

# Optimization of sensor deployment using multi-objective evolutionary algorithms

Arouna Ndam Njoya<sup>1</sup> · Wahabou Abdou<sup>2</sup> · Albert Dipanda<sup>2</sup> · Emmanuel Tonye<sup>3</sup>

Received: 8 June 2016 / Accepted: 3 October 2016 / Published online: 18 October 2016  
© Springer International Publishing Switzerland 2016

**Abstract** Many designs of wireless sensor network applications require the determination of the optimal locations of sensor nodes to be placed in a sensor field. Coverage enables us to evaluate the supervision quality of each point within an area of interest. In this paper, we address the problem of target coverage in wireless sensor networks. This concern is trivial if each target must be covered by a single sensor. However, it becomes an NP-complete problem when the choice of the position of the sensor must take into account the targets that it should cover in its vicinity. Using a multi-objective evolutionary-based approach, we propose a stochastic method to search for network configurations that achieve good coverage with the fewest sensors. A comparative experimental study of the model with well-known multi-objective algorithms such as NSGA-II, SPEA2, SMSEMOA and MOEA/D indicate that NSGA-II performs better than others on most of the test instances.

**Keywords** Sensor deployment · Target coverage · Multi-objective evolutionary algorithms

✉ Arouna Ndam Njoya  
ndanjoa@gmail.com

Wahabou Abdou  
wahabou.abdou@u-bourgogne.fr

Albert Dipanda  
albert.dipanda@u-bourgogne.fr

Emmanuel Tonye  
tonyee@hotmail.com

<sup>1</sup> University of Ngaoundéré, IUT, PO box 455, Ngaoundéré, Cameroon

<sup>2</sup> LE2I UMR6306, CNRS, Arts et Métiers, Univ. Bourgogne Franche-Comté, 21000 Dijon, France

<sup>3</sup> University of Yaoundé I, PO box 337, Yaoundé, Cameroon

## 1 Introduction

Wireless sensor networks are one of the technologies developed to address the growing needs of the observation and control of certain physical and biological phenomena of the environment. Originally, the expansion of wireless sensor network applications is conducted by military applications [1]. Nowadays, sensor networks are used in a wide variety of applications including medicine, environmental risk monitoring, traffic control, habitat monitoring and control of industrial processes [2–6]. A sensor is defined as a device capable of transforming a physico-chemical quantity to a measurable electrical signal [2]. Sensors are characterized by a low processing capacity and data storage, a limited battery, and a limited coverage range and communication range. Therefore, the implementation of wireless sensor networks (WSN) applications like urban air quality monitoring in our case requires the deployment of hundreds, even thousands, of source nodes, relay nodes and collector nodes for a finer spatial granularity measurement of pollutants. The heterogeneity of nodes enables us to segment the network into subnets around the collector nodes (e.g., cluster head) for data collection within each monitoring area. This subdivision offers the possibility of limiting the communications links between the nodes and consequently reducing network energy consumption and improving the quality of service [7–9]. Thus, WSN designs take the form of large complex networks, consisting of spatial distributions of sensor nodes that cooperate with each other by radio message for the observation and the controlling of certain physical phenomena.

In general, there are three types of sensor deployment scenarios, namely target coverage, area coverage and barrier coverage [2–4, 6]. Target coverage focuses on how to cover the set of points defined into the monitoring region. Area

coverage seeks to find a minimum number of sensors capable of completely covering the area of interest. Instead of fully covering the space, barrier coverage finds the way to place the sensors along a boundary defined by two parallel curves to limit intrusions into the surveillance zone.

Depending on the objectives and the implementation step of the network, coverage problem can be formulated in several manner. When designing the network, coverage can be built around the determination of the minimum number of sensors required to cover the entire region, for example [3]. In the deployment phase, according to the accessibility of the region (hostile or not), the interest may be focused on random or deterministic deployment. Once the network is deployed, the cover problem can be formulated in terms of network lifetime extension, connectivity and quality of service [4, 10, 11].

Most of the work on network lifetime extension focuses on finding the maximum number of cover sets (disjoint or not) and their scheduling once sensors are randomly deployed [12–17]. However, by knowing the position of the target and using the deterministic deployment approach, the extension of the network lifetime can be taken into account in the step of sensor nodes placement and the scheduling phase [18].

In contrast to the sequential approach proposed in [18] for target coverage, we introduce in [19] an approach that aims to optimize the network lifetime by finding simultaneously the “better” positions of the sensor nodes and the maximum number of disjoint cover sets. This case of study includes two main processes: (1) placing sensors around the targets, (2) generating the maximum number of disjoint cover sets to lengthen the network lifetime. To tackle this, we designed a chromosome in which the genes contain both the position and the identifier of the sensor owning group. The proposed approach denoted as Combined Approach for Sensor Deployment and Disjoint Cover sets problems (CASEDISC) outperformed existing methods on most well-known test instances. We also observed that efficient networks with extended lifetime can be created by finding a single optimized cover and placing redundant sensors at each of the indicated sensor locations. Thus, the objective of this paper is to design a method to obtain covers with fewer sensors and better coverage. The proposed method computes the number of “free sensors”, as introduced in [19], and the number of “uncovered targets” to reduce sensor number while maintaining good coverage.

Given that the questions raised in the present study involve trade-offs between the number of uncovered targets and the number of free sensors, the objectives set can be tackled by using multi-objective optimization techniques. Without loss of generality, suppose that  $x_1$  and  $x_2$  are two solutions of a multi-objective minimization problem with  $l$  objectives. The solution  $x_2$  is dominated by  $x_1$ , denoted by  $x_1 < x_2$ , if  $\forall i \in$

$\{1, \dots, l\}$ ,  $f_i(x_1) \leq f_i(x_2) \wedge \exists i \in \{1, \dots, l\}$ ,  $f_i(x_1) < f_i(x_2)$ . The set of non-dominated solutions forms the Pareto front. If the solutions in the front dominate all others in the search space, this front is called Pareto-optimal or globally Pareto-optimal set [20].

This paper proposes a method based on a multi-objective evolutionary algorithm. To achieve the objectives, namely minimizing the number of sensors while maintaining good target coverage, a representation scheme is used where chromosomes are vectors which give  $(x, y)$  coordinates of multiple sensors.

The rest of the article is organized as follows. In Sect. 2, we briefly present the related work on sensor deployment. In Sect. 3, the problem definition is provided, and the proposed approach is detailed. The simulation results and discussions are presented in Sect. 4. Finally, Sect. 5 concludes this paper and presents some future work.

## 2 Related work

Several researchers have proposed solutions for the sensor placement problem for target monitoring in various situations. This problem is a kind of classical set-covering problem which is known to be NP-hard [2]. To achieve this, in most cases assumptions are made about the sensor coverage model, the obstacles, and the number of targets and sensors to be used. Thereafter, greedy algorithms and approximation algorithms are used to solve the problem. For example, in [21], the authors assume that the detection of targets can be influenced by the nature of the terrain and the distance. They propose greedy algorithms to find a minimum number of sensors that can cover all targets with a minimum degree of confidence within a grid. In [22], a simulated annealing algorithm is used for sensors placement within a grid to minimize the error of maximum distances between target monitored by the same sensors under the constraint of cost. Some genetic algorithms have also been used to cope with the sensors placement problem [3, 23–30]. In [30], a genetic algorithm is proposed to find the position and the minimum number of mobile sensors that has to be deployed to fill the holes of coverage generated by the random deployment of static sensors. In [3], three sensors with different sensing radii are used to solve the area coverage problem by applying genetic algorithms. Each solution (chromosome) is encoded as a vector that contains the sensors’ geographic coordinates (in a plane). The solution found by the algorithm is improved by applying a local search algorithm using a modified version of the virtual force algorithm outlined in [31].

In addition to the above methods, some genetic frameworks have also been used to find the cover sets. A multi-objective genetic algorithm NSGA-II is used in [16] for area coverage. The area to cover is discretized into a grid of size

$m \times n$  and the chromosome is encoded as a binary string of size  $n$  that represents the number of sensors. Two objective functions are defined. The first one maximizes the coverage rate of the constructed cover set and the second one seeks to minimize the rate or the number of sensors used in the same cover set.

Unlike the previous works, the main contributions of this paper can be summarized as follows: (1) the definition of two antagonist objective functions: minimizing the number of uncovered target and maximizing the number of free (unused) sensors. These sensors can be used to increase the redundancy and lengthen the network lifetime; (2) the comparison of results provided by different multi-objective algorithms in the context of sensor deployment.

### 3 The proposed sensor deployment method

#### 3.1 Problem description

The work carried out in this paper focuses on two main objectives: (1) to compute the maximum number of “free sensors” and (2) to determine the minimum number of uncovered targets.

There exists several abstract models to represent sensor networks, especially unit disc graph and statistical channel models [32].

The unit disc graph model (UDG) represents a wireless sensor network as an intersection graph of equal size circle. Each circle corresponds to a sensor that is located at the center. The radius of the circle corresponds to the coverage range of the sensor. Each sensor can monitor targets that are inside the unit disc. The coverage function is defined in Eq. 1 [3].

$$f(s, t) = \begin{cases} 1 & \text{if } d(s, t) \leq r_s \\ 0 & \text{otherwise} \end{cases}, \tag{1}$$

where  $d(s, t)$  is the Euclidian distance between a sensor  $s$  and a target  $t$ .

Statistical channel models take into account non-deterministic characteristics of wireless communications. They consider that the coverage area of sensors is not uniform; it varies depending on obstacles, interference, signal fading, etc. They introduce the concept of transition regions where  $f(s, t)$  (Eq. 1) takes values form 0 to 1.

In the work presented in this paper, we use the UDG model. This model is known to be extremely useful for simplifying the analysis of large graphs [32]. Our work has as its final goal to be extended to the deployment of sensors for monitoring pollution in big cities.

We formulate sensor deployment for target coverage in the following manner. There are  $m$  targets  $T = \{t_j, j \in \{1, \dots,$

$m\} | t_j$  is the target indexed by  $j\}$  where each target has a definite location within a given square region. The objective is to find the locations of  $n$  ( $n \leq m$ ) sensor nodes  $S = \{s_i, i \in \{1, \dots, n\} | s_i$  is the sensor indexed by  $i\}$  in such a way that the number of sensors can be minimized and each target is monitored by at least one sensor node.

To extend the network lifetime for target coverage, the technique usually used is to deploy a large number of sensors and activate redundant sensors at different times, so that the maximum number of cover sets (disjoint or not) can be formed. Alternating the sleep and awake phases leads to save energy consumption. Thus, in this work, we try to favor the redundancy of sensors to cope with the energy preservation.

#### 3.2 Problem formulation

Sensors placement problem is NP-hard. This has been pointed out in [2,3]. To tackle it, a good identification of parameters among a lot that exist need to be done to reduce the size of the problem. Thus, we need to select parameters able to perform both efficient sensor deployment and network lifetime extension. Because of the complexness of the problem, approximation algorithms need to be used for the resolution. Genetic algorithms are part of these algorithms. Their implementation requires to design objective functions. The objective functions allows to evaluate the quality of solutions with respect to the considered problem. To set up the objective functions of the model, we include the following definitions:

- $I$  a possible solution of the problem.
- $\delta_{ij}$  denotes the variable that equals to 1, if the sensor  $s_i$  covers the target  $t_j$  and 0 otherwise.
- $fsensors(I)$  the number of free sensors in the solution  $I$ . A sensor is considered as free if it does not cover any target.
- $uncovered(I)$  the number of uncovered targets in the solution  $I$ .
- $\theta(s_i)$  denotes the variable that equals to 1, if  $s_i$  does not cover any target, and 0 otherwise.

The objective functions used by the genetic algorithm to solve the problem are given by Eq. (2):

$$\begin{cases} f_0 : uncovered(I) & \text{to minimize} \\ f_1 : fsensors(I) & \text{to maximize} \end{cases}, \tag{2}$$

where

$$uncovered(I) = m - \sum_{j=1}^m \delta_{ij}, \tag{3}$$

$$f_{\text{sensors}}(I) = \sum_{i=1}^n \theta(s_i). \tag{4}$$

$f_0$  defines the number of uncovered targets in the solution  $I$ , and  $f_1$  the number of free sensors.

### 3.3 Pareto front analysis

For our optimization problem, the expected shape of the Pareto front is illustrated in Fig. 1.  $X_A$  and  $X_C$  are extreme solutions. Solution  $X_A$  provides complete coverage (since  $f_0(X_A) = 0$ ), but few free sensors. Solution  $X_C$  as all free sensors (hence, no sensor in the cover) and no covered targets. The solution  $X_B$  is a balance between the number of free sensors and the number of uncovered targets.

### 3.4 Evolutionary approach

#### 3.4.1 Evolutionary framework

Evolutionary algorithms are stochastic optimization methods that are based on the theory of natural evolution proposed by Charles Darwin. This theory relies on the idea that individuals who best fit their environment have more chance to survive and to reproduce. Through the reproduction process, parents transmit their characters to their children. This theory is transposed in the optimization field. Individuals are represented by solutions and the environment is replaced by the problem definition. At the beginning of the optimization process, a solution set (population) is generated. These solutions are evaluated using some objective functions to determine their ability to solve the problem. Thereafter, some solutions are selected for the recombination step. This selection is based on the quality of the solutions: good solutions will have greater probability of being selected. Thus, good characters (or genes) will be transmitted from parents to offspring (new solutions). The main reproduction steps are crossover and

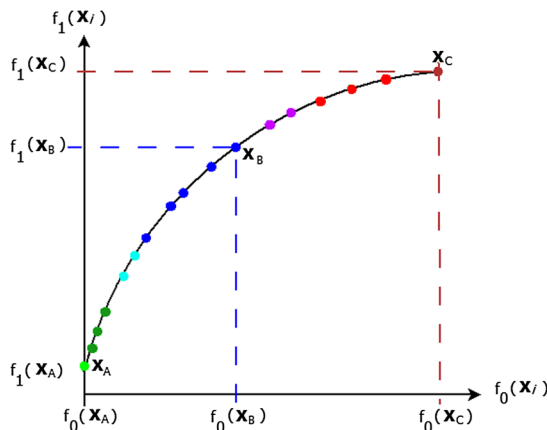


Fig. 1 Expected shape of the Pareto front

mutation. Finally, a replacement strategy allows to regulate the population size. The goal here is to choose among parents and offspring who will survive. All these steps are repeated over generations until a stop criterion is met (for example, a number of evaluations or a convergence metric).

Since the main goal of this paper is the sensor deployment, it does not intend to introduce a new evolutionary algorithm. Hence, the well-known multi-objective algorithms NSGA-II, SPEA2, SMSEMOA, MOEA/D [33–36] are used, because they give good results in various types of optimization problems, and especially when dealing with multiple objective functions

Applying an evolutionary algorithm to solve a given problem requires firstly to design a chromosome representing the solution of the problem.

#### 3.4.2 Chromosome encoding

The topic addressed in this paper aims to find the good positions of sensor nodes. To achieve this goal, we consider that each sensor  $s_i$  of a solution  $I$  is characterized by two information: its coordinates  $(x_i, y_i)$  in the plan. Therefore, we design a solution (chromosome) as a vector of integers of size  $2 \times m$ , where  $m$  represents the initial number of targets. The genes representing the coordinates of sensor  $s_i$  take the values between 0 and the length of side of the square region defining the sensor field. An example of the structure of chromosome is shown in Fig. 2 where we have five sensors. The coordinates  $(x_1, y_1)$  of sensor  $s_1$  are expressed by the couple (89, 69) for example.

#### 3.4.3 Population initialization

The initialization phase involves generating initial solutions, which can be done by a dedicated heuristic or by randomly assigning values to the genes of the chromosome. In the initialization phase, for a population of size  $N$ , we randomly generate  $N - 1$  solutions (they may or may not cover all the targets). Thereafter, we generate a single solution that guarantees the coverage of all the targets. We empirically observed that adding this solution improves the exploration of the search space.

89	69	20	19	122	25	22	145	23	52
$s_1$		$s_2$		$s_3$		$s_4$		$s_5$	

Fig. 2 Chromosome encoding

**Table 1** Test instances

Instance	Number of sensors	Sensor field size	Number of targets
S1-25	25	500 × 500	25
S2-25	25	1000 × 1000	25
S1-50	50	1000 × 1000	50
S2-50	50	1500 × 1500	50
S1-100	100	5000 × 5000	100
S1-150	150	5000 × 5000	150
S1-200	200	5000 × 5000	200

### 3.5 Test data set

Since the minimum number of sensors can give the information about the network lifetime extension and the investment cost, we investigated here how to calculate it. For this purpose, we used four types of square regions as sensor fields. The sensing radius is equal to 75 m. Seven random test instances given in Table 1 were generated. For each test instance, the algorithms are executed 30 times for each target locations. At each execution, the number of uncovered targets and the number of free sensors are saved. The monitoring area is a square region that varies from 500 to 5000 m. In the first group of the test instance, the number of targets is 25, the number of sensors is 25 and the region varies from 500 to 1000 m with increments 500. The second group of the test instance uses 50 targets, 50 sensors, and the region that varies from 1000 to 1500 m with increments 500. In the third group of the test instance, the region is 5000 m × 5000 m (25 km<sup>2</sup>), and the number of targets (receptively, the number of sensors) varies from 100 to 200 with increments 50.

## 4 Experiments

In this section, we describe the experiments we conducted to assess the validation and the efficiency of the proposed model. First, we show the optimal results obtained with the best multi-objective algorithms, i.e., NSGAI. Second, we detail the comparison results between methods. And third, we present the results provided by indicators such as inverted generational distance (IGD), spread and hypervolume. The parameters of multi-objective algorithms used in the simulation are summarized in Table 2.

### 4.1 Performance indicators

The indicators we used to compare the set of solutions of different algorithms are described in this section. Let  $l$  be the number of objectives; let us set  $P^*$ ,  $P$  to be, respectively, the

**Table 2** Parameters of NSGAI, SPEA2, SMSEMOA, MOEA/D

NSGAI, SPEA2, and SMSEMOA parameters	
Crossover:	simulated binary crossover (SBX) with probability:0.9, distribution:20
Mutation:	polynomial mutation, with probability: $\frac{1.0}{(2 \times m)}$ , distribution: 20
Selection:	binary tournament
Number of evaluations:	100,000. Population Size:100
SPEA2 Archive size:	100
SMSEMOA Offset size:	100
MOEA/D parameters	
Crossover:	differential evolution crossover with probability:0.9, distribution:20
Mutation:	polynomial mutation, with probability: $\frac{1.0}{(2 \times m)}$ , distribution: 20
Number of evaluations:	150,000. Population size:300
Independent runs number:	30

Pareto-optimal and the Pareto front returned by an algorithm. Let  $x_i$  be the solution of the problem.

The Inverted generational distance (IGD) [37] is an indicator of convergence. The best algorithm is the one associated with the smallest value of IGD. The IGD computes the lowest distance from every point of  $P^*$  toward each point of  $P$ , as follows:

$$IGD = \frac{\left(\sum_{i=1}^{|P^*|} d_l^*(\vec{x}_i^*)\right)^{1/l}}{|P^*|}, \tag{5}$$

where

$$d_l^*(\vec{x}_i^*) = \min_{j=1}^{|P|} \sqrt{\sum_{k=1}^l (f_k(\vec{x}_i^*) - f_k(\vec{x}_j))^2}. \tag{6}$$

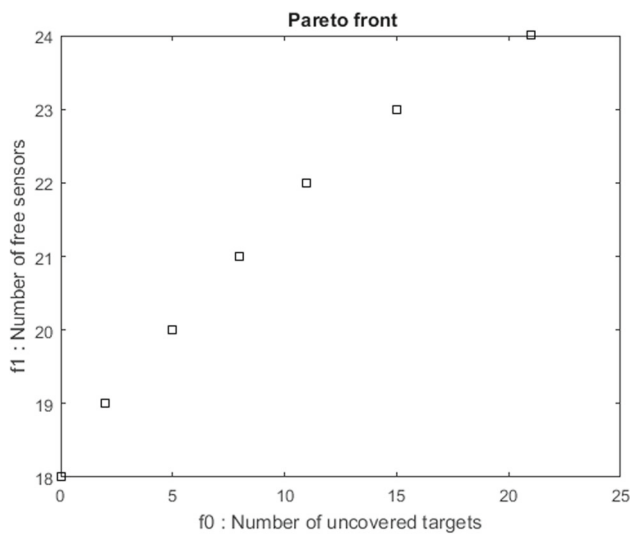
The distribution of the solutions into the Pareto front is evaluated by the spread. The spread uses the minimal distance between two consecutive solutions to assess the distribution of the solutions in the search space [37]. The best algorithm in terms of spread is the one associated with the lowest value of spread. Equation (7) gives the expression for the spread.

$$S = \sqrt{\frac{1}{|P|} \sum_{i=1}^{|P|} (D_i - \bar{D})}, \quad \bar{D} \text{ is the mean of } D_i, \tag{7}$$

where

$$D_i = \min_{j \in P \wedge i \neq j} \left( \sum_{k=1}^l |f_k(\vec{x}_i) - f_k(\vec{x}_j)| \right). \tag{8}$$

To measure the volume covered by each point of  $P$  in the objective space, we choose a reference point  $\vec{W}$  and define



**Fig. 3** Pareto Front for test instance S1-25

$v_i$  to be the rectangular region whose diagonal is given by the segment  $[\vec{x}_i, \vec{W}]$ . The hypervolume indicator is defined as the volume of the union of  $v_i$  for all  $i$  in  $P$  [37]:

$$HV = \text{volume} \left( \bigcup_{i=1}^{|P|} v_i \right). \tag{9}$$

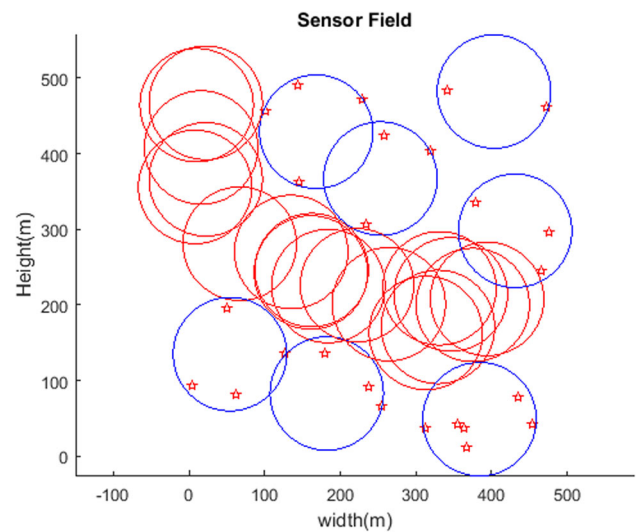
The hypervolume indicator measures both the convergence and the diversity. A bigger hypervolume indicates a better algorithm.

### 4.2 Validation tests

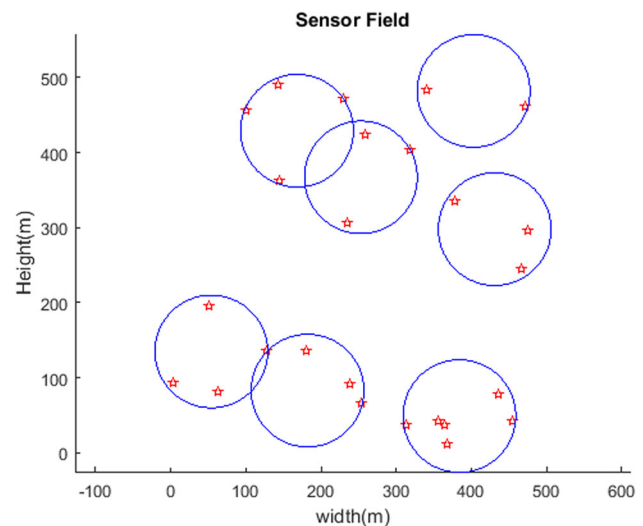
For all figures of sensor deployment presented in this section, the red stars indicate targets deployed into the monitored area. Free sensors and sensors included in the cover solution are represented by red and blue circles, respectively.

#### 4.2.1 Simulation for test instance S1-25

Figure 3 shows the Pareto front of the present test instance where each solution is a couple of the number of uncovered targets and the number of free sensors. To illustrate the distribution of the sensor nodes in the sensor field, among the solutions in the Pareto front, we take the one which has  $f_0 = 0.0$  and  $f_1 = 18$  in this case. Sensor deployment with free sensors and without free sensors is shown in Figs. 4 and 5, respectively. Figure 4 clearly shows that the position assigned to the sensor takes into account the targets within its vicinity. We can also observe that we need only seven sensors to cover all targets. Thus, because the initial number of sensors was 25, we can say that the number of disjoint cover sets can be 3.



**Fig. 4** Sensor deployment for test instance S1-25 with 18 free sensors



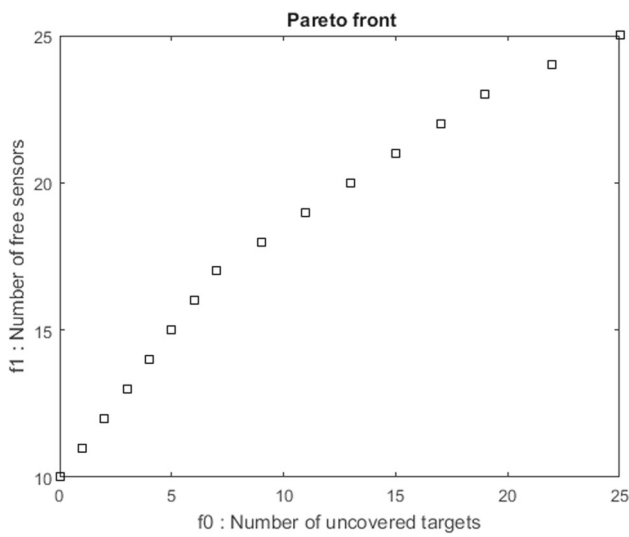
**Fig. 5** Sensor deployment for test instance S1-25

#### 4.2.2 Simulation for test instance S2-25

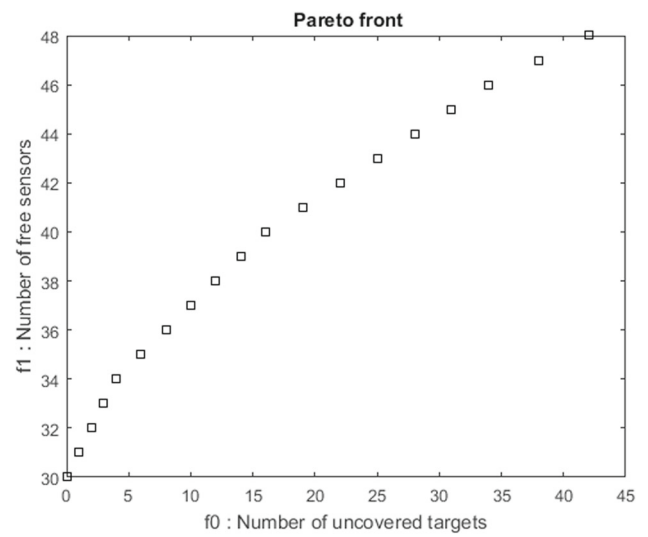
The Pareto distribution of S2-25 is presented in Fig. 6. The minimum number of sensor nodes returned is 15. This value is obtained by selecting the solution in the Pareto front where the value of  $f_0 = 0.0$ , i.e.,  $f_0 = 0.0$  and  $f_1 = 15$ . Figures 7 and 8 illustrate, respectively, sensor deployment with free sensors and without free sensors. For this test instance, the number of cover sets can be 1.

#### 4.2.3 Simulation for test instance S1-50

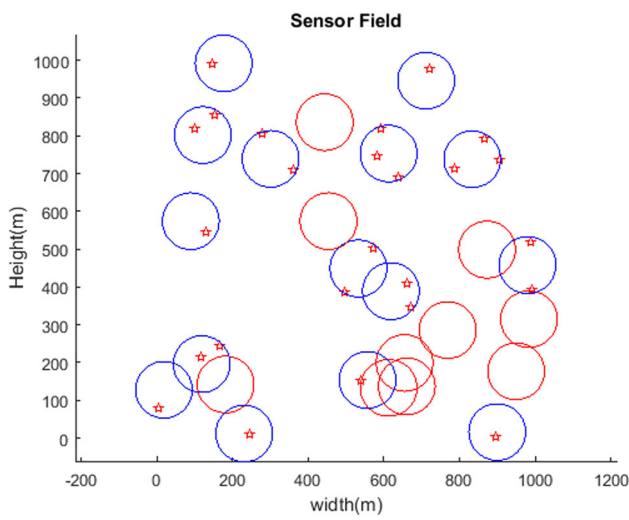
The Pareto front and sensor deployment obtained are given in Figs. 9 and 11, respectively. Figure 10 shows the sensor deployment with 30 free sensors. The minimum number of



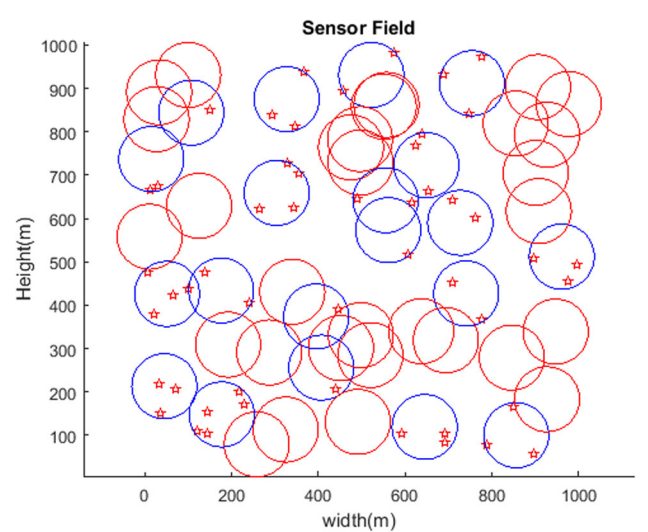
**Fig. 6** Pareto Front for test instance S2-25



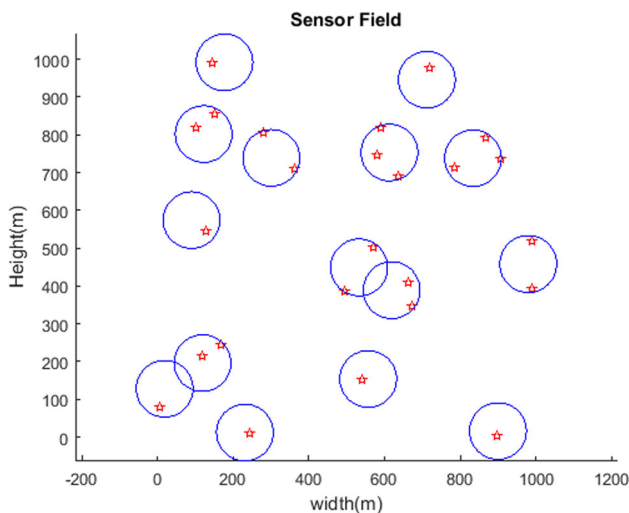
**Fig. 9** Pareto Front for test instance S1-50



**Fig. 7** Sensor deployment for test instance S2-25 with ten free sensors



**Fig. 10** Sensor deployment for test instance S1-50 with 30 free sensors



**Fig. 8** Sensor deployment for test instance S2-25

sensor nodes obtained is 20. As the initial number of sensors is 50, two disjoint cover sets can be found.

*Simulation for test instance S2-50*

In Fig. 12, we present the Pareto front. Figures 13 and 14 are dedicated for sensor deployment. The optimal number of sensor nodes found is 29. Thus, only one disjoint cover set can be generated from the initial sensors.

*4.2.4 Simulation for test instance S1-100, S1-150, and S1-200*

When the number of target is equal to 100, 150 and 200, the illustration of sensor deployment is no longer visible. From the Pareto front in Figs. 15, 16 and 17, we observed that the

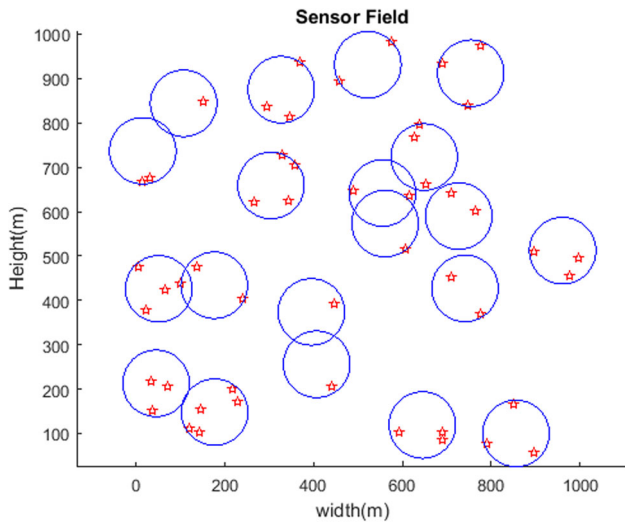


Fig. 11 Sensor deployment for test instance S1-50

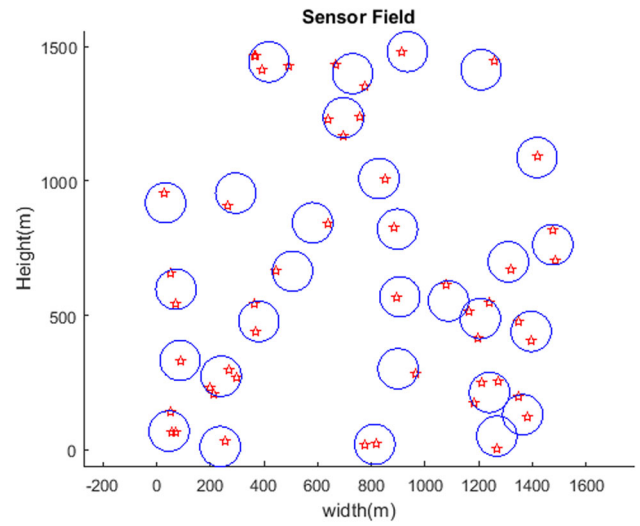


Fig. 14 Sensor deployment for test instance S2-50

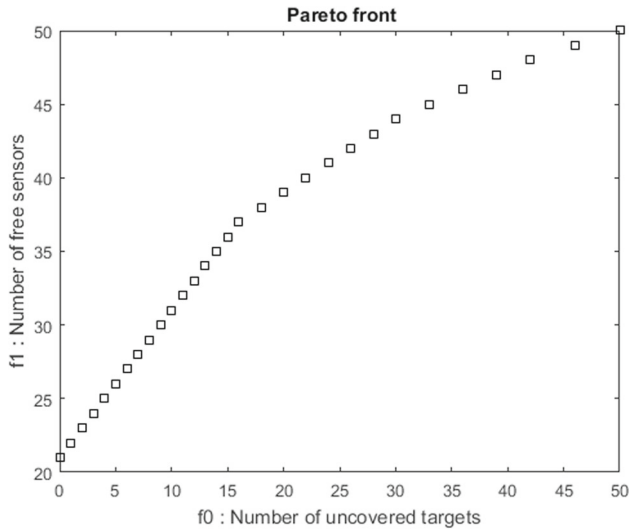


Fig. 12 Pareto Front for test instance S2-50

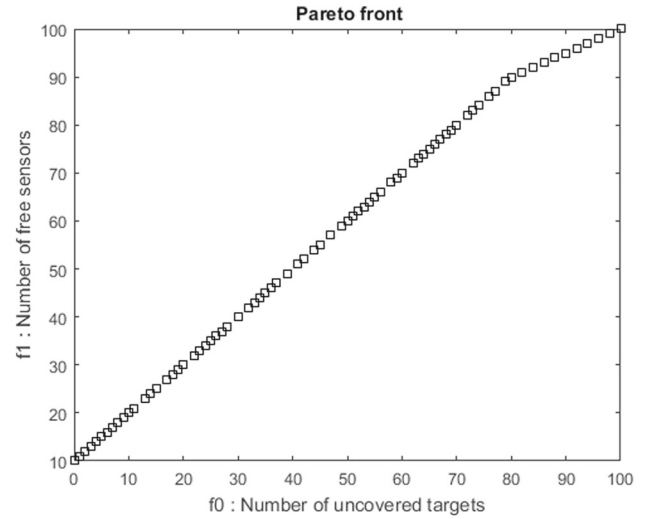


Fig. 15 Pareto Front for test instance S1-100

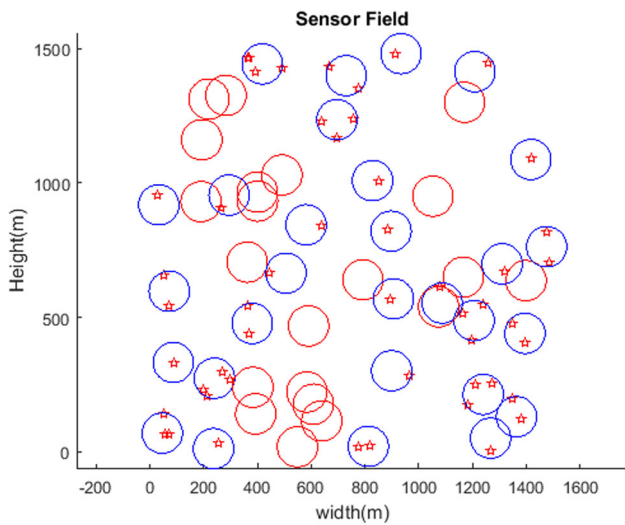


Fig. 13 Sensor deployment for test instance S2-50 with 21 free sensors

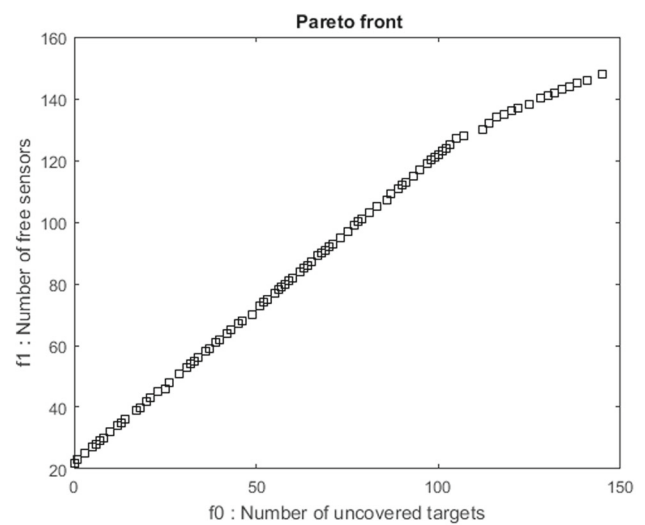
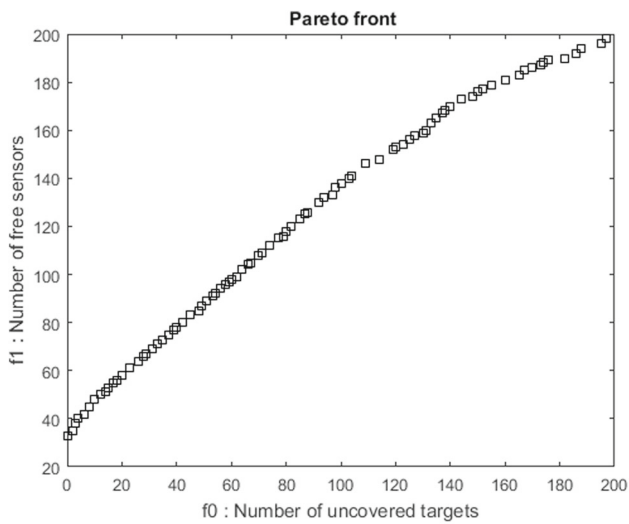
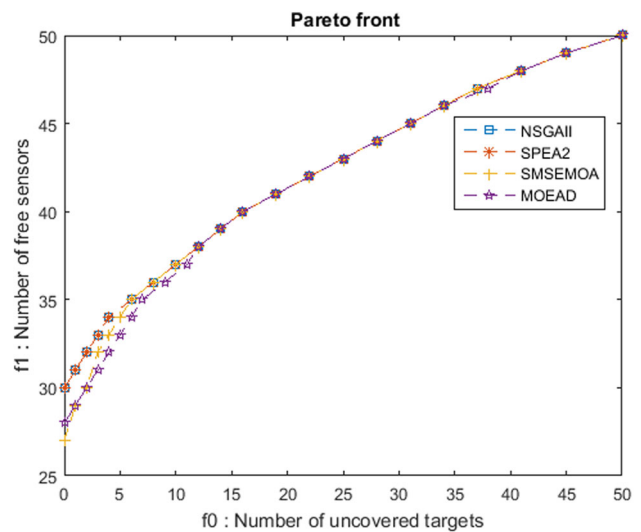


Fig. 16 Pareto Front for test instance S1-150

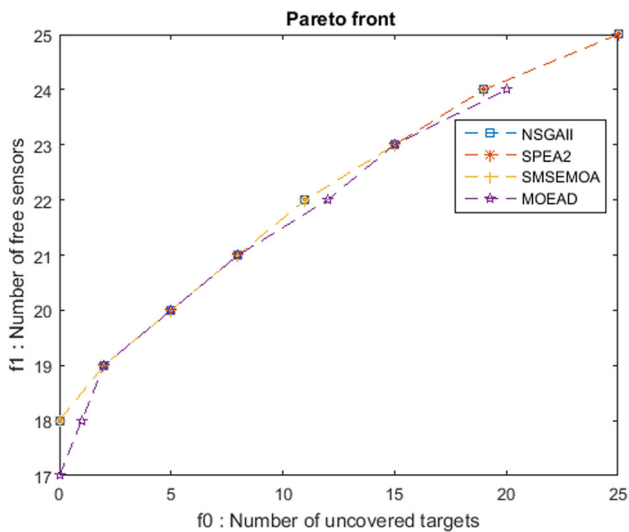




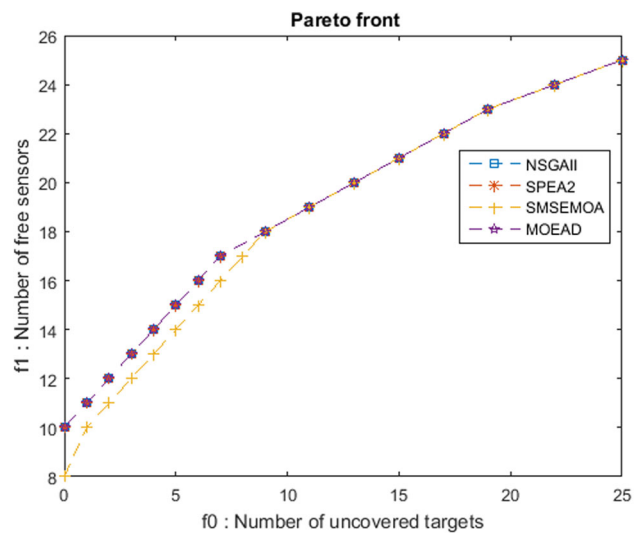
**Fig. 17** Pareto Front for the test instance S1-200



**Fig. 19** Pareto front obtained by NSGAI, SPEA2, SMSEMOA and MOEA/D for test instance S1-50



**Fig. 18** Pareto front obtained by NSGAI, SPEA2, SMSEMOA and MOEA/D for test instance S1-25



**Fig. 20** Pareto front obtained by NSGAI, SPEA2, SMSEMOA and MOEA/D for test instance S2-25

minimum number of sensors needed are 90, 128 and 167 for test instances S1-100, S1-150 and S1-200, respectively.

### 4.3 Comparative study of Pareto fronts

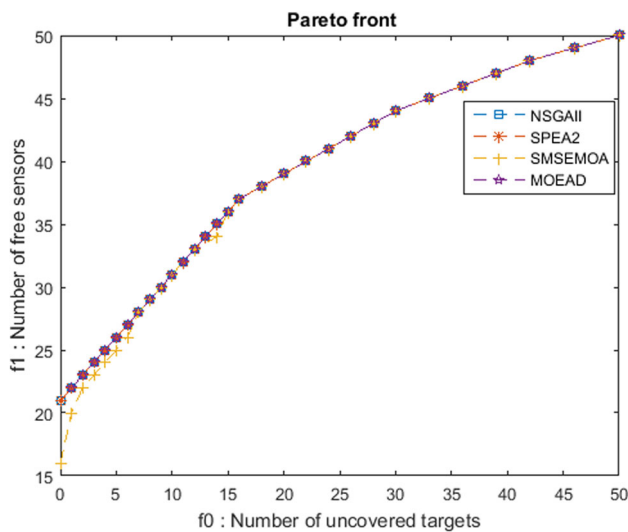
This section compares the performance of multi-objective algorithms NSGAI [33], SPEA2 [34], SMSEMOA [35] and MOEA/D [36]. The seven test instances are those presented previously in Table 1.

Figures 18, 19, 20, 21, 22, 23 and 24 show that NSGAI always gives the best Pareto front for all test instances. The performance of NSGAI is nearly matched by SPEA2 in almost all cases. For test instances S1-100 to S1-200, Figs. 22, 23, and 24 show that MOEA/D gives the poorest Pareto front.

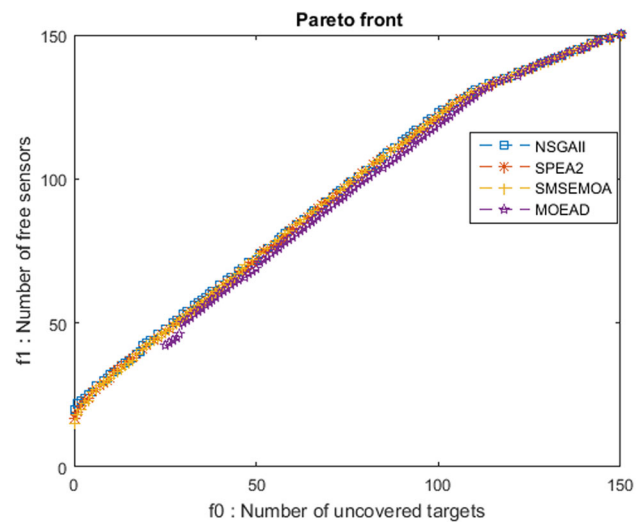
### 4.4 Indicators study

In this section, we use IGD, spread and hypervolume indicators to compare the performance of NSGAI, SPEA2, SMSEMOA and MOEA/D. We keep the same test instances and algorithm parameters presented previously.

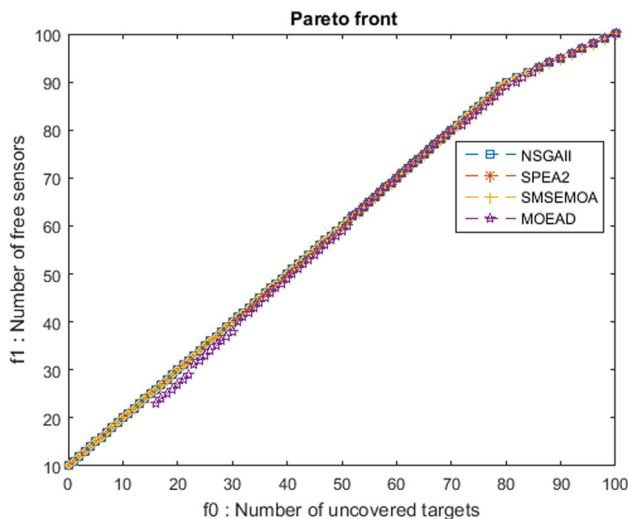
Tables 3, 4 and 5 compare means and standard deviations for IGD, spread and hypervolume, respectively. Except for test instance S1-100, Table 3 illustrates that for over all test instances, NSGAI achieves lower standard deviations and means in IGD than SPEA2, SMSEMOA and MOEA/D. In Table 4, the best values of spread are provided by SMSEMOA followed by SPEA2. Table 5 indicates that NSGAI provides the best values of hypervolume on most test instances.



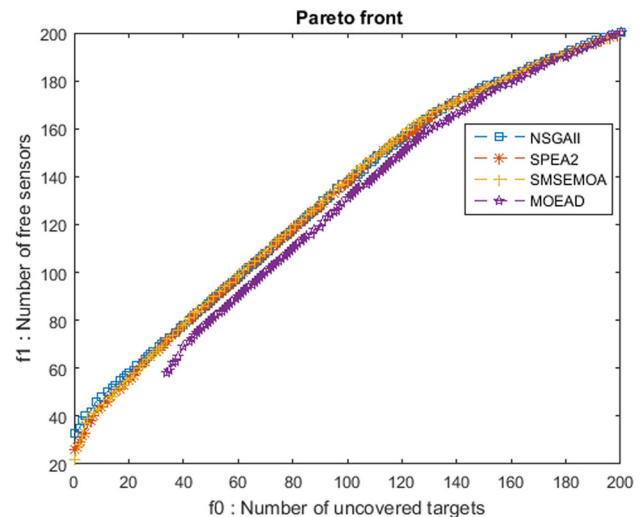
**Fig. 21** Pareto front obtained by NSGAI, SPEA2, SMSEMOA and MOEA/D for test instance S2-50



**Fig. 23** Pareto front obtained by NSGAI, SPEA2, SMSEMOA and MOEA/D for test instance S1-150



**Fig. 22** Pareto front obtained by NSGAI, SPEA2, SMSEMOA and MOEA/D for test instance S1-100



**Fig. 24** Pareto front obtained by NSGAI, SPEA2, SMSEMOA and MOEA/D for test instance S1-200

The simulation results can be summarized as follows:

- For cases with few targets, all algorithms find solutions which cover all targets. This also occurs when targets are close to each other.
- When the region size grows, the density of targets decreases and thus the number of sensors required to maintain full coverage of targets grows (see Figs. 5, 8, 11 and 14).
- Statistical analysis from Tables 3, 4 and 5 shows that NSGAI provides better results in comparison to SPEA2, SMSEMOA and MOEA/D. IGD and hypervolume indicator results support this conclusion.

## 5 Conclusion

In this paper, we optimize the locations of sensor nodes for target detection using multi-objective evolutionary algorithms such that the number of sensors is minimized while maintaining good target coverage. We have formulated the sensor deployment problem as a multi-objective optimization problem which includes two main objectives: (1) maximize the number of free sensors and (2) minimize the number of uncovered targets. Computational results were used to validate the proposed model. The comparative studies between various multi-objective evolutionary algorithms show that NSGAI outperforms other algorithms and can converge to the approximate Pareto front. Our future work will improve the proposed model and extend it to solve the problem raised

**Table 3** IGD. Mean and standard deviation study between NSGAI, SPEA2, SMSEMOA and MOEA/D

Test instances	NSGAI		SPEA2		SMSEMOA		MOEA/D	
	Avg.	Stdev.	Avg.	Stdev.	Avg.	Stdev.	Avg.	Stdev.
S1-25	1.27e-03	8.8e-04	<b>9.70e-04</b>	<b>6.9e-04</b>	4.89e-03	2.3e-03	2.48e-03	7.7e-04
S2-25	<b>1.08e-04</b>	<b>9.3e-05</b>	1.16e-04	1.0e-04	4.08e-04	1.2e-04	2.29e-04	1.3e-04
S1-50	6.97e-04	3.5e-04	8.73e-04	4.5e-04	2.14e-03	1.2e-03	<b>5.19e-04</b>	<b>1.2e-04</b>
S2-50	<b>1.73e-04</b>	<b>9.6e-05</b>	2.38e-04	1.0e-04	4.38e-04	2.2e-04	2.69e-04	1.2e-04
S1-100	6.87e-05	1.2e-05	4.65e-05	3.8e-05	<b>3.80e-05</b>	<b>2.9e-05</b>	9.26e-04	1.9e-04
S1-150	<b>4.22e-04</b>	<b>1.2e-04</b>	5.93e-04	2.4e-04	6.35e-04	2.2e-04	3.01e-03	7.3e-04
S1-200	<b>8.89e-04</b>	<b>2.0e-04</b>	1.82e-03	5.8e-04	1.80e-03	4.5e-04	3.52e-03	9.1e-04

**Table 4** Spread. Mean and standard deviation study between NSGAI, SPEA2, SMSEMOA and MOEA/D

Test instances	NSGAI		SPEA2		SMSEMOA		MOEA/D	
	Avg.	Stdev.	Avg.	Stdev.	Avg.	Stdev.	Avg.	Stdev.
S1-25	1.60e+00	9.6e-02	1.64e+00	9.0e-02	<b>1.35e+00</b>	<b>1.3e-01</b>	1.55e+00	6.7e-02
S2-25	1.65e+00	3.9e-02	1.65e+00	4.2e-02	<b>1.57e+00</b>	<b>1.8e-02</b>	1.79e+00	2.7e-02
S1-50	1.44e+00	4.0e-02	1.42e+00	4.0e-02	<b>1.38e+00</b>	<b>2.8e-02</b>	1.62e+00	3.2e-02
S2-50	1.37e+00	2.6e-02	1.35e+00	2.6e-02	<b>1.32e+00</b>	<b>2.3e-02</b>	1.69e+00	3.5e-02
S1-100	5.18e-01	1.1e-01	<b>1.99e-01</b>	<b>1.7e-02</b>	2.10e-01	2.2e-02	1.39e+00	1.3e-02
S1-150	5.34e-01	4.6e-02	<b>3.61e-01</b>	<b>1.9e-02</b>	3.69e-01	3.0e-02	1.22e+00	2.1e-02
S1-200	4.79e-01	4.2e-02	<b>2.97e-01</b>	<b>3.0e-02</b>	3.42e-01	3.3e-02	1.03e+00	3.6e-02

**Table 5** Hypervolume. Mean and standard deviation study between NSGAI, SPEA2, SMSEMOA and MOEA/D

Test instances	NSGAI		SPEA2		SMSEMOA		MOEA/D	
	Avg.	Stdev.	Avg.	Stdev.	Avg.	Stdev.	Avg.	Stdev.
S1-25	4.76e-01	4.0e-02	<b>4.83e-01</b>	<b>3.2e-02</b>	2.52e-01	1.4e-01	3.81e-01	5.3e-02
S2-25	<b>5.71e-01</b>	<b>7.7e-03</b>	5.68e-01	1.0e-02	5.35e-01	1.5e-02	5.56e-01	1.8e-02
S1-50	<b>5.67e-01</b>	<b>1.1e-02</b>	5.58e-01	1.5e-02	5.29e-01	3.2e-02	5.49e-01	9.4e-03
S2-50	<b>6.07e-01</b>	<b>1.0e-02</b>	5.97e-01	1.3e-02	5.78e-01	1.7e-02	5.98e-01	9.5e-03
S1-100	5.36e-01	2.2e-03	5.35e-01	4.5e-03	<b>5.37e-01</b>	<b>3.1e-03</b>	4.84e-01	1.1e-02
S1-150	<b>5.53e-01</b>	<b>5.3e-03</b>	5.51e-01	6.0e-03	5.51e-01	6.7e-03	5.02e-01	1.7e-02
S1-200	5.63e-01	8.1e-03	5.59e-01	7.4e-03	<b>5.63e-01</b>	<b>7.3e-03</b>	5.10e-01	1.6e-02

in this paper when different types of sensors are used. It would be also interesting to investigate how to use variable-size chromosomes to address very large-scale applications of sensor networks. The principal challenge here is to design an approach with low computational complexity.

**Acknowledgements** This work was supported by Agence Universitaire de la Francophonie (AUF).

**References**

- Cai W, Chen M, Hara T, Shu L, Kwon T (2011) A genetic algorithm approach to multi-agent itinerary planning in wireless sensor networks. *Mob Netw Appl* 16(6):782–793
- Wang B (2011) Coverage problems in sensor networks: survey. *ACM Comput Surv* 43(4):32-1–32-53
- Yoon Y, Kim Y-H (2013) An efficient genetic algorithm for maximum coverage deployment in wireless sensor networks. *IEEE Trans Cybern* 43(5):1473–1483
- Zorbas D, Glynos D, Kotzanikolaou P, Douligieris C (2010) Solving coverage problems in wireless sensor networks using cover sets. *Ad Hoc Netw* 8(4):400–415
- Deif DS, Gadallah Y (2014) Classification of wireless sensor networks deployment techniques. *IEEE Commun Surv Tuts* 16(2):834–855
- Liu X (2015) A deployment strategy for multiple types of requirements in wireless sensor networks. *IEEE Trans Cybern* 45(10):2364–2376
- Zhu H, Luo H, Peng H, Li L, Luo Q (2009) Complex networks-based energy-efficient evolution model for wireless sensor networks. *Chaos Solitons Fractals* 41(4):1828–1835

8. Li S, Li L, Yang Y (2011) A local-world heterogeneous model of wireless sensor networks with node and link diversity. *Phys A* 390(6):1182–1191
9. Ruela A, Cabral R, Aquino A, Guimaraes F (2009) Evolutionary design of wireless sensor networks based on complex networks. In: *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, pp 237–242
10. Gu Y, Liu H, Zhao B (2007) Target coverage with QoS requirements in wireless sensor networks. In: *The 2007 International Conference on Intelligent Pervasive Computing*, 2007. IPC, pp 35–38
11. Singh A, Rossi A (2013) A genetic algorithm based exact approach for lifetime maximization of directional sensor networks. *Ad Hoc Netw* 11(3):1006–1021
12. Slijepcevic S, Potkonjak M (2001) Power efficient organization of wireless sensor networks. In: *IEEE international conference on Wireless Communications*, vol 2, Helsinki, Finland, pp 472–476
13. Cardei M, Thai MT, Li Y, Wu W (2005) Energy-efficient target coverage in wireless sensor networks. In: *Proceedings IEEE INFOCOM, (2005) 24th Annual Joint Conference of the IEEE Computer and Communications Societies*, vol 3, pp 1976–1984
14. Cardei M, Du DZ (2005) Improving wireless sensor network lifetime through power aware organization. *Wirel Netw* 11(3):333–340
15. Lai C-C, Ting C-K, Ko R-S (2007) An effective genetic algorithm to improve wireless sensor network lifetime for large-scale surveillance applications. In: *IEEE Congress on Evolutionary Computation, CEC*, pp 3531–3538
16. Chen J, Jia J, Wen Y, Zhao D, Liu D (2009) Modeling and extending lifetime of wireless sensor networks using genetic algorithm. In: *Proceedings of the 1st ACM/SIGEVO Summit on Genetic and Evolutionary Computation*, Shanghai, China, pp 47–54
17. Gil J-M, Han Y-H (2011) A target coverage scheduling scheme based on genetic algorithms in directional sensor networks. *Sensors* 11(2):1888–1906
18. Mini S, Udgata SK, Sabat SL (2014) Sensor deployment and scheduling for target coverage problem in wireless sensor networks. *IEEE Sens J* 14(3):636–644
19. Njoya AN, Abdou W, Dipanda A, Tonye E (2015) Evolutionary-based wireless sensor deployment for target coverage. In: *Proc. IEEE SITIS*, 2015, pp 739–745
20. Zdemir S, Baraa AA, Khalil A (2013) Multi-objective evolutionary algorithm based on decomposition for energy efficient coverage in wireless sensor networks. *Wirel Pers Commun* 71(1):195–215
21. Dhillon SS, Chakrabarty K (2003) Sensor placement for effective coverage and surveillance in distributed sensor networks. In: *IEEE Wireless Communications and Networking Conference*, 2003, pp 1609–1614
22. Lin F, Chiu P (2005) A near-optimal sensor placement algorithm to achieve complete coverage-discrimination in sensor networks. *IEEE Commun Lett* 9(1):43–45
23. Jourdan DB, de Weck OL (2004) Layout optimization for a wireless sensor network using a multi-objective genetic algorithm. In: *Proceedings 59th IEEE Veh. Technology Conference*, vol 5, pp 2466–2470
24. Y. Xu and X. Yao, “A ga approach to the optimal placement of sensors in wireless sensor networks with obstacles and preferences,” in *Proc. 3rd IEEE Consumer Commun. Networking Conf.*, 2006, pp. 127–131
25. Zhao JC, Yu Z, Chen P (2007) Optimal deployment of nodes based on genetic algorithm in heterogeneous sensor networks. In: *Proceedings of Sept, IEEE International Conference Wireless Communication Networking Mobile Comput.*, pp 2743–2746
26. Wu Q, Rao NSV, Du X, Sitharama S, Vaishnavi VK (2007) On efficient deployment of sensors on planar grid. *Comput Commun* 30:2721–2734
27. Seo J-H, Kim Y-H, Ryou H-B, Cha S-H, Jo M (2008) Optimal sensor deployment for wireless surveillance sensor networks by a hybrid steady-state genetic algorithm. *IEICE Trans Commun E91–B(11):3534–3543*
28. Yildirim K, Kalayci T, Ugur A (2008) Optimizing coverage in a k-covered and connected sensor network using genetic algorithms. In: *9th WSEAS International Conference on Evolutionary Computing (EC08)*, Sofia, Bulgaria, pp 21–26
29. Kalayci TE, Ugur A (2011) Genetic algorithm-based sensor deployment with area priority. *Cybern Syst* 42(8):605–620
30. Banimelhem O, Mowafi M, Aljoby W (2013) Genetic algorithm based node deployment in hybrid wireless sensor networks. *Commun Netw* 5(4):273–279
31. Zou Y, Chakrabarty K (2003) Sensor deployment and target localization based on virtual forces. In: *Proceedings of IEEE Infocom Conference*, pp 1293–1303
32. Akyildiz IF, Vuran MC (2010) *Wireless sensor networks*. Wiley, New York
33. Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: Nsga-ii. *Trans Evol Comput* 6(2):182–197. doi:[10.1109/4235.996017](https://doi.org/10.1109/4235.996017)
34. Zitzler E, Laumanns M, Thiele L (2001) *Spea2: Improving the strength pareto evolutionary algorithm*. Technical Report 103, Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH), Zurich, Switzerland
35. Beume N, Naujoks B, Emmerich M (2007) Sms-emoa: Multiobjective selection based on dominated hypervolume. *Eur J Oper Res* 181(3):1653–1669
36. Zhang Q, Li H (2007) Moea/d: a multiobjective evolutionary algorithm based on decomposition. *IEEE Trans Evol Comput* 11(6):712–731
37. Abdou W (2011) *Optimisation évolutionniste multi-objectifs pour des réseaux ad hoc mobiles : la diffusion robuste et le routage multi-chemins non recouvrant (Ph.D.thesis)*. Université de Franche-Comté, France