Efficient coverage for grid-based mobile wireless sensor networks

Valeria Loscrì, Enrico Natalizio, Francesca Guerriero, Nathalie Mitton

▶ **To cite this version:**

Valeria Loscrì, Enrico Natalizio, Francesca Guerriero, Nathalie Mitton. Efficient coverage for grid-based mobile wireless sensor networks. ACM symposium on Performance evaluation of wireless ad hoc, sensor, & ubiquitous networks, Sep 2014, Montreal, Canada. pp.53-60, 10.1145/2653481.2653489 . hal-01121280

HAL Id: hal-01121280
https://inria.hal.science/hal-01121280

Submitted on 27 Feb 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Efficient Coverage for Grid-Based Mobile Wireless Sensor Networks

Valeria Loscrí*, Enrico Natalizio†, Francesca Guerriero‡, Nathalie Mitton*
* Inria Lille - Nord Europe, France, firstname.lastname@inria.fr
† Université de Technologie de Compiègne, France, enrico.natalizio@hds.utc.fr
‡ University of Calabria, Italy, guerriero@unical.it

***Abstract*—In this paper we propose a new coverage technique for mobile wireless sensor networks in which the sensors are arranged in a square grid. Our technique is totally distributed and based only on local information. We show how this extremely simple technique is able to reach good performance in terms of coverage even when sensor fields with different features are considered. Specifically, we focus on sensor fields with and without obstacles for different densities of nodes. We show the effectiveness of the proposed technique by comparing it with the results obtained by an optimization model and two coverage algorithms, based on a genetic approach and virtual forces, respectively. Results of simulations show how, in the worst case, the coverage obtained by our scheme is 14% smaller than the coverage obtained with the optimization model. When the size of the field increases the searching space of our scheme increases too and we obtain a better performance in terms of coverage. In fact, for larger sensor fields, both with obstacles and without obstacles, our scheme outperforms the other two algorithms.**

I. Introduction

In distributed Wireless Sensor Networks (WSNs), design of the placement of sensors devices in the sensor field is a very important issue. Efficient sensor placement strategies allow to minimize cost and reach a high level of accuracy [5]. In the art gallery problem, cameras are deployed such that the whole gallery is thief-proof [19], [11]. A wireless sensor network must achieve the specified coverage level of the application so that the quality of service provided by the wireless sensor network can be guaranteed. The coverage is considered a critical issue that has been the focus of many research activities with different associated issues, such as presence of obstalce, coverage of specific areas, etc. In [26], authors introduce the concept of transparent obstacle, that is a specific zone where a node cannot be deployed, but the signal is allowed to pass. Their proposal consists in the reduction of the original coverage problem in a discrete approach named Minimum Geometric Disk Cover with Candidate position (MGDCCP) and they present a polynomial-time approximation scheme instead of directly considering the MGDCCP, that is proved to be NP-hard. Some other approaches consider the coverage problem in combination with Particle Swarm Optimization (PSO) and Differential Evolution [15], by considering as primary QoS parameter the reduction of the energy consumption. Many sensor deployment algorithms attempt to fully cover a sensor field by using the minimum number of sensors or by minimizing the cost associated to the nodes. In [6] authors propose a method to deploy sensors to provide full coverage of a sensor field in presence of obstacles. In [8] authors propose a distributed algorithm in order to activate a subset of sensors to fully cover the entire sensor field all at once. In [1] authors try to ensure that each point in a sensor field is covered by at least k sensors. In order to cover the field in an effective way, they select a subset of sensors for ensuring k-coverage and k'-connectivity. Some coverage techniques consider the components in the sensor field such as sensors, obstacles or preferential fields as virtual forces sources and try to balance virtual forces through the deployment of the devices [10] and [12]. In [27] authors propose CCAN that selects a connected dominating set in a dense wireless sensor network such that the coverage probabilities of the given points (e.g. specific targets) are larger than a given parameter. In many applications, some specific areas need to have a larger coverage than others, in this case the problem is that of deploying a wireless sensor network to fully cover these areas as well as not leaving uncovered the rest of the field. In [13] authors consider the problem of deploying a wireless sensor network to fully cover critical grids of equilateral triangles by deploying the minimum number of sensors on grid points and they show that this problem is NP-Complete. In the circle covering problem, usually homogeneous circles are used to fully cover equilateral triangles [17], rectangle [16], and squares [18]. The circle covering problem is different from the covering problem considered in this paper since the circles are independent and can be moved anywhere. Moreover, we have to consider the constraint of connectivity, so the potential available position in the sensor field are limited. Usually a sensor field is divided into square grids [5], [25] and in [14] the problem of deploying the minimum number of sensors on grid points to construct a wireless sensor network fully covering critical square grids is considered and authors show that this problem is NP-Complete. Several of the mentioned schemes consider sensors equipped with mobility capability. In this paper we focus on the deployment of mobile sensor nodes on a grid sensor field where a certain number of obstacles can limit transmission and mobility capabilities of the nodes. The objective of our study is to design a simple heuristic able in a totally distributed way to improve the coverage of the mobile sensor network by using local information only.

In order to show the effectiveness of the proposed technique we introduce an optimization model that is able to find the best position for a sensor node in terms of coverage and

two algorithmic approaches: a centralized technique based on a genetic algorithm and a virtual forces based approach. Simulation results show how our technique is able to produce a coverage that is close to the optimization model, and when the size of the field increases it overcomes the virtual forces technique for both the scenarios, with and without obstacles.

In Section II we consider square grids. In Section III we present the optimization model used as a benchmark. In Section IV we present the details of the simple heuristic we propose and we present two comparison centralized approaches: the former based on a genetic approach and the latter based on the Virtual Forces concept. Moreover, we explain the results in Section V and finally, we conclude the paper and consider a discussion of future research in Section VI. In the Appendices, we introduce the concept of Zone of Interest (ZoI), that is a specific zone in the field that requires a larger coverage by the sensor network, and we show how our mathematical model and heuristic can be easily extended in order to consider different ZoI in the field. For space reasons, we have omitted the results of this variant of the heuristic that we will present in an extended version of the work.

II. Square Grid Networks

A wireless sensor network is said *grid-based* if it consists of a (potentially unbounded) number of identical sensors arranged in a square grid. A cell can be occupied by at most one sensor node. Classically, if a sensor has a maximum transmission range said t_r, then each node is able to communicate with all nodes within the circle of radius t_r in a direct way. In a grid-based network we need to model the connectivity among nodes in a different way, i.e. using a *Lee sphere* of appropriate radius [9]. Let the length of the sides of the squares of our grid be of unitary length. The distance among the squares in the grid can be measured in terms of Manhattan metric, that is the distance between two squares is the sum of the horizontal and vertical distances between the centers of the squares. With "horizontal" and "vertical" we indicate the two perpendicular directions parallel to the sides of the squares in the grid.

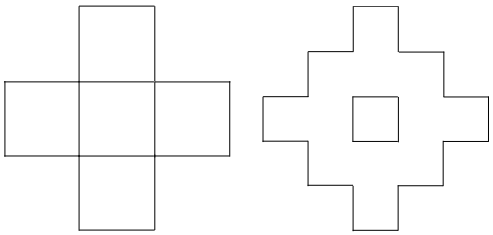
Definition 1. *A Lee sphere of radius r centered at a given square consists of the set of squares that lie at Manhattan distance at most r from that square as shown in Fig. 1.*

This approximation facilitates the representation of the coverage in the grid-based networks. In this paper we refer to the Lee sphere of radius 2 as shown in Fig. 1(b). In practice, all the points that are within the squares are considered covered by a node at the center of the Lee sphere.

In the rest of the paper we will refer to the farthest frame of cells around the sensor node as *exterior frame* and to the closest as *interior frame*. Also we will take account of the possibility to have a physical obstruction in the cells of the grid, we will refer to this obstruction that occupies either part or the whole cell as an *obstacle*.

III. The optimization models

In this section, we present different formulations for the problem under consideration, defined in such a way to handle several practical scenarios. It is assumed that the sensor field is represented by a two-dimensional grid. The parameters used in the formulations are the following: h: the grid height, w: the grid width, d: the discretization step, n: the number of available sensors, r: the coverage radius, M: a large positive number. The variables of the proposed models are: (x_k, y_k), $k = 1, \ldots, n$ the Cartesian coordinates that indicate the location of the sensor k in the sensor field; ϕ_{ijk}, $i = 1, \ldots, \lceil h/d \rceil$, $j = 1, \ldots, \lceil w/h \rceil$, $k = 1, \ldots, n$ a binary variable that takes the value one if the location (i, j) is covered by sensor k, and zero otherwise; δ_{ij}, $i = 1, \ldots, \lceil h/d \rceil$, $j = 1, \ldots, \lceil w/d \rceil$, a binary variable that takes the value one if the location (i, j) is covered by at least one sensor, and zero otherwise.



(a) (b)

Fig. 1: Lee spheres of radius 1 (a) and 2 (b).

A. Field without obstacles

The first mathematical formulation whose aim is to deploy the sensors in a field without obstacles such that the coverage is maximized is given in what follows:

$$max \quad \sum_{i=1}^{\lceil h/d \rceil} \sum_{j=1}^{\lceil w/d \rceil} \delta_{ij} \tag{1}$$

$$r - \sqrt{(i - x_k)^2 + (j - y_k)^2} \geq M\ (\phi_{ijk} - 1), \quad \forall i, j, k \tag{2}$$

$$\delta_{ij} \leq \sum_{k=1}^{n} \phi_{ijk}, \quad \forall i, j \tag{3}$$

$$M\ \delta_{ij} \geq \sum_{k=1}^{n} \phi_{ijk}, \quad \forall i, j \tag{4}$$

$$0 \leq x_k \leq \lceil h/d \rceil, \quad 0 \leq y_k \leq \lceil w/d \rceil, \quad \forall k \tag{5}$$

$$x_k, \ \ y_k \ \ integer, \quad \forall k \tag{6}$$

$$\phi_{ijk} \quad binary, \quad \forall i, j, k \tag{7}$$

$$\delta_{ij} \quad binary, \quad \forall i, j \tag{8}$$

The objective function in (1) maximizes the number of locations covered by at least one sensor. Conditions (2) state that if the euclidean distance between the sensor k and the location (i, j) is lower than or equal to the coverage radius r than the variable ϕ_{ijk} takes the value one, otherwise it is set to zero. Constraints (3) and (4) are logical constraints and ensure that the indicator variable δ_{ij} takes on a value of one if the location (i, j) is covered by at least one sensor and zero otherwise. Finally, conditions (5)-(8) represent the variable domain constraints. The mathematical formulation reported

above is an integer nonlinear programming model, where the nonlinearity is confined to the constraints (2).

In order to derive an integer linear model, the euclidean distance has been replaced by the following expression: $d_{x_{ik}} + d_{y_{jk}} - 0.5min(d_{x_{ik}}, d_{y_{jk}})$, where $d_{x_{ik}} = |i - x_k|$ and $d_{y_{jk}} = |j - y_k|$. This approximation, introduced in [20], overestimates distance and yields error in interval 0% to 12%.

To eliminate the terms with the absolute value, we introduce the additional constraints reported below ([3]):

$$d_{x_{ik}} \geq i - x_k \quad \forall i, k \tag{9}$$

$$d_{x_{ik}} \geq -i + x_k \quad \forall i, k \tag{10}$$

$$d_{y_{jk}} \geq j - y_k \quad \forall j, k \tag{11}$$

$$d_{y_{jk}} \geq -j + y_k \quad \forall j, k \tag{12}$$

Thus, constraints (2) are replaced by the following conditions:

$$r - d_{x_{ik}} - d_{y_{jk}} + 0.5\, min_{ijk} \geq M\ (\phi_{ijk} - 1), \quad \forall i, j, k \tag{13}$$

where the variables min_{ijk}, $\forall i, j, k$ take the minimum value between $d_{x_{ik}}$ and $d_{y_{jk}}$. Consequently, the following constraints need to be satisfied:

$$min_{ijk} \leq d_{x_{ik}} \quad \forall i, j, k \tag{14}$$

$$min_{ijk} \leq d_{y_{jk}} \quad \forall i, j, k \tag{15}$$

The proposed integer linear model (referred in the rest of the paper as no-obstacles optimization model *NObOM*) optimizes the function (1) subject to constraints (9)-(15) and (3)-(8).

B. Field with obstacles

It is worth observing that the mathematical formulation reported above can be easily extended to handle the specific situation in which some obstacles are present in the sensor field. In this particular case, it is sufficient to impose that the locations occupied by the obstacles cannot be considered as feasible positions where the sensors can be localized.

Let $\mathcal{OBS} = \{(\tilde{i}, \tilde{j}) \mid \tilde{i} \geq 1 \ and \ \tilde{i} \leq \lceil h/d \rceil \ and \ \tilde{j} \geq 1 \ and \ \tilde{j} \leq \lceil w/h \rceil\}$ be the set of the positions in which the obstacles are located, the mathematical formulation of the problem in which the obstacles are considered (referred in the rest of the paper as obstacles optimization model *ObOM*), is obtained by replacing in the model reported above the constraints (5) with the following conditions:

$$0 \leq x_k \leq \lceil h/d \rceil, \quad 0 \leq y_k \leq \lceil w/d \rceil, \ (x_k, y_k) \notin \mathcal{OBS} \quad \forall k \tag{16}$$

IV. Distributed algorithm

In this section we present the details of the coverage map assisted technique. It is worth noting that this technique is totally distributed and based on local information only. The proposed algorithm is based on a local coverage grid map as depicted in green shaded cells in Fig. 2. The letters call a cell, the number represents their coverage degree. For instance, the upper right cell is cell a and has a coverage degree of zero. We assume that the transmission range is larger than the sensing range so that every node is able to communicate with nodes in its local coverage grid map. Thus, a node can exploit information that come from nodes placed within the *exterior frame*. In any case, the coverage will be computed by considering the *Lee sphere* model as shown in Fig. 1 (b). Each node, by communicating with its neighbors, is able to calculate the local map needed to compute the movement to improve the coverage. In Fig. 2, the Lee sphere of cell ac is represented by the grey-dashed cells, the bue-shaded cells are the cells occupied by nodes, while the number in each white cell is the number of nodes that cover that cell. Henceforth, we will refer to this number as the *coverage degree* of a cell.

Definition 1 (Coverage degree)**.** *The coverage degree of a cell represents the number of nodes that are in the neighborhood of this cell and can "sense" it. In our case, a node covers a cell if it lays in the Lee sphere of this cell.*

a 0	b 1	c0	d0	e0	f1	g1	h1
i 1	j 1	k1	l0	m2	n1	o1	p1
q1	r1	s1	t2	u1	v2	w1	x1
y1	z1	aa2	ab1	ac 1	ad1	ae2	af1
ag0	ah1	ai1	aj1	ak1	al2	am1	an1
ao0	ap1	aq1	ar1	as2	at1	au1	av1
aw1	ax1	ay1	az1	ba1	bb2	bc1	bd1

Fig. 2: Coverage grid example, the blue cells indicate the presence of a sensor in the cell. The letters call a cell, the number represents their coverage degree. The dash grey cells represent the Lee sphere of node in cell ac. North and South-East borders are highlighted in black.

The basic idea of the algorithm is to base nodes movement on the information transmitted by the nodes placed in the cells of the exterior frame of the local map, i.e. in cells $k, m, n, l, o, s, aa, ai, aq, ar, as, at, au, am$ and ae for a node in cell ac. Specifically, we individuate 8 possible directions of coverage degree computation and movement: North (N), North-East(NE), East (E), South-East (SE), South (S), South-West (SW), West(W), North-West (NW). In Fig. 2 the North and the South-East borders of cell ac are highlighted by a black border. North border is composed of cells k, m, n, l and o. south-east border is composed by cells ae, am, au, at and as. The algorithm sums up the coverage degree obtained on each border and makes the node move according to the highest difference between geographically opposite borders, as explained in the following 4 steps:

STEP 1: Local Coverage Map Construction. The construction of the local coverage map is performed by each node using position information provided by neighboring nodes. This information can be regularly sent by using classical HELLO messages that contain the position of the sensor. It is important to notice here that sensors may have different coverage capabilities. In this case, the HELLO messages may also contain the identification of the covered cells. In this work we assume homogeneous capabilities of coverage and leave

this issue as a future work. Of course, wireless environment is unreliable and it can happen that some information is missed, but the frequency of message exchange is such that the impact of some missed information can be neglected. The inclusion in HELLO messages is also possible, since the amount of additional information considered (e.g. the overhead of the protocol) is really minimal. This step allows every node to set up of a $n \times n$ matrix later referred as map, where n is the radius of Lee sphere. If x, y is the position of the current node, $map(x-i, y-j)$ with $i, j \in [0...n]$ is the coverage degree of cell at position $x-i, j-i$. For instance on Fig. 2, in the local map of node ac in position $5, 4$, we have $map(4, 4) = 1$.

STEP 2: Coverage Degree Computation. The coverage degree computation is performed at each node by summing up the coverage degree of nodes belonging to the same border. For example in Fig. 2 the North border coverage degree of node in ac is $2 + 2 + 3 + 2 + 2 = 11$ and the South-East border coverage degree is $1 + 0 + 0 + 1 + 1 = 3$.

STEP 3: Direction Computation. In order to increase and balance the coverage provided by our algorithm, each sensor decides its movement direction depending on the difference of coverage measured on geographically opposite borders. Specifically, we compute the difference between the North and South borders, East and West borders, North-East and South-West border and North-West and South-East borders. Based on these differences, the sensors will move along the appropriate axis by choosing the highest absolute differential value.

STEP 4: Movement. Upon the computation of the axis is completed, nodes decide their movement by choosing the lowest coverage density along the chosen axis and move in that direction. In case of equality, a random direction is chosen between candidates except if all 8 directions have the same degree. In this latter step, the algorithm terminates.

The details of the algorithm are given in the following pseudo-code. It is important to notice here that the algorithm is constantly, at a regular interval, run on each sensor. Moreover, we also add the termination condition to avoid oscillation. The termination condition holds whether there is equality between gradients on borders or if the current position has already been visited during the four previous steps except the last one. Indeed, our algorithms accepts that a node may take a bad decision once. We verified experimentally that four previous steps is enough to ensure the convergence of the algorithm.

It is worth noticing that the simple heuristic proposed in this work is a kind of greedy technique. This implies that in every iteration each node tries to reach its better solution that does not correspond to the global optimal. This kind of approach could imply the generation of loop that we will avoid by storing the last "four" positions of each node. We dealt simulations with different number of positions in order to make a kind of sensitivity analysis. Specifically, we perform tests by considering the last 3, 4, 5 and 6 positions, and we noticed that from 3 to 4 we could obtain different performance in terms of positioning, but from 4 to 5 and also 6, results were the same with the only difference that the greater the number of positions "stored"to be considered, the greater the number of iterations and for that we considered to store the last 4 positions. If a node occupies a position already occupied in the last iterations, it will stop moving again and the total algorithm will globally stop when all the nodes stopped. One important property of our algorithm is that it uses only local information. This makes it scalable and gives it the possibility to easily manage the obstacle avoidance. Indeed, obstacles can be easily taken into account for the movement decision by artificially setting the coverage degree value of the cell that they occupy to $-\infty$ to induce a movement toward the obstacle or to $+\infty$ to induce a movement against the obstacle. We will refer to this value as *obstacle avoidance*. The same parameter is used also to consider the border of the field.

Algorithm 1 Coverage Map based Movement Algorithm

• **Local variables:** current position of node (x, y); coverage radius r; coverage degree East C_e; West C_w; North C_n; South C_s; North-East C_{ne}; North-West C_{nw}; South-East C_{se}; South-West C_{sw};

Part I — Local Coverage Map Construction:

Get current position (GPS or any other localization system) (x, y);
Every HelloFrequency seconds; Broadcast an HELLO message m with id, (x, y);
Upon reception of an HELLO message, update coverage map;

Part II — Coverage Degree Computation:

$C_n = \sum_{j=x-r}^{x+r} map(y + r, j)$;
$C_e = \sum_{i=y-r}^{y+r} map(i, x + r)$;
$C_s = \sum_{j=x-r}^{x+r} map(y - r, j)$;
$C_w = \sum_{i=y-r}^{y+r} map(i, x - r)$;
$C_{ne} = \sum_{j=x}^{x+r} map(y + r, j) + \sum_{i=y+r}^{y} map(i, x + r)$;
$C_{se} = \sum_{j=x}^{x+r} map(y - r, j) + \sum_{i=y-r}^{y} map(i, x + r)$;
$C_{sw} = \sum_{j=x-r}^{x} map(y - r, j) + \sum_{i=y-r}^{y} map(i, x - r)$;
$C_{nw} = \sum_{j=x-r}^{x} map(y + r, j) + \sum_{i=y}^{y+r} map(i, x - r)$;

Part III — Direction Computation:

$S_N = (C_n - C_s) * (C_n - C_s)$;
$E_W = (C_w - C_e) * (C_w - C_e)$;
$NE_SW = (C_{sw} - C_{ne}) * (C_{sw} - C_{ne})$;
$NW_SE = (C_{se} - C_{nw}) * (C_{se} - C_{nw})$;
$DIR = \max(S_N, E_W, NE_SW, NW_SE)$
in case of tie choose a random direction between winners

Part IV — Movement:

if $(DIR == S_N)$ **then**
  if $(C_n \geq C_s)$ **then** Move South
  else Move North
else if $(DIR == E_W)$ **then**
  if $(C_w \geq C_e)$ **then** Move East
  else Move West
else if $(DIR == NE_SW)$ **then**
  if $(C_{sw} \geq C_{ne})$ **then** Move North-East
  else Move South-West
else if $(DIR == NW_SE)$ **then**
  if $C_{nw} \geq C_{se})$ **then** Move South-East
  else Move North-West
end if

PartV — Termination condition:

Stop movement if $(x(t), y(t)) == (x(t-2), y(t-2))$**or**$(x(t), y(t)) == (x(t-3), y(t-3))$**or**$(x(t), y(t)) == (x(t-4), y(t-4))$ **or** all directions have the same coverage degree.

V. Performance Evaluations

In this section, we evaluate the performance of the proposed heuristic by comparing it with the optimization model presented in Section III and two algorithms that use a centralized genetic approach and a virtual forces based approach, respectively. All the techniques considered in this paper have been

adapted to use the grid-based model described in Section II as far as the sensing model is concerned. As already outlined, we consider Lee sphere of radius 2.

For the performance assessment, we consider three increasing sensor field sizes and a variable density of nodes in the grid in order to evaluate the scalability of the simulated schemes. Also we present two simulation campaigns for sensor fields with and without obstacles. In this case we fix the number of node and make the percentage of obstacles in the field grow in order to show the effect of limiting factors on nodes' transmission and mobility.

A. Other Coverage Techniques

In the following we present the most important details of two algorithms taken from the literature that will be uses in the comparison with our scheme. In Table I we reported the values of the most important parameters used from these two schemes as well as our heuristic.

TABLE I: Simulation Parameters

Searching Space	10×10, 15×15, 25×25 cells
# nodes	26, 38, 120
Coverage radius (r)	2 m
Transmission radius	(2 × r) m
Cell size	1x1 m^2
Obstacle Avoidance Factor (Heuristic)	0
Attractive Force (VFA)	0.01
Repulsive Force (VFA)	1000
Threshold Distance (VFA)	(2 × r) m
Forces Exclusion Threshold (VFA)	4 × r m
# Generations (GA)	2000
Population Size (GA)	500
Crossover Fraction (GA)	0.8
Mutation Factor (GA)	1
Confidence Interval	95 %

1) Genetic Algorithm: This technique is taken from [7] and adapted to fit our grid-based coverage problem. The genetic algorithm is used as an optimization method in order to find the optimal solution by reducing the solution space through the evolutionary approach. A particular placement is coded as a member of a population by the sequence of x and y coordinates of all sensors in the grid as illustrated in Fig. 3.

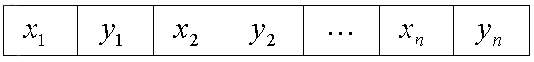


Fig. 3: Placement coded in a chromosome

The objective of the optimization problem is formalized as the minimization of uncovered cells, consequentially, the fitness function used for the evolution of the population is:

$$f = \#_total_cells - \#_covered_cells \tag{17}$$

This function takes a given placement and the positions of obstacles (if any) in the grid as input, and it gives a fitness value that represents the number of holes in the grid as output. Substantially, the population evolves through the genetic algorithm using the fitness function as the supervisor of the evolution process. The steps of the algorithm are:

1) random initialization of a population;
2) ranking of population members according to the fitness function;
3) selection, mutation and crossover of best members of the population;
4) repeat from point (1) until termination condition (maximum number of generations) is satisfied.

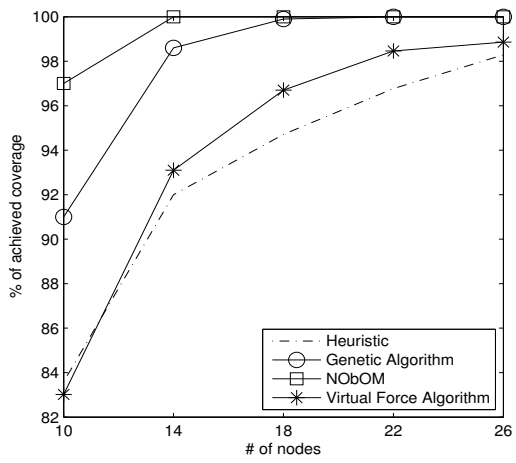
We tuned the algorithm by running some preliminary simulations in order to achieve the best results for the coverage. Firstly, a number of experiments were conducted to determine appropriate population size, size ranging from 100 to 1000 individuals. we obtained best performance in correspondence of 500. We also considered several simulations with different crossover methodologies, and we selected a crossover fraction equal to 0.8. Similarly, a Gaussian mutation methodology with scale and shrink factor equal to 1 was found as the best value in terms of performance. The values used for the mentioned parameters in the simulations are reported in Table I.

2) Virtual Forces Algorithm (VFA): VFA is a well-known technique used for deployment in mobile wireless sensor networks. It achieves good performance in terms of coverage by starting from a random placement, and by using a weighted combination of repulsive and attractive forces based on mutual distances among nodes. The main advantage of this approach is its flexibility to adapt to different scenarios also in presence of obstacle. The drawbacks regard the difficulty to tune opportunely some parameters that depend on the particular scenario and the fact that performance of VFA are strictly related to the tuning of the parameters for the specific scenario. We adapt the algorithm proposed in [2] to fit assumptions and properties used in this work and obtain the best performance related to the specific scenarios considered. The values used for VFA in the simulations are reported in Table I.

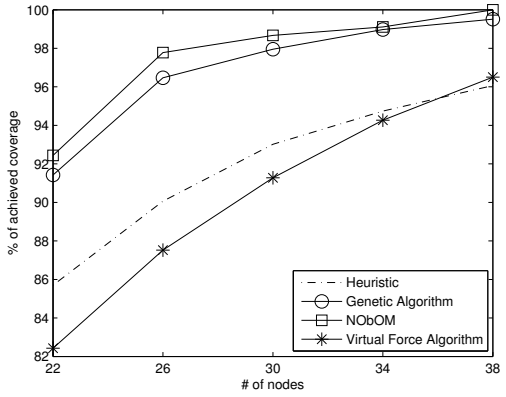
B. Results

As already said, results show two simulation campaigns, which consider sensor grid with and without obstacles.

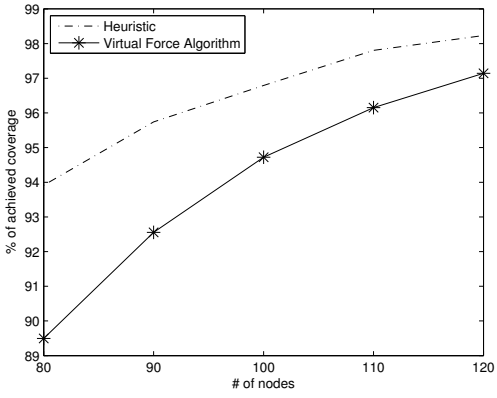
1) Field with no obstacles: Fig. 4 plots the coverage achieved by the mathematical model and the three algorithms for the three different grid sizes, when the number of nodes in the grid varies and no obstacles are present on the sensor field. We can see that for small grid size the coverage achieved by our technique is the worst, whereas for the intermediate size, it overcomes the results obtained by the VF algorithm. Furthermore, when we compare the coverage obtained with our technique and the virtual forces approach, we can observe how the heuristic outperforms the virtual force technique for the biggest size of the grid. This result is encouraging because it implies that our heuristic is scalable. The improved performance is related to an increased movement space. In practice, with the 15×15 grid, the heuristic can exploit more information. These considerations are confirmed by considering also a 25×25 grid, as shown in 4 (c).



(a)



(b)

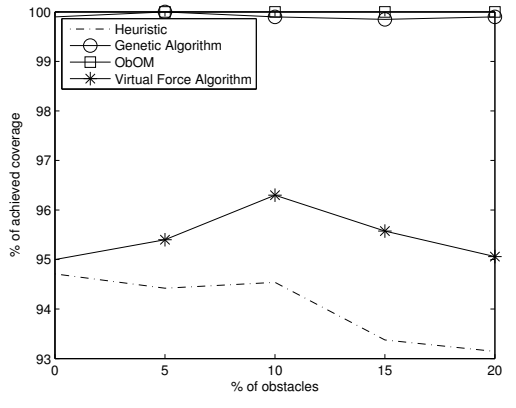


(c)

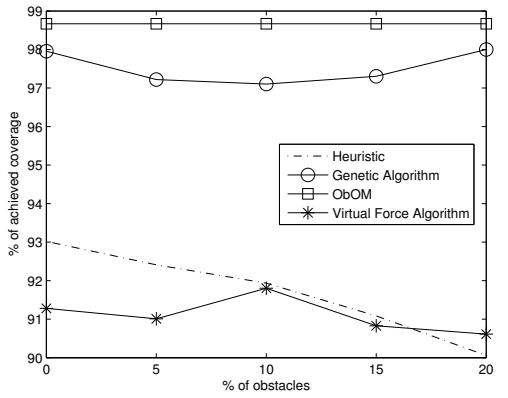
Fig. 4: Percentage of coverage when the grid size is 10×10 (a) 15×15 (b) and 25×25 (c).

It is worth noticing that for a 25×25 sensor field size we reported results of neither the Optimization Model (*NObOM*) nor the Genetic Algorithm (GA). The reason is that when the size of the grid and the number of nodes increase, the number of potential solutions to be evaluated increases too and both approaches are extremely inefficient and time-consuming.

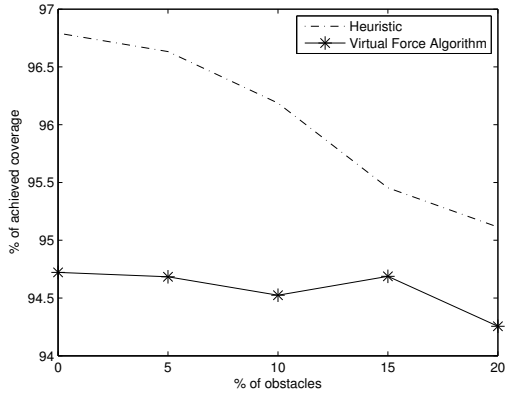
2) Field with obstacles: Fig. 5 plots the coverage achieved by the mathematical model and the three algorithms for the three different grid sizes, when the percentage of obstacles in the field varies. Specifically, we consider a number of obstacles



(a)



(b)



(c)

Fig. 5: Percentage of coverage when the grid size is 10×10 (a) 15×15 (b) and 25×25 (c) and obstacles are considered.

ranging from 0 to 20 % in respect of the total number of cells in the field. In this case the coverage is computed by eliminating the squares occupied by the obstacles.

Fig 5 shows the results of the 4 techniques when a certain percentage of obstacles is present in the grid. The Optimization Model (*ObOM*) and the genetic approach are able to reach the maximum achievable coverage. This is due to the fact that both of them are based on global information and know perfectly the positions already occupied by obstacles. In this case we can observe as the perfromance of the heuristic outperform the Virtual Force Algorithm when we consider the increased size of the network (cases $15x15$ and $25x25$), and more

specifically, the distance in terms of coverage improves when the network size increases.

VI. Conclusion

We proposed a new heuristic technique to opportunistically move sensor devices in order to accomplish different coverage requests. This heuristic is extremely flexible, adaptive and works in a distributed fashion. Based on local information it is able to "capture" the situation in the surrounding and nodes move in a greedy fashion by trying to cover the lesser covered zones. The surrounding information can both represent the information of how much a close point is covered and how much a node needs to be covered. This latter point allows to introduce the concept of "more interesting zone", that is a zone that for many reasons needs to be highly monitored. The heuristic can easily be implemented in simple sensor devices, because does not require much computation resources. In order to evaluate the effectiveness of this simple technique we considered three coverage approaches: an optimization model that gives the optimal solution in terms of coverage, a genetic based approach that works in a centralized way and a virtual forces based technique. We show that our approach allows to reach results close to those of the optimization model and the genetic approach and in certain scenarios, when the size of the sensor field increases, our scheme is able to outperform the virtual based approach. Moreover, we introduced some obstacles in the sensor field in order to understand if our approach is able to adapt the movement of nodes in an effective way when obstacles are present. Results are encouraging also in this specific condition.

Appendix

In the following appendices we present the preliminary study we performed by introducing the concept of Zone of Interest (ZoI). In a realistic field covered by mobile wireless sensors, it could happen that some regions of the field are "more" interesting than others. Hence some regions need to be monitored more precisely. We assume that a higher precision in sensing the area is given by a higher number of nodes that monitor it. Specifically, we show this situation in Fig. 6, where we the *degree of interest* of a specific cell in the grid is represented by the intensity of the filling color. The darker the color, the higher is the interest in monitoring that specific cell, and consequently more nodes are needed in the area.

In order to deal with this new concept, we present an extended version of both the mathematical model presented in Section III and the heuristic presented in Section IV.

Mathematical model for field with ZoI

In some applications, there are the needs to guarantee that some zones (the so-called Zones of Interest) are covered up to a certain degree. In what follows, we present three different formulations to represent the sensor deployment problem with zones of interests.

It is important to observe that both *NObOM* and *ObOM* can be easily extended to handle this specific situation. However, for the sake of simplicity, in what follows, we assume that the field is free of obstacles and thus *NObOM* can be used as basic optimization model.

In the first formulation (referred in the sequel as *ZoI-Min*), it is assumed that for each location (i,j), $i = 1,\ldots,\lceil h/d \rceil$, $j = 1,\ldots,\lceil w/h \rceil$, the minimum number of sensor $cover_{ij}$ that have to cover (i,j) is given. Thus, the aim is to find a sensor placement such that the number of locations covered by at least one sensor is maximized and the coverage constraints of the regions of interest are satisfied.

The related mathematical formulation is obtained by adding to *NObOM* $\sum_{k=1}^{n} \phi_{ijk} \geq cover_{i,j}, \ \ \forall i,j.$

An alternative way to address the problem of cover ZoI is to maximize the number of sensors that cover these zones, by ensuring that each location is covered by at least one sensor. In this second formulation (denoted as *ZoI-Max*), let $\mathcal{ZOI}$ be the set of locations corresponding to the zone of interests, the objective function to be maximized assumes that : $\sum_{k=1}^{n} \ \sum_{(\tilde{i},\tilde{j})\in\mathcal{ZOI}} \phi_{\tilde{i}\tilde{j}k}$ subject to constraints (3)-(15) and the condition reported in: $\sum_{i=1}^{\lceil h/d \rceil} \sum_{j=1}^{\lceil w/d \rceil} \delta_{ij} \geq \gamma$ where γ is set to $\lceil h/d \rceil \times \lceil w/d \rceil$, if each location has to be covered by at least one sensor. It is important to observe that it is possible to carry out a sensitive analysis on the value of the parameter γ in order to find the best trade-off between maximizing the number of sensors to cover $\mathcal{ZOI}$ and ensuring an acceptable coverage for the remaining locations.

ZoI-Max can be easily extended to ensure that each location belonging to $\mathcal{ZOI}$ is covered by exactly the same number of sensors. The corresponding optimization model (denoted in the sequel as *ZoI-Eq*) is obtained by adding to *ZoI-Max*

$$\sum_{k=1}^{n} \phi_{\tilde{i}\tilde{j}k} = \sum_{k=1}^{n} \phi_{\bar{i}\bar{j}k}, \quad \forall(\tilde{i},\tilde{j}) \ \ and \ \ (\bar{i},\bar{j}) \in \mathcal{ZOI}, \ \ (\tilde{i},\tilde{j}) \neq (\bar{i},\bar{j})$$

Heuristic extension for Zone of Interest

The *degree of interest* of a certain region of the sensor field can be defined in different ways, i.e. we can explicitly declare the number of nodes needed to cover a specific point/zone such as in the Optimization Model (*ZoI-Min*) defined above, or we can simply define a certain degree of interest through a weight associated to the zone. This second approach coincides with our basic heuristic (Basic-Heuristic in the following) proposed

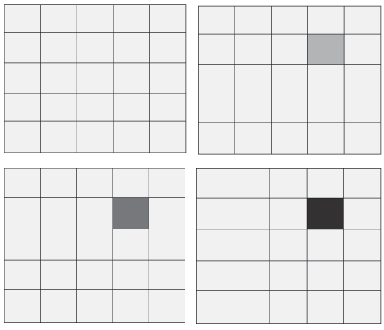


Fig. 6: An interesting zone (point) in the sensor field. The different degrees of gray correspond to different degrees of interest.

in Section IV and by the variants *ZoI-Max* and *ZoI-Eq* of the mathematical model *NObOM*. In practice, we associate a certain "weight" to a specific area. The weight represents the interest to cover the zone without explicitly associating a specific number of sensor. We will refer to the extension of the Basic-Heuristic as Ext-Heuristic (Extended-Heuristic). Specifically, in the Ext-Heuristic we take into account the information about the exterior frame as in the Basi-Heuristic but also the information contained in the interior frame. This extension is necessary when different degrees of interest are taken into account, since an increased necessity to cover a certain zone could not be captured by only considering the exterior frame. Specifically, we apply the same principle of the Basic-Heuristic and we calculate the total coverage degree of a frame. In this case, nodes will move towards the zone with smaller coverage. In practice, let us consider the field as shown in Fig. 2 and focus on the gray node ac inside the blue square, with the Basic-Heuristic we only take into account the external frame and we make the difference between each couple. This difference gives us the axis along which the current node has to move, whereas the direction will be determined by considering the smallest value of the weights. In our extended version, we also take into account the contribution of the weights of the interior frame by applying the same concept of the Basic-Heuristic. The largest difference between the opposite borders will determine the axis and direction of movement.

In Fig. 7 we show the coverage situation with the presence of ZoI placed at location [4,4] and [15,15] of the sensor field. The figure shows the results before running the heuristic (a) and after the heuristic completed its work (b). We can observe that the effect of the heuristic is to produce a more uniform distribution of nodes in the field but also an increased coverage for the specific ZoIs. A more quantitative evaluation will be presented in an extended version of the paper.

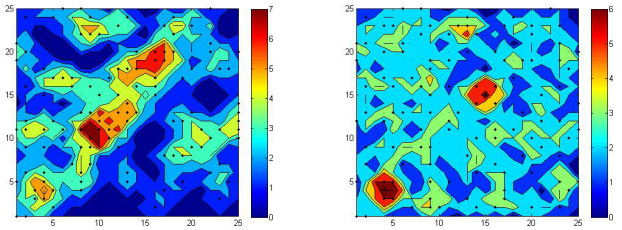


Fig. 7: Coverage before (a) and after (b) the heuristic.

ACKNOWLEDGMENT

This work has been carried out in the framework of the Labex MS2T, which is funded by the French Government, through the program Investments for the future, managed by the National Agency for Research (Reference ANR-11-IDEX-0004-02).

REFERENCES

[1] D. Simplot-Ryl A. Gallais, J. Carle and I. Stojmenovic. Localized sensor area coverage with low communication overhead. *IEEE Transactions on Mobile Computing (TMC)*, 5(7):661–672, 2008.

[2] M. A. Batalin and G. S. Sukhatme. Spreading out: A local approach to multi-robot coverage. In *Proc. of 6th International Symposium on Distributed Autonomous Robotic Systems*, pages 373–382, 2002.

[3] D. Bertsimas and J. N. Tsitsiklis. *Introduction to Linear Optimization*. Athena Scientific, 1997.

[4] T. Bokareva, W. Hu, S. Kanhere, B. Ristic, N. Gordon, T. Bessel, M. Rutten, and S. Jha. Wireless sensor networks for battlefield surveillance. In *Proc. of LWC*, 2006.

[5] K. Chakrabarty, S. S. Iyengar, H. Qi, and E. Cho. Grid coverage for surveillance and target location in distributed sensor networks. *IEEE Transactions on Computers*, 51(12), 2002.

[6] C. Y. Chang, C. T. Chang, Y. C. Chen, and H. R. Chang. Obstacle-resistant deployment algorithms for wireless sensor networks. *IEEE Trans. Vehicular Technology*, 58(6):2925–2941, 2009.

[7] C. Costanzo, V. Loscri, E. Natalizio, and T. Razafindralambo. Nodes self-deployment for coverage maximization in mobile robot networks using an evolving neural network. *Computer Communications*, 2011.

[8] A. Gallais, J. Carle, D. Simplot-Ryl, and I. Stojmenovic. Localized sensor area coverage with low communication overhead. *IEEE Transactions on Mobile Computing*, 5(7):661–672, 2008.

[9] S. Golomb and L. Welch. Perfect codes in the lee metric and the packing of polyominoes. *SIAM J. Appl Math.*, 18:302–317, 1970.

[10] N. Heo and P. K. Varshney. An intelligent deployment and clustering algorithm for a distributed mobile sensor network. In *Proc. SMC*, 2003.

[11] F. Hoffmann, M. Kaufmann, and K. Kriegel. The art gallery theorem for polygons with holes. In *Proc. of IEEE FOCS*, 1991.

[12] A. Howard, M. J. Mataric, and G. S. Sukhatme. Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem. In *Proc. of DARS*, 2002.

[13] W. C. Ke, B. H. Liu, and M. J. Tsai. Constructing a wireless sensor network to fully cover critical grids by deploying minimum sensors on grid points is np-complete. *IEEE Trans. Computers*, 56(5), 2007.

[14] W.C. Ke, B. H. Liu, and M. J. Tsai. The critical-square-grid coverage problem in wireless sensor networks is np-complete. *Computer Networks*, 55(9):2209–2220, 2011.

[15] I. Maleki, S.R. Khaze, M.M. Tabrizi, and A. Bagherinia. A new Approach for Area Coverage Problem in Wireless Sensor Networks with Hybrid Particle Swarm Optimization and Differential Evolution Algorithms . *International Journal of Mobile Network Communications and Telematics (IJMNCT)*, 3(6):61–76, December 2013.

[16] J. B. M. Melissen and P. C. Schuur. Covering a rectangle with six and seven circles. *Discrete Applied Math.*, 99(1):149–156, 2000.

[17] K. J. Nurmela. Conjecturally optimal coverings of an equilateral triangle with up to 36 equal circles. *Experimental Math.*, 9(2):241–250, 2000.

[18] K. J. Nurmela and P. R. J. Ostergard. Covering a square with up to 30 equal circles. Technical report, TECHNISCHE UNIVERSITT HELSINKI, 2000.

[19] J. O'Rourke. *Art Gallery Theorems and Algorithms*. Oxford University Press, 1987.

[20] A. W. Paeth. *Graphics Gems V, Chapter II*. Academic Press, 1995.

[21] J. Polastre, R. Szewczyk, A. Mainwaring, D. Culler, and J. Anderson. *Wireless Sensor Networks, Chapter VI*. Springerlink, 2004.

[22] K. K. Rachuri and C. Murthy. Energy efficient and scalable search in dense wireless sensor networks. *IEEE Trans. Computers*, 58(6), 2009.

[23] P. Ray and P. Varshney. Estimation of spatially distributed processes in wireless sensor networks with random packet loss. *IEEE Trans. Wireless Communications*, 8(6):3162–3171, 2009.

[24] L. Schwiebert, S. K. S. Gupta, and J. Weinmann. Reseach challenges in wireless networks of biomedical sensors. In *Proc. MobiCom 2001*, pages 151–165, 2001.

[25] X. Shen, J. Chen, and Y. Sun. Grid scan: a simple and effective approach for coverage problem in wireless sensor networks. In *IEEE ICC*, 2006.

[26] SH. Tan, X. Hao, Y. Wang, F. C. M. Lau, and Y. Lv. An Approximate Approach for Area Coverage in Wireless Sensor Networks . *Procedia Computer Science*, 19(0):240–247, 2013.

[27] Y. Zou and K. Chakrabarty. A distributed coverage- and connectivity-centric technique for selecting active nodes in wireless sensor networks. *IEEE Trans. Computers*, 54(8):978–991, 2005.