

# A Multi-objective Disjoint Set Covers for Reliable Lifetime Maximization of Wireless Sensor Networks

Bara'a A. Attea · Enan A. Khalil · Suat Özdemir · Oktay Yıldız

Published online: 12 November 2014  
© Springer Science+Business Media New York 2014

**Abstract** An important challenge facing many large-scale surveillance applications is how to schedule sensors into disjoint subsets to maximize the coverage time span. Due to its NP-hard complexity, the problem of finding the largest number of disjoint set covers (DSC) of sensors has been addressed by many researchers. Majority of these studies employs the Boolean sensing model where a sensor covers a target if it lies within its sensing range. In reality, however, the sensing reliability may be affected by several parameters, e.g., strength of the generated signals, environmental conditions and the sensor's hardware. To the best of our knowledge, improving coverage reliability of Wireless Sensor Networks (WSNs) has not been explored while solving DSC problem. This paper addresses the problem of improving coverage reliability of WSNs while simultaneously maximizing the number of DSC. Thus, in the context of WSNs design problem, our main contribution is to turn the definition of single-objective DSC problem into a multi-objective problem (MOP) by adopting an additional conflicting objective to be optimized. Specifically, we investigate the performance of two multi-objective evolutionary algorithms in terms of diversity and quality of the Pareto optimal set for the modeled MOP. The simulation results indicate that multi-objective approach results in achieving reliable coverage and large number of DSC compared to a single-objective approach.

---

B. A. Attea (✉)  
Department of Computer Science, Baghdad University, Baghdad, Iraq  
e-mail: baraaali@yahoo.com

B. A. Attea  
e-mail: baraaali@scbaghdad.edu.iq

E. A. Khalil · S. Özdemir · O. Yıldız  
Department of Computer Engineering, Gazi University, Ankara, Turkey  
e-mail: enanameen@yahoo.com

S. Özdemir  
e-mail: suatozdemir@gazi.edu.tr

O. Yıldız  
e-mail: oyildiz@gazi.edu.tr

**Keywords** DSC · MOO · Non-dominated solution · NP-hard · Probabilistic coverage · Reliability · WSNs

## 1 Introduction

The Wireless Sensor Network (WSN) paradigm is an emerging and fast growing technological platform in myriad working environments. To name just a few, WSNs can be used in military and agriculture applications, harsh environment monitoring, emergency search-and-rescue operations. One of the important characteristics of WSNs is their ability to densely deploy in ad-hoc manner. For successful operation of WSNs, sensors should effectively cover the required region of interest for a long period of time. Energy-aware sensing and communication mechanisms have been substantially pursued by the research community in order to prolong the lifespan of WSNs. Energy saving techniques can generally be classified in the following categories; (1) Energy-efficient data aggregation, gathering and routing; (2) power management by adjusting the transmission and/or sensing range of sensor nodes; and (3) sensor wake-up scheduling to alternate between active and idle state.

In this paper, we consider the third approach to achieve energy-efficient WSN that monitors all the targets at all times. In this approach, sensors are divided into disjoint sets of covers (DSC) such that every set completely cover a set of targets being located with known locations. We assume that the lifetime of a WSN is divided into intervals and at each interval only one set is active while the remaining set covers are in the low-energy *sleep* mode. The sensors from the active set are in *active* state, monitoring all targets within their sensing ranges. Once the active set cover runs out of energy, another set cover will be selected to enter the active mode and provide the functionality continuously. As all targets should be monitored by every set cover, the goal of this approach is to determine a maximum number of DSC. The more set covers we can find, the longer sensor network lifetime is prolonged. It has been proven that this problem (also called the SET  $k$ -cover problem) is a generalization of the minimum cover problem [1] and proves its NP-completeness [2,3].

Many attempts in the literature have been made to solve DSC problem in WSNs using either heuristic or meta-heuristic methods (like genetic algorithms). Different scheduling rules determine when sensors change to be active or sleep mode. In localized and distributed realizations, sensors periodically investigate their neighborhood and decide whether to change their operation modes [4–11].

In [2], a heuristic approach called the “most constrained-minimally constraining covering (MCMCC)” is proposed to select and successively activate mutually exclusive sets of covers, where every set completely covers the entire monitored area. This method gives priority to sensors which cover a high number of uncovered fields, cover sparsely covered fields and do not cover fields redundantly. This method achieves energy savings by increasing the number of disjoint covers. Also, the DSC problem and its extension to the maximum set cover (MSC) problem have been solved in [12] and [3] using integer programming, linear programming, and greedy heuristics. In [12], the DSC problem is reduced to a maximum flow problem and solved using mixed integer programming. By a branch and bound method, the maximum covers based on mixed integer programming algorithm (MC-MIP) acts as an implicit exhaustive search to guarantees finding the optimal solution.

The definition of DSC problem has been re-formulated in [3,13], and [14] to include additional coverage constraints. The definition of DSC problem has been generalized in [3] to a maximum set covers (MSC) problem and solved it using linear programming and greedy techniques. The extended problem in MSC lets the sensors to participate in multiple sets. In

[13], the DSC problem has been extended to include connectivity constraint as well. Then, the Connected Set Covers (CSC) problem has as objective finding a maximum number of set covers such that each sensor to be activated should be connected to the base station. In [14], DSC problem has been extended to include sensor coverage-failure probability. Each sensor is associated with sensor's failure probability (comes from several facts, e.g., manufacture, weather in the monitoring area, interferences to the sensors, or unexpected accidents). The proposed Maximum Reliability Sensor Covers (MRSC) problem has been solved in [14] using a heuristic greedy algorithm to compute the maximal number of set covers that satisfy a user specified coverage-reliability threshold.

The works in [15–17] also provide solutions to the DSC problem in WSNs by employing the meta-heuristic framework of evolutionary and genetic algorithms. Like the previous mentioned heuristic methods, the *single-objective* genetic algorithms proposed in [15–17] assume simple and common isotropic (i.e., disc) sensing model. Each sensor in this model is associated with a sensing area which is represented by a circle with the same radius. According to [18], the upper bound,  $ub$ , of the maximum number of disjoint complete set covers depends on the size of the target area, the total number of sensors, the sensors' locations, and their *sensing ranges*.  $ub$  can be estimated as the minimum number of sensors covering the *most sparsely* covered target, and thus can be computed by polynomial-time algorithms. It has been shown that for the same deployment of sensors and targets,  $ub$  can be increased by enlarging the sensing radius of each sensor [15–18]. In reality, however, the sensing region of a sensor could be irregular, resulting in imprecise target coverage. The coverage in this case could be expressed in probabilistic terms [19–21]. In probabilistic sensing model, there is a measure of uncertainty in sensor signal-detection being expressed by a value from 0 to 1. For reliable coverage with threshold  $c_{th}$ , the detection uncertainty of each target should not exceed  $1 - c_{th}$ . This in turn should effect on the number of sensors *reliably* covering the most sparsely covered target. As more reliability degree to cover a target is required,  $ub$  becomes smaller. Thus, there should be a trade-off between the coverage reliability (also denoted throughout this paper as *covers reliability*) and  $ub$ .

Like many other sensor network design problems, maintaining reliable coverage while maximizing the number of DSC (i.e., extending network lifetime) often require optimizing two conflicting decision variables. Due to the conflicting nature of many sensor network design problems, multi-objective optimization (MOO) has, recently, attracted several researchers in formulating multi-objective optimization problems (MOPs). In MOO, a set of alternative solutions (called non-dominated solutions) can simultaneously be obtained providing the decision maker with an optimal tradeoff between the conflicting objectives. As multi-objective evolutionary algorithms (MOEAs) possess several interesting characteristics for tackling MOPs, there are many MOEA approaches for solving different sensor design problems. For organizing energy efficient WSNs, the literature provides a collection of MOO schemes in which sensing coverage and network lifetime are the main design issues to be optimized. However, all of these methods fall in one of the first two categories; data aggregation, routing, and sensor clustering [22–25], or sensors' power management [26–29]. For the reliable DSC design problem in WSNs, our work in [30] formulates a single objective evolutionary algorithm to tackle it. To the best of our knowledge, however, MOO concept has not been exploited. The contributions of this paper are as follows:

1. The paper turns the *de-facto* definition of single-objective DSC in WSN design problems to a multi-objective DSC. The goal of which becomes to cope with contradictory objectives. The formulated optimization problem will be directed to effectively maintaining reliable coverage with large number of cover sets in WSNs.

2. The paper presents two MOEAs with a new heuristic perturbation operator (instead of the traditional *bi*-perturbation of crossover and mutation operators) to solve the modeled problem. The results show that the Pareto-optimization can provide the decision maker with a set of non-dominated solutions that effectively render the tradeoff between the two conflicting objectives (i.e., network lifetime and coverage probability).

The rest of this paper is outlined as follows. Section 2 describes the traditional formulation of DSC problem in WSNs as well as the new formulation of multi-objective DSC by adding probabilistic coverage means to the original problem. Section 3 provides the preliminaries to the concept of multi-objective optimization and introduces the formulation of the proposed perturbation operator. The results of multi-objective DSC optimizations are then compared with a related work of single-objective genetic algorithm in Sect. 4. Finally, Sect. 5 concludes the current work and hints some further ramifications.

## 2 Problem Definition

### 2.1 System Model and Assumptions

In order to model the system, we assume that the investigated WSNs have 2D sensing area  $\mathcal{A}$  with known size  $(X_{max}, Y_{max})$ . We also assume that  $\mathcal{A}$  has a set  $\mathcal{T}$  (i.e., target set) of  $n$  targets with known locations, i.e.,  $\mathcal{T} = \{(x_{t1}, y_{t1}), (x_{t2}, y_{t2}), \dots, (x_{tn}, y_{tn})\}$ . There are  $m$  homogenous sensors  $\mathcal{S} = \{(x_{s1}, y_{s1}), (x_{s2}, y_{s2}), \dots, (x_{sm}, y_{sm})\}$  having the same sensing range  $r_s$ . All the sensors are dropped randomly in  $\mathcal{A}$  ( $1 \leq \forall i \leq m | (x_{si}, y_{si}) = ([0, X_{max}], [0, Y_{max}])$ ). Depending on the sensing range  $r_s$ , each sensor is responsible for sensing and covering a part of  $\mathcal{A}$ . We consider a probabilistic sensing model [20,21] to define the notion of the probabilistic coverage of a target  $\mathcal{T}_j = (x_{tj}, y_{tj})$  by a sensor  $s_i$ .

$$Coverage(s_i, t_j) = \begin{cases} 0 & \text{if } r_s + r_u \leq d(s_i, t_j) \\ e^{-\lambda a^\beta} & \text{if } r_s - r_u < d(s_i, t_j) < r_s + r_u \\ 1 & \text{if } r_s - r_u \geq d(s_i, t_j) \end{cases} \quad (1)$$

where  $r_u$  is a measure of the uncertainty in sensor detection.  $d(s_i, t_j)$  is the Euclidean distance  $\sqrt{(x_{si} - x_{tj})^2 + (y_{si} - y_{tj})^2}$  between sensor  $s_i$  and target  $t_j$ .  $a = d(s_i, t_j) - (r_s - r_u)$ , and  $\lambda$  and  $\beta$  are probabilistic detection parameters to measure detection strength when a target point lies within the interval  $\{r_s - r_u, r_s + r_u\}$ . It causes coverage value to exponentially decrease as the distance increase. All points that lie within a distance of  $r_s - r_u$  from the sensor are said to be 1-covered. Beyond the distance  $r_s + r_u$ , all the points have 0-coverage by this sensor.

To save energy and prolong WSN’s lifetime, sensors in the sensor set  $\mathcal{S}$  should be divided into duty-cycling *sensor cover* (also called *set cover*) subsets, each of which can cover all the interested targets in  $\mathcal{T}$ . Thus, in the traditional Boolean sensing model, the definition of the sensor cover could be formulated as:

**Definition 1** (*Sensor Cover*). Given a WSN consists of target set  $\mathcal{T}$  and sensor set  $\mathcal{S}$ , where each sensor  $s_i \in \mathcal{S}$  can be represented as a subset  $\mathcal{T}_i \subset \mathcal{T}$ , such that  $t_j \in \mathcal{T}_i$  if and only if  $Coverage(s_i, t_j) = 1$ . Any subset  $\mathcal{S}_i \subset \mathcal{S}$  that can completely cover all the target set  $\mathcal{T}$  is termed as a sensor cover.

However, considering probabilistic sensing model, the definition of the traditional sensor cover needs to be re-formulated as:

**Definition 2** (*Reliable Sensor Cover*). Given a WSN consists of target set  $\mathcal{T}$  and sensor set  $\mathcal{S}$ , where each sensor  $s_i \in \mathcal{S}$  can be represented as a subset  $\mathcal{T}_i \subset \mathcal{T}$ , such that  $t_j \in \mathcal{T}_i$  if and only if  $Coverage(s_i, t_j) \geq c_{th}$ . Any subset  $\mathcal{S}_i \subset \mathcal{S}$  that can satisfy a user coverage constraint  $c_{th}$  to cover all the targets in  $\mathcal{T}$  is termed as a reliable sensor cover or reliable set cover.

Now, the problem of finding the maximum number of disjoint set covers (DSC) could be turned into the problem of finding the maximum number of disjoint reliable sensor covers (DRSC), and can be formulated as:

**Definition 3** (*Disjoint Reliable Sensor Covers Problem—DRSC*). Given a collection  $\mathcal{S}$  of reliable subsets of a finite set  $\mathcal{T}$ , find the maximum number of disjoint reliable covers for  $\mathcal{T}$ . Every cover  $\mathcal{S}_i$  is a subset of  $\mathcal{S}$ ,  $\mathcal{S}_i \subseteq \mathcal{S}$ , such that every element  $t_j$  of  $\mathcal{T}$  belongs to at least one member of  $\mathcal{S}_i$ , and for any two covers  $\mathcal{S}_i$  and  $\mathcal{S}_j$ ,  $\mathcal{S}_i \cap_j^S = \emptyset$ . Now, the upper bound  $ub$  of the maximum number of disjoint reliable covers is determined by the most sparsely covered target and can be calculated as:

$$ub = \min_{i=1,2,\dots,n} (|\mathcal{S}_i|) \tag{2}$$

where  $\mathcal{CS} = \mathcal{S}_i$ ,  $|\mathcal{S}_i| = ub$  denotes the set of sensors (called *Critical sensors Set*) covering target  $t_i$  with certainty equal to or larger than a user specified threshold  $c_{th}$ . As the most sparsely covered target only covered by  $ub$ . Sensors, the maximum number of disjoint reliable covers  $N$  is no larger than  $ub$  i.e.,  $N \leq ub$ ).

### 2.2 Problem Formulation

In this section, we formulate the problem of getting high coverage percentage with maximum number of DSC as *multi-objective Disjoint Reliable Sensor Covers Problem*:

Given:

- $\mathcal{A}$  : 2-D plane area with size  $(X_{max}, Y_{max})$ .
- $\mathcal{T}$  : set of  $n$  targets being uniformly distributed in  $\mathcal{A}$ .
- $\mathcal{S}$  : set of  $m$  sensors with probabilistic sensing capability, being uniformly distributed in  $\mathcal{A}$ .
- $r_s$  : sensing range.
- $r_u, \lambda$  and  $\beta$ : uncertainty-parameters adjusted according to the physical properties of the sensor.

*Decision (Design) variables of a solution:*

The scheduling sets of sensors in  $\mathcal{S}$ , i.e.,  $\{\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_m\} | \mathcal{S}_i \in [1, ub]$ .

*Objective:*

Two objectives are considered as optimization functions. The first objective is to maximize the number of scheduling sets of the sensors in  $\mathcal{S}$ , such that each scheduling set  $\mathcal{S}_i$  can form a complete reliable sensor cover (see Eq. 3). Thus, Eq. 3 computes the number of reliable covers (i.e., without and coverage holes) being formed by a particular solution. Detection of coverage holes is formulated in Eq. 4. The second objective is to maximize the reliability of the whole disjoint sensor covers (see Eq. 5). In each sensor cover, we consider only the highest coverage of the sensors belong to this cover to each target in set  $\mathcal{T}$ .

$$max\ DRSC = \sum_{\mathcal{S}_i \in [1, ub]} Cover(\mathcal{S}_i, \mathcal{T}) \tag{3}$$

where:

$$Cover(S_i, T) = \begin{cases} 1 & \text{if } \forall t \in T \rightarrow \exists s \in S_i | Coverage(s, t) \geq c_{th} \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

$$max\ Reliability = \sum_{S_i | Cover(S_i, T)=1} Coverage(S_i, T) \tag{5}$$

where:

$$Coverage(S_i, T) = \frac{\sum_{t \in T} \max_{s \in S_i} Coverage(s, t)}{n} \tag{6}$$

### 3 Multi-objective Evolutionary Algorithms for DRSC Problem

#### 3.1 Preliminaries

The general formalization of MOP is defined as [31,32]: finding a vector  $\mathbb{X}^* = [x_1^*, x_2^*, \dots, x_n^*]^T$  optimizing (in terms of *domination*) the vector function  $\mathbb{F}(\mathbb{X}) = [f_1(\mathbb{X}), f_2(\mathbb{X}), \dots, f_k(\mathbb{X})]^T$  where  $\mathbb{X} = [x_1, x_2, \dots, x_n]^T$  is the decision variables vector. Optimizing  $\mathbb{F}(\mathbb{X})$ , here, considers the domination concept. Let us consider two solution  $\mathbb{U}$  and  $\mathbb{V}$  from the solution space  $P$ . Then, solution  $\mathbb{U}$  is said to dominate  $\mathbb{V}$  if and only if the following two conditions hold:

1. The solution  $\mathbb{U}$  is no worse than  $\mathbb{V}$  in all objectives, or  $f_i(\mathbb{U}) \not\prec f_i(\mathbb{V})$  for all  $i = 1, 2, \dots, k$ . For example in maximization, the word “no worse” means  $f_i(\mathbb{U}) \not< f_i(\mathbb{V})$ .
2. The solution  $\mathbb{U}$  is strictly better than  $\mathbb{V}$  in at least one objective, or  $f_i(\mathbb{U}) \triangleleft f_i(\mathbb{V})$  for at least one  $i \in 1, 2, \dots, k$ . For example in maximization, the word “strictly better” means  $f_i(\mathbb{U}) > f_i(\mathbb{V})$ .

The notation  $(i \triangleleft j)$  is used to denote that solution  $i$  is better than solution  $j$  regardless of the type of the optimization problem at hand (maximization or minimization). Also, the notation  $(i \triangleright j)$  is used in the same way to express that solution  $j$  is better than solution  $i$ . Hence, a dominated set can be defined as: among a set of solutions  $P$ , the non-dominated solutions set  $\hat{P}$  are those that are not dominated by any member of the set  $P$ .

In this paper, we consider two multi-objective evolutionary algorithms (MOEAs). These are: multi-objective evolutionary algorithm with Tchebycheff decomposition (MOEA/D) and non-dominated sorting genetic algorithm (NSGA-II). In what follows, we only present these two algorithms focusing on the proposed perturbation operator. We refer interested readers to [31–35] for more detailed description of the general framework of MOEA/D and NSGA-II.

#### 3.2 The Proposed Meta-heuristic Framework

As MOEA/D and NSGA-II are population-based optimization algorithms, let us consider a population  $\rho$  of  $K$  solutions. The choice of a good solution representation is a critical issue for the applicability and performance of evolutionary algorithm. Solution representation is highly problem dependent and related to the evolution operations. In our algorithm design, each individual solution  $\mathbb{P}_{1 \leq i \leq K} \in \rho$  is represented as a fixed-length vector of size  $m$ , where each element (i.e., gene) value indicates the index of the set cover that the corresponding sensor joins, specifying that the index does not exceed the upper limit  $ub$ . Thus,  $\forall i \in \{1, \dots, N\}$  and  $\forall j \in [1, m] : \mathbb{P}_i = (\mathbb{P}_{i1}, \mathbb{P}_{i2}, \dots, \mathbb{P}_{im})$  s.t.  $\mathbb{P}_{ij} \in [1, ub]$ . Then, each of the implemented MOOs can be described as a process formulated in an iterative evolution

function  $\Psi : \{\rho, EP\} \rightarrow \{\rho', EP'\}$  with  $\Psi(\rho_i) = \rho_{i+1}$ , where  $i$  is the iteration index and  $\rho_i$  is the population in iteration  $i$ .  $EP$  is an external archive for the generated non-dominated solutions. The population starts with an initial random population  $\rho_0$  and continues until a maximum number of iterations  $max_i$  has been reached. Each iteration  $i$  consists of a three main operators: parents selection, heuristic perturbation, and  $EP$  update.

Based on the chromosome representation, we adopted an intuitive and fast random initialization mechanism, where each gene can take a random value that does not exceed the upper limit  $ub$ . Formally speaking:

$$(\forall i \in \{1, \dots, N\}, \forall j \in [1, m]) : \mathbb{P}_{ij} \sim [1, ub] \tag{7}$$

where  $\sim [1, ub]$  means a uniform random integer number between 1 and  $ub$ .

For selection operator, we imitate similar functions to that found in the implementation of the traditional MOEA/D and NSGA-II. For the perturbation function, we proposed one evolution operator  $\Psi$  that provides heuristic for improving a solution coverage reliability or number of disjoint covers. The process of the proposed heuristic perturbation operator is presented next (see Algorithm 1). It takes as input two parent individuals  $\mathbb{P}_1, \mathbb{P}_2$  being selected from  $\rho$  by the selection operator and perturbs them with probability  $p_{per} = 1.0$  to return one child  $\mathbb{C}$  with its decoded set covers  $C_1, C_2, \dots, C_k$ . For each child, the proposed

---

**Algorithm 1: Heuristic-Perturbation** ( $\mathbb{P}_1, \mathbb{P}_2, \mathbb{C}, F, \mathcal{CS}$ )

---

```

1  set number of covers:
    $k \leftarrow \max(\mathbb{P}_{11}, \mathbb{P}_{12}, \dots, \mathbb{P}_{1m}, \mathbb{P}_{21}, \mathbb{P}_{22}, \dots, \mathbb{P}_{2m})$ 
2  set  $\mathbb{C} \leftarrow \emptyset$ 
3  set cover counter  $i \leftarrow 1$ 
4  while  $i \leq k$  do
5      /* a new cover  $C_i$  will be formed */
6      set  $S$  to the sensors from both  $\mathbb{P}_1$  and  $\mathbb{P}_2$  being grouped with  $C_i$ 
        $S \leftarrow \{\mathbb{P}_{1j} | \mathbb{P}_{1j} = i\} \cup \{\mathbb{P}_{2j} | \mathbb{P}_{2j} = i\}$ 
7      set contributed sensors set  $S' \leftarrow \emptyset$ 
8      if  $Cover(S, \mathcal{T}) == 1$  /* no coverage hole */
9          set  $Target \leftarrow \mathcal{T}$ 
10         while  $Target \neq \emptyset$ 
11             /* in  $C_i$  more targets need to be covered */
12             if  $F == \text{reliability-heuristic}$  then
13                 select a sensor  $s \in S$  that contributes to the most
                   reliable coverage to a target  $t \in \mathcal{T}$ 
14             else /* heuristic towards maximizing set covers */
15                 select a sensor  $s \in S$  and  $s \notin \mathcal{CS}$  that contributes
                   to cover more number of targets  $t \in \mathcal{T}$ .
                   Otherwise, select a sensor  $s \in S$  and  $s \in \mathcal{CS}$  that
                   contributes to cover more number of targets
                    $t \in \mathcal{T}$ .
16             end if
17             set the index of  $s$  in  $\mathbb{C}$  to  $C_i$ 
18             set  $Target \leftarrow Target - t; S' \leftarrow S' \cup s$ 
19         end while
20         remove  $\{S - S'\} \rightarrow C_k; i \leftarrow i + 1$ 
21     else /* coverage holes exist after using the sensors in  $S$  */
22         if  $i < k$ 
23             /* draw the sensors from the last cover and collect
                   them with the current cover */
                $S \leftarrow S \cup \{\mathbb{P}_{1j} | \mathbb{P}_{1j} = k\} \cup \{\mathbb{P}_{2j} | \mathbb{P}_{2j} = k\}; k \leftarrow k - 1$ 
24         else
25             set the index of  $s \in S$  in  $\mathbb{C}$  to  $C_k$ 
26         end if
27     end if
28 end while

```

---



heuristic will be directed (with equal probability) either towards maximizing the reliability of the generated set covers or towards maximizing the number of sensor covers  $k$ . Thus, for each child  $\mathbb{C}_i$ ,  $1 \leq i \leq K$ , the heuristic procedure also takes as input a flag  $F$  indicating whether to prefer maximizing reliability or maximizing number of sensor covers. The proposed heuristic also takes as input the set of the critical sensors  $CS$ .

The proposed heuristic recursively builds sensor covers in the child. For each cover  $i$ , the set  $S$  (in line 6) collects sensors from both parents that have an index value equal to  $i$ . The sensors in  $S$  can then participate in creating the cover  $i$  in the child. When sensors in  $S$  can cover all the targets without any coverage hole (in line 8), then (and according to the flag  $F$ ), the procedure will select from  $S$  those sensors having the highest contribution to the heuristic (line 12–18). In order to maximize number of sensor covers towards its upper limit  $ub$ , a heuristic process is maintained in line 15. Note that this process is guided *to restrict the choice of sensors, for the current sensor cover  $i$ , to those that do not belong to the set of critical sensors (i.e.,  $s \notin CS$ ) or at least bias the choice in their support*. The biasing depends on the number of targets being covered by each sensor  $s \notin CS$ . However, if there is still coverage hole in the current cover  $i$ , the heuristic process turns the selection to the critical sensors  $s \in CS$ . The remaining sensors can then contribute in the next covers (line 20). They will migrate to the last set cover  $k$  and used again (line 23) when there is a coverage hole (line 21) while forming a new cover  $i$ . Finally, the last cover (line 25) may form a complete set cover or incomplete one. The generated child  $\mathbb{C}$  may modify the contents of the external archive  $EP$  according to the non-domination condition described previously.

#### 4 Simulation Results and Discussion

In this section, we evaluate the performance of the proposed MOO algorithms in terms of number of set covers obtained and coverage reliability. The results are obtained after setting WSNs and algorithm parameters into the most commonly used in the related literature. The simulation area is square-shaped with side length  $X_{max} = 500$  m of 10 randomly distributed targets. Unless otherwise stated, the simulation is mainly divided into five groups according to five different settings of sensor density: 50, 60, 70, 80, and 90. For each group, we vary the sensing range of the sensor nodes  $R_s$  to three different values {100, 200, 300}, to get a different test instance. Each test instance includes 10 random WSNs with different configurations. Thus the overall simulation examines a total of 150 random networks. As we have random WSN configurations, we may get different value for the upper bound of set covers  $ub$ , for the same parameter settings. Thus in each test instance, the presented results also indicate the average number of upper bound of set covers (denoted by  $\overline{ub}$ ). Uncertainty level  $R_u$  is set to  $R_s * 0.5$  units, both  $\lambda$  and  $\beta$  are set to 0.5, and  $c_{th}$  is set to 0.001. The setting of the probabilistic coverage parameters also influences the overall network's coverage reliability. Since studying the impact of varying these parameters is out of the scope of this paper, we fixed these parameters to one setting and evaluate the average coverage reliability (denoted by  $\bar{r}$ ) of all network configurations in each test instance. Population size is set to 50 and will be allowed to evolve 500 times. Perturbation probability is fixed to 1.0. For MOEA/D, the size of neighboring solutions that participate in the evolution of each individual solution is fixed to 5.

Both MOEA/D and NSGA-II have been compared with Genetic Algorithm for Maximum Disjoint Set Covers (GAMDSC). GAMDSC has been proposed in [15] to extend the lifetime of WSNs. GAMDSC is single-objective genetic algorithm that use integer representation for chromosome representation, traditional uniform crossover, creep mutation, and a scatter



operator to evolve their chromosome solutions. The proposed fitness function in GAMDSC has been devoted to maximize number of complete set covers in each chromosome. The performance of GAMDSC has been compared in [15] against the most constrained-minimum constraining heuristic (MCMCC) [2] and the exhaustive heuristic of Maximum Covers using Mixed Integer Programming (MCMIP) [12] for solving the DSC problem in WSNs. Simulation results in [15] show that GAMDSC can get near-optimal solutions and improve the performance of MCMCC by 16 % in terms of the obtained sensor covers. As compared with the exhaustive search of MCMIP, GAMDSC can get results with acceptable computation time.

#### 4.1 Comparison of Multi-objective Versus Single-objective DSC

First, results are presented in Tables 1 and 2 to compare the quality of results obtained by GAMDSC, MOEA/D, and NSGA-II. Table 1 presents number of set covers being obtained by the three algorithms. Table 2 presents the coverage reliability being obtained by the algorithms. Also, to examine the quality of solutions provided by the three algorithms, we present  $\overline{ub}$  and  $\bar{r}$  as reference values in Tables 1 and 2, respectively.

For both MOEA/D and NSGA-II, Tables 1 and 2 present the two extreme non-dominated solutions (in terms of number of set covers and reliability, respectively). The solution providing the closest value to  $\overline{ub}$  in each test instance in Table 1 has been presented in bold. Also, the solution providing the maximum reliability in each test instance in Table 2 has been bolded. Note that in Table 1, closest solutions to  $\overline{ub}$  do not necessarily imply best solutions. In terms of the algorithm's independent single-objective measure of solution's quality, closest solutions to  $\overline{ub}$  should result in high quality solutions. That is GAMDSC provides more sensor covers in four test instances while NSGA-II provides more sensor covers in almost all of the remaining test instances. However, seeking for tradeoff solutions that balance between the sensor covers and coverage reliability (see Table 2), the performance of GAMDSC in these four test instances is not good enough. The coverage reliability of GAMDSC gives the lowest score while both MOEA/D and NSGA-II outperform GAMDSC in all test instances. Similarly, in the third and tenth test instances (referred with asterisk), GAMDSC gives equal values of sensor covers to NSGA-II or MOEA/D, respectively (as presented in Table 1), however, both NSGA-II MOEA/D outperform GAMDSC in terms of reliability (as presented in Table 2). Thus, the combined results of Tables 1 and 2 signify the importance of utilizing the two objectives as optimizing criteria rather than maximizing only the number of sensor covers. Moreover, in all test instances, Table 2 presents that NSGA-II performs better than MOEA/D even in the case of providing an equal number of set covers as in the sixth test instance, referred with asterisk in Table 1.

Figure 1 compares the performance of GAMDSC, MOEA/D, and NSGA-II in terms of average number of disjoint sensor covers obtained while fixing sensing radius. In this simulation, we fixed sensing range  $R_s$  to 500 and vary number of sensors from 50 to 90 with an incremental value of 5. Each test instance of this simulation includes 10 random networks, and thus the total simulation is experimented under test-bed of 90 different network topologies. This simulation clarifies the impact of varying sensing ranges, of a given fixed number and fixed locations of sensors, on the total number of disjoint sensor covers. The results of this simulation summarize that for a fixed sensing radius, the number of sensor covers increases with increasing number of sensors. Also, from the figure we can that NSGA-II provides more number of sensor covers than both MOEA/D and GAMDSC.

On the other hand, Fig. 2 clarifies the impact of varying sensing range  $R_s$  while fixing number of sensors. In this simulation, the number of sensors is set to 70 and let  $R_s$  to vary

**Table 1** Comparison of the maximum number of the generated set covers for 10 WSNs in each test case

Test instance	No. of sensors	$R_s$	$\overline{ub}$	GAMDSC	MOEA/D	NSGA-II
1	50	100	5.00	<b>4.80</b>	4.70 1.10	4.60 1.30
2		200	17.30	<b>15.80</b>	15.10 6.80	15.00 5.90
3		300	23.10	21.20	21.10 9.50	<b>21.20*</b> 8.20
4	60	100	6.50	5.50	5.60 1.30	<b>5.60</b> 1.80
5		200	22.30	18.90	19.20 8.90	<b>19.60</b> 7.30
6		300	30.00	25.80	26.30 13.40	<b>26.30*</b> 9.90
7	70	100	7.70	6.70	6.70 1.80	<b>6.80</b> 1.80
8		200	23.80	20.80	20.60 10.10	<b>20.90</b> 8.30
9		300	32.10	28.10	29.00 14.60	<b>29.40</b> 11.10
10	80	100	9.50	8.30	<b>8.30*</b> 2.10	8.20 1.80
11		200	31.20	25.80	25.90 13.70	<b>26.50</b> 10.60
12		300	41.00	34.10	36.10 18.50	<b>36.40</b> 14.50
13	90	100	9.10	<b>8.50</b>	8.40 3.00	8.30 3.40
14		200	30.80	<b>27.10</b>	26.50 14.80	26.40 11.40
15		300	42.10	36.30	36.70 20.40	<b>37.90</b> 16.20

For MOEA/D and NSGA-II, results are presented in terms of *max* and *min* DRSC solutions in EP

from 100 to 500 with an incremental value of 50. Also, in this simulation, each test instance includes 10 random networks with a total of 90 different network topologies. The results of the figure indicates that increasing sensing radius have a positive impact on the number of sensor covers, even when fixing the number of sensors to a certain value. Also, we can see that NSGS-II provides more number of sensor covers than both MOEA/D and GAMDSC.

#### 4.2 Quality and Diversity of Pareto Front Solutions

Let us now turn to quantitatively compare the performance of MOEA/D and NSGA-II. Here, the comparison will be according to the quality of the non-dominated solutions set being obtained by each algorithm. The performance of MOEA/D and NSGA-II can also be

**Table 2** Comparison of the maximum covers reliability provided for 10 WSNs in each test case

Test instance	No. of sensors	$R_g$	$\bar{r}$	GAMDSC	MOEA/D	NSGA-II
1	50	100	0.1867	0.3932	0.6971 0.4038	<b>0.7027</b> 0.4052
2		200	0.2114	0.3365	0.6447 0.3463	<b>0.7542</b> 0.3489
3		300	0.3216	0.4805	0.8049 0.4783	<b>0.9000</b> 0.4933
4	60	100	0.1731	0.3295	0.6066 0.3661	<b>0.6168</b> 0.3584
5		200	0.2000	0.3175	0.6069 0.3273	<b>0.7062</b> 0.3187
6		300	0.3206	0.4533	0.7501 0.4545	<b>0.8784</b> 0.4708
7	70	100	0.1754	0.3020	0.6275 0.3502	<b>0.6531</b> 0.3482
8		200	0.2090	0.3367	0.6265 0.3357	<b>0.7340</b> 0.3422
9		300	0.3156	0.4644	0.7706 0.4701	<b>0.8932</b> 0.4768
10	80	100	0.1888	0.3245	0.6221 0.3620	<b>0.6753</b> 0.3711
11		200	0.2072	0.3220	0.5968 0.3422	<b>0.7229</b> 0.3365
12		300	0.3267	0.4799	0.7665 0.4764	<b>0.8895</b> 0.4857
13	90	100	0.1972	0.3796	0.6549 0.4263	<b>0.6787</b> 0.4336
14		200	0.2214	0.3683	0.6163 0.3695	<b>0.7521</b> 0.3910
15		300	0.3384	0.5168	0.7822 0.4999	<b>0.8772</b> 0.5176

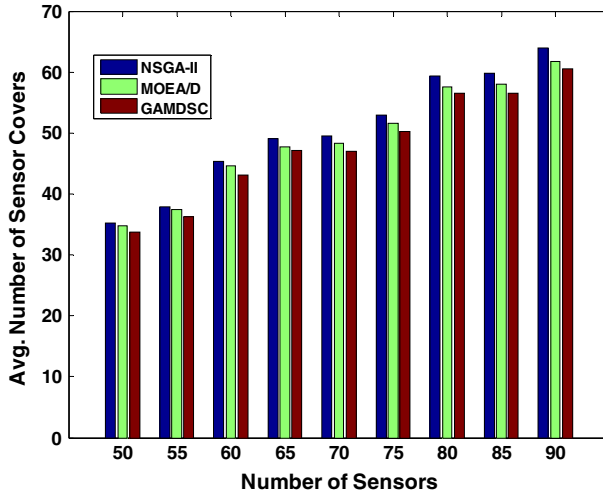
For MOEA/D and NSGA-II, results are presented in terms of *max* and *min Reliability* extreme solutions in *EP*

verified visually by depicting the Pareto Front *EP*. According to [36], if *A* and *B* are two non-dominated sets resulted from two MOEAs *A* and *B*, respectively, then, the set coverage quality metric (C-metric) will express the fraction of solutions in one non-dominated solution set that are dominated by those solutions of the second algorithm. The C-metric is defined as:

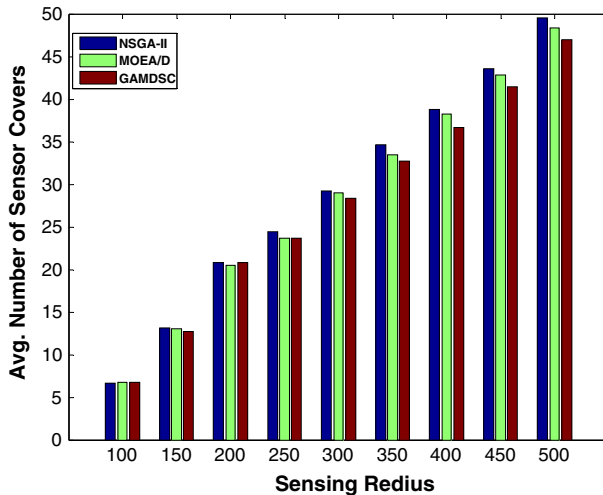
$$C(A, B) = |b \in B | \exists a \in A : a > b| / |B| \tag{8}$$

where *>* reads *dominate*. Note that:

- $C(A, B) \neq 1 - C(B, A)$ , and
- *A* is better than *B* if  $C(A, B)$  is found to be higher than  $C(B, A)$  over many trials.



**Fig. 1** Average number of sensor covers for 90 networks with  $R_s = 500$  and number of sensors = {50, 55, ..., 90}. In each group of bars: left, middle, and right bar corresponds to NSGA-II, MOEA/D, and GAMDSC results, respectively



**Fig. 2** Average number of sensor covers for 90 networks with number of sensors = 70 and  $R_s = \{100, 150, \dots, 500\}$ . In each group of bars: left, middle, and right bar corresponds to NSGA-II, MOEA/D, and GAMDSC results, respectively

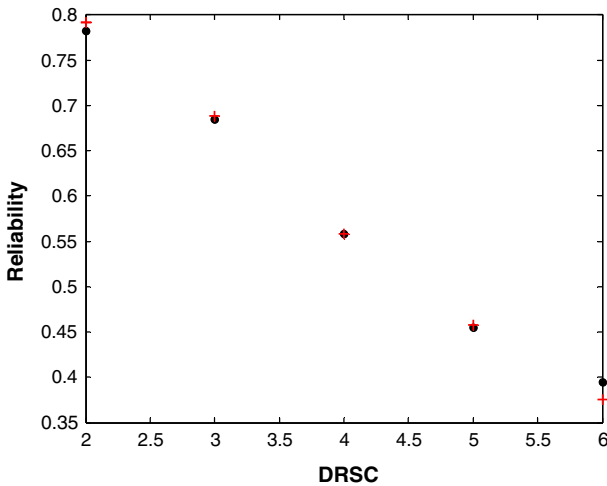
The second measure is the domination metric *Dom* which is defined as:

$$Dom(A, B) = \frac{d(A, B)}{d(A, B) + d(B, A)} \tag{9}$$

where  $d(X, Y) = \sum_{x \in X} |\{y \in Y | x \succ y\}|$ , is known as the domination factor. Mutually non-dominating pairs are ignored in calculating the dominance factor  $d(A, B)$ . If each solution in *A* dominates every solution in *B*, then  $Dom(A, B) = 1$ . Thus,  $Dom(A, B) = 1 - Dom(B, A)$ . Table 3 presents these two metrics when comparing the quality of non-

**Table 3** Comparison of coverage set and domination metric of *MOEA/D* versus *NSGA-II* for 10 WSNs in each test case

Test instance	No. of sensors	$R_s$	$C(A, B)$	$C(B, A)$	$Dom(A, B)$	$Dom(B, A)$
1	50	100	<b>0.4897</b>	0.4100	<b>0.5168</b>	0.4832
2		200	0.3895	<b>0.4998</b>	0.4466	<b>0.5534</b>
3		300	0.2609	<b>0.6408</b>	0.3059	<b>0.6941</b>
4	60	100	<b>0.4450</b>	0.3226	<b>0.5386</b>	0.4614
5		200	0.3137	<b>0.5704</b>	0.3595	<b>0.6405</b>
6		300	0.2330	<b>0.6198</b>	0.2844	<b>0.7156</b>
7	70	100	0.3185	<b>0.5356</b>	0.4022	<b>0.5978</b>
8		200	0.2277	<b>0.6786</b>	0.2632	<b>0.7368</b>
9		300	0.2327	<b>0.5813</b>	0.3387	<b>0.6613</b>
10	80	100	0.3829	<b>0.4071</b>	0.4981	<b>0.5019</b>
11		200	0.3152	<b>0.4757</b>	0.4468	<b>0.5532</b>
12		300	0.1732	<b>0.6750</b>	0.2125	<b>0.7875</b>
13	90	100	0.2883	<b>0.6171</b>	0.2994	<b>0.7006</b>
14		200	0.3394	<b>0.4477</b>	0.4597	<b>0.5403</b>
15		300	0.2394	<b>0.5576</b>	0.2664	<b>0.7336</b>

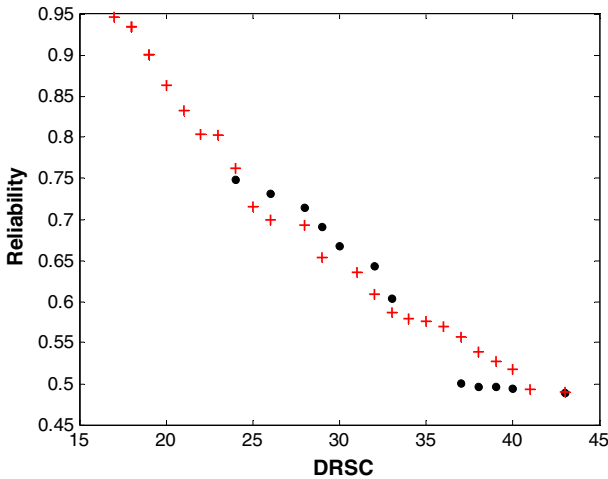


**Fig. 3** Pareto solutions of *MOEA/D* (black ball) and *NSGA-II* (+), for one network with number of sensors = 50 and  $R_s = 100$

dominated set obtained by *MOEA/D* (denoted as *A*) against those of *NSGA-II* (denoted as *B*). Also, here best results are given in bold.

From the results, we can consider that on overall, *NSGA-II* performs better than *MOEA/D*, except for two test instances (i.e., test instance 1 and 4) where there is a slight difference between the solutions quality attained by the two algorithms.

For the purpose of illustration, Figs. 3 and 4 depict the non-dominated solution sets provided by the both algorithms for two WSNs belong to the two extreme test instances experimented in our simulations (i.e. test instance 1 and 15, respectively).  $x, y$  – coordinates



**Fig. 4** Pareto solutions of *MOEA/D* (black ball) and *NSGA-II* (+), for one network with number of sensors = 90 and  $R_s = 300$

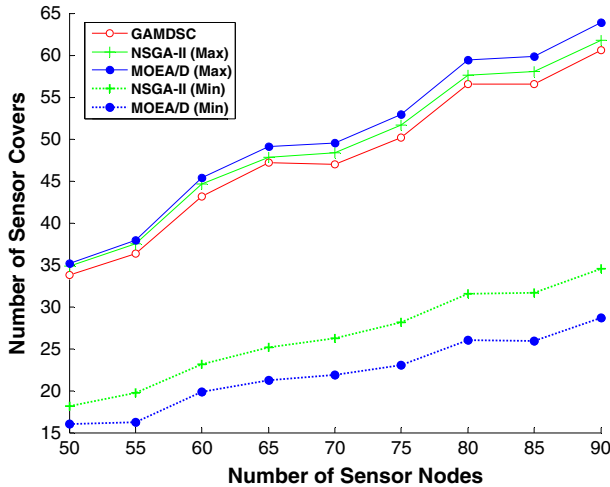
in the figures correspond to number of set covers (DRSC) and total coverage reliability. In the first test instance, the average number of upper bound of sensor covers  $\overline{ub}$  is 5.00. Projection in Fig. 3 shows the cardinality and quality of both *MOEA/D*'s and *NSGA-II*'s Pareto Fronts. The number of non-dominated solutions being obtained by both algorithms is similar (i.e.,  $|A| = |B| = 5$ ). Also, we can see that both *A* and *B* have nearly equal quality.

However, the dissimilarity in the quality and number of solutions in *A* and *B* becomes more apparent while increasing  $\overline{ub}$  to its extreme in the last test instance (i.e., 42.10). For this test instance, Fig. 4 clarifies that *NSGA-II* discovers more non-dominated solutions ( $|B| = 24$  against  $|A| = 12$ ). Also, we can see that more than 70% of solutions in *B* are with better quality than those in *A*. On the other hand, about 25% of *MOEA/D* solutions are with better quality than those obtained by *NSGA-II*.

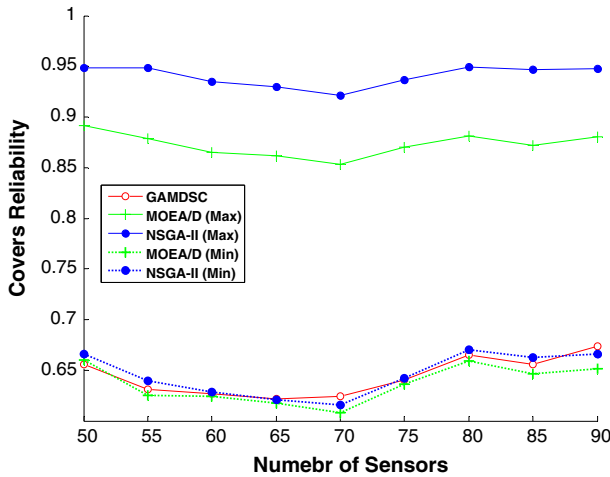
Figures 5 and 6 also compare the quality and diversity (or width) of Pareto Front solutions of both *NSGA-II* and *MOEA/D* in terms of the extremes solutions found there (i.e., minimum and maximum values). Here, 9 simulations examine 90 different WSNs (each simulation with 10 networks) with the same sensing radius setting ( $R_s = 500$ ), but with different number of sensor nodes ( $\{50, 55, \dots, 90\}$ ). Thus, each simulation carries out a couple of extremes averaged over 10 different networks with certain number of sensors and fixed value of  $R_s$ . Figure 5 depicts the extremes in terms of number of sensor covers, while Fig. 6 depicts the corresponding extremes' coverage reliability (covers reliability). For complete illustration, we also present the best results delivered by *GAMDSC* in both figures. Moreover, for more clarity, the overall statistics of *NSGA-II*, *MOEA/D* and *GAMDSC* are presented in Table 4.

According to the tradeoff between the two optimization objectives (i.e., number of sensor covers and covers reliability), an extreme solution (in Fig. 5) which is dedicated on maximizing the number of sensor covers is connected to the extreme solution (in Fig. 6) that is dedicated to the minimum coverage reliability and vice versa.

From the results of both figures, we can see that *NSGA-II* provides a wider Pareto Front than *MOEA/D* and with better quality. The detailed results of Fig. 5 clarifies that the average distance between the extremes of *NSGA-II* is about 28.26 sensor covers, while for the extremes of *MOEA/D* is 22.68 sensor covers. Also, the results reveals that the maximum



**Fig. 5** Pareto extreme solutions (in terms of number of sensor covers) of *MOEA/D* (+) and *NSGA-II* (filled circle). *GAMDSC*'s best solutions (empty circle) are also included in the figure. The simulation is experimented under 90 networks with  $R_s = 500$  and number of sensors = {50, 55, ..., 90}



**Fig. 6** Pareto extreme solutions (in terms of covers reliability) of *MOEA/D* (+) and *NSGA-II* (filled circle). *GAMDSC*'s best solutions (empty circle) are also included in the figure. The simulation is experimented under 90 networks with  $R_s = 500$  and number of sensors = {50, 55, ..., 90}

number of sensor covers provided by *MOEA/D* ranges from 34.8 to 61.8 with an average of 49.12. On the other hand, *NSGA-II* provides more number of sensor covers ranges from 35.2 to 63.9 with average of 50.34. Both *MOEA/D* and *NSGA-II* gain more sensor covers than *GAMDSC*. In Fig. 5, we can see that *GAMDSC* can provide from 33.8 to 60.6 with average of 47.91 sensor covers.

Similarly, Fig. 6 shows that *NSGA-II* provides wider Pareto Front than *MOEA/D*. Also, in Fig. 6 we can see that all algorithms have nearly equal reliability performance represented by *min* extremes of *MOEA/D* and *NSGA-II* and by the best results of *GAMDSC*. Knowing that



**Table 4** Overall statistics of Pareto extremes of MOEA/D and NSGA-II and the best solutions of GAMDSC given in Figs. 5 and 6

Objective	Algorithm	Extreme	max	min	average
No. of sensor covers	GAMDSC	–	60.6	33.8	47.91
	MOEA/D	<i>max</i>	61.8	34.8	49.12
		<i>min</i>	34.5	18.1	26.44
	NSGA-II	<i>max</i>	63.9	35.2	50.34
		<i>min</i>	28.7	16	22.08
Covers reliability	GAMDSC	–	0.6737	0.6217	0.644
	MOEA/D	<i>max</i>	0.8913	0.853	0.8726
		<i>min</i>	0.6602	0.6077	0.6364
	NSGA-II	<i>max</i>	0.949	0.9211	0.9403
		<i>min</i>	0.6699	0.6156	0.6456

these reliability results are associated with the *max* extremes (in terms of number of sensor covers) of MOEA/D and NSGA-II and with the best results of GAMDSC in Fig. 5, we can then realize that NSGA-II facilitate more qualitative sensor covers. In contrast, GAMDSC is incurred by the lowest number of sensor covers. This reflects the positive impact of the proposed heuristic perturbation operator over the combined collaboration of crossover, mutation, and the scattering operators of GAMDSC algorithm.

### 4.3 Comparison of Worst Case Time Complexity

We also compare the worst-case computational complexity of the single-objective GAMDSC algorithm and the multi-objective MOEA/D and NSGA-II algorithms for solving Disjoint Reliable Sensor Covers Problem (DRSC). Due to optimizing multiple objectives instead of only one objective, multi-objective evolutionary algorithms, in general, are more expensive than single-objective algorithms. Without loss of generality, the computational complexity of single-objective genetic algorithms, such as GAMDSC is  $O(max_i \times K)$  where  $K$  is the size of solutions to be evolved, via certain evolution operators, for  $max_i$  iterations. On the other hand, the general computational complexity of an MOEA/D is  $O(max_i \times 2 \times K \times K_{neighbor})$  (where 2 means number of objective functions and  $K_{neighbor} < K$  means size of neighboring solutions for each individual solution) [35]. The computational complexity of NSGA-II is  $O(max_i \times 2 \times K^2)$  [32,33].

Now, let us consider the computation time needed for the most critical parts of the above algorithms when applied for solving DRSC problem. It is stated in [15] that in each generation of GAMDSC, the fitness function is the most critical part which represents the computation time bottleneck. The evaluation of the fitness function is linearly related to the number of sensors and the number of targets. Thus, the worst-case time complexity of GAMDSC is:

$$O(GAMDSC) = max_i \times K \times |n| \times |m| \tag{10}$$

where  $|n|$  is the number of targets and  $|m|$  is the number of sensors. On the other hand, for MOEA/D and NSGA-II, the proposed heuristic perturbation operator (Algorithm 1) costs the most computation time and in the worst-case time it linearly related to the maximum number of sensors and targets, and upper bound of sensor covers *ub*. Thus, the worst-case

time complexity of MOEA/D and NSGA-II are, respectively, as:

$$O(\text{MOEA/D}) = \max_i \times 2 \times K \times K_{neighbor} \times |n| \times |m| \times |ub| \quad (11)$$

$$O(\text{NSGA-II}) = \max_i \times 2 \times K^2 \times |n| \times |m| \times |ub| \quad (12)$$

## 5 Conclusions

Maximum DSC is a well-known WSN design which is NP-hard. Unlike all of the existing work that rely on the traditional single-objective definition, this paper introduces a *multi-objective Disjoint Reliable Sensor Covers Problem* (DRSC). The formulated problem is then tackled by two well-known and competitor multi-objective evolutionary algorithms, namely MOEA/D and NSGA-II. Instead of the traditional use of two perturbation operators (i.e., crossover and mutation), one heuristic operator is proposed that involves the incorporation of *problem-specific knowledge* in its mechanism. The proposed heuristic operator is proved to be of positive effect on the overall performance of the implemented MOEAs. The results over 150 different WSN topologies signifies the importance of utilizing the two contradictory criteria of getting large number of disjoint sensor covers and more coverage reliability and optimizing them simultaneously rather than optimizing only number of disjoint sensor covers. Good overall performance of the multi-objective algorithms for the modeled DRSC highlights that it may be motivated to consider another WSN's design criterion while solving disjoint sensor covers problem. As an example, we can include communication constraints (e.g., sensor-sink connectivity) to the multi-objective definition of disjoint sensor covers. Also, by satisfying the sensor energy constraints, a sensor can participate in multiple sensor covers and the definition of DRSC could be reformulated to satisfy a multi-objective non-disjoint sensor covers (*nDRSC*) problem.

**Acknowledgments** This work is partially supported by TUBITAK under Grant No. 113E328.

## References

1. Garey, M. R., & Johnson, D. S. (1979). *Computers and intractability. A guide to the theory of NP-completeness*. New York: Freeman.
2. Slijepcevic, S., & Potkonjak, M. (2001). Power efficient organization of wireless sensor networks. In *Proceedings of IEEE international conference communications*, Vol. 2. Finland, pp. 472–476.
3. Cardei, M., Thai, M., Li, Y., & Wu, W. (2005). Energy-efficient target coverage in wireless sensor networks. In *IEEE INFOCOM 2005*, Mar. 2005, pp. 1976–1984.
4. Tian, D., & Georganas, N. D. (2002). A coverage-preserving node scheduling scheme for large wireless sensor networks. In *Proceedings of 1st association computing machinery international workshop wireless sensor networks and applications*, pp. 32–41.
5. Wang, X., Xing, G., Zhang, Y., Lu, C., Pless, R., & Gill, C. (2003). Integrated coverage and connectivity configuration in wireless sensor networks. In *Proceedings of 1st international conference on embedded network sensor systems*, Los Angeles, CA, pp. 28–39.
6. Liu, Y., & Liang, W. (2005). Approximate coverage in wireless sensor networks. In *Proceedings of IEEE conference on local computing network, 30th anniversary*, pp. 68–75.
7. Gupta, H., Zhou, Z., Das, S. R., & Gu, Q. (2006). Connected sensor cover: Self-organization of sensor networks for efficient query execution. *IEEE, Association for Computing Machinery Transactions on Networking*, 14(1), 55–67.
8. Ai, J., & Abouzeid, A. A. (2006). Coverage by directional sensors in randomly deployed wireless sensor networks. *Journal of Combinatorial Optimization*, 11(1), 21–41.
9. Funke, S., Kesselman, A., Kuhn, F., Lotker, Z., & Segal, M. (2007). Improved approximation algorithms for connected sensor cover. *Wireless Networks*, 13(2), 153–164.

10. Zhou, Z., Das, S. R., & Gupta, H. (2009). Variable radii connected sensor cover in sensor networks. *Association Computing Machinery Transactions on Sensor Networks*, 5(1, article 8), 1–36.
11. Martins, F. V. C., Carrano, E. G., Wanner, E. F., Takahashi, R. H. C., & Mateus, G. R. (2009). A dynamic multiobjective hybrid approach for designing wireless sensor networks. In *Proceedings on IEEE congress on evolutionary computation*, pp. 1145–1152.
12. Cardei, M., & Du, D.-Z. (2005). Improving wireless sensor network lifetime through power aware organization. *Wireless Networks*, 11(3), 333–340.
13. Cardei, I., & Cardei, M. (2008). Energy-efficient connected-coverage in wireless sensor networks. *International Journal on Sensor Networks*, 3(3), 201–210.
14. He, J., Xiong, N., Xiao, Y., & Pan, Y. (2010). A reliable energy efficient algorithm for target coverage in wireless sensor networks. In *IEEE 30th international conference on distributed computing systems workshops (ICDCSW)*, 2010, pp. 180–188.
15. Lai, C., Ting, C., & Ko, R. (2007). An effective genetic algorithm to improve wireless sensor network lifetime for large-scale surveillance applications. In *IEEE congress on evolutionary computation, 2007. CEC 2007*. IEEE, 2007.
16. Hu, X. M., Zhang, J., Yu, Y., Chung, H. H., Li, Y. L., Shi, Y. H., et al. (2010). Hybrid genetic algorithm using a forward encoding scheme for lifetime maximization of wireless sensor networks. *IEEE Transactions on Evolutionary Computation*, 14(5), 766–781.
17. Gil, J. M., & Han, Y. H. (2011). A target coverage scheduling scheme based on genetic algorithms in directional sensor networks. *Sensors* 11, 11(2), 1888–1906.
18. Huang, C.-F., & Tseng, Y.-C. (2005). The coverage problem in a wireless sensor network. *Mobile Networks and Applications*, 10(4), 519–528.
19. Elfes, A. (1987). Sonar-based real-world mapping and navigation. *IEEE Journal of Robotics and Automation*, RA-3(3), 249–265.
20. Ghosh, A., & Das, S. K. (2006). Coverage and connectivity issues in wireless sensor networks. In *Mobile, wireless, and sensor networks: Technology, applications, and future directions*, pp. 221–256.
21. Hossain, A., Biswas, P. K., & Chakrabarti, S. (2008). Sensing models and its impact on network coverage in wireless sensor network. In *IEEE region 10 and the third international conference on Industrial and information systems, 2008. ICIIS 2008*, (pp. 1–5). IEEE.
22. Rajagopalan, R. (2010). A multi-objective optimization approach for data fusion in mobile agent based distributed sensor networks. In *Instrumentation and measurement technology Conference (I2MTC)*, 2010 IEEE, pp. 208–212.
23. Rajagopalan, R., Mohan, C. K., Varshney, P., & Mehrotra, K. (2005). Multi-objective mobile agent routing in wireless sensor networks. In *The 2005 IEEE congress on evolutionary computation, 2005*, Vol. 2, pp. 1730–1737.
24. Özdemir, S., Attea, B. A., & Khalil, Ö. A. (2012). *Multi-objective clustered-based routing with coverage control in wireless sensor networks*. *Soft computing*, (pp. 1–12). Berlin: Springer. doi:10.1007/s00500-012-0970-x.
25. Özdemir, S., Attea, B. A., & Khalil, Ö. A. (2012). *Multi-objective evolutionary algorithm based on decomposition for energy efficient coverage in wireless sensor networks*. *Wireless personal communications*. Berlin: Springer. doi:10.1007/s11277-012-0811-3.
26. Konstantinidis, A. (2009). *Multiobjective deployment and power assignment in wireless sensor networks using metaheuristics*. University of Essex.
27. Jia, J., Chen, J., Chang, G., Wen, Y., & Song, J. (2009). Multi-objective optimization for coverage control in wireless sensor network with adjustable sensing radius. *Computers and Mathematics with Applications*, 57(11), 1767–1775.
28. Konstantinidis, A., Yang, K., Zhang, Q., & Zeinalipour-Yazti, D. (2009). A multi-objective evolutionary algorithm for the deployment and power assignment problem in wireless sensor networks. *Computer Networks*, 54(6), 960–976.
29. Konstantinidis, A., & Yang, K. (2011). Multi-objective energy-efficient dense deployment in wireless sensor networks using a hybrid problem-specific MOEA/D. *Applied Soft Computing*, 11(6), 4117–4134.
30. Abdulhalim, M. F., & Attea, B. A. (2014). *Multi-layer genetic algorithm for maximum disjoint reliable set covers problem in wireless sensor networks*. Accepted in *Wireless Personal Communications*. Berlin: Springer.
31. Coello Coello, C. A., Van Veldhuizen, D. A., & Lamont, G. B. (2002). *Evolutionary algorithms for solving multi-objective problems*. New York: Kluwer.
32. Srinivas, N., & Deb, K. (1994). Multi-objective function optimization using non-dominated sorting genetic algorithms. *Evolutionary Computation*, 2(3), 221–248.
33. Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2000). A fast and elitist multi-objective genetic algorithm: NSGA-II. In *Proceedings parallel problem solving from nature VI*, pp. 849–858.

34. Deb, K. (2001). *Multi-objective optimization using evolutionary algorithms*. Chichester: Wiley.
35. Zhang, Q., & Li, H. (2007). MOEA/D: A multi-objective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation*, 11(6), 712–731.
36. Zitzler, E. (1999). *Evolutionary algorithms for multi-objective optimization: Methods and applications*. Doctoral dissertation, Zurich: Swiss Federal Institute of Technology.



**Bara'a A. Attea** (a.k.a. Baraa A. Atiyah) received the BS and MS degrees in computer science from University of Baghdad, Baghdad, Iraq in 1993 and 1996, respectively, and the PhD degree in computer science from University of Technology, Baghdad in 2002. She is an associate professor in Computer Science Department - University of Baghdad. She was a visiting researcher in Gazi University, Ankara, Turkey from 2011 to 2012. Her main research interests are computational intelligence, evolutionary algorithms, data mining, and applications of bio-inspired algorithms in wireless sensor networks.



**Enan A. Khalil** received the BS, and MS in computer science from University of Baghdad, Baghdad, Iraq in 2008 and 2011, respectively. He is currently a PhD student in Computer Engineering Department - Gazi University, Ankara, Turkey. His main interests are computational intelligence, evolutionary algorithms, and wireless sensor networks.



**Suat Özdemir** has been with the Computer Engineering Department in Gazi University, Ankara, Turkey. He received his MSc degree from Syracuse University and PhD degree from Arizona State University in 2001 and 2006, respectively. Dr. Ozdemir's research areas mainly include wireless and sensor networks, network security, and data mining. He is a member of IEEE and currently serving as editorial board or TPC member for various leading IEEE and ACM journals and conferences.



**Oktay Yıldız** received his M.Sc. degree in Institute of Science from Gazi University, in 2004 and Ph.D. degree in Institute of Information Sciences from Gazi University, in 2012. He has been with the Computer Engineering Department at Gazi University, Ankara, Turkey since 2009. His research interests include machine learning, data mining and bioinformatics.