space, which makes the optimization harder.

In the present paper, the combination of inter-node reachability with throughput is further explored and a novel placement approach is proposed. Firstly, the optimization of a discrete function f_c is a constraint satisfaction problem (CSP) [40] that should be tackled with combinatorial/discrete optimization techniques rather than continuous techniques. When necessary, the solution found for the CSP can be further improved using continuous optimization, by performing an smoothing operation to the shape of the discrete performance function. This smoothing operation considers both reachability and throughput. The details of the approach will be introduced in the next section. The remaining of the present one is devoted to discuss the smoothing transformation of the performance function.

The smooth performance function for continuous optimization should be defined so that:

1. It is unique, continuous and smooth.
2. It resembles the shape of the connectivity/reachability in Eq. (5) as much as possible.

Following these constraints, the performance function is defined as:

$$g(G') = \sum_{\forall u_i, u_j \in U: j > i} id(G'_i, i, j) \tag{6}$$

where G'_i is the spanning tree starting in the i-th UN and minimizing the distance of the largest edge of each path in the network. The function $id(G'_i, i, j)$ is the inverse of the length of the longest edge in the path from i to j in G'_i network. Function $g(G')$ implies a redefinition of the network as a complete graph G', so that:

$$G' = (N, E') \tag{7}$$

10

$$E' := \{\mathbf{e}_{ij} \mid \forall n_i, n_j \in N\} \tag{8}$$

As mentioned before, $g(G')$ function is a smoothed version of the reachability $f_c(G)$. This is shown in Figure 1, in which the values of both functions are computed for a scenario where two UNs are 3 units away and a RN is located in different points along the line defined between them. The maximum range for a link has a distance of 2 units, *i.e.* $r = 2$. In the abscissas axis we represent the different locations of the RN, UNs being located in 1 and 4. The three nodes (UNs and RN) are connected when the RN is located inside the interval [2,3]. There is a flat region wherein no differences between different locations in terms of $f_c(G)$ are found. Besides, intervals [1,2] and [3,4] also present flat regions of $f_c(G)$. However, function $g(G')$ is continuous in the entire interval [1,4] and also smooth except in the optimum point 2.5. This function has derivative information to lead the optimization to the optimum solution no matter the initial point chosen.

4. Relay node placement proposal

As stated before in Section 1, a relay placement solution based on two steps, initial distribution and solution selection, is developed. In the first step, a higher number of RNs than those actually available are distributed around the scenario. Then, a subset of the theoretical RNs are only selected. This scheme of solutions generation and selection can be found in many CSP approaches, such as in genetic algorithms. Moreover, in our case the resulting locations can be additionally introduced into an optimization algorithm for further improvement of the finally selected relay nodes.

The functional structure of the placement system is shown in Figure 2. Beyond this general modular architecture, different specific algorithms can be used. For experimentation in present paper, we will consider a LOO selection procedure and a PSO algorithm for steps 2 and 3, respectively. See the Appendix for more information about the latter.

4.1. Initial distribution

In this step, the primary theoretical RNs are homogeneously distributed along the edges connecting partitions in the network, according to a distance-based spanning tree. To select these RNs the user-defined parameter $\lambda \in (0, 1]$ is selected, so that:
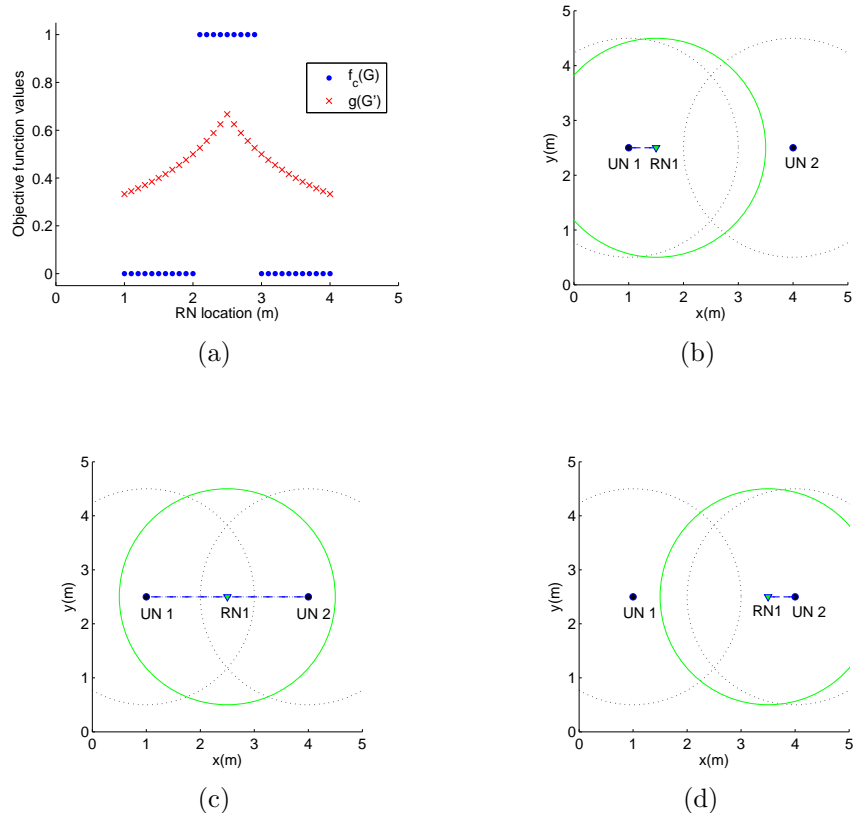
11

Figure 1: Comparison of optimization functions $f_c(G)$ and $g(G')$. Sub-figure (a) shows the function evolution due to the RN movements from UN 1 to UN 2. For the sake of clarification, the RN movements shown through sub-figures (b), (c) and (d) represents how the network becomes disconnected, connected and disconnected again, respectively.
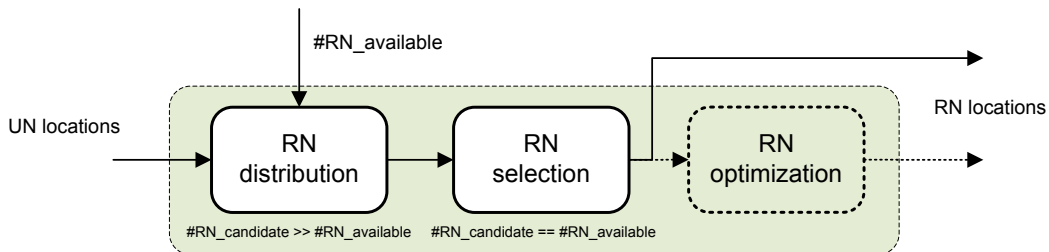

Figure 2: Main functional blocks of the devised RN placement system.

$$l_{ij} = \left\lceil \frac{\|\mathbf{e}_{ij}\|}{\lambda \cdot r} \right\rceil, \ \forall u_i, u_j \in U \ s.t. \ \|\mathbf{e}_{ij}\| > r \tag{9}$$

where l_{ij} is the number of RNs distributed in edge $\mathbf{e}_{ij}$, r the coverage range, and $\lceil \ \rceil$ stands for the ceil operation. Thus, parameter λ controls the number of relays in the network. For $\lambda = 1$, the minimum number of relays to connect the nodes in the network is considered. The number of initial RNs increases as λ decreases. The distance between equidistant relays in the edge $\mathbf{e}_{ij}$ is $d_{ij} = \|\mathbf{e}_{ij}\|/l_{ij}$, and it holds:

$$\lambda \cdot r \geq d_{ij}, \ \forall u_i, u_j \in U \ s.t. \ \|\mathbf{e}_{ij}\| > r \tag{10}$$

The total number of initial RNs in the network is:

$$\sum_{\forall u_i, u_j \in U} l_{ij}, \ s.t. \ \|\mathbf{e}_{ij}\| > r \tag{11}$$

In Figure 3 we show the effect of λ on the number of candidate RNs, which are represented as inverted filled triangles.

In terms of computational complexity, the execution time of this algorithm will be directly related with the number of available RNs, which is in turn related with the λ parameter. Since the value of λ is previously fixed, the computation time will be mainly influenced by the algorithm used to conform the minimum spanning tree to distribute the RNs. After analyzing the worst execution cases for each of the algorithms involved, we can conclude that the upper bound of the complexity, expressed in Big $\mathcal{O}$ notation, is $\mathcal{O}(n^2 \cdot \log n)$ which corresponds to the Prim's algorithm [41].

4.2. Solution selection

After the initial RN distribution, an iterative procedure is carried out to discard the worst RNs in the previous set, in terms of the $f_c(G)$ function. In case that several candidates to be discarded attain the same value for $f_c(G)$, then the $g(G')$ function is considered. This is repeated until the number of RNs coincides with the number of actually available RNs in the environment.

To evaluate how $f_c(G)$ and $g(G')$ are affected by omitting a given RN, a LOO approach is applied. This way, the effect in terms of performance in the removal of each specific RN is evaluated and the RN with the least influence in $f_c(G)$ and/or $g(G')$ is selected. To the best of our knowledge, this is the first time that LOO is used for this purpose.
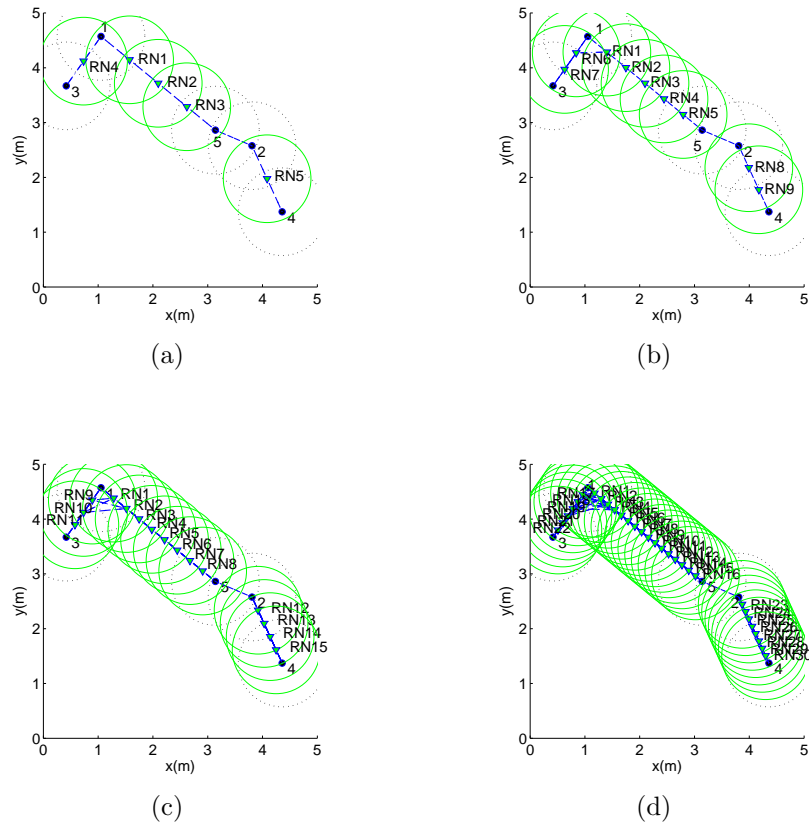
13

Figure 3: Effect of λ on the number of initial candidate RNs along the spanning tree: (a) $\lambda = 1 \to 5$ RNs, (b) $\lambda = 0.6 \to 9$ RNs, (c) $\lambda = 0.4 \to 15$ RNs, and (d) $\lambda = 0.2 \to 30$ RNs.

14

Algorithm 1: PSO-based algorithm

Data: Current U and R positions.
Result: Optimal R^* positions ($gbest$)

$[X, V] \leftarrow initialPopulation(\#R, \#particles)$;
$[gbest, pbest] \leftarrow evaluatePopulation(X, R, U)$;
while $k \leq maxIterations$ **do**
$\quad w = w * 0.98$;
$\quad$ **for** $i = 1 \cdots \#particles$ **do**
$\quad\quad V(i) \leftarrow getVelocityIncrements(X, V, w, pbest(i), gbest)$;   /* see Eq. (23) */
$\quad\quad V(i) \leftarrow setVelocityClamp(V(i), V_{max})$;   /* see Eq. (24) */
$\quad\quad X(i) \leftarrow X(i) + V(i)$;   /* see Eq. (22) */
$\quad\quad [X(i), V(i), gbest, pbest(i)] \leftarrow evaluateSolution(X(i), V(i), gbest, pbest(i), U, R)$;
$\quad$ **end**
$\quad k = k + 1$;
end
$R^* \leftarrow gbest$;

In each iteration, the set of worst cases R_* is determined as:

$$R_* := \arg\max_{\mathbf{r}_j \in R} f_c(G(U \cup \{R - r_j\}, E)) \qquad (12)$$

If R_* contains a unique RN r_*, this is the one to be discarded. Otherwise r_* is obtained as:

$$r_* := \arg\max_{\mathbf{r}_j \in R_*} g(G(U \cup \{R - r_j\}, E)) \qquad (13)$$

Then, R is updated following $R = \{R - r_*\}$, and the operation is repeated until the number of RNs in R meets the number of available RNs.

The selection of the best RNs is not a trivial task, with a complexity of $\mathcal{O}((k^2 - m^2) \cdot n^4)$ where the number of theoretical and available RNs are represented by k and m, respectively. The term n^4 in the expression is mainly due to the inherent complexity in computing $g(G')$ in Eq. (6).

4.3. Optimal location

Once the most promising RNs in terms of $f_c(G)$ and $g(G')$ are selected, an optimization algorithm can be executed to locate the previously selected RNs with best locations in terms of $g(G')$. As it will be shown in Section 6, this step leads the system to improve the network flow while maintaining the maximum reachability.

Without loss of generality, we have considered a PSO-based optimization algorithm (see Appendix), which is presented in Algorithm 1. Firstly, a random initial set of particles X is distributed around the search space as well as their initial velocity increments V. From this set we evaluate afterwards the best global and particle solutions, *gbest* and *pbest* respectively. Then the new position of each particle is updated and again evaluated. This procedure is iteratively repeated a number of times equal to *maxIterations*, which is the predefined stopping criterion. Once the algorithm is finished, the best RN locations R^* are returned.

Main factors in the computational complexity of Algorithm 1 are the number of particles and iterations considered. Also the inner functions, like that in charge of evaluating each candidate solution, have a clear impact on the overall complexity. This is concluded to be $\mathcal{O}((n^3 + m) \cdot p \cdot i)$, where p and i stand for the number of particles and iterations, respectively. The highest complexity term, n^3, is due to the fact that the cost function $g(G')$ is evaluated for each iteration and particle.

Figure 4 shows the optimization procedure at each stage considering 5 UNs (filled circles) and 4 available RNs (filled inverted triangles). Figure 4(a) depicts the initial UNs distribution. Figure 4(b) shows the initial RNs distribution along the spanning tree among UN partitions. Figure 4(c) depicts the selected RNs after the LOO procedure according to the system main goals: network reachability and throughout. Finally, the optimization results in terms of the best positions for RNs are shown in Figure 4(d). The optimization step moves node RN2 to a new location in order to connect UN number 1, increasing the overall network connectivity in terms of reachability and throughput. In this specific example, a higher number of RNs would be needed to achieve full connectivity. Nevertheless, with the current amount of RNs, the system locates them in their best positions according to the system performance goals.

5. Experimental set-up

In this section the experimental scenario considered to evaluate our RN placement approach is introduced. For purposes of flexibility and low cost, a simulation-based framework is chosen.
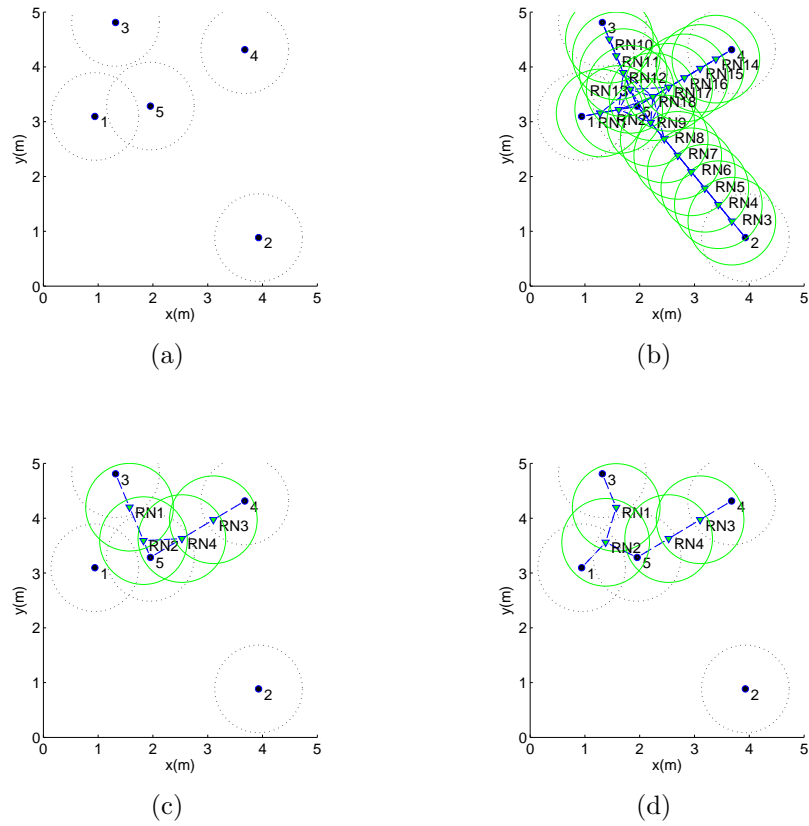
Figure 4: RNs location after each stage considering 5 UNs (filled circles) and 4 available RNs (filled inverted triangles): (a) initial UNs distribution, (b) initial candidate RNs locations along the spanning tree minimizing the inter-partitions distance, (c) RNs selection, and (d) RNs optimal locations after the PSO-based optimization.
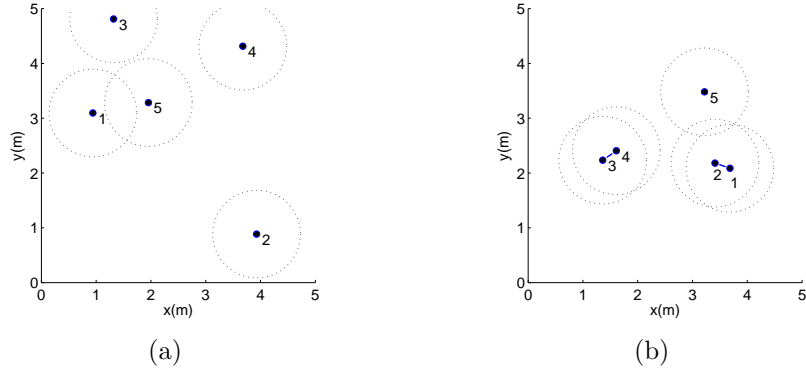
17

Figure 5: UNs distribution patterns, where 5 UNs (filled circles) are distributed throughout a $5m \times 5m$ squared area. All of them have $r = 0.8m$ coverage radii. Subfigure (a) shows the MRWP UNs distribution, while (b) shows the RPGM UNs distribution creating groups of 2 nodes.

5.1. Simulation framework

The experimental scenario has been developed in Matlab. As usually assumed elsewhere [11], low-level network related details are skipped and the focus is restricted to the functions of interest, reachability/connectivity and throughput.

This concern has not been discussed previously, but the UNs distribution throughout the area is a main issue too. Although static scenarios are considered here, the simulation makes use of some well-known UNs mobility patterns to establish the associated distribution. In particular, two different mobility models are deployed: a modified random way point mobility (MRWP) model [11], and the reference point group mobility model (RPGM) [42]. The former, MRWP, slightly modifies the widely used random way point (RWP) mobility model [43]. Similarly, RPGM is also based on RWP but it creates groups of UNs at the beginning of the simulation moving them in a similar way like RWP does. For the sake of clarification, Figure 5 shows the simulation scenario where there exist 5 UNs (filled circles) and different UNs distribution patterns are considered. Figure 5(a) shows the UNs distribution by using MRWP, while Figure 5(b) depicts a RPGM scenario where groups of 2 UNs are created.

In order to simulate partitions in the network, only 5 UNs are deployed through a $5m \times 5m$ squared area, where a coverage range of $r = 0.8m$ is assumed. Without loss of generality, this is the same communication range

for RNs. The performance of the relay placement solutions are evaluated in terms of the number of RNs. Regarding the proposal of the paper, parameter λ determines the number of initial RNs and their locations, see Figure 5. We choose $\lambda = 0.5$ for setting up a higher number of candidate RNs than those available.

Each solution is assessed in terms of reachability values and throughput. The first is measured through Eq. (5) and the second is in turn computed by the following equation:

$$th(G') = \sum_{\forall u_i, u_j \in U: j > i} ie(G'_i, i, j) \tag{14}$$

where G'_i is the spanning tree starting in user node i and minimizing the distance of the largest edge of each path in the network. The function $ie(G'_i, i, j)$ measures the throughput between nodes i to j in G'_i. Each graph edge capability is measured through

$$c_{ij} = \frac{1}{1 + e^{(10 \cdot d_{ij} - 0.5)}} \tag{15}$$

where d_{ij} represents the distance between nodes i to j in G'_i. Each RN placement solution is assessed through the average results obtained from 25 different repetitions considering each of the aforementioned UNs distribution patterns.

Additionally, each solution is evaluated in terms of the execution time. For that, the average time value of 25 repetitions for 25 different scenarios is obtained. The number of RNs is fixed to 3, and the number of UNs is varied from 5 to 30 to study the impact of this parameter on the execution time.

Finally, it is important to remark the different RN placement schemes to be analyzed and compared. Basically, we will evaluate the improvements in system performance introduced by the addition of each stage of the proposal in a incremental way. Besides, each one will also be compared among them and with DKS method. In summary, they are:

1. DKS. A complete RNs placement proposal as stated in [11]. The authors proposed a system for maximizing the connectivity in MANETs through the use of a PSO-based solver and several objective functions. The solver switches among the functions depending on the network state.
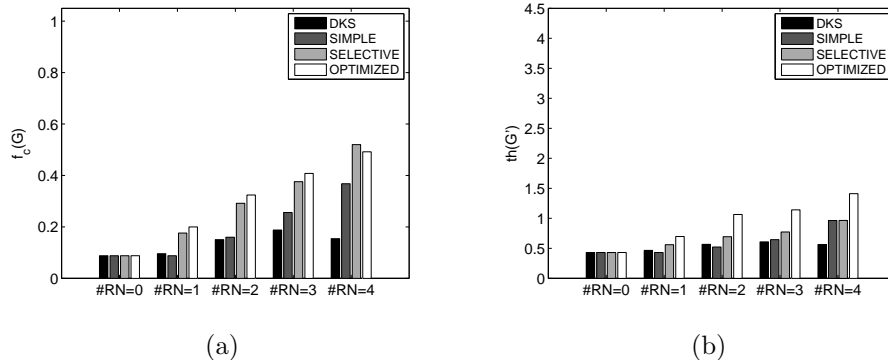
19

Figure 6: Reachability (a) and throughput values (b) obtained in the simulation environment with the MRWP distribution.

2. SIMPLE. It is based on the homogeneously RN distribution along the edges connecting partitions in the network, according to a distance-based spanning tree (step 1 of the proposed architecture in Fig. 2). To select these RNs the user-defined parameter $\lambda \in (0, 1]$ is selected.

3. SELECTIVE. This solution iteratively runs a LOO based procedure in order to discard the worst RN locations in terms of $f_c(G)$ and $g(G')$ from those locations obtained through SIMPLE scheme (step 2 in Fig. 2).

4. OPTIMIZED. From the RN locations obtained in SELECTIVE scheme, a PSO-based optimization algorithm is executed to locate the previously selected RNs in terms of $g(G')$ (step 3 in Fig. 2).

6. Performance and result analysis

This section presents the average results obtained from 25 repetitions per node distribution pattern and placement method in the experimental scenario previously described. Figure 6 shows the results in terms of node reachability, $f_c(G)$, and throughput, $th(G')$, for different numbers of RNs and using the MRWP distribution for the UNs. The figure shows that the introduction of RNs is more effective in terms of reachability, as expected. With the MRWP distribution, the problem is very challenging and less than 10% of the links are connected when no RN is introduced. In this situation, a fully connected network cannot be obtained with 4 or less RNs, and the best solution can only maintain around 50% of the links.

20

Regarding the comparison of the four placement methods, the worst performance is obtained by the DKS approach. In this case the performance is very low and it is clearly outperformed by the naive SIMPLE approach, which does not even consider the communication range of the devices. Both SELECTIVE and OPTIMIZED methods outperform the other two approaches, SELECTIVE being mainly focused on optimizing the reachability. In comparison to SIMPLE scheme, SELECTIVE attains an improvement of more than 10% of reachability, and doubles it for low number of RNs. OPTIMIZED approach yields similar improvements in reachability and also an important improvement in terms of throughput. This refers to the result of the optimization step over the smoothed version of the connectivity discussed in Section 3. Since reachability itself is not of great use when throughput is reduced, the solution provided by OPTIMIZED scheme is the most promising one.

Figure 7 illustrates the differences among the above methods in a specific example. The low performance achieved by DKS is explained in Figure 7(a). DKS is only capable to locate one RN in what the authors of [11] call attraction points: the inverted triangles on dotted lines in the figure. The rest of RNs remain unused. Besides, the definition of the attraction points is naive and useless in most situations, since it does not take into account the communication range. See [44] for a detailed explanation of these problems. The SIMPLE solution for the same example is shown in Figure 7(b). RNs are distributed now in the spanning tree connecting the UNs. Again, since the communication range is not considered, there are two RNs that remain unused. However, if an enough number of RNs are available, this approach would lead to a fully connected network, while the DKS would not. SELECTIVE scheme shows in Figure 7(c) to perform a more intelligent RNs distribution, where the communication range is implicitly taken into account in the LOO selection procedure. Finally, OPTIMIZED approach in Figure 7(d) yields a very similar solution to SELECTIVE scheme, since the PSO did not find any solution outperforming the SELECTIVE one.

Figure 8 presents the results using the RPGM distribution for the UNs. Clearly, this problem is simpler than that of Figure 6, since some UNs are already grouped. With the addition of RNs, the fully connected network is approached in averaged terms. Again, both SELECTIVE and OPTIMIZED schemes outperform the other two methods, and DKS yields the lowest performance. SELECTIVE approach attains the best reachability results but at the expense of low throughput values. OPTIMIZED method provides a
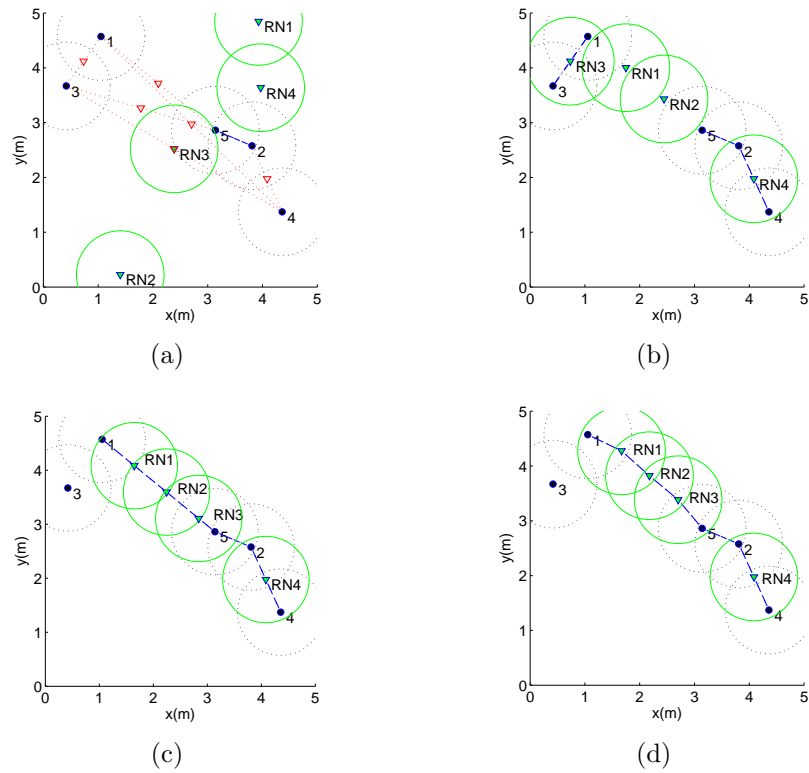
Figure 7: RN locations achieved by DKS (a), SIMPLE (b), SELECTIVE (c) and OPTI-MIZED (d) RNs placement proposals. There are 4 RNs available (inverted filled triangles) and 5 UNs (filled circles), distributed throughout the area by using the MRWP pattern. In sub-figure (a) the imaginary attraction points are represented by inverted triangles on dotted lines.
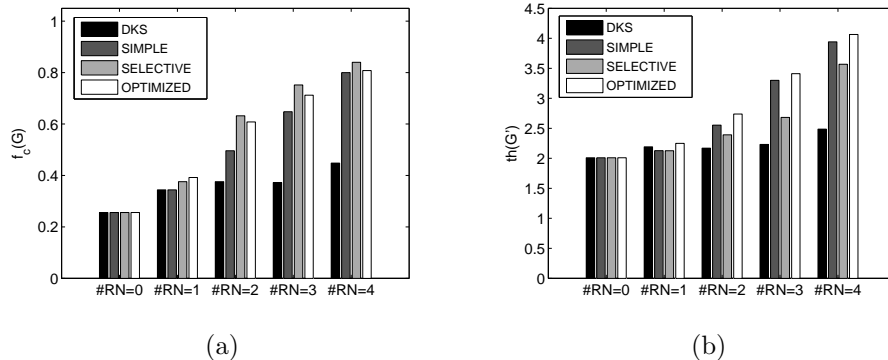
Figure 8: Reachability (a) and throughput values (b) obtained in the simulation environment with the RPGM distribution with groups of 2 UNs.

more complete solution since improvements in reachability and throughput are obtained. The good outcomes of SIMPLE are noticeable, which remain close in both reachability and throughput to those of OPTIMIZED. This is a consequence of the simplification of the problem using RPGM, and it is not expected to happen in more complex scenarios.

Aimed at comparing the time spent by each placement procedure, Figure 9 shows the evolution of the execution time with the number of UNs. We can observe how the execution time increases as the number of UNs does. Moreover, it is noticeable that both SELECTIVE and OPTIMIZED are much more time demanding than DKS and SIMPLE. In particular, the complexity of the former ones is $\mathcal{O}((k^2 - m^2) \cdot n^4)$ and $\mathcal{O}((n^3 + m) \cdot p \cdot i)$, respectively. This way, OPTIMIZED repeats $p \cdot i$ times an operation of n^3 order, where $p = 50$ and $i = 100$. Instead, SELECTIVE repeats $(k^2 - m^2)$ times an operation of n^4 order, the number of theoretical RNs (k) growing with the number of UNs (until a certain value since the network area remains equal). As a consequence, OPTIMIZED has a higher execution time values than SELECTIVE. These behaviors are observed in Figure 9(a), while Figure 9(b) shows better details about how SIMPLE solution involves lower execution time than DKS as it is expected from the associated complexity expressions (see Section 4).

According to the complexity of each placement solution (see Table 1) and the associated performance results previously discussed (see Figures 6 and 8), the overall performance improves as the complexity of the corresponding placement solution does. The most influencing parameter that affects the
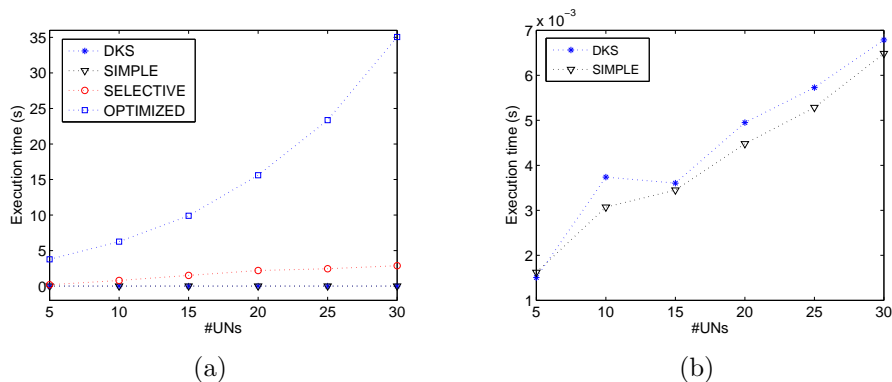
23

Figure 9: Dependence of the execution time with the number of UNs. Subfigure (a) shows a comparison for the four algorithms studied, while subfigure (b) provides a closer inspection for DKS and SIMPLE, as the execution time for them is much lower (near to zero) than for the other algorithms.

	O(.)
DKS	$O(n^2 \cdot m)$
SIMPLE	$O(n^2 \cdot \log n)$
SELECTIVE	$O((k^2 - m^2) \cdot n^4)$
OPTIMIZED	$O((n^3 + m) \cdot p \cdot i)$

Table 1: Complexity of placement algorithms. Parameters m and n stand for #RNs and #UNs, respectively; k represents the number of candidate RNs to be selected; p is the number of particles considered in the PSO optimization algorithm; and i the maximum number of iterations in PSO.

complexity is the number of UNs, n. Despite the fourth degree term of this parameter in SELECTIVE solution in comparison with the first degree in DKS, two main aspects should be remarked here in defense of our approach. First, in static scenarios (like those generally studied in the literature) the RNs placement algorithm is executed just once at the beginning of the network functioning, thus making computational time a secondary concern. Instead, the performance improvement is principal. Second, DKS is originally devised to be applied in dynamic scenarios, having much higher complexity than in static ones (see more details about the DKS solution in [11]), which constitutes an extra justification for considering our solution better.

In summary, experimentation demonstrates how our multi-stage RN placement approach satisfies the expected performance objectives maintaining a

reasonable running time for this kind of scenarios. Moreover, it must be noticed that the general functional architecture allows to consider different specific schemes at each step.

7. Open challenges

As presented, the placement solution introduced here is intended to improve the global connectivity understood as a mixture of reachability and throughput in multi-hop wireless networks (*e.g.*, WSNs or mesh networks) by locating a number of RNs in certain optimal positions around the target environment. After analyzing the performance obtained, we can conclude the suitability of the approach. However, there are a number of interesting issues that should be studied in the future. Some of these are briefly discussed in the next.

The RN placement solution is centralized in the sense that a central engine is in charge of: (a) collecting the node positions (*e.g.*, they provided through a tracking system), (b) estimating the optimal locations of the RNs to improve the system performance goals, and (c) positioning the RNs on such locations. However, carrying out first and third steps will require the deployment of a communication protocol to exchange the location information specially in real environments. A publisher-subscriber communication based system can be a good and versatile solution. For example, the lightweight communications marshaling (LCM) [45] is a suitable choice for a practical deployment of the approach. Moreover, faults in nodes, the existence of attackers or some unreachability situations due to nodes mobility may have a severe impact on the centralized solution. Distributed solutions may be an interesting alternative in those situations. However, the potential loss of performance when solving the placement in a distributed, local manner should be carefully studied to determine the circumstances under which this is a suitable solution.

Also related with the previous issue and in regards with dynamic scenarios (*e.g.*, MANETs), nodes mobility constitutes a relevant topic. In such environments, the placement performance in terms of the frequency in re-estimating the RNs position and the velocity in moving them to the optimal positions should also be carefully analyzed and evaluated. This way, the faster the changes in network topology the faster the re-estimation of the RNs position is necessary. As a consequence, as mentioned at the end of Section 6, both the complexity of the solution and the performance achieved will

need to be analyzed to conclude the real benefits of the potential placement solutions.

Future work also includes the extension of the proposal described here to combine connectivity with other interesting goals, such as coverage or fault-tolerance. Alternative parameters such as high-level QoS related ones can be considered too.

8. Conclusions

The relay node (RN) placement topic is a complex engineering challenge. Although several approaches to this problem exist, the number of potential performance functions to be consider and the size of the associated search space, have fostered the development of a large amount of heuristic solutions for specific situations and scenarios, such as WSNs. Besides, several of these approaches are based on constraining the search space to simplify the optimization.

In this paper, a novel approach to relay placement for connectivity optimization in general multi-hop wireless networks is proposed. The approach is composed of three steps, the third one being optional. The first step is intended to identify a number of potential locations for the RNs, which is higher than that of the actually available RNs. After that, the best locations are identified using a certain selection algorithm. In a third and last (optional) phase , the locations are further optimized through an optimization algorithm. For the illustration of the approach in this paper, a leave-one-out (LOO) selection procedure and a particle swarm optimization (PSO) algorithm are used, but other combinations of methods may be evaluated in the future. Moreover, a Steiner-based approach is considered for initially distributing the theoretical RNs. It is also relevant to notice that two distributions for UNs are considered: MRWP and RPGM.

The approach is specially suited to optimize connectivity, which is defined in terms of inter-node reachability and network throughput and constitutes a challenging task due to the constraints imposed by the coverage rank. Besides, a procedure to smooth the constrained problem is proposed to run the second and third steps. The approach was evaluated and compared to other approaches in simulation, outstanding results being obtained.

Acknowledgments

Appendix: Particle swarm optimization

Bio-inspired optimization algorithms mimic special and smart behaviors in nature. The procedure for searching food in ant colonies, bird flocks and animal herds are some examples of this. The *swarm intelligence* paradigm [46] studies the collective behavior and properties of social structures in nature and how their components interact for a common objective. This section introduces and describes the fundamentals of one of the most used bio-inspired algorithms: the PSO algorithm. Some fields where PSO has been successfully applied are machine learning, dynamic system, bio-informatics and optimal RN placement, among others [47].

PSO was devised under the intelligence paradigm principles where a given situation, or *swarm*, is defined in terms of solutions, or *particles*. This way, PSO is an iterative population-based algorithm aimed at solving a problem through the evaluation of multiple alternative candidate solutions (the particles) by means of a common *objective function*. For that, PSO takes into account the experience acquired by each particle to generate new solutions.

Let us put these abstract concepts into a formal mathematical description. Assume $A \subset \mathbb{R}^n$ to be the specific target space, and $f : A \to Y \subseteq \mathbb{R}$ the objective function. The swarm can be defined as a set of N particles (candidate solutions)

$$\mathbf{S}^k = \left\{ x_1^k, x_2^k, \cdots, x_N^k \right\}, \tag{16}$$

each one defined as

$$x_i^k = (x_{i1}^k, x_{i2}^k, \cdots, x_{in}^k)^T \in A, \ i = 1, 2, \cdots, N \tag{17}$$

where the k index represents the k-th iteration of the algorithm and n, as defined, the dimension of the space, $\mathbb{R}^n$.

In order to move the particles around the target space to find an optimal solution, PSO excites them by applying a *velocity* term to each:

$$v_i^k = (v_{i1}^k, v_{i2}^k, \cdots, v_{in}^k)^T \in A, \ i = 1, 2, \cdots, N \tag{18}$$

The velocity is updated at each iteration of the algorithm depending on the previous best solutions for each particle, p_i^k, as well as on the global best solution computed so far. The best solution for each individual particle is obtained as:

$$p_i^k := \arg\min_k f(x_i^k) \tag{19}$$

Similarly, the best global solutions can be chosen from the set of best particle solutions:

$$p_g^k := \arg\min_i p_i^k \tag{20}$$

Finally, the complete (early) PSO version [48, 49] is obtained as:

$$v_{ij}^{k+1} = v_{ij}^k + c_1 \cdot R_1 \cdot (p_{ij}^k - x_{ij}^k) + c_2 \cdot R_2 \cdot (p_{gj}^k - x_{ij}^k) \tag{21}$$

$$x_{ij}^{k+1} = x_{ij}^k + \Delta t \cdot v_{ij}^{k+1} \tag{22}$$

where the updating process for the velocity and the positions of the particles (v_{ij}^{k+1} and x_{ij}^{k+1} respectively, where $i \in [1, N]$, $j \in [1, n]$, and $\Delta t = 1$ between two consecutive iterations) are shown. To update the velocity in Eq. (21) three terms are involved. The first one is the velocity of the particle in the previous iteration. The other two components exploit the knowledge acquired both by the particle (p_{ij}^k) and by the global procedure (p_{gj}^k). They are weighted by two multiplying factors: the *cognitive* factor, c_1, and the *social* factor, c_2, which goal is to tune the behavior of the algorithm from local to global search. Finally, R_1 and R_2 are random variables uniformly distributed in the range $[0, 1]$.

In order to avoid problems like the so-called *swarm explosion* [47], the velocity increments need to be clamped (*velocity clamping*). The clamping procedure depends on the problem variables and their magnitudes, which should be carefully studied. Another key issue in an optimization algorithm is the convergence. Convergence can be improved by the introduction of the parameter w known as *inertia coefficient* [50, 51]. This is applied to the particle velocity value at the previous iteration and it decreases with time. This decreasing trend of w promotes the algorithm solution exploration at early stages and solution exploitation when close to ending. According to the previous discussion, Eq. (21) is slightly modified redefining the PSO update law as:

$$v_{ij}^{k+1} = w \cdot v_{ij}^k + c_1 \cdot R_1 \cdot (p_{ij}^k - x_{ij}^k) + c_2 \cdot R_2 \cdot (p_{gj}^k - x_{ij}^k) \tag{23}$$

where it must be taken into account that this value is constrained by

$$v_{ij}^{k+1} = \begin{cases} v_{max}, & \text{if } v_{ij}^{k+1} > v_{max} \\ -v_{max}, & \text{if } v_{ij}^{k+1} < -v_{max} \end{cases} \tag{24}$$

References

[1] A. Alfaqdhly, U. Baroudi, M. Younis, "Optimal node repositioning for tolerating node failure in wireless sensor actor network." In *Proceedings of the 25th. Biennal Symposium on Communications*, pp. 67–71, 2010. [doi: 10.1109/BSC.2010.5473000]

[2] I.F. Senturk, K. Akkaya, S. Yilmaz, "Relay placement for restoring connectivity in partitioned wireless sensor networks under limited information." In *Ad Hoc Networks*, vol. 13, pp. 487–503, 2014. [doi: 10.1016/j.adhoc.2013.09.005]

[3] S. Evans, "Telecommunications Network Modelling, Planning and Design." The Institution of Engineering and Technology, London, 2003. [doi: 10.1049/PBBT006E]

[4] G. Held, S.R. Jagannathan, "Practical Network Design Techniques. A Complete Guide for WANs and LANs." Auerbach Publication - CRC Press, Boca Raton, 2004.

[5] M. Younis, K. Akkaya, "Strategies and techniques for node placement in wireless sensor networks: A survey." In *Ad Hoc Networks*, vol. 6, pp. 621–655, 2008. [doi: 10.1016/j.adhoc.2007.05.003]

[6] Z.A. Eu, H.P. Tan, W.K.G. Seah, "Routing and relay node placement in wireless sensor networks powered by ambient energy harvesting." In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 2003-2008, 2009. [doi: 10.1109/WCNC.2009.4917956]

[7] H. Xu, L. Huang, W. Gang, Y. Zhang, "Spanner-aware relay node placement in wireless ad hoc sensor networks." In *Proceedings of the 5th. International Conference on Mobile Ad-hoc and Sensor Networks (MSN)*, pp. 180–186, 2009. [doi: 10.1109/MSN.2009.14]

[8] B. Han, J. Li, J. Su, "Optimal relay node placement for multi-pair coorperative communication in wireless networks." In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 4724–4729, 2013. [doi: 10.1109/WCNC.2013.6555340]

[9] A. Barolli, F. Xhafa, M. Takizawa, "Optimization problems and resolution methods for node placement in wireless mesh networks." In *Proceedings of the 14th. International Conference on Network-Based Information Systems (NBiS)*, pp. 126–134, 2011. [doi: 10.1109/NBiS.2011.28]

[10] H. Safa, W. El-Hajj, H. Zoubian, "Particle swarm optimization based approach to solve the multiple sink placement problem in WSNs." In *Proceedings of the IEEE International Conference on Communications (ICC) - Wireless Network Symposium*, pp. 5445–5450, 2012. [doi: 10.1109/ICC.2012.6363906]

[11] O. Dengiz, A. Konak, A.E. Smith, "Connectivity management in mobile ad hoc networks using particle swarm optimization." In *Ad Hoc Networks*, vol. 9, n. 7, pp. 1312–1326, 2011. [doi: 10.1016/j.adhoc.2011.01.010]

[12] S. Misra, S.D. Hong, G. Xue, J. Tang, "Constrained relay node placement in wireless sensor networks: Formulation and approximations." In *IEEE/ACM Transactions on Networking*, vol. 18, n. 2, pp. 434–447, 2010. [doi: 10.1109/TNET.2009.2033273]

[13] E.L. Lloyd, G. Xue, "Relay node placement in wireless sensor networks." In *IEEE Transactions on Computers*, vol. 56, pp. 134–138, 2007. [doi: 10.1109/TC.2007.250629]

[14] W. Zhang, G. Xue, S. Misra, "Fault-tolerant relay node placement in wireless sensor networks: Problems and algorithms." In *Proceedings of the 26th. IEEE International Conference on Computer Communications (INFOCOM)*, pp. 1649–1657, 2007. [doi: 10.1109/INFCOM.2007.193]

[15] J.L. Bredin, E.D. Demaine, M. Hajiaghayi, D. Rus, "Deploying sensor networks with guaranteed fault tolerance." In *IEEE/ACM Transactions on Networking*, vol. 18, pp. 216–228, 2010. [doi: 10.1109/TNET.2009.2024941]

[16] X. Han, X. Cao, E.L. Lloyd, C.S. Shen, "Fault-tolerant relay node placement in heterogeneous wireless sensor networks." In *IEEE Trans-*

actions on Mobile Computing, vol. 9, n. 5, pp. 643–656, 2010. [doi: 10.1109/TMC.2009.161]

[17] A. Kashyap, S. Khuller, M. Shayman, "Relay placement for fault tolerance in wireless networks in higher dimensions." In *Computational Geometry*, vol. 44, n. 4, pp. 206–215, 2011. [doi: 10.1016/j.comgeo.2010.11.002]

[18] B. Hao, J. Tang, G. Xue, "Fault-tolerant relay node placement in wireless sensor networks: Formulation and approximation." In *2004 Workshop on High Performance Switching and Routing (HPSR)*, pp. 246–250, 2004. [doi: 10.1109/HPSR.2004.1303479]

[19] H. Liu, P.J. Wan, X. Jia, "Fault-tolerant relay node placement in wireless sensor networks." In *Computing and Combinatorics*, L. Wang, Ed. Springer Berlin Heidelberg, vol. 3595, pp. 230–239, 2005. [doi: 10.1007/11533719_25]

[20] K. Xu, Q. Wang, H. Hassanein, G. Takahara, "Optimal wireless sensor networks (WSNs) deployment: Minimum cost with lifetime constraint." In *Proceedings of the IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, vol. 3, pp. 454–461, 2005. [doi: 10.1109/WIMOB.2005.1512937]

[21] Q. Wang, G. Takahara, H. Hassanein, K. Xu, "On relay node placement and locally optimal traffic allocation in heterogeneous wireless sensor networks." In *Proceedings of the IEEE Conference on Local Computer Networks (LCN)*, pp. 657–664, 2005. [doi: 10.1109/LCN.2005.95]

[22] Q. Wang, K. Xu, H. Hassanein, G. Takahara, "Minimum cost guaranteed lifetime design for heterogeneous wireless sensor networks (WSNs)." In *Proceedings of the 24th. IEEE International Performance, Computing, and Communications Conference*, pp. 599–604, 2005. [doi: 10.1109/PCCC.2005.1460645]

[23] Q. Wang, K. Xu, G. Takahara, H. Hassanein, "Locally optimal relay node placement in heterogeneous wireless sensor networks." In *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM)*, vol. 6, pp. 3549–3553, 2005. [doi: 10.1109/GLOCOM.2005.1578433]

[24] J. Tang, B. Hao, A. Sen, "Relay node placement in large scale wireless sensor networks." In *Computer Communications*, vol. 29, n. 4, pp. 490–501, 2006. [doi: 10.1016/j.comcom.2004.12.032]

[25] S. Misra, S.D. Hong, G. Xue, J. Tang, "Constrained relay node placement in wireless sensor networks to meet connectivity and survivability requirements." In *Proceedings of the 26th. IEEE International Conference on Computer Communications (INFOCOM)*, pp. 879–887, 2008. [doi: 10.1109/INFOCOM.2008.65]

[26] A. Nigam, Y.K. Agarwal, "Optimal relay node placement in delay constrained wireless sensor network design." In *European Journal of Operational Research*, vol. 233, n. 1, pp. 220–233, 2014. [doi: 10.1016/j.ejor.2013.08.031]

[27] A. Nigam, Y.K. Agarwal, "Optimal relay placement in wireless sensor networks using node cut inequalities." In *Proceedings of the 4th. International Conference on Communication Systems and Networks (COMSNETS)*, pp. 1–8, 2012. [doi: 10.1109/COMSNETS.2012.6151353]

[28] X. Cheng, D.Z. Du, L. Wang, B. Xu, "Relay sensor placement in wireless sensor networks." In *Wireless Networks*, vol. 14, n. 3, pp. 347–355, 2008. [doi: 10.1007/s11276-006-0724-8]

[29] S. Lee, M. Younis, "Optimized relay node placement for connecting disjoint wireless sensor networks." In *Computer Networks*, vol. 56, n. 12, pp. 2788–2804, 2012. [doi: 10.1016/j.comnet.2012.04.019]

[30] D. Yang, S. Misra, X. Fang, G. Xue, J. Zhang, "Two-tiered constrained relay node placement in wireless sensor networks: Computational complexity and efficient approximations." In *IEEE Transactions on Mobile Computing*, vol. 11, n. 8, pp. 1399–1411, 2012. [doi: 10.1109/TMC.2011.126]

[31] L. Wang, X. Fu, J. Fang, H. Wang, M. Fei, "Optimal node placement in industrial wireless sensor networks using adaptive mutation probability binary particle swarm optimization algorithm." In *Proceedings of the 7th. International Conference on Natural Computation (ICNC)*, pp. 2199–2203, 2011. [doi: 10.1109/ICNC.2011.6022417]

[32] N. Aziz, A.W. Mohemmed, M.Y. Alias, "A wireless sensor network coverage optimization algorithm based on particle swarm optimization

and Voronoi diagram." In *Proceedings of the International Conference on Networking, Sensing and Control (ICNSC)*, pp. 602–607, 2009. [doi: 10.1109/ICNSC.2009.4919346]

[33] H.Z. Abidin, N. Din, "Provisioning WSN coverage via minimax based sensor node placement scheme." In *Proceedings of the International Conference on Wireless Communications and Applications*, pp. 1–5, 2012.

[34] H. Liu, X. Chu, Y.W. Leung, R. Du, "Simple movement control algorithm for bi-connectivity in robotic sensor networks." In *IEEE Journal on Selected Areas in Communications*, vol. 28, n. 7, pp. 994–1005, 2010. [doi: 10.1109/JSAC.2010.100904]

[35] B. Wang, H. Xu, W. Liu, H. Liang, "A novel node placement for long belt coverage in wireless networks." In *IEEE Transactions on Computers*, vol. 62, n. 12, pp. 2341–2353, 2013. [doi: 10.1109/TC.2012.145]

[36] E.F. Flushing, G.A. Di Caro, "A flow-based optimization model for throughput-oriented relay node placement in wireless sensor networks." In *Proceedings of the 28th. Annual ACM Symposium on Applied Computing*, pp. 632–639, 2013. [doi: 10.1145/2480362.2480482]

[37] I. Bekmezci, O.K. Sahingoz, S. Temel, "Flying ad-hoc networks (FANETs): A survey." In *Ad Hoc Networks*, vol. 11, n. 3, pp. 1254–1270, 2013. [doi: 10.1016/j.adhoc.2012.12.004]

[38] T. Saitou, M. Nukada, Y. Uchimura, "Deployment control of mobile robots for wireless network relay based on received signal strength." In *Proceedings of the IEEE Workshop on Advanced Robotics and its Social Impacts (ARSO)*, pp. 237–242, 2013. [doi: 10.1109/ARSO.2013.6705535]

[39] J. Nocedal, S. Wright, "Numerical Optimization." Springer, New York, 2006.

[40] E. Tsang. *Foundations of Constraint Satisfaction*. Academic Press, 1993.

[41] R. C. Prim, "Shortest Connection Networks And Some Generalizations." Bell System Technical Journal, vol. 36, no. 6, pp. 1389–1401, 1957. [doi: 10.1002/j.1538-7305.1957.tb01515.x]

[42] X. Hong, M. Gerla, G. Pei, C.C. Chiang, "A group mobility model for ad hoc wireless networks." In *Proceedings of the 2nd. ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, pp. 53–60, 1999. [doi: 10.1145/313237.313248]

[43] D.B. Johnson, D.A. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks." In *Mobile Computing*, The Kluwer International Series in Engineering and Computer Science. Springer US, pp. 153–181, 1996. [doi: 10.1007/978-0-585-29603-6_5]

[44] R. Magán-Carrión, J. Camacho-Páez, P. García-Teodoro, "A multiagent self-healing system against security incidents in MANETs." In *Proceedings of the Practical Applications of Heterogeneous Multi-Agent Systems (PAAMS)*, vol. 430, pp. 321–332, 2014. [doi: 10.1007/978-3-319-07767-3_29]

[45] A. S. Huang, E. Olson, and D. C. Moore, "LCM: Lightweight Communications and Marshalling." In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4057–4062, 2010.

[46] R.C. Eberhart, Y. Shi, J. Kennedy, "Swarm Intelligence." Elsevier, 2001.

[47] K.E. Parsopoulos, M.N. Vrahatis (Eds.), "Particle Swarm Optimization and Intelligence: Advances and Applications." IGI Global, 2010.

[48] J. Kennedy, R. Eberhart, "Particle swarm optimization." In *Proceedings of the IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948, 1995. [doi: 10.1109/ICNN.1995.488968]

[49] R. Eberhart, J. Kennedy, "A new optimizer using particle swarm theory." In *Proceedings of the the 6th. International Symposium on Micro Machine and Human Science*, pp. 39–43, 1995. [doi: 10.1109/MHS.1995.494215]

[50] Y. Shi, R. Eberhart, "A modified particle swarm optimizer." In *IEEE World Congress on Computational Intelligence - Proceedings of Evolutionary Computation*, pp. 69–73, 1998. [doi: 10.1109/ICEC.1998.699146]

[51] Y. Shi, R.C. Eberhart, "Parameter selection in particle swarm optimization." In *Proceedings of the 7th. International Conference on Evolutionary Programming VII (LNCS 1447)*, pp. 591–600, 1998. [doi: 10.1007/BFb0040810]