Terrain Partitioning Based Approach for Realistic Deployment of Wireless Sensor Networks

Mostefa Zafer[1](✉), Mustapha Reda Senouci[2](✉), and Mohamed Aissani[2]

[1] Ecole nationale Supérieure d'Informatique,
BP 68M, 16309 Oued-Smar, Alger, Algérie
m_zafer@esi.dz

[2] Ecole Militaire Polytechnique, BP 17, 16046 Bordj El-Bahri, Alger, Algérie
mrsenouci@gmail.com, maissani@gmail.com

Abstract. This paper addresses the NP-hard problem of deploying wireless sensor networks on 3D terrains. On the contrary to previous works that place the sensors without any analysis of the terrain, we propose a two-phase solution based on terrain partitioning. The main idea is to estimate the number of sensors to be used and simplify the sensors deployment by partitioning the terrain according to topographic criteria. Simulation results based on real-world terrains confirm the efficiency of our solution in terms of coverage quality.

Keywords: Wireless Sensor Networks · 3D terrain · Deployment

1 Introduction

A Wireless Sensor Network (WSN) is a special type of ad hoc networks, consisting of a set of sensors deployed in a region of interest (RoI) to monitor a target event. Each sensor is battery-powered and has a sensing unit, a processing unit, and a wireless communication interface. The primary task of a sensor is to take measurements related to the target event and to communicate them wirelessly to a collection point (sink), using a one-hop or a multi-hop routing protocol [1].

For a WSN to be fully functional, at least two conditions must be met: coverage and connectivity [1]. Coverage suggests, in its simplest form, that each point of the RoI is covered by at least one sensor. This gives the network the ability to detect, at any point, the target event occurrence. Connectivity implies the existence, for each sensor, of at least one path connecting it to the sink.

To satisfy the two aforementioned conditions, it is necessary that the sensors must be well distributed in the RoI. The usually used solution consists in deploying the sensors in a random but very dense manner [1]. This practice is recommended in some cases, especially when the RoI is inaccessible. Nevertheless, for a three-dimensional RoI, characterized by a complex topography (e.g. mountainous areas), this random deployment produces many coverage holes, which

A. Amine et al. (Eds.): CIIA 2018, IFIP AICT 522, pp. 423–435, 2018.
https://doi.org/10.1007/978-3-319-89743-1_37

are generally hidden parts of the RoI with very low sensor density; and therefore a very low coverage quality, which gravely impairs the WSN performances.

The limitations of the random deployment of 3D WSNs, notably in terms of coverage, have been highlighted in several works [2–4] that recommended the use of a deterministic sensor placement. In this latter, the sensors' positions are precomputed, while taking into consideration not only the factors related to the sensors characteristics but also the topography of the RoI. This requires first a reliable representation of the RoI and a rigorous modeling of its influence on the network performances.

The problem of WSNs deployment on a 3D terrains, proved NP-hard [5,6], has been the subject of several researches [2–15] that have proposed, under different hypotheses and formulations, approximates solutions based on heuristics or meta-heuristics, aiming to find the best nodes positions, that meet the network objectives. However, these works treat the entire RoI in the same way, without taking into account the differences in topographic complexity between its different sub-regions. To contribute in removing this weakness and then improve coverage and connectivity, we propose a new deployment approach, in which the novelty is to partition the RoI into sub-regions characterized by simple topographies, to guide the deployment of the network composed of sensors and relays. In this work, we tackle the part related to the sensors deployment in order to maximize coverage. For the relays deployment, which are used to overcome the connectivity problem, we just give an indication that will be detailed and evaluated in a future work.

The rest of this paper is organized as follows. The related works and their limits are summarized in Sect. 2. The proposed approach is described in Sect. 3. The coverage quality produced by the proposed approach is evaluated by simulation in Sect. 4. The conclusion of this work is given in Sect. 5.

2 Related Works

The main difference between the existing approaches for the problem at hand lies in the terrain representation. As such, most approaches [2–4,7–15] adopt the matrix model, in which the terrain is represented by a matrix, where the value of each cell corresponds to the elevation of the place that it represents on the terrain. Another digital model, used by other approaches [5,6,16–18], consists in representing the terrain by triangles network form. This model, called TIN (Triangulated Irregular Network), can be deduced from the matrix model by using triangulation methods. Few simplistic approaches [19–21] adopt a mathematical model, where the coordinates of each point are calculated directly according to a mathematical formula.

The aforementioned approaches consider different assumptions and models to represent the sensor capabilities. The sensors can be considered homogeneous or heterogeneous, directional or omnidirectional, mobile or static. Their coverage capacities can be represented according to several parameters, with binary or probabilistic models. The parameters considered include, besides sensors-related ones, those related to the RoI, in particular its topography that is introduced, by the most of these approaches, via the visibility condition, which is verified in different ways, depending on the adopted terrain model [22]. Also, these approaches address various objectives such as connectivity [11,19,20], sensors stealth [16,17], and coverage that is the primary objective for all of these approaches.

In the literature, different heuristics and meta-heuristics are employed to find the best positions (and orientations) of the sensors to meet the planned objectives. In the sequel, these approaches are classified according to the adopted terrain model, and described in terms of assumptions and resolution schemes.

2.1 Deployment Approaches Based on a Mathematical Terrain Model

All the algorithms of this class are based on the binary coverage model. To improve the quality of the random deployment in terms of coverage and connectivity, two sensors redistribution algorithms have been proposed in [19,20], where the terrain is modeled as a cone, and sensors are assumed to be homogeneous and omnidirectional. The algorithm proposed in [19] is based on the virtual forces technique, under the constraint that the sensors move only on the surface of the terrain. In [20], the proposed solution consists in moving the sensors to positions situated on a spiral curve surrounding the terrain.

Assuming that the sensors are directional, the SA (Simulated Annealing) algorithm is used in [21] to relocate and reorient the sensors, initially deployed in a uniform manner, with the aim of improving the coverage.

2.2 Deployment Approaches Based on a Matrix Terrain Model

Some deployment solutions of this class adopt the binary coverage model, assuming that the sensors are omnidirectional [4,7,9] or directional [8]. To maximize coverage, the SA is implemented in [7], in both centralized and distributed manners. For the same goal, the S-GA (Simple Genetic Algorithm) and the CMA-ES (Covariance Matrix Adaptation-Evolution Strategy) algorithms are used in [4] and [8], respectively. In [9], the Voronoï diagram reinforced by the use of mobile nodes is made more realistic by taking into account the presence of obstacles. Other solutions adopt the probabilistic coverage model and the sensors are supposed to be omnidirectional [2,3,10–12] or directional [13–15].

To maximize coverage, the S-GA and SS-GA (Steady State Genetic Algorithm) algorithms are used in [2] where they were reinforced by the 2D-DWT (Discrete Wavelet Transform) transformation used to locate the coverage holes. For the same goal, this same transformation is adopted to reinforce the CSO (Cat Swarm Optimization) and ABC (Artificial Bee Colony) algorithms, used in [3] and [10], respectively.

In [11], in addition to coverage, connectivity is taken into account by using the link evaluation model adopted in [23]. To maximize these two criteria, the S-GA algorithm, reinforced by the 2D-DWT transformation, is used. The authors in [12] proposed a greedy algorithm to achieve k-coverage. Also, the work done in [8] is revised in [13–15] by adopting a probabilistic coverage model. In [13,14], SA and CMA-ES are the main methods used to maximize coverage. In [15], the PSO (Particle Swarm Optimization) method, consolidated by the virtual force technique, is used to maximize coverage of critical points on the terrain.

2.3 Deployment Approaches Based on a TIN Terrain Model

Assuming that the sensors are homogeneous and omnidirectional and by adopting a binary coverage model, a greedy algorithm is proposed in [5,6] based on the partitioning of the terrain in a regular grid, whose cells are smaller than the sensors coverage range. This algorithm consists in choosing, in a progressive (iterative) way, the cells that should host sensors in order to maximize coverage. With the same assumptions and by adopting a probabilistic coverage model, another solution is proposed in [18]. This latter is based on the Centroidal Voronoï Tessellation adapted to 3D terrain and guided by Ricci one-to-one mapping, with the aim to ensure coverage.

Table 1. A comparison between various deployment approaches.

Ref	Objectives	Resolution approach	RoI model	Deployment parameters		
				Sensor type	Visibility	Coverage model
[19]	Coverage + Connectivity	Virtual forces	Mathematical	Omnidirectional	✗	Binary
[20]	Coverage + Connectivity	Computational geometry	Mathematical	Omnidirectional	✗	Binary
[21]	Coverage	SA	Mathematical	Directional	✗	Binary
[7]	Coverage	SA	Matrix	Omnidirectional	✓	Binary
[9]	Coverage	Voronoï diagram	Matrix	Omnidirectional	✗	Binary
[4]	Coverage	S-GA	Matrix	Omnidirectional	✓	Binary
[8]	Coverage	CMA-ES	Matrix	Directional	✓	Binary
[2]	Coverage	S-GA + SS-GA	Matrix	Omnidirectional	✓	Probabilistic
[3]	Coverage	CSO	Matrix	Omnidirectional	✓	Probabilistic
[10]	Coverage	ABC	Matrix	Omnidirectional	✓	Probabilistic
[11]	Coverage + Connectivity	S-GA	Matrix	Omnidirectional	✓	Probabilistic
[12]	k-Coverage	Greedy Algorithm	Matrix	Omnidirectional	✗	Probabilistic
[13]	Coverage	SA + CMA-ES	Matrix	Directional	✓	Probabilistic
[15]	Coverage	PSO	Matrix	Directional	✓	Probabilistic
[14]	Coverage	SA + CMA-ES	Matrix	Directional	✓	Probabilistic
[18]	Coverage	Centroidal Voronoï Tessellation	TIN	Omnidirectional	✗	Probabilistic
[5,6]	Coverage	Greedy Algorithm	TIN	Omnidirectional	✗	Binary
[16,17]	Coverage + Sensors stealth	SS-GA	TIN	Directional	✓	Probabilistic

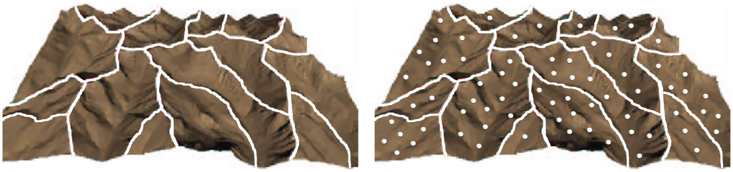
By adopting a probabilistic coverage model and assuming that the sensors are directional, the authors in [16,17] used the SS-GA algorithm to find the minimum number of sensors, their positions and their orientations, to maximize coverage while ensuring sensors stealth. Table 1 provides a comparative summary of the aforementioned deployment approaches.

It is worth to mention that most of the above-discussed approaches focused on the coverage of the RoI without considering other deployment-related aspects, such as connectivity, which is as important as coverage. In addition, some of these approaches are based on a very simplified and unrealistic terrain model, where its effective influence on coverage and connectivity has not been well taken into account. This observation concerns in particular works based on a mathematical terrain model. Also, the complexity of the RoI is an important factors to be taken into consideration to estimate the number of sensors to be deployed. This issue has not been addressed in these approaches. In addition, the performance results presented in these works show that the most effective approaches are those that use digital models to represent the terrain and exploit meta-heuristics to find better positions of the sensors. However, their weakness lies in the fact that they treat the entire RoI in the same way, without taking into account the differences in topographic complexity between the different sub-regions of the RoI.

3 The Proposed Approach

In order to face the topography challenges and simplify the deployment of the network, consisting of sensors and relays, our key idea is to partition the RoI according to the visibility criterion. Our approach is composed of three phases described as follows. The first phase consists of partitioning the RoI into relatively simple topography sub-regions by using a simple heuristic designed for this purpose and based on the visibility analysis. Thus, in each built sub-region, the visibility factor is less pronounced and its influence on detection and communication operations is greatly degraded. The second phase is intended for the deployment of the sensors on each sub-region, with the aim of maximizing its coverage, by using a method designed for this purpose. The first and second phases of our deployment approach are depicted on Fig. 1.

The third and final phase is intended to consolidate the network connectivity by using relays that will be placed at positions with wide visibility, located on the crest lines separating the constructed sub-regions. Our goal is to build a 2-tier architecture for the network, whose advantage lies in the fact that the sensors do not participate in data routing, which increases their lifetimes. Visibility, distance and load balancing are the most important factors that will be taken into consideration when building this architecture. We recall that this paper focuses only on the first and the second phases. The third phase is the subject of an upcoming work.



(a) Phase 1

(b) Phase 2

Fig. 1. First and second phases of our deployment approach.

3.1 RoI Partitioning

In this section, we describe our approach for partitioning the RoI into sub-regions with relatively simple topographies, which should facilitates the deployment of sensors on each of these sub-regions. We assume that the RoI, denoted $\mathcal{T}$, is represented by a TIN model that is composed of n triangles $t_i (1 \leq i \leq n)$, thus $\mathcal{T} = \{t_i, 1 \leq i \leq n\}$. To construct a partition $\mathcal{P}$ of $\mathcal{T}$, we apply an iterative fusion of triangles based on visibility analysis. The number of sub-regions constructed, denoted $\|\mathcal{P}\|$, is not fixed beforehand, and it is strongly depended to the RoI complexity. Each constructed sub-region, denoted $\mathcal{R}_k (1 \leq k \leq \|\mathcal{P}\|)$, is composed of one or more triangles and each triangle belongs only to one sub-region. To explain this partitioning heuristic, we define the following notions.

Definition 1 (Intervisibility triangle-triangle). *Two triangles t_i and t_j are considered intervisible if and only if they are adjacent; that is to say that they share a common edge, and their non-common vertices are intervisible. Therefore, any point $p \in t_i$ is visible to any other point $q \in t_j$.*

If the visibility between two points p and q is verified by the binary function $v(p, q)$ and the adjacency between two triangles t_i and t_j is verified by the binary function $adj(t_i, t_j)$, then the visibility between t_i and t_j is verified by the binary function $\mathcal{V}_\triangle(t_i, t_j)$ where $\mathcal{V}_\triangle(t_i, t_j) \iff adj(t_i, t_j) \wedge (\forall p \in t_i, \forall q \in t_j : v(p, q))$.

The visibility between two triangles (see Fig. 2) means that the coverage of one of the triangles could be ensured by the sensors deployed on the other, hence the opportunity to consider both as a single entity when deploying the sensors.

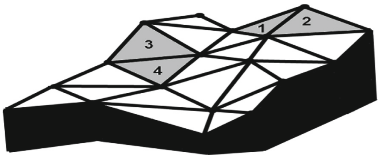


Fig. 2. Intervisibility triangle-triangle ($\mathcal{V}_\triangle(t_1, t_2) = 0, \mathcal{V}_\triangle(t_3, t_4) = 1$).

Definition 2 (Intervisibility triangle-sub-region). *A triangle t_i is considered visible to a sub-region $\mathcal{R}_k$ if and only if it does not belong to $\mathcal{R}_k$ and it is visible to a triangle $t_j \in \mathcal{R}_k$. This relation is modeled by the binary function $\mathcal{V}_{\mathcal{R}}(t_i, \mathcal{R}_k)$ where $\mathcal{V}_{\mathcal{R}}(t_i, \mathcal{R}_k) \iff (t_i \notin \mathcal{R}_k) \wedge (\exists t_j \in \mathcal{R}_k : \mathcal{V}_{\triangle}(t_i, t_j))$.*

Similarly, the visibility between a triangle t_i and a sub-region $\mathcal{R}_k$ means that the coverage of t_i can be ensured by the sensors deployed in $\mathcal{R}_k$, hence the opportunity to consider them as a single entity when deploying the sensors.

Definition 3 (Free triangle). *A triangle t_i is considered free if and only if it is not assigned to any sub-region.*

Definition 4 (Front of a sub-region). *The front of a sub-region $\mathcal{R}_k$ is the set of free triangles visible to $\mathcal{R}_k$. If the set of free triangles is represented by Ω, then the front of $\mathcal{R}_k$ is $\mathcal{F}(\mathcal{R}_k) = \{t_i \in \Omega / \mathcal{V}_{\mathcal{R}}(t_i, \mathcal{R}_k) = 1\}$.*

Description. Initially, the set of free triangles Ω is equal to the set of triangles composing the RoI $\mathcal{T}$ (all the triangles of $\mathcal{T}$ are free). We assign to the first sub-region to be constructed, denoted $\mathcal{R}_1$, a triangle t arbitrarily selected from Ω ($t \longleftarrow rand_{\triangle}(\Omega)$). The sub-region $\mathcal{R}_1$ will be extended in an iterative manner, adding to it, at each iteration, the set $\mathcal{F}(\mathcal{R}_1)$. When there is no free triangle to add to $\mathcal{R}_1$; that is to say that $\mathcal{F}(\mathcal{R}_1) = \emptyset$, the construction of $\mathcal{R}_1$ is completed, and the construction of a new sub-region in the same way is launched. The sub-region construction algorithm stops when the set of free triangles becomes empty ($\Omega = \emptyset$). Obviously, the number of sub-regions produced by this heuristic depends on the complexity of the topography. In the ideal case, we will have a single sub-region including all the n triangles of $\mathcal{T}$. In extremely complex cases, we will have a number of sub-regions equal to the number of triangles of $\mathcal{T}$. It is easy to confirm that the temporal complexity of this heuristic is $\mathcal{O}(n^2)$, but it can be reduced to $\mathcal{O}(n \times log(n))$. Indeed, adding a triangle to a sub-region requires confirming that it is still free, by checking its presence in Ω, which is done in a sequential way in our basic heuristic, and can be improved by adopting a dichotomy search, under the condition that the triangles in Ω are sorted (by their numbers for example). The partitioning produced by our heuristic has two interesting characteristics: (i) two triangles lying in two different sub-regions are necessarily not inter-visible; and (ii) for every point in a sub-region, there are necessarily other points in that sub-region, which are visible to it.

3.2 Sensors Deployment

This section describes the sensors deployment phase, which includes the formulation of the desired objective according to the considered parameters and the adopted assumptions, an estimate of the number of sensors to be used, as well as the design of a resolution method, allowing to select the appropriate positions of these sensors.

Algorithm 1. RoI partitioning

```
Input: $\mathcal{T} = \{t_i / 1 \leq i \leq n\}$
Output: $\mathcal{P}$
$\mathcal{P} \longleftarrow \emptyset$; $\Omega \longleftarrow \mathcal{T}$; $i \longleftarrow 0$;
while $(\Omega \neq \emptyset)$ do
    $i \longleftarrow i+1$;
    $t \longleftarrow rand_{\triangle}(\Omega)$; /* random selection of a free triangle */
    $\mathcal{R}_i \longleftarrow \{t\}$; $\Omega \longleftarrow \Omega - \{t\}$;
    while $(\mathcal{F}(\mathcal{R}_i) \neq \emptyset)$ do
        $\mathcal{R}_i \longleftarrow \mathcal{R}_i \cup \mathcal{F}(\mathcal{R}_i)$; $\Omega \longleftarrow \Omega - \mathcal{F}(\mathcal{R}_i)$;
    end
    $\mathcal{P} \longleftarrow \mathcal{P} \cup \mathcal{R}_i$;
end
```

Hypotheses and Objective. We aim to achieve the minimum coverage (1-coverage) level of the RoI $\mathcal{T}$, wherever every point must be covered by at least one sensor. For simplicity, we assume that the sensors are homogeneous and omnidirectional. Also, we adopt the binary coverage model, where the coverage capacity of the sensors is influenced by the distance and the terrain topography, which is introduced via the visibility factor. Thus, a point p_i is considered to be covered by a sensor s_j if and only if: (1) p_i and s_j are intervisible and (2) p_i is within the coverage range of s_j. This last condition is examined by the binary function $\mu_d(p_i, s_j)$, defined by Eq. 1, where r_s represent the coverage range of the sensors and $d(p_i, s_j)$ the Euclidean distance between p_i and s_j.

$$\mu_d(p_i, s_j) = \begin{cases} 1 & \text{if } d(p_i, s_j) \leq r_s \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Similarly, the visibility between p_i and s_j is modeled by a binary function $\mu_v(p_i, s_j)$, calculated by the Bresenham's algorithm [22]. Thus, the coverage of p_i by s_j is modeled by the binary function $\mathcal{C}(p_i, s_j) = \mu_d(p_i, s_j) \times \mu_v(p_i, s_j)$.

In the sequel, we describe our approach for the deployment of sensors on an arbitrary sub-region $\mathcal{R}_k \in \mathcal{P}$, constructed by the above-described partitioning phase. The same deployment approach is applied for the other sub-regions.

Deployment of Sensors on a Sub-region $\mathcal{R}_k$. We assume that the sensors deployed on $\mathcal{R}_k$ only contribute to the coverage of $\mathcal{R}_k$. In the presence of $\mathcal{N}_k$ sensors in $\mathcal{R}_k$, the coverage of a point $p_i \in \mathcal{R}_k$, is modeled by the function $\mathcal{C}ov(p_i, \mathcal{N}_k) = \max_{1 \leq j \leq \mathcal{N}_k} \mathcal{C}(p_i, s_j)$.

We consider by approximation that the coverage $\mathcal{C}ov_{\triangle}(t_i, \mathcal{N}_k)$ of a triangle $t_i \in \mathcal{R}_k$ is the mean of the coverage value of its vertices p_1^i, p_2^i and p_3^i (Eq. 2). The objective is therefore to find the number $\mathcal{N}_k$ and the positions of the sensors to be deployed in the sub-region $\mathcal{R}_k$, to maximize its coverage quality $\mathcal{C}ov_{\mathcal{R}}(\mathcal{R}_k, \mathcal{N}_k)$ calculated according to Eq. 3, where $\|\mathcal{R}_k\|$ represents the number of triangles composing the sub-region $\mathcal{R}_k$.

$$\mathcal{C}ov_{\triangle}(t_i, \mathcal{N}_k) \approx \frac{1}{3} \times \sum_{j=1}^{3} \mathcal{C}ov(p_j^i, \mathcal{N}_k) \tag{2}$$

$$\mathcal{C}ov_{\mathcal{R}}(\mathcal{R}_k, \mathcal{N}_k) = \frac{1}{\|\mathcal{R}_k\|} \times \sum_{t_i \in \mathcal{R}_k} \mathcal{C}ov_{\triangle}(t_i, \mathcal{N}_k) \tag{3}$$

Estimation of the number of sensors. The number of sensors ($\mathcal{N}_k$) to be deployed on the sub-region $\mathcal{R}_k$ to ensure its coverage depends on the coverage range r_s of the sensors, the surface area of $\mathcal{R}_k$, and the arrangement of the triangles of $\mathcal{R}_k$. For simplicity, we take into account only the first and the second factors. Therefore, we compute $\mathcal{N}_k$ by dividing the surface of $\mathcal{R}_k$, which is equal to the sum of the surfaces of all its triangles, on the surface covered by a sensor in an ideal (flat) terrain. The number $\mathcal{N}_k$ is given by Eq. 4, where $\|t_i\|$ represents the surface of the triangle t_i and $\mathcal{I}$ denotes the integer part function.

$$\mathcal{N}_k = \mathcal{I}\Big(\frac{\sum_{t_i \in \mathcal{R}_k} \|t_i\|}{\pi \times r_s^2}\Big) + 1 \tag{4}$$

Algorithm 2. Sensors deployment on sub-region $\mathcal{R}_k$

```
Input: $X_r$
Output: $X$
$X \longleftarrow X_r$; $i \longleftarrow 0$;
while $i < i_s$ do
    $j \longleftarrow rand_{\mathbb{N}}\{1, ..., \mathcal{N}_k\}$; /* random selection of a sensor $s_j$ */
    $X_i \longleftarrow X_r + rand_{s_j}(\mathcal{R}_k)$ /* generation of $X_i$ from $X_r$ by randomly
        changing the position of $s_j$ in $\mathcal{R}_k$ */
    if $\mathcal{C}ov_{\mathcal{R}}(X_i) > \mathcal{C}ov_{\mathcal{R}}(X_r)$ then
        $X_r \longleftarrow X_i$;
        if $\mathcal{C}ov_{\mathcal{R}}(X_i) > \mathcal{C}ov_{\mathcal{R}}(X)$ then
            $X \longleftarrow X_i$;
        end
    else
        $\zeta \longleftarrow rand_{\mathbb{R}}[0, 1]$; /* generating a random number */
        if $\zeta < \mathfrak{T}(i)$ then
            $X_r \longleftarrow X_i$;
        end
    end
    $i \longleftarrow i+1$;
end
```

Deployment of sensors. In order to maximize the coverage of $\mathcal{R}_k$ by the $\mathcal{N}_k$ sensors, we use the simulated annealing to determine the positions to be occupied by the sensors. Thus, a possible solution to our problem is coded in the form of a vector containing the positions of the $\mathcal{N}_k$ sensors. The method is based on the follow-up of the evolution of a current solution, that we note it X_r. Let $\mathcal{C}ov_{\mathcal{R}}(X_r)$ be the coverage quality $\mathcal{C}ov_{\mathcal{R}}(\mathcal{R}_k, \mathcal{N}_k)$ generated by X_r. Initially, X_r is generated in an arbitrary manner, where the sensors occupy random positions in $\mathcal{R}_k$. Thereafter, X_r evolves in several iterations. At an iteration i, a new solution X_i is generated from X_r, by randomly changing the position of an arbitrary sensor in X_r ($X_i \longleftarrow X_r + rand_{s_j}(\mathcal{R}_k)$), under the constraint that its new position is always in $\mathcal{R}_k$. Let $\mathcal{C}ov_{\mathcal{R}}(X_i)$ be the coverage quality $\mathcal{C}ov_{\mathcal{R}}(\mathcal{R}_k, \mathcal{N}_k)$ produced by X_i. The solution X_r is updated to X_i ($X_r \leftarrow X_i$) if X_i improves the coverage quality; that is to say that $\mathcal{C}ov_{\mathcal{R}}(X_i) > \mathcal{C}ov_{\mathcal{R}}(X_r)$. In the contrary case, this update ($X_r \leftarrow X_i$) is carried out with a probability $\mathfrak{T}(i)$ calculated according to Eq. 5. This probability is impaired as a function of the number of iterations performed, which is limited to a threshold value i_s. The best value taken by X_r during all its evolution on the i_s iterations is retained as a solution to the problem.

$$\mathfrak{T}(i) = 2^{-(\frac{2i}{i_s}+1)} \tag{5}$$

The total number $\mathcal{N}$ of sensors to be used for the terrain $\mathcal{T}$, and its coverage quality, are computed using Eqs. 6 and 7, respectively.

$$\mathcal{N} = \sum_{k=1}^{\|\mathcal{P}\|} \mathcal{N}_k \tag{6}$$

$$\mathcal{C}ov_{\mathcal{T}}(\mathcal{T}, \mathcal{N}) = \frac{1}{\|\mathcal{P}\|} \times \sum_{k=1}^{\|\mathcal{P}\|} \mathcal{C}ov_{\mathcal{R}}(\mathcal{R}_k, \mathcal{N}_k) \tag{7}$$

It should be noted that the sensor deployment phase can be easily upgraded, by eliminating the assumption that the sensors are omnidirectional and replacing the binary coverage model with a probabilistic model, which is more realistic. Whatever the choices adopted concerning the coverage model, the main contribution proposed in this paper, which consists in preceding the phase of deployment of the sensors by a phase of partitioning the terrain according to topographic criteria, remains intact.

4 Performance Evaluation

To evaluate the coverage quality produced by our approach, we have chosen four real RoIs of size 1200 m × 1200 m with highly distinguishable and progressive complexities (RoI1 < RoI2 < RoI3 < RoI4). Each of these RoI is represented by a TIN model, built by the application of a Delaunay triangulation on altimetry data of these terrains, retrieved via "http://www.zonums.com/gmaps/terrain.php?action=sample". We set the coverage range of each sensor (r_s) to 30 m, and

Table 2. Simulation results.

RoI	# Sensors	Coverage quality		
		Without partitioning	With partitioning	Gain
1	723	0.7460	0.8537	+0.1077
2	779	0.7187	0.8263	+0.1076
3	897	0.6832	0.8038	+0.1206
4	1081	0.6325	0.7923	+0.1598

we apply Eq. 6 to determine the number of sensors to use for each terrain, where the results obtained are shown on Table 2.

For the simulated annealing used for sensor deployment, we set the maximum iteration count (i_s) to 100. We note that the terrain partitioning heuristic and the simulated annealing are implemented on Matlab. The main objective is to analyze the impact of the partitioning phase on the efficiency of the simulated annealing in terms of coverage quality. Thus, for each RoI, the deployment of the sensors by the simulated annealing is carried out according to two scenarios. In the first scenario, all the $\mathcal{N}$ sensors are deployed on the entire RoI, without partitioning it. In the second scenario, the RoI is partitioned using the partitioning heuristic, after which the $\mathcal{N}_k$ sensors allocated to each sub-region $\mathcal{R}_k$ are deployed on it. The coverage quality produced by the simulated annealing implemented for the four RoIs according to these two scenarios (with and without partitioning) is summarized in Table 2.

The obtained results indicate the following. (1) *the number of sensors increases according to the complexity of the RoI.* This is justified by the fact that this complexity reflects the number of peaks. Each peak contributes to the increase of the triangles surfaces of the TIN model, and consequently to the increase of the surface of RoI, used as a factor to calculate the number of sensors; (2) *the coverage quality produced by the SA, in both scenarios (with and without partitioning), degrades according to the RoI complexity, despite the increase in the number of sensors.* This is justified by the fact that this complexity decreases in some way the factor of inter-visibility (notably for the third and the fourth RoI), which obviously has an influence on the coverage quality produced by SA, especially since it was launched with the same number of iterations (set at 100) for the four RoI; (3) *the quality of coverage is improved by the partitioning phase, and this improvement becomes more significant for the third and fourth RoI.* This is justified by the fact that this partitioning has already been based on the visibility analysis, in such a way that the increase in RoI complexity implies first and foremost the increase in the number of sub-regions constructed, which has allowed to minimize the influence of the terrain complexity on the visibility criterion in each constructed sub-region. Also, this partitioning allowed to subdivide the global problem, which consists of deploying the total number of sensors over the entire RoI, to very small sub-problems, where each one consists of deploying on each constructed sub-region, only the number of sensors allocated to it.

5 Conclusion

Examining previously proposed solutions for the deployment of WSNs on realistic terrains allowed us to remark that these solutions are "blind", because they proceed to the deployment of the sensors without analyzing the terrain and without taking into account the topographical differences that may exist between its different sub-regions. From this observation, in this paper, we have proposed a new deployment approach, which consists of partitioning the terrain into simple topography sub-regions, to simplify the estimation of the number of sensors to be used and guide their deployment. The obtained simulation results confirm the efficiency of our approach in terms of coverage quality. As a future work, we aim to design a relay placement algorithm to ensure the connectivity between the constructed sub-regions, and a logical topology management algorithm in order to maximize the network lifetime.

References

1. Senouci, M.R., Mellouk, A.: Deploying Wireless Sensor Networks: Theory and Practice. Elsevier, Amsterdam (2016)
2. Unaldi, N., Temel, S., Asari, V.K.: Method for optimal sensor deployment on 3D terrains utilizing a steady state genetic algorithm with a guided walk mutation operator based on the wavelet transform. Sens. J. **12**, 5116–5133 (2012)
3. Temel, S., Unaldi, N., Kaynak, O.: On deployment of wireless sensors on 3D terrains to maximize sensing coverage by utilizing cat swarm optimization with wavelet transform. IEEE Trans. Syst. Man Cybern. Syst. **44**, 111–120 (2014)
4. Ko, A.H.-R., Gagnon, F.: Process of 3D wireless decentralized sensor deployment using parsing crossover scheme. EACI **11**, 89–101 (2015)
5. Zhao, M.-C., Lei, J., Wu, M.-Y., Liu, Y., Shu, W.: Surface coverage in wireless sensor networks. In: IEEE INFOCOM, pp. 109–117 (2009)
6. Kong, L., Zhao, M.-C., Liu, X.-Y., Lu, J., Liu, Y., Wu, M.-Y., Shu, W.: Surface coverage in sensor networks. IEEE TPDS **25**, 234–243 (2014)
7. Veenstra, K., Obraczka, K.: Guiding sensor-node deployment over 2.5D terrain. In: IEEE ICC, London, pp. 6719–6725, June 2015
8. Akbarzadeh, V., Ko, A.H.-R., Gagné, C., Parizeau, M.: Topography-Aware Sensor Deployment Optimization with CMA-ES. In: Schaefer, R., Cotta, C., Kołodziej, J., Rudolph, G. (eds.) PPSN 2010. LNCS, vol. 6239, pp. 141–150. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15871-1_15
9. Doodmana, S., Afghantoloee, A., Mostafavi, M.A., Karimipour, F.: 3D extention of the VOR algorithm to determine and optimize the coverage of geosensor networks. In: ISPRS, Tehran, Iran, pp. 103–108, November 2014
10. Hang, Y., Xunbo, L., Zhenlin, W., Wenjie, Y., Bo, H.: A novel sensor deployment method based on image processing and wavelet transform to optimize the surface coverage in WSNs. Chin. J. Electron. **25**, 495–502 (2016)
11. Unaldi, N., Temel, S.: Wireless sensor deployment method on 3D environments to maximize quality of coverage and quality of network connectivity. In: Proceedings of WCECS, San Francisco, USA, October 2014
12. Song, T., Gong, C., Liu, C.: A practical coverage algorithm for wireless sensor networks in real terrain surface. Int. J. Wirel. Mob. Comput. **5**, 671–680 (2012)

13. Akbarzadeh, V., Gagné, C., Parizeau, M., Argany, M., Mostafavi, M.A.: Probabilistic sensing model for sensor placement optimization based on line-of-sight coverage. IEEE ToIM **62**, 293–303 (2013)
14. Akbarzadeh, V., Gagné, C., Parizeau, M., Mostafavi, M.A.: Black-box optimization of sensor placement with elevation maps and probabilistic sensing models. In: IEEE International Symposium, pp. 89–94 (2011)
15. Tam, N.T., Thanh, H.D., Son, L.H., Le, V.T.: Optimization for the sensor placement problem in 3D environments. In: 12th International Conference on Networking, Sensing and Control, Taipei, Taiwan, pp. 327–333 (2015)
16. Topcuoglu, H.R., Ermis, M., Sifyan, M.: Positioning and utilizing sensors on a 3D terrain Part I: theory and modeling. IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. **41**, 376–382 (2011)
17. Topcuoglu, H.R., Ermis, M., Sifyan, M.: Positioning and utilizing sensors on a 3D terrain Part II: solving with a hybrid evolutionary algorithm. IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. **41**, 470–480 (2011)
18. Jin, M., Rong, G., Wu, H., Shuai, L., Guo, X.: Optimal surface deployment problem in wireless sensor networks. In: IEEE INFOCOM, pp. 2345–2353 (2012)
19. Boufares, N., Khoufi, I., Minet, P., Saidane, L.: 3D surface covering with virtual forces. In: PEMWN, Hammamet, Tunisia, pp. 103–108, November 2015
20. Kim, K.: Mountainous terrain coverage in mobile sensor networks. IET Commun. **9**(5), 613–620 (2015)
21. Xiao, F., Yang, X., Yang, M., Sun, L., Wang, R., Yang, P.: Surface coverage algorithm in directional sensor networks for three-dimensional complex terrains. Tsinghua Sci. Technol. **21**, 397–406 (2016)
22. Proctor, M.D., Gerber, W.: Line-of-sight attributes for a generalized application program interface. Def. Model. Simul. Appl. Method. Technol. **1**(1), 43–57 (2004)
23. De Marco, G.: MOGAMESH: a multi-objective algorithm for node placement in wireless mesh networks based on genetic algorithms. In: 6th International Symposium on Wireless Communication Systems, pp. 388–392, September 2009