Artificial potential field approach in WSN deployment: Cost, QoM, connectivity, and lifetime constraints

Nadjib Aitsaadi, Nadjib Achir, Khaled Boussetta, Guy Pujolle

Abstract

In this paper, we address a wireless sensor network deployment problem. It is considered when the deployment field is characterized by a geographical irregularity of the monitored event. Each point in the deployment area requires a specific minimum guarantee of event detection probability. Our objective is to generate the best network topology while minimizing cost of deployment, ensuring quality of monitoring and network connectivity, and optimizing network lifetime. The problem is formulated as combinatorial optimization problem, which is NP-complete. Unfortunately, due to the large solution state space and the exponential computational complexity, the exact methods can be applied only in the case of small-scale problem. To overcome the complexity of an optimal resolution, we propose new scalable deployment heuristics based on artificial potential field and Tabu search metaheuristic, namely potential field deployment algorithm (PFDA) and multi-objective deployment algorithm (MODA). We compare our proposal to the related deployment strategies, the obtained results show that PFDA and MODA obtain the best performances.

1. Introduction

The monitoring and the control of physical environment is recently becoming a hot spot in the technology landscape. Consequently, a large number of companies are proposing new small devices with different phenomena’s monitoring capabilities such as temperature, humidity, vibration, pressure, and several other factors. These devices, called sensors, can be deployed to build a wireless sensor network (WSN). In this case, in addition to the phenomenon detection capability, the sensors should be able to communicate between each other in order to propagate the event notifications. Basically, each sensor sends, when an event is detected, a message to one specific node working as a sink (base station) which is in charge to inform the user-application that an event is occurred.

While the set of challenges introduced by the wireless sensor networks are diverse, researchers have mainly focused on fundamental networking challenges, which include: routing protocols, energy minimization, sensor localization, data gathering, etc. [1,2]. Whatever, the performances of the various solutions strongly depend on the network topology. Indeed, most of studies assume that the environment is unknown and the deployment area is inaccessible (e.g. battle field). Consequently, sensors are deployed randomly. On the other hand, some works address specifically the WSN deployment problem and assume that the sensors are likely to be deployed via hand placement. However, most of these works consider that the phenomenon are uniformly distributed within the area. Unfortunately, in more realistic environments, the latter assumption does not hold. In this case, the deployment process must be done manually, non-uniformly and
according to the target area and phenomenon characteristics. A fire detection is a typical example of such irregular geographic event distribution. The detection accuracy strongly depends on the importance of the area. High surveillance accuracy is required for important regions such as near habitats, and low accuracy for less important regions such as rock-ground. Up to now, to the best of our knowledge, few studies have specifically addressed this problem.

In the related deployment strategies, a sensor communication range is not considered. The authors focused only on the quality of monitoring problem and assumed a large communication range to guarantee the network connectivity. However, a large communication range implies that the transmission power is set to the maximum value. Hence, the energy consumption is not optimized.

In this work, we specifically address the issue of wireless sensor network deployment in an area characterized by the geographical irregularity of the sensed event. The main goal is to deploy a wireless sensor network while optimizing the cost of deployment, ensuring the required quality of monitoring and network connectivity, and optimizing network lifetime. When the network is deployed, the event detection probability at any point in the deployment field should be greater than a predefined threshold.

The problem is formulated as combinatorial optimization problem. To overcome the computational complexity, we propose two scalable wireless sensor network deployment strategies based on the Tabu search metaheuristic and the artificial potential field, namely potential field deployment algorithm (PFDA) and multi-objectives deployment algorithm (MODA). Our proposals assume an initial over dimensioned sensor deployment. Then, we calculate for each sensor its virtual force that will be exerted in order to move it. The virtual force computation (i.e. angle and magnitude) takes into account the cost of deployment (reduce the number of deployed sensors), and the required event detection threshold required at each point belonging to the sensor event detection circle. We compare our proposal to the related strategies, the simulations show that we obtain the best performances. The early work of PFDA was published in [3]. This paper includes significant extensions in the objectives tackled and simulations.

The rest of this paper is structured as follows: the next section provides a state of the art on WSN deployment issue. The generalized problem is formulated in Section 3. Our potential field deployment algorithm (PFDA) is detailed in Section 4. Performance evaluation is discussed in Section 5. Then, network lifetime objective is considered and we describe MODA in Section 6. Finally, Section 7 summarizes our contributions.

2. Related works

Up to now, few WSN deployment strategies have been proposed in the literature. This section contains two main parts. Firstly, we introduce the mobile WSN deployment strategies which are based on virtual forces paradigm. Secondly, we summarize the static non-uniform deployment strategies.

2.1. Mobile sensor network

In [4–7], the authors assumed a uniform event detection requirement and focused mainly on full coverage problem. Proposals are based on the virtual force approach in aim to move sensors at the best positions. All the methods are deterministic, the virtual forces (i.e. angle and magnitude) are calculated with a deterministic procedure.

In [4], the authors studied mobile WSN deployment considering an unknown environment. The main objective is to ensure the full coverage of the deployment area. For this purpose, initially a certain number of sensors are concentrated in a small place within the deployment area. Then, sensors are spread out by applying virtual forces. To compute the virtual force that will be applied in the sensor, the authors used the potential field concept.

In [5], the authors proposed the virtual force algorithm (VFA) to spread out sensors and enhance coverage after an initial random deployment. The VFA process is based on the disk packing technique [8] and virtual force field. The authors considered that each sensor is the source of forces. If the distance separating two sensors is less than a predefined threshold, then the sensors exert a repulsive force on each other. Otherwise, sensors exert attractive forces. The final force in each sensor is equal to the sum of repulsive and attractive forces generated by its neighbors.

In [6], the authors proposed three distributed deployment algorithms VEC (VECtor-based), VOR (VORonoi-based) and Minimax to deploy mobile WSN. The strategies are based on the construction of Voronoi diagrams to determine the existence of coverage holes. Basically, each sensor broadcasts its locations. Based on the received informations, the node builds its local Voronoi polygons. Then, in order to increase the global coverage, each sensor moves and the motion (direction and distance) is calculated according to the coverage holes in its vicinity. A sensor’s motion is determined only by one strategy among the three strategies proposed: (i) VEC pushes sensors away from area with high sensor density, (ii) VOR pulls sensors to the scarcely covered area, (iii) Minimax moves sensors to their local center area. The main difference between VOR and Minimax resides in the fact that Minimax introduces a new mechanism to avoid sensor movement oscillation.

In [9], the authors focused on the area coverage problem and assumed that wireless sensor network contains both static and mobile sensors. In this context, a deployment strategy based only on virtual forces is not powerful, due to the fact that static sensors can exert virtual forces on mobile nodes. To avoid the later drawback, the authors proposed the Virtual Force directed co-evolutionary particle swarm optimization (VFCP\textit{PSO}) algorithm. This algorithm is a combination of Virtual Forces and co-evolutionary particle swarm optimization (\textit{PSO}) algorithm. \textit{CPSO} is an improvement of the particle swarm optimization (\textit{PSO}) algorithm, which is a population based stochastic optimization technique. Basically, \textit{VFCP\textit{PSO}} performs a local optimization by using a virtual force method and global optimisation by using \textit{CPSO}. Hence, the running time is reduced compared to the solution obtained only by \textit{CPSO}.
All the deployment strategies based on virtual forces, introduced above, assumed uniform detection requirement and focused only on the area coverage problem. However, if we take into consideration that the deployment area is characterized by a geographical irregularity of the sensed event detection probability, then we need to guarantee a required event detection probability at each point of the deployment field.

2.2. Static sensor network

In [10–14], the authors assumed a non-uniform event detection requirement and proposed different strategies based on regular structures or pseudo-random strategies.

In [10], the authors proposed two deterministic deployment algorithms. A probabilistic event detection model is assumed. It is expressed as an exponential function of the separating distance between the detected event and the sensor. This model was considered in the proposal of two new deployment algorithms. Both approaches seek toward an optimal area coverage under the constraints of imprecise detections. The strategies are based on a grid structure. The first algorithm, Max-Avg-Cov, aims to maximize the average coverage of the grid points. The second one, Max-Min-Cov aims to maximize the coverage of the grid points which are the least effectively covered. Both Max-Avg-Cov and Max-Min-Cov were initially designed for an area with uniform detection probability, however they can easily adapted in the case of an area with non-uniform detection probability. Unfortunately, the two strategies suffer from high computational complexity, it is equal to $O(n^6)$.

In [11], the authors proposed a new deterministic deployment strategy called Min-Miss. This strategy is an iterative algorithm, one sensor is deployed at each step. The authors associate for each point in the deployment field a new metric named over miss probability. The later quantifies the benefit in coverage when a new sensor is deployed. Min-Miss operates as following. Firstly, all the free grid points (i.e. no sensors are deployed) are selected and build a set. Thereafter for each free point in set, the authors calculate its over miss probability metric. Finally, a new sensor is deployed at the point that minimizes the over miss probability. That means, a sensor is deployed in a position that maximizes the event detection probabilities in the deployment field. The main drawback of this approach is the huge computational complexity, which is equal to $O(n^6)$.

In [12], the authors addressed explicitly a WSN deployment problem for non-uniform detection requirement. As in [10], they assume a probabilistic event detection model. They formulated the differentiated deployment problem as an integer linear programming deployment problem, which has been proved to be NP-complete. A new deterministic deployment heuristic, named Diff-Deploy, is proposed. The disadvantage of this solution is the computational complexity which is equal to $O(4^n n^4)$.

In [13], we proposed a deterministic deployment algorithm named differentiated deployment algorithm (DDA). It is based on mesh representation used in image processing and 3D modeling. Generally, mesh representation allows convenient modeling of arbitrary surfaces, where meshes serve as basic primitives to approximate a surface. In the WSN deployment context, we used an unstructured triangle mesh representation of the target area. Basically, the meshes nodes represent sensor positions and each arc is the Euclidean distance between the sensors. The main idea of DDA is to allow meshes division as long as the required detection probability is not reached or the mesh division is not benefit. The mesh division consists to deploy a new sensor. Unfortunately, DDA suffers form high computational complexity, which is equal to $O(n^6)$.

In [14], we proposed a pseudo-random deployment strategy based on Tabu search metaheuristic named Bernoulli Deployment Algorithm (BDA). Basically, the decision to deploy or remove a sensor in the area is made according to the Bernoulli distribution. For this reason, we defined the Bernoulli parameter as the gap between the required and generated detection probabilities. BDA is extended in [15] to take into consideration the network connectivity. The complexity of the method is equal to $O(Cn^2)$, where $C$ depends mainly on the Tabu search parameters and the sensor radius coverage. The computational complexity of the method is quadratic. However, the main weakness of the proposal is the pseudo-random character in the generation of the topologies. Consequently, we cannot guarantee that all points in the deployment area will obtain the required event detection probability.

Network connectivity is not tackled in all the deployment strategies proposed above. The authors assumed that the communication range is so large in order to ensure connectivity. Coverage and connectivity are related since they are both affected by the position of sensors. In [16–18], the authors formulated sufficient conditions to guarantee the coverage and network connectivity. The sufficient conditions are impacted by the sensor positions, communication range ($R_{\text{com}}$), and sensing range ($C_{\text{max}}$). For example in [18], the authors prove that if $R_{\text{com}} \geq 2C_{\text{max}}$ and the area is fully covered then the communication graph is connected. However, having a large communication range is not realistic and expensive in terms of energy. The assumption $R_{\text{com}} \geq 2C_{\text{max}}$ may not hold in practice because the reliable communication range is often 60–80% of the claimed value as developed in [16]. In this paper, our objective is to propose a new WSN deployment strategy that ensures the required QoM and network connectivity for any value of communication and sensing ranges as considered in [18].

3. Problem statement

3.1. Sensor event detection model

An important factor in the deployment issue is the sensor detection capabilities. Depending on the employed technology, sensing devices have generally different physical characteristics. Hence, numerous theoretical detection models of varying complexity can be proposed based on application needs and device features. Generally, we can distinguish between two models. In the first model, called unit disc model, a sensor is supposed to be able to detect an
event if and only if their separating distance is less than a particular sensing range. This model was mainly considered in research works addressing area coverage problems, such as target detection or k-coverage problem. The model aims to simplify the analysis, but unfortunately it does not reflect really the sensing capabilities of sensors. The second model is based on Elfes model [19] and introduced in WSN [20–22]. It assumes that sensing ability diminishes as the separating distance to the monitored point increases (e.g. sensing a temperature). Using this model, a confidant and maximum sensor monitoring circles are defined. If any event occurs within the confidant circle (event (1) in Fig. 1), then the event is detected. If the event occurs outside the confidant circle but within the maximum circle (event (2) in Fig. 1), then the event detection probability is inversely proportional to the euclidian distance separating the sensor from the event. Finally, when the distance is larger than the radius of the maximum circle (event (3) in Fig. 1), the event is no longer detected. Based on the above assumption, we consider, as in [23], the following expression of the general detection probability \( P_s \) of a sensor \( s \) at an arbitrary point \( p \):

\[
P_s = \begin{cases} 
1 & \text{if } ||sp|| \leq C_{\text{con}}, \\
\frac{s}{||sp||} & \text{if } C_{\text{con}} < ||sp|| \leq C_{\text{max}}, \\
0 & \text{if } C_{\text{max}} < ||sp||, 
\end{cases} \tag{1}
\]

where,

- \( C_{\text{con}} \) and \( C_{\text{max}} \) denote respectively, the confidant and the maximum radius coverage circle of a sensor,
- \( ||sp|| \) is the euclidian distance between a sensor \( s \) and a point \( p \),
- \( \alpha \) is a sensor technological related parameter,
- \( \beta \) is an event characteristic-dependant parameter (e.g. temperature, pressure, etc.).

We believe that this model is more realistic than the unit disc model. Therefore, we assume it for our deployment problem, which we formulate in the next section.

### 3.2. Problem formulation

We consider a square target deployment area denoted \( T \). The deployment field is discretized into cells of square units. A unit is defined as a normalized physical distance. Thus, the area \( T \) is a square with a side equals to \( n \) units.

We will refer to each square unit, cell, of \( T \) by its **barycenter** point. In other words, when a sensor is deployed at point \( p \in T \) then this means that a sensor is placed at the barycenter of the cell containing \( p \) as illustrated in Fig. 2. We associate for each point \( p \) a binary variable \( D_p \). It is equal to 1, if a sensor is deployed at point \( p \). Otherwise, \( D_p \) is equal to 0. Similarly, the detection probability of a unit square is calculated considering the detection probability of its barycenter. Finally, we consider that 

\[
C_{\text{con}} = \sqrt{\frac{1}{2}} \text{unit}^2.
\]

Indeed, any event occurring inside a unit square is detected with probability 1 by a sensor which would be placed in its barycenter point. The analysis, conducted in the rest of the paper, will be based on the set of cells’ barycenter points denoted \( A \).

We consider a probabilistic event detection model. We assume that to each point \( p \in A \) is associated a required minimum event detection probability threshold denoted \( R_p \). It is calculated according to the event historic at point \( p \) by identifying the events spatio-temporal correlation [24]. Thus, the first objective of our deployment problem is that \( \forall p \in A \), the measured event detection probability of that point is greater or equal than \( R_p \).

The event detection probability at point \( p \) is estimated by all the sensors in its vicinity. However, we do not consider a collaborative event detection model. The collaboration between sensors needs a specific protocol in order to exchange information to make event detection decision (i.e. it depends on the neighbours). We assume that the event detection decision is made locally in a sensor without any influence from its neighbours. The event detection probability at point \( p \), denoted \( P_p \), is estimated as

\[
P_p = 1 - \prod_{s \in V_p} \left( 1 - P_s^p \right), \tag{2}
\]

where, \( P_s^p \) is the detection probability of a sensor \( s \) at point \( p \), expressed by Eq. (1). \( V_p \) is the set of deployed sensors in the vicinity of \( p \). Therefore, \( \forall p \in A \), \( V_p \) is defined as the set of sensors located in the circle with a center \( p \) and radius \( C_{\text{max}} \).

Our first objective is to minimize the number of sensors. More formally, the objective can be expressed as

\[
\min \sum_{s \in A} D_s, \tag{3}
\]

Once the sensors are deployed, all the generated event detection probabilities must be greater than or equal to the threshold \( R_p \). These constraints model our second objective and can be formulated as follows:

\[
P_p \geq R_p \quad \forall p \in A \tag{4}
\]

We replace \( P_p \) by

\[
1 - \prod_{s \in A} \left( 1 - P_s^p \right),
\]
hence we obtain the following inequation:
\[
\ln \left( \prod_{s \in A} \left( 1 - P_p^s \right)^{D_s} \right) \leq \ln(1 - \mathcal{R}_p) \quad \forall p \in A.
\] (5)

Making several transformation, we obtain the following constraint:
\[
\sum_{s \in A} x_p^s D_s \leq \beta_p \quad \forall p \in A
\] (6)

where
\[
x_p^s = \ln \left( 1 - P_p^s \right),
\]
\[
\beta_p = \ln(1 - \mathcal{R}_p).
\]

Moreover, network connectivity must be guaranteed in order to notify the sink when an event is detected. For this aim, the graph of connectivity \(\mathcal{G}(V, E)\) must be connected. Where, \(V\) is the set of vertex formed by the deployed sensors, including the sink. \(E\) is the set of communication edges between sensors. \(\mathcal{C}P_p\) denotes the set of connected components of \(\mathcal{G}(V, E)\). To guarantee network connectivity, the cardinal of \(\mathcal{C}P_p\) must be equal to 1. Formally, network connectivity constraint can be expressed as following:
\[
|\mathcal{C}P_p| = 1
\] (7)

To summarize, our differentiated WSN deployment problem can be formalized as following:

**Problem 1:** WSN deployment problem: Cost, QoM, and network connectivity

\[
\begin{align*}
\text{minimize} & \quad \sum_{s \in A} D_s \\
\text{subject to:} & \quad \sum_{s \in A} x_p^s D_s \leq \beta_p \quad \forall p \in A \\
& \quad x_p^s = \ln \left( 1 - P_p^s \right) \\
& \quad \beta_p = \ln(1 - \mathcal{R}_p) \\
& \quad |\mathcal{C}P_p| = 1 \\
& \quad D_s \in \{0, 1\}
\end{align*}
\]

Problem 1 is an integer optimization problem (binary), which is NP-complete [25]. Hereafter, we propose a new scalable deployment strategy, named PFDA based on Tabu search metaheuristic.

### 4. Potential field deployment algorithm – PFDA

As in our previous work [15], our proposal, named potential field deployment algorithm PFDA, relies on the Tabu search metaheuristic [26,27]. Our algorithm starts with a judiciously chosen initial solution and then iteratively explores the solution space until that a predetermined number \(N\) of iterations is reached. During each iteration, a given number of candidate solutions are generated from the actual solution. Then, using a cost function, a new solution will be selected among the candidates. The latter will be inserted in a Tabu list TL which will contain all the visited solutions that could not be selected again as candidate during the \(T\) next exploration stages. Once the stop criterion is reached, the algorithm returns the best solution memorized in TL. In the following, we will detail the fundamental elements of our algorithm: the initialization procedure, the candidate exploration stages, and the cost function.

#### 4.1. Initialization

At this stage, the main objective of our initialization procedure is to fulfill the both objectives: (i) \(P_p \geq \mathcal{R}_p\), \(\forall p \in A\) and ii) network connectivity. Minimizing the number of deployed sensors is not a priority in this stage. Given that the detection probability at point \(p\) is estimated by all sensors located in its vicinity, see Eq. (2). The potential locations for our initial deployment are the points \(p \in A\) where the ratio of cells \(v \in V_p\) with unsatisfied event detection probabilities are high (i.e. close to 1), and the points \(p\) are within the communication range \(\mathcal{R}_{com}\) of at least one deployed sensor. To express the event detection criteria we define \(\forall p \in A\) the following metric:
\[
x_p = \frac{1}{|V|} \sum_{v \in V_p} 1_{\{p \in \mathcal{R}_v\}}
\] (8)

where \(1_{\{\text{cond}\}}\) is the indicating function, which is equal to 1 if the condition “cond” is true and 0 otherwise.

Our initialization procedure starts by assuming that no sensors are yet deployed. Thus, a first node deployed is the sink, its coordinates are chosen by the user. Thereafter, a new sensor is deployed at point \(p\) with the highest value of \(x_p\) and can communicate with the nodes already deployed (i.e. sink). After updating the parameter \(x_p\), \(\forall p \in A\), this process is repeated until achieving a full event detection probability. The connectivity is thus guaranteed by construction (gradually). The deployed sensor topology resulting from our initialization procedure is considered as the initial solution, which will be inserted in the TL.

#### 4.2. Neighborhood exploration functions

##### 4.2.1. Artificial potential field approach

The initialization stage will lead to an initial deployment solution, which will satisfy the network connectivity and the event detection probabilities objectives. This will be probably achieved at the cost of an over-dimensioned deployment solution: the number of deployed sensors is much larger than what is really necessarily to solve our optimization problem.

Now, the aim of the exploration stages is to adjust the number and the positions of the deployed sensors. The fundamental mechanism behind this adjustment is the motion of sensors. The basic idea of our proposal is that the sensor motion is induced from an artificial virtual forces. Despite its popularity, there is no general framework based on virtual forces that can be applied to address any optimization problem. The approach has to be adapted to the problem’s specificities.
In our case study, we propose to follow the artificial potential field approach, which was introduced in [28]. The authors proposed a method that aims to find the optimal trajectory of robot from an initial point to a target point in presence of obstacles. The robot’s movements are derived according to the virtual force \( \vec{F}_r \) exerted on the robot. Formally, \( \vec{F}_r \) is given by the following formula [28]:

\[
\vec{F}_r = -\nabla \mathcal{F}_r = - \left( \frac{\partial \mathcal{F}_r}{\partial x_r}, \frac{\partial \mathcal{F}_r}{\partial y_r} \right)
\]  

(9)

where, \((x_r, y_r)\) are the coordinates of the robot and \( \mathcal{F}_r \) is the potential field function, which depends on the location \( r \) of the robot. This function quantifies the degree of attraction or repulsion on the trajectory of the robot. Basically, since the target \( t \) is an attractive pole for the robot and the obstacles represent the repulsive surfaces, then the potential field function in areas close to \( t \) are mainly attractive, while the potential field function for the points close to obstacles are mainly repulsive.

In the follow, we propose an adaptation of the artificial potential field approach to our case study. Precisely, for each deployed sensor \( s \in \mathcal{D} \) we introduce two separates artificial potential field functions: \( \mathcal{F}^a(s) \) and \( \mathcal{F}^s(s) \). Each field induces a virtual force on the sensor \( s \), an attractive virtual force denoted \( \vec{F}^a_s \) and a repulsive virtual force denoted \( \vec{F}^r_s \). The calculation of the virtual forces is detailed in the following sections.

4.2.2. Repulsive virtual force

The artificial repulsive forces aim to minimize the number of deployed sensors. To reach this aim, the separating distance between sensors is maximized (i.e. sensor density is decreased) and at the same time the generated event detection probabilities and network connectivity should be maintained. This action could lead to push some sensors outside the area \( T \) or merge two neighbor nodes, which will correspond to reducing the number of sensors.

We propose the following artificial potential field function \( \mathcal{F}^a(s) \) of a sensor \( s \), generated by the deployment field:

\[
\mathcal{F}^a(s) = \sum_{p \in \mathcal{A}_s} \left[ P_{p/s} - \min(\mathcal{R}_p, P_p) \right]^2,
\]

(10)

where \( P_{p/s} \) is the generated detection probability at point \( p \) assuming that a sensor is deployed at point \( s \). Formally,

\[
P_{p/s} = 1 - \left( 1 - P^p_s \right) \prod_{s \in \mathcal{D}_{p/s}} \left( 1 - P^p_s \right).
\]

(11)

The repulsive potential field function, \( \mathcal{F}^r(s) \), quantifies the gap between the generated and required event detection probabilities according to the coordinates of a sensor \( s \). Our objective is to move a sensor \( s \) in order to maximize the distance between sensors and to minimize the repulsive potential field function.

Then, similarly to [28], we can derive the virtual repulsive force as follows:

\[
\vec{F}^r_s = -\nabla \mathcal{F}^r(s) = \left( \frac{\partial \mathcal{F}^r(s)}{\partial x_s}, \frac{\partial \mathcal{F}^r(s)}{\partial y_s} \right).
\]

(12)

Replacing Eqs. (10), (11), and (2) in (12) we find that:

\[
\frac{\partial \mathcal{F}^r(s)}{\partial x_s} = \sum_{p \in \mathcal{A}_s} \left[ 2 \left( \frac{\alpha}{||sp||} - \min(\mathcal{R}_p, P_p) \right) \right. \\
\left. \left( \frac{\alpha(b(x_p - x_s))}{||sp||^{p+2}} \right) \prod_{s \in \mathcal{D}_{p/s}} \left( 1 - P^p_s \right) \right],
\]

(13)

\[
\frac{\partial \mathcal{F}^r(s)}{\partial y_s} = \sum_{p \in \mathcal{A}_s} \left[ 2 \left( \frac{\alpha}{||sp||} - \min(\mathcal{R}_p, P_p) \right) \right. \\
\left. \left( \frac{\alpha(b(y_p - y_s))}{||sp||^{p+2}} \right) \prod_{s \in \mathcal{D}_{p/s}} \left( 1 - P^p_s \right) \right],
\]

(14)

where

\[
\mathcal{A}_s = \{ \forall p \in \mathcal{A} | \mathcal{R}_p > P_p \text{ and } C_{con} < ||sp|| \leq C_{max} \}.
\]

(15)

4.2.3. Attractive virtual force

The aim of the attractive virtual force is to move sensors toward the points where the detection probability constraint is unsatisfied. The violation of this constraint could happen in some points of \( \mathcal{A} \) as a consequence of the motion resulting from the repulsive forces. To readjust the position of sensors, we define the following artificial attractive potential field function of a sensor \( s \) generated by the deployment field:

\[
\mathcal{F}^a(s) = \sum_{p \in \mathcal{A}_s} (\mathcal{R}_p - P_p/s) \cdot 1_{\{p < \mathcal{R}_p\}}
\]

(16)

The attractive potential field function, \( \mathcal{F}^a(s) \), quantifies the event detection probabilities deficiency in the neighborhood region of a sensor \( s \). The miss detection depends on the coordinates of a sensor \( s \). Similarly to the repulsive virtual force, the artificial attractive virtual force \( \vec{F}^a \) exerted on the sensor \( s \) is derived using the gradient of the attractive potential field function, \( \mathcal{F}^a(s) \). Formally, \( \vec{F}^a = -\nabla \mathcal{F}^a(s) \). The gradient \( \nabla \mathcal{F}^a(s) \) is obtained after replacing Eqs. (16), (11), and (2) in (12). Precisely,

\[
\frac{\partial \mathcal{F}^a(s)}{\partial x_s} = \sum_{p \in \mathcal{A}_s} \left[ \alpha(b(x_s - x_p)) \prod_{s \in \mathcal{D}_{p/s}} \left( 1 - P^p_s \right) \right],
\]

(17)

\[
\frac{\partial \mathcal{F}^a(s)}{\partial y_s} = \sum_{p \in \mathcal{A}_s} \left[ \alpha(b(y_s - y_p)) \prod_{s \in \mathcal{D}_{p/s}} \left( 1 - P^p_s \right) \right],
\]

(18)

where the set \( \mathcal{A}_s \) is defined in Eq. (15).

4.2.4. Force magnitude

From the previous section, one can easily derive the direction and the magnitude of the attractive and repulsive virtual forces. Unfortunately, when we applied the calculated magnitude values to some test scenarios, we observed that the algorithm was unable to converge toward a steady state. Besides the use of Tabu List in order to ensure the convergence and skip the local optimum and oscillations, we limit the movement of a sensor to its adjacent cells. Precisely, the sensor is moved to the first adjacent cell located in the direction calculated according to our procedure described in the previous section. In a certain way, we limit the magnitude of the resulting force to one unit.
When a sensor is moved, we check the network connectivity. If the moved sensor breaks the network communication graph then we do not allow this movement. Hence, the network connectivity is maintained during all the Tabu search iterations.

4.2.5. Tabu alternating exploration stages

During an exploration stage, we do not calculate simultaneously the repulsive and attractive virtual forces, which are exerted over all the sensors. Instead, we will consider during an iteration only one type of force.

For instance, during an iteration we will calculate only the repulsive forces exerted over the deployed sensors. These forces might lead to move several sensors, but we allow the motion of only one sensor during an iteration. Basically, sensors in \( \mathcal{D} \) are considered role by role. On each considered sensor \( s \in \mathcal{D} \), we determine the new location resulting from the application of the repulsive virtual force. If the obtained topology is already present in the TL or the network connectivity is broken then this movement is considered as unauthorized. Otherwise, the generated detection probabilities for all points in \( \mathcal{A} \) are evaluated assuming the new location of \( s \). Finally, the moved sensor is replaced back to its initial location, before considering the next sensor in \( \mathcal{D} \).

After having considered all the sensors in \( \mathcal{D} \) and evaluated the impact of their movement, only one sensor movement will be realized. Among all the simulated and authorized movements, we select the motion that maximizes the satisfaction rate. This metric corresponds to the percentage of points \( p \in \mathcal{A} \) where the constraints: \( P_p \geq R_p \) are satisfied. If several solutions provide the same maximum satisfaction rate, then the elected solution among them is the one that minimizes the number of deployed sensors.

Then, in the next Tabu exploration stage, the same procedure is executed assuming this time the attractive virtual forces, instead of repulsive ones. The exploration of solutions continues by alternating between attractive and repulsive stages until that the number of iterations reaches its maximum number \( N \).

The PFDA pseudo-code is illustrated in Algorithm 1. The computation complexity is equal to \( \mathcal{O}(vn^2) \). \( v \) is the initial number of deployed sensors and \( N \) is the maximum number of iterations. \( N \) and \( v \) are constants. We define a con-
constant C as the product between N and v. Hence, PFDA computation complexity is quadratic and it is equal to O(Cn^2).

Algorithm 1: PFDA pseudo-code

1: Compute initial solution x0
2: xout ← x0
3: TL ← (x0)
4: Routine = {Repulsive, Attractive}
5: Routine: MVF
6: MVF ← Repulsive
7: for i = 0 to N do
8: if MVF = Repulsive then
9: for all s ∈ D do
10: Compute repulsive virtual force F_s^r
11: end for
12: else
13: for all s ∈ D do
14: Compute attractive virtual force F_s^a
15: end for
16: end if
17: for all s ∈ D do
18: Simulate the sensor movement s
19: end for
20: Select a sensor s_j to move it, which
1: maximizes the satisfaction rate,
2: reduces the number of sensors after moving,
3. the result topology x_j^r ≠ TL
21: x_{i+1} ← x_j^r
22: if Satisfaction Rate (x_j^r) = 100% then
23: MVF ← Repulsive
24: else
25: MVF ← not (MVF)
26: end if
27: if x_j^r is better than xout then
28: xout ← x_j^r
29: end if
30: %Update the tabu list TL%
31: TL ← TL ∪ {x_{i+1}}
32: end for

4.3. Simplified potential field deployment algorithm

When we relax network connectivity constraint, we can apply PFDA but with a minor modification. In the initialization stage, we do not take into account if the next potential sensor’s locations could reach the already deployed sensors. Only the unsatisfied event detection probabilities criterion is considered. In the exploration stage, we accept the motions breaking network connectivity. The simplified version of PFDA is noted S-PFDA.

5. Performance analysis

In this section, we evaluate the performance of PFDA and S-PFDA. We implemented our proposal and the related deployment strategies in C++. We run our proposed deployment strategy PFDA and we compare it to the deployment strategies found in literature Random-Uniform, Random-Bernoulli, Grid, Max-Min-Cov [10], Max-Avg-Cov [10], Min-Miss [11], Diff-Deploy [12], DDA [13], and EDA [14,15]. The related strategies cannot deploy more sensors than PFDA and assume the identical required QoM distribution as used in PFDA.

The Random-Uniform strategy consists to deploy the sensors in the deployment field according to the uniform distribution. However, the decision to deploy a sensor in any cell with a Random-Bernoulli strategy follows a Bernoulli distribution. The Bernoulli parameter is equal to the cell’s required event detection probability threshold.

We set the maximum number of iterations of Tabu search N and the size of the Tabu List to 5000 and 20, respectively. We set a confidence level of stochastic deployment strategies to 99.70%. The sink is deployed at the center of the deployment area.

5.1. PFDA results

We test the WSN deployment strategies according to two types of requested event detection probabilities distributions: (i) non-uniform distribution and (ii) uniform distribution. In the following, we present the obtained results.

5.1.1. Non-uniform distribution

We test our proposal PFDA considering two scenarios (i) \( R_{com} < C_{con} \), and (ii) \( R_{com} > C_{con} \). We set the sensor parameters \( \alpha \) and \( \beta \) to 1. We have considered an area of 50 x 50 units. We assume that the requested event detection probabilities are non-uniformly distributed in the deployment field. The distribution is illustrated in Fig. 3.

1. \( R_{com} < C_{con} \)

We set the sensor parameters \( R_{com}, C_{con}, C_{max} \) to 90, 14.15, 140 m, respectively. The unit length is set to 20 m. Our proposal PFDA deploys 117 sensors. In Fig. 4, we illustrate the network topology. We notice that a network is connected and we obtain a full satisfaction rate (100%). In all the cells, the generated event detection probability is greater than the requested. In Fig. 5, we show the generated event detection probabilities. We compare PFDA to the related deployment strategies. We set the number of sensors to 117 as obtained in PFDA. Table 1 summarizes the satisfaction rate generated by the different methods. We notice that our proposal PFDA ensures network connectivity and outperforms all the related deployment strategies. In spite of connectivity, PFDA does not need a large number of nodes and guarantees a full satisfaction rate. Fig. 6 shows the cumulative distribution function (cdf), obtained by the related deployment strategies. We notice that our proposal PFDA provides the best performances.

In Fig. 7, we show the 10 largest connected components generated by the related strategies. A largest connected component is the one which contains the greatest number of nodes. We remark that only the graphs resulting
from PFDA, BDA, Max-Avg-Cov, and Grid contain one connected component. In the case of the Max-Avg-Cov and Grid, we ensure the connectivity but the performances in term of event detection probabilities are mediocre. BDA does not ensure a full satisfaction rate. The remainder methods generate more than one connected component, hence the network is not connected. For example, we remark that Diff-Deploy contains many small connected component. Our proposal PFDA is the only method which ensures the connectivity and a full satisfaction rate while minimizing the number of required sensors.

2. \( R_{\text{com}} > C_{\text{con}} \)

We set the sensor parameters \( R_{\text{com}}, C_{\text{com}}, C_{\text{max}} \) to 50, 3.53, 25 m, respectively. The unit length is set to 5 m. Table 2 summarizes the obtained results.

Since the communication range is large, all the deployment strategies generate connected network topolo-

gies. However, only PFDA ensures a full satisfaction rate while minimizing the number of sensors.

5.1.2. Uniform distribution

We test our proposal PFDA and the related deployment strategies with three uniform distributions scenarios. In scenarios 1, 2, and 3 we request an event detection probability equal to 0.3, 0.6, and 0.9, respectively in the whole area. The requested event detection probabilities of each scenario are shown in Fig. 8.

### Table 1

<table>
<thead>
<tr>
<th>Deployment strategy</th>
<th>Number of sensors</th>
<th>Satisfaction rate %</th>
<th>Connectivity</th>
</tr>
</thead>
<tbody>
<tr>
<td>PFDA</td>
<td>117</td>
<td>100</td>
<td>ok</td>
</tr>
<tr>
<td>BDA</td>
<td>117</td>
<td>98.95 ± 0.20</td>
<td>ok</td>
</tr>
<tr>
<td>Diff-Deploy</td>
<td>117</td>
<td>99.84</td>
<td>–</td>
</tr>
<tr>
<td>DDA</td>
<td>117</td>
<td>82.92</td>
<td>–</td>
</tr>
<tr>
<td>Min-Miss</td>
<td>117</td>
<td>82.52</td>
<td>–</td>
</tr>
<tr>
<td>Max-Min-Cov</td>
<td>117</td>
<td>78.32</td>
<td>–</td>
</tr>
<tr>
<td>Max-Avg-Cov</td>
<td>117</td>
<td>74.96</td>
<td>ok</td>
</tr>
<tr>
<td>Grid</td>
<td>117</td>
<td>74.04 ± 1.75</td>
<td>–</td>
</tr>
<tr>
<td>Random-Uniform</td>
<td>117</td>
<td>80.29 ± 1.65</td>
<td>–</td>
</tr>
</tbody>
</table>

### Fig. 5.

PFDA – generated event detection probabilities, non-uniform distribution.

### Fig. 6.

PFDA – cumulative distribution function comparison, non-uniform distribution.
PFDA deploys 87, 127 and 254 sensors in scenarios 1, 2 and 3, respectively. In Figs. 9 and 10, we illustrate the network topologies and the generated event detection probabilities obtained in the above scenarios. We notice that the connectivity and the requested quality of monitoring are guaranteed.

We compare PFDA to the related deployment strategies, the obtained results are illustrated in Tables 3–5.

**PFDA satisfies rate comparison, non-uniform distribution, $R_{con} \geq C_{con}$.**

<table>
<thead>
<tr>
<th>Deployment strategy</th>
<th>Number of sensors</th>
<th>Satisfaction rate %</th>
<th>Connectivity</th>
</tr>
</thead>
<tbody>
<tr>
<td>PFDA</td>
<td>149</td>
<td>100</td>
<td>ok</td>
</tr>
<tr>
<td>BDA</td>
<td>149</td>
<td>98.05 ± 0.28</td>
<td>ok</td>
</tr>
<tr>
<td>Diff-Deploy</td>
<td>149</td>
<td>95.96</td>
<td>ok</td>
</tr>
<tr>
<td>DDA</td>
<td>149</td>
<td>84.64</td>
<td>ok</td>
</tr>
<tr>
<td>Min-Miss</td>
<td>149</td>
<td>81.24</td>
<td>ok</td>
</tr>
<tr>
<td>Max-Min-Cov</td>
<td>149</td>
<td>76.04</td>
<td>ok</td>
</tr>
<tr>
<td>Max-Avg-Cov</td>
<td>149</td>
<td>76.96</td>
<td>ok</td>
</tr>
<tr>
<td>Grid</td>
<td>149</td>
<td>76.32</td>
<td>ok</td>
</tr>
<tr>
<td>Random-Uniform</td>
<td>149</td>
<td>71.25 ± 1.17</td>
<td>ok</td>
</tr>
<tr>
<td>Bernoulli</td>
<td>149</td>
<td>74.56 ± 1.17</td>
<td>ok</td>
</tr>
</tbody>
</table>

We notice that our proposal PFDA obtains the best performance when the required event detection probabilities are medium (scenario 2). Nevertheless, PFDA guarantees the connectivity in scenario 1 but the cost of deployment is not minimized. When the required event detection probability increases (scenario 3), the density of sensors grows and the network topology tightens to the grid topology. In Figs. 11–13, we compare the number of connected components generated by the related strategies. The connectivity is ensured in all scenarios only with Grid and our proposals PFDA and BDA.

**S-PFDA results**

S-PFDA strategy does not tackle the network connectivity. We compare the WSN deployment strategies according to two types of requested event detection probabilities distributions: (i) non-uniform distribution and (ii) uniform distribution.

**5.2.1. Non-uniform distribution**

We split our analysis into two steps. In the first step, we evaluate how far is the S-PFDA solution compared to the optimal sensor deployment. To calculate the optimal solution, the Branch and Bound (B&B) algorithm [29] is used.
This is a general algorithm that finds the optimal solution to various optimization problems, especially in discrete and combinatorial optimization. Branch and Bound models the candidate solutions in a tree structure and intelligently visits the tree nodes. Only a set of admissible solutions are visited. Large subsets of candidate solutions are discarded when the cost is sure (mathematically proven) to be greater than a current solution.

However, due to the high computational complexity of B&B algorithm we focused only on small-scale deployment area. In the second step, we focused on large scale deployment area and we compared our proposal to several other algorithms proposed in the literature.

5.2.1.1. Optimal comparison. In this step, we carried on three test scenarios. Each one is characterized by a

<table>
<thead>
<tr>
<th>Deployment strategy</th>
<th>Number of sensors</th>
<th>Satisfaction rate</th>
<th>Connectivity</th>
</tr>
</thead>
<tbody>
<tr>
<td>PFDA</td>
<td>87</td>
<td>100</td>
<td>ok</td>
</tr>
<tr>
<td>BDA</td>
<td>85 ± 1</td>
<td>99.72 ± 0.09</td>
<td>ok</td>
</tr>
<tr>
<td>Diff-Deploy</td>
<td>67</td>
<td>100</td>
<td>–</td>
</tr>
<tr>
<td>DDA</td>
<td>80</td>
<td>100</td>
<td>–</td>
</tr>
<tr>
<td>Min-Miss</td>
<td>87</td>
<td>98</td>
<td>–</td>
</tr>
<tr>
<td>Max-Min-Cov</td>
<td>63</td>
<td>100</td>
<td>–</td>
</tr>
<tr>
<td>Max-Avg-Cov</td>
<td>87</td>
<td>79.56</td>
<td>–</td>
</tr>
<tr>
<td>Grid</td>
<td>87</td>
<td>69.80</td>
<td>ok</td>
</tr>
<tr>
<td>Random- Uniform</td>
<td>87</td>
<td>88.70 ± 1.31</td>
<td>–</td>
</tr>
<tr>
<td>Random- Bernoulli</td>
<td>87</td>
<td>89.33 ± 1.96</td>
<td>–</td>
</tr>
</tbody>
</table>

### Table 3

<table>
<thead>
<tr>
<th>Deployment strategy</th>
<th>Number of sensors</th>
<th>Satisfaction rate %</th>
<th>Connectivity</th>
</tr>
</thead>
<tbody>
<tr>
<td>PFDA</td>
<td>127</td>
<td>100</td>
<td>ok</td>
</tr>
<tr>
<td>BDA</td>
<td>127</td>
<td>98.87 ± 0.26</td>
<td>ok</td>
</tr>
<tr>
<td>Diff-Deploy</td>
<td>127</td>
<td>99.48</td>
<td>–</td>
</tr>
<tr>
<td>DDA</td>
<td>127</td>
<td>86.76</td>
<td>–</td>
</tr>
<tr>
<td>Min-Miss</td>
<td>127</td>
<td>92.88</td>
<td>ok</td>
</tr>
<tr>
<td>Max-Min-Cov</td>
<td>127</td>
<td>98.96</td>
<td>–</td>
</tr>
<tr>
<td>Max-Avg-Cov</td>
<td>127</td>
<td>71.36</td>
<td>ok</td>
</tr>
<tr>
<td>Grid</td>
<td>127</td>
<td>81.64</td>
<td>ok</td>
</tr>
<tr>
<td>Random- Uniform</td>
<td>127</td>
<td>76.07 ± 1.51</td>
<td>–</td>
</tr>
<tr>
<td>Random- Bernoulli</td>
<td>127</td>
<td>77.91 ± 1.45</td>
<td>–</td>
</tr>
</tbody>
</table>

Fig. 9. PFDA – network topology, uniform distribution.

Fig. 10. PFDA – generated event detection probabilities, uniform distribution.
required event detection probabilities distribution for a certain area dimension ($5 \times 5, 6 \times 6, \text{ and } 7 \times 7$ as illustrated in Figs. 14(a), Fig. 15(a), and Fig. 16(a), respectively). We set the maximum coverage range $C_{\text{max}}$, communication range $R_{\text{com}}$, $\alpha$, and $\beta$ for each sensor to 3, 2, 1, and 1 respectively. We calibrate the maximum number of iterations $N$ to 100 and the tabu list TL size to 20.

We summarized in Table 6, the running time and the number of deployed sensors for B&B, S-PFDA, and PFDA. We can clearly notice that the running time of PFDA and S-PFDA is much lower than B&B’s running time necessary to converge. Moreover as expected, the B&B running time increases significantly over the three scenarios. This is mainly due to the size space solutions which rises very quickly according to the number of the optimization problem variables.

We plot in Figs. 14–16 the obtained sensor positions for the three scenarios generated by B&B, S-PFDA, and PFDA. We can clearly observe that B&B and S-PFDA deploys the same number of sensors in all scenarios. Thus, our proposal achieves the optimal performance for low scale area size and when considering only the required detection probability guarantee. Finally, in spite of considering the connectivity, PFDA deploys the same number of sensors as B&B in scenario $6 \times 6$, and just adds one sensor in scenarios $5 \times 5$ and $7 \times 7$.

5.2.1.2. Large scale comparison. The distributions of the required event detection probabilities are illustrated in Fig. 3. S-PFDA deploys 112 sensors, Fig. 17 illustrates the network topology. We notice a clear concentration of sensors in areas requesting high event detection probabilities. As the network connectivity constraint is relaxed, we remark that S-PFDA deploys less sensors than PFDA.

![Fig. 11. PFDA – number of connected components comparison, uniform distribution – scenario 1.](image)

![Fig. 12. PFDA – number of connected components comparison, uniform distribution – scenario 2.](image)
The satisfaction rate is equal to 100%. In Fig. 18, we show the generated event detection probabilities. We compare S-PFDA to the related deployment strategies. We set the number of sensors to 112. Table 7
summarizes the satisfaction rate generated by the different methods. We notice that our proposal S-PFDA ensures a full satisfaction rate and outperforms all the related deployment strategies.

Fig. 19 illustrates the cumulative distribution function (cdf), obtained by the related deployment strategies. We conclude that our proposal S-PFDA provides the best performances and ensures a full satisfaction rate.

5.2.2. Uniform distribution

We assume a uniform distribution of required event detection probabilities as illustrated in Fig. 8. We test and compare S-PFDA with the related strategies. The results obtained are summarized in Tables 8–10. In Fig. 20, we illustrate the generated event detection probabilities. S-PFDA outperforms all the related deployment strategies. It minimizes the cost of deployment and ensures the requested quality of monitoring.

6. Network lifetime

In this section, we will investigate the wireless sensor network deployment problem by addressing the objectives
of the cost of deployment, the quality of monitoring, network connectivity, and network lifetime. We will propose a new deployment strategy that does not require the protocol stack to be improved. Indeed, in order to minimize energy consumption, our aim is to generate the best network topology according (i) to the protocols installed in the nodes (routing, MAC); (ii) to the monitored event characteristics (frequency of occurrence); (iii) to the position of the sink; and finally (iv) to the environment (noise, signal propagation).

To this end, we will propose a new wireless sensor network deployment strategy based on the Multi-Objective Tabu search metaheuristic and virtual forces. The proposal is named the multi-objective deployment algorithm (MODA). It minimizes the cost of deployment, guarantees full satisfaction of the requested quality of monitoring and connectivity, and also maximizes the network lifespan. To run our proposal, the implemented protocol stack must be known. Hence, in this chapter we assume the use of ZigBee-based nodes. We will show that our MODA strategy outperforms all the existing related strategies.

6.1 Problem formalization

As formalized in Section 3, we keep the same model but we enrich it by the network lifetime objective. We will reach this target by adapting the network topology to the routing, MAC, and physical layers used in the network. Thus, instead of proposing any improvement to the protocol stack, we will seek to find the best network topology (i.e. the number and position of sensors) that:

1. Minimizes the traffic load transited over the nodes and favors the load balancing (routing layer).
2. Minimizes collisions (MAC layer).
3. Minimizes the bit error rate (physical layer).

In the following, we will outline how we set about tackling the above objectives in each layer.

6.1.1 Minimizing the negative impact of the routing protocol

The traffic load of each node depends heavily on the routing protocol and the network topology. This traffic
can be considered as an aggregation of two types of traffic. The first type is the traffic generated locally by each sensor when it detects an event. The second type involves forwarding the traffic received from a sensor’s neighbors towards the sink. If we denote the generated and the forwarded packet rates as $a_i$, $b_i$, respectively, the global packet rate $k_i$ transmitted by each sensor $s_i$, can be calculated as follows:

$$k_i = a_i + b_i.$$  \hspace{1cm} (19)

Reducing the negative impact of the routing protocol in turn minimizes the packet rate $\lambda_i$. Hence, our aim is to minimize the maximum load on the network, which can be formulated as

$$\min \left[ \max_{s_i \in V} (k_i) \right].$$  \hspace{1cm} (20)

At the same time, maximizing the load balancing that can be obtained by minimizing the standard deviation of the sensor packet rates:

$$\min \left[ \text{std}_{s_i \in V} (\text{set of } (k_i)) \right].$$  \hspace{1cm} (21)

In order to estimate the generated packet rate, the event occurrence distribution must be known as it is an input of our problem. Let $E_p$ denotes the probability that an event appears within a short period of time $\Delta_t$ at point $p$. $\Delta_t$ quantifies the period of time during a sensor runs its sensing algorithm and notifies the application layer if an event is detected. This time depends on the sensor hardware characteristics and the type of event being monitored. We assume a non-uniform event appearance probability, thus the calculation of $E_p$ depends on the event coordinates and can be fixed by identifying the event’s spatio-temporal correlation [24].
From the assumptions detailed above, we can compute the probability that a sensor \( s_i \) generates a packet during a period \( D_t \), as follows:

\[
E_{s_i} = \frac{1}{C_0 \Psi_{s_i}^{p_{2A}} C_3^{p_{16/C_17}}}.
\]  

Note that \( E_{s_i} \) follows a Bernoulli distribution. Hence, the generated packets process in a sensor \( s_i \) is Poisson with an arrival rate equal to:

\[
\lambda_i = -\ln(1 - E_{s_i}).
\]  

However, to calculate \( \beta_i \), the routing protocol must be known in order to estimate the forwarded traffic in each node. In this research, we assume ZigBee-based nodes [30] are being used, with AODV (peer-to-peer communication) as a routing protocol. The specification [30] recommends that a link cost should be less than 7, and also specifies that implementers are free to apply their ingenuity. Hence, we propose the following formula to calculate the link cost:

\[
\text{cost} = 7 - \frac{\text{LQI}}{36.43}.
\]

Above, LQI [31] is the Link Quality Indicator. This indicator can be calculated using physical layer receiver energy detection (ED), a signal-to-noise ratio or a combination of these to measure the strength and/or quality of a link from which a packet is received. We also assume the Two-ray-ground signal propagation model. Hence, the received signal energy depends mainly on the distance between the transmitter and the receiver. The receiving signal energy is inversely proportional to the separating distance. To simplify the formalization, the effect of noise energy in the antenna is not taken into account. Consequently, the path built by AODV between any sensor \( s_i \) and the sink can be approximated as static, and depends only on the sensors’ separating distance. Then, the different paths between sensors and the sink constitute a tree, in which the root is the sink. Once the approximate forwarding tree has been built and the generating packet loads \( \lambda_i \) have been calculated, the forwarding packet rate \( \beta_i \) can be easily set as follows:

\[
\beta_i = \sum_{j=\text{Children}(i)} \beta_j.
\]

This means that we calculate the \( \beta_i \) of nodes at the deep depth \( d \), then the nodes at depth \( d - 1 \), etc. We run recursively the same process until reaching the root (sink).

### 6.1.2. Minimizing the negative impact of the MAC layer

ZigBee nodes assume the IEEE 802.15.4 MAC layer [30]. In this research, we adopt the IEEE 802.15.4 non-beacon enabled mode is being used, which allows peer-to-peer topology. In this mode, a sleep period with a transceiver off does not exist and the lowest energy state is the idle mode. Specifically, the radio default mode is idle. When

![Fig. 19. S-PFDA – cumulative distribution function comparison, non-uniform distribution.](image)
it detects a signal with the same carrier proprieties, it switches to the receiving state even if the node is not addressed by the transmitter. This phenomenon is known as the overhearing effect. It leads to an increased of packet collisions, hence the energy consumed is grown. Indeed, to reduce the energy consumption, the number of sensors in the connectivity area of each node should be minimized. Formally, this leads to minimizing the maximum degree of connectivity graph:

\[
\min \max_{s_i \in V} \text{Deg}(s_i)
\]  

(26)

6.1.3. Minimizing the negative impact of the physical layer

As specified above, the Two-ray-ground signal propagation is assumed. The receiving signal energy depends mainly on the distance between nodes. In order to reduce the bit error rate, we propose to minimize the sensor’s separating distance. Formally, the maximum distance between sensors can be minimized as follows:

\[
\min \max_{s_i, s_j \in V} \langle \text{Dist}(s_i, s_j) \rangle
\]  

(27)

To summarize, the multi-objective WSN deployment problem can be formalized as follows:

**Problem 2:** Multi-Objective WSN deployment problem

\[
\begin{align*}
\text{minimize} & \quad \sum_{s \in A} D_s \\
\text{minimize} & \quad \max_{s \in V}(\lambda_i) \\
\text{minimize} & \quad \text{std}_{s \in V}(\lambda_i) \\
\text{minimize} & \quad \max_{s_i \in V} \text{Deg}(s_i) \\
\text{minimize} & \quad \max_{s_i \in V} \langle |s_i s_j| \rangle \\
\text{subject to} & \quad \mathcal{P}_p \supseteq \mathcal{R}_p, \forall p \in A \\
& \quad |\mathcal{P}_p| = 1 \\
& \quad D_s \in \{0, 1\}
\end{align*}
\]

It is worth noting that the problem in hand is multi-objective combinatorial optimization problem and that the objectives are contradictory. Our aim is to find the global Pareto-optimal solutions, where it is not possible to minimize one of the objectives without deteriorating at least one of the remaining objectives. Then, we will select one solution among the Pareto-optimal solutions as the best case performance.

6.2. Multi-Objective deployment algorithm – MODA

In multi-objective optimization problem, the usual meaning of the optimum makes no sense because a solution optimizing all objectives does in general not exist. The resolution seeks for a feasible solution yielding the best compromise among objectives on a set of efficient solutions (Pareto optimal). The identification of the best solution among the Pareto optimal solutions requires the preferences expressed by the user (decision maker).

To resolve the problem, we propose a combination of neighborhood and evolutionary metaheuristics. A metaheuristic refers to an iterative master strategy that guides and modifies the operations of subordinate methods by combining intelligently different concepts for exploring and exploiting the search space. A metaheuristic is a solution concept, we have to adapt it to resolve the specific problem.

An evolutionary method manages a population of solutions rather than a single feasible solution. A population based-method searches a global convergence. A neighborhood method manages one individual solution and searches a local convergence. It presents an aggressive convergence because the search is less dispersed and requires more effort in diversification in order to cover the efficient frontier completely.

We use a Multi-Objective Tabu search (MOTS) metaheuristic [32]. It is a local neighborhood search algorithm. To overcome the aggressive convergence and more diversification in search space, MOTS operates on population of solutions instead on individual solution. MODA starts with a judiciously chosen initial solution and then iteratively explores the solution space until that a predetermined number \(N_{iter}\) of iterations is reached. During each iteration,
for each solution from the population a given number of candidates solutions (neighbours) are generated. Then, we purify the whole generating solutions set by deleting the solutions that are non-Pareto or belong to the Tabu list. The latter, noted \( L_{\text{tabu}} \), contains all the visited solutions that cannot be selected again as candidate in the next \( T \) exploration stages. Once the stop criterion is reached, the algorithm returns the non-dominated solutions set.

MODA generates the initial solution and the neighborhood as described in PFDA. For instance, during an iteration we will calculate the virtual forces exerted over the deployed sensors. For each solution in the population, (i) we save it on the tabu list, (ii) we apply a virtual force for each sensor separately. Hence, for each motion we generate a new topology that we include in the population of the next iteration. When the new population is generated, we eliminate all the solutions that we find in the Tabu list \( L_{\text{tabu}} \), or the network connectivity is broken, or the solution is not efficient (dominated).

Then, in the next Tabu exploration stage, the same procedure is executed assuming this time attractive virtual forces, instead of repulsive ones. The exploration of solutions continues by alternating between attractive and repulsive forces until that the number of iterations reaches its maximum number \( N_{\text{iter}} \).

Once the MODA process is stopped, we elect one topology among the non-dominated solutions. For this purpose, we define a priority (i) high, ..., (v) low) between the objective functions: (i) maximum of degree of connectivity objective (Eq. 26), (ii) maximum of packet rate objective (Eq. 20), (iii) the standard deviation of packet rate objective (Eq. 21), (iv) the number of sensors objective (Eq. 3), finally (v) the maximum of separating distance between sensors objective (Eq. 27).

### 6.3. Performance evaluation

In order to evaluate the performance of MODA, we implemented with C++ and compared the satisfaction rate and the connectivity of our proposal with several other WSN deployment strategies. To compare the energy consumption, we simulate the generated topologies with NS2. We assume ZibBee nodes. We modified the cost link code of AODV as defined in Eq. 24. The MAC protocol is IEEE 802.15.4 non-beacon enabled mode. The physical signal propagation model is the two-ray-ground model. As mentioned in the ZigBee transceiver (chipcon CC2420) specification, we fixed the transmitted power \( p_t \) to 0 dBm (1 mW), consequently the communication range \( R_{\text{max}} \) is approximately equal to 90 m. We fixed the Receiver Threshold \( \text{RXThresh} \), Carrier Sensing Threshold \( \text{CSThresh} \), and Capture Threshold \( \text{CPThresh} \) to \( 3.98 \times 10^{-13} \), \( 3.98 \times 10^{-13} \), and 10, respectively. We fixed the transmission, reception, and idle power consumption to 52.2 mW, 59.1 mW, and 0.06 mW, respectively.

As assumed in PFDA evaluation, we calibrate the unit length to 20 m. We set \( C_{\text{max}}, x, \) and \( \beta \) for each sensor to 140 m, 1, and 1, respectively. We have considered an area of \( 50 \times 50 \) units (1 km²). We assume the same geographic distribution for required minimum event detection probability thresholds and the event appearance probabilities.

The distribution is illustrated in Fig. 3 The sink is deployed at the center of deployment field.

We run MODA and we compare it to the deployment strategies proposed in literature (Random, Grid, Max-Min-Cov, Max-Avg-Cov, Min-Miss, Diff-Deploy, DDA, EDA, and PFDA). We set the maximum number of iterations \( N \) to 500 and the tabu list size to 100.

Fig. 21 illustrates the graph of connectivity and the deployed sensors’ positions obtained by MODA approach. The number of deployed sensors is equal to 123 sensors. Fig. 21 demonstrates a clear concentration of sensors in the areas requiring high detection probabilities thresholds (top right, top left, and down right). Moreover, we can observe that the network forms a connected graph. The resulting event detection probabilities are represented in Fig. 22. We obtained a full satisfaction rate 100%, all the generated event detection probabilities are greater than thresholds.

In Table 11, we compare the number of deployed sensors, satisfaction rate, and connectivity performances of the different methods.

We notice that only our proposal MODA, Diff-Deploy, and PFDA ensure a full satisfaction rate (100%). However, Diff-Deploy does not guarantee the network connectivity. The generated topologies with MODA, PFDA, EDA, Grid, and Max-Avg-Cov are connected. However Grid and Max-Avg-Cov do not ensure a full satisfaction rate.

To compare the energy consumption (lifetime), we simulate with NS2 all the connected topologies generated by MODA, Grid, Max-Avg-Cov, EDA, and PFDA. We cannot compare MODA with the remainder strategies because the generated topologies contain more than 1 connected component. We set the initial energy of nodes to 1 Joule.

In the literature, we found many WSN lifetime definitions. Most of works consider the network is dead when the first sensor is fallen down. In this work, we consider the network is dead when no sensors could communicate with a sink. In Fig. 23, we show the percentage of sensors

![Fig. 21. MODA – network topology.](image-url)
which can communicate with the sink. We remark just after few seconds all the deployment strategies, except our proposal, the percentage of sensors falls from 100% to 0%. Max-Avg-Cov after 7.10 s, the percentage falls from 61.47% to 0%. However with our method, the percentage of sensors falls to 0% after 152 s. During this period, at least 36.88% of sensors can communicate with a sink. We notice that MODA improves widely the network lifetime compared to the related works (from 7.10 s to 152 s).

The results obtained in Fig. 23 are the consequent of the expensive energy consumption of sensors near to the sink. This nodes forward all the packets sent from the rest of sensors in the network to the sink. Hence, the near sink's sensors fall first. We compare the residual energy of sensors located at 1, 2, 3, and 4 hops from the sink. The plots in Fig. 24(a) and (b) illustrate that all methods except MODA, sensors located at 1 and 2 hops from the sink exhaust their energy quickly. Consequently, the number of sensors, which can communicate with the sink, deployed by MODA strategy is larger compared to the related strategies. We explain the best results of MODA by the fact that the maximum node's packet load forwarded (Eq. 20) is considered in the problem formalization. We remark when sensors are far (3 and 4 hops) from the sink (Figs. 24(c) and (d)), nodes consume less energy. The obtained residual energy with our method is, approximately, equal to the related strategies (PPDA, Max-Avg-Cov).

<table>
<thead>
<tr>
<th>Deployment strategy</th>
<th>Number of sensors</th>
<th>Satisfaction rate %</th>
<th>Connectivity</th>
</tr>
</thead>
<tbody>
<tr>
<td>MODA</td>
<td>123</td>
<td>100</td>
<td>ok</td>
</tr>
<tr>
<td>PPDA</td>
<td>121</td>
<td>100</td>
<td>ok</td>
</tr>
<tr>
<td>BDA</td>
<td>123</td>
<td>99.66 ± 0.07</td>
<td>ok</td>
</tr>
<tr>
<td>Diff-Deploy</td>
<td>119</td>
<td>100</td>
<td>–</td>
</tr>
<tr>
<td>DDA</td>
<td>123</td>
<td>85.60</td>
<td>–</td>
</tr>
<tr>
<td>Min-Miss</td>
<td>123</td>
<td>79.48</td>
<td>–</td>
</tr>
<tr>
<td>Max-Min-Cov</td>
<td>123</td>
<td>77.48</td>
<td>ok</td>
</tr>
<tr>
<td>Max-Avg-Cov</td>
<td>123</td>
<td>81.44</td>
<td>ok</td>
</tr>
<tr>
<td>Grid</td>
<td>123</td>
<td>75.93 ± 1.32</td>
<td>–</td>
</tr>
<tr>
<td>Random-Uniform</td>
<td>123</td>
<td>81.58 ± 1.62</td>
<td>–</td>
</tr>
</tbody>
</table>

Fig. 22. MODA – generated event detection probabilities.

Fig. 23. MODA – percentage of sensors connected to the sink.
7. Conclusion

We provided in this paper a new differentiated deployment wireless sensor network algorithm, named PFDA, inspired from the artificial potential field used in robotic. We assumed that a sensor event detection model is probabilistic and the sensed events present a geographical irregularity. The obtained results show that our differentiated deployment strategy PFDA outperforms the related approaches found in the literature.

Then, we proposed an extension of PFDA, named multi-objective deployment algorithm (MODA), to maximize network lifetime and at the same time keeping the cost of deployment, QoM and network connectivity objectives. The proposed strategy is based on multi-objectives Tabu search metaheuristic and artificial potential field. We assumed the use of ZigBee-based nodes and the protocol stack is simulated with NS2 simulator. The obtained results show that MODA outperforms the existing strategies. The network lifetime is maximized, the required quality of monitoring and network connectivity are guaranteed, and the cost of deployment is minimized.

References
