A Genetic Algorithm for Minimum Set Covering Problem in Reliable and Efficient Wireless Sensor Networks

Bara'a A. Attea*, Sarab M. Hameed
Department of Computer Science, College of Science, Baghdad University, Baghdad, Iraq.

Abstract

Densely deployment of sensors is generally employed in wireless sensor networks (WSNs) to ensure energy-efficient covering of a target area. Many sensors scheduling techniques have been recently proposed for designing such energy-efficient WSNs. Sensors scheduling has been modeled, in the literature, as a generalization of minimum set covering problem (MSCP) problem. MSCP is a well-known NP-hard optimization problem used to model a large range of problems arising from scheduling, manufacturing, service planning, information retrieval, etc. In this paper, the MSCP is modeled to design an energy-efficient wireless sensor networks (WSNs) that can reliably cover a target area. Unlike other attempts in the literature, which consider only a simple disk sensing model, this paper addresses the problem of scheduling the minimum number of sensors (i.e., finding the minimum set cover) while considering a more realistic sensing model to handle uncertainty into the sensors' target-coverage reliability. The paper investigates the development of a genetic algorithm (GA) whose main ingredient is to maintain scheduling of a minimum number of sensors and thus to support energy-efficient WSNs. With the aid of the remaining unassigned sensors, the reliability of the generated set cover provided by the GA, can further be enhanced by a post-heuristic step. Performance evaluations on solution quality in terms of both sensor cost and coverage reliability are measured through extensive simulations, showing the impact of number of targets, sensor density and sensing radius.

Keywords: Coverage, energy efficiency, genetic algorithm, probabilistic sensing model, SCP, wireless sensor networks.

الخوارزمية الجينية لمشكلة المجموعة الادنى للتغطية في شبكات الاستشعار اللاسلكية الموثوقة والفعالة

براء علي عطيه، سراب مجيد حميد
قسم الحاسبات، كلية العلوم، جامعة بغداد، بغداد، العراق.

الخلاصة

يستخدم عادة نشر أجهزة الاستشعار بكثافة في شبكات الاستشعار اللاسلكية (WSNs) لضمان تغطية كفوءه وأقتصادية في المنطقة المستهدفة. حديثأ تم اقتراح العديد من تقنيات جدولة أجهزة الاستشعار لتصميم WSNs كفوء من ناحية استخدام الطاقة وأصبحت هذه المشكل تعميم لمشكله المجموعة الادنى للتغطيه (MSCP). MSCP هي مشكلة NP-Hard والمطبقة في حل العديد من المشاكل الناجمة مثل التصنيع، وتخطيط الخدمة، واسترجاع المعلومات، وما إلى ذلك من المشاكل.

*Email: baraaali@yahoo.com

في هذ البحث ، تم نمذجة MSCP لتصميم شبكات الاستشعار اللاسلكية (WSNs) والتي يمكن أن تغطي المنطقة المستهدفة بثقة وبطريقة أقتصادية. على عكس محاولات أخرى في هذا المجال، يتناول هذا البحث مشكلة جدولة الحد الأدنى لعدد من أجهزة الاستشعار (إيجاد الحد الأدنى لغطاء مجموعة)، في نموذج الاستشعار أكثر واقعية للتعامل مع حالة عدم اليقين في موثوقية أجهزة الاستشعار في تغطية الهدف. تدارس هذا البحث تطوير الخوارزمية الجينية (GA) للحفاظ على جدوله لعدد أدنى من أجهزة الاستشعار، لدعم WSNs كفوء في استخدام الطاقة . وبمساعدة من أجهزة الاستشعار غير المعينة المتبقية، يمكن زيادة موثوقية تغطية المجموعة التي تقدمها GA بعد خطوة الكشف عن مجريات الأمور. تم قياس تقييم الأداء على نوعية الحل من حيث تكلفة الاستشعار وموثوقية التغطية من خلال محاكاة واسعة النطاق، والتي تبين أثر عدد الأهداف، وكثافة اجهزة الاستشعار ونصف قطر الاستشعار عن بعد.

1.Introduction

Recently, many applications, ranging from remote harsh field monitoring to surveillance and smart homes, have been directed towards studying and building their backbones based on Wireless sensor networks (WSNs). The dense ad-hoc deployment of such sensors from an aircraft into the monitoring area can results in network configurations with adequate targets coverage level. However, recharging or replacing a sensor's battery is generally infeasible. Hence, efficient utilization of the limited energy is one of the critical design considerations in WSNs. Energy-aware mechanism has been substantially pursued by the research community in order to form long lived 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 will consider the third approach to design energy-efficient WSNs while completely monitoring the targets. In this class of techniques, sensor activities are scheduled into disjoint *set covers* (DSC), and each set cover (hereinafter, interchangeably called, *sensor cover*) needs to satisfy the coverage constraints. At each interval of the whole WSN's lifetime, only one sensor cover (*active sensor cover*) is working to provide the required sensing functionality while the remaining sensor covers are in the low-energy *sleep* mode. Once the active sensor cover runs out of energy, another sensor cover will be selected to enter the active mode and provide the functionality continuously. It has been proven that this problem is a generalization of the minimum set cover problem (MSCP) [1] and showing its NP-completeness in [2], [3].

Many attempts in the literature have been proposed for solving disjoint sensor subsets problem in WSNs using either heuristic or meta-heuristic (like genetic algorithms) approaches [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 area. Their 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. The DSC problem has been solved in [12] using integer programming. 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 also 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 non-disjoint set covers (MSC) problem and solved it using, linear programming, and greedy heuristics. 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 work in [15] – [17] also provides solutions to the DSC problem in WSNs but using the meta-heuristic framework of evolutionary and genetic algorithms. Like the previous mentioned heuristic methods, the genetic algorithms (GAs) proposed in [15] – [17] assume simple and common isotropic (i.e., disc) sensing model. Each sensor in this definite range law approximation model is associated with a sensing area which is represented by a circle and it successfully detects anything falling only within its sensing range. In a more realistic scenario, the sensing region of a sensor could be irregular, resulting in imperfect sensor approximation model. The coverage in this case could be expressed in probabilistic terms [18] – [20]. 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 certainty threshold c_{th}, the detection uncertainty of each target should not exceed $1 - c_{th}$.

Unlike other related works, this paper concerns with the applicability of the genetic algorithm for solving the MSCP problem while assuming a probabilistic sensing model to reflect the uncertainty in sensor readings. The main contributions of this paper are as follows:

1. With the de-facto definition of the simple genetic algorithm, a set cover can be identified that should maintain low cost in terms of number of sensors to reliably cover the all the targets within the specified certainty threshold c_{th}.

2. With the incorporation of unassigned sensors, the coverage reliability of the network reliability can be further improved. A post-heuristic operator weighs each assigned and/or unassigned sensor to the membership of the constructed set cover.

In what follow we first briefly describe the MSCP in WSNs and its related system model. Then, in section 3, we introduce the proposed genetic algorithm and a post-heuristic operator tailored for solving MSCP in WSNs. The results of the proposed genetic algorithm are then evaluated in Section 4. Finally, Section 5 concludes the current work and hints some further ramifications.

2.Minimum Reliable Set Cover Problem (MRSCP) in WSNs

In order to model the system, we will assume that the investigated WSNs have 2D sensing area $\mathcal{A}$ with known size (X_{max}, Y_{max}). We will 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 $\mathcal{A}$ $(1 \le \forall i \le m \mid (x_{si}, y_{si}) = ([0, X_{max}], [0, Y_{max}])$. Depending on the sensing range R_s, each sensor will be responsible for sensing and covering a part of $\mathcal{A}$. We will consider a probabilistic sensing model [19], [20] to define the notion of the probabilistic coverage of a target $\mathcal{T}_j = (x_{tj}, y_{tj})$ by a sensor s_i.

$$Coverage(s_i, t_j) = \begin{cases} 0 & if\ R_s + R_u \le d(s_i, t_j) \\ e^{-\lambda a^{\beta}} & if\ R_s - R_u < d(s_i, t_j) < R_s + R_u \\ 1 & if\ R_s - R_u \ge d(s_i, t_j) \end{cases} \quad (1)$$

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 λ and β 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 (see Figure 1).

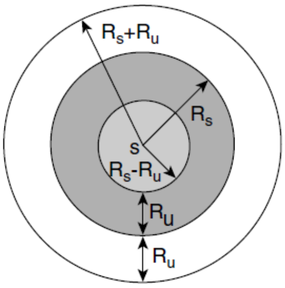

Figure 1 -Probabilistic sensing model.

To save energy, a subset of sensors from the sensor set $\mathcal{S}$ should be activated into a duty-cycling *sensor cover* subset, to cover all the interested targets in $\mathcal{T}$. In the literature and under the traditional Boolean sensing model, the definition of the sensor cover could be formulated as:

Definition 1: (**Sensor Cover - SC**). Given a WSN consists of target set $\mathcal{T}$ and sensor set $\mathcal{S}$, where each sensor $s_i \in S$ can be represented as a subset $\mathcal{T}_i \subset \mathcal{T}$, such that $tj \in \mathcal{T}_i$ 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, here, as:

Definition 2 (**Reliable Sensor Cover - RSC**). Given a WSN consists of target set $\mathcal{T}$ and sensor set $\mathcal{S}$, where each sensor $s_i \in S$ can be represented as a subset $\mathcal{T}_i \subset \mathcal{T}$, such that $tj \in \mathcal{T}_i$ 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. Formally speaking:

$$Cover(\mathcal{S}_i, \mathcal{T}) = \begin{cases} 1 \; if \; \forall t \in \mathcal{T} \rightarrow \; \exists s \in \; \mathcal{S}_i | Coverage(s, t) \geq c_{th} \\ 0 \; otherwise \end{cases} \quad (2)$$

Now, the problem of finding the minimum number of sensors that reliably cover all the targets (here we called it minimum reliable set cover problem - MRSCP) can be formulated as:

Definition 3 (**Minimum Reliable Sensor Cover Problem - MRSCP**). Given a collection $\mathcal{S}$ of sensors, find the minimum number of sensors that reliably covers $\mathcal{T}$. Every cover $\mathcal{S}_i$ is a subset of $\mathcal{S}$, $\mathcal{S}_i \subseteq \mathcal{S}$, such that every element tj of $\mathcal{T}$ belongs to at least one member of $\mathcal{S}_i$. A cover $\mathcal{S}_{min}$ is said to contain a minimum number of sensors if for any other cover $\mathcal{S}_i$, $|\mathcal{S}_{min}| < |\mathcal{S}_i|_i$. Formally speaking:

$$|\mathcal{S}_{min}| = \underset{i=1,2,\dots}{\arg\min}(|\mathcal{S}_i|) \quad (3)$$

The proposed Genetic Algorithm

The GA simulates the biological processes of natural selection, reproduction, and mutation to iteratively evolve species of individual solutions to become more and more adapted to the problem environment. The proposed GA can be described as a process formulated in the following formula-fashion. Let $GA \;: \mathcal{S} \rightarrow \{\mathcal{S}_{sleep}, \mathcal{S}_{active}\}$, be the process that iteratively evolve a population ρ of solutions, using genetic operators, toward the best *set cover* solution in terms of minimum number of sensors (i.e., sensor cost) that reliably cover all targets. Thus, the objective function Φ of GA can be formulated for the minimum set cover problem as:

$$\Phi^i: Minimize \; |\mathcal{S}_{active}| \quad (4)$$

The best set cover solution will show up both active and sleep sensor sets.

3.1 Space and Solution Configurations

The choice of a good solution representation is a critical issue for the applicability and performance of any evolutionary algorithm. Solution representation is highly problem dependent and related to the evolution operations. In our algorithm design, each individual solution $\mathbb{P}$ of GA is represented as a fixed-length vector of size $m = |\mathcal{S}|$, where each $gene_j$ controls the *active/sleep scheduling* of the corresponding i^{th} sensor in $\mathcal{S}$. Thus, $\forall k \in \{1, \dots, K\} \; and \; \forall i \in \{1, \dots, m\}: \mathbb{P}_k = \left(\mathbb{P}_{k,1}, \mathbb{P}_{k,2}, \dots, \mathbb{P}_{k,m}\right) \; s.t.:$

$$\mathbb{P}_{k,i} = \begin{Bmatrix} 1, & if \; s_i \; is \; active \\ 0, & if \; s_i \; is \; sleep \end{Bmatrix} \quad (5)$$

Then, the whole configuration space $\mathcal{S}$ for the GA can be created by the Cartesian product of activation/inactivation of all m unassigned sensors:

$$\mathcal{S} = \prod_{i=1}^{m}(\{0,1\}) = 2^m \qquad (6)$$

where 0 means inactive (i.e., unassigned) sensor, while 1 means active (i.e., assigned) sensor. Let us to consider that GA handles only $K \ll \mathcal{S}$ different individual solutions at a time. It starts with an initial random population $\rho_0 \subset \mathcal{S}, |\rho_0| = K$ and continues until a maximum number of generations max_{gen} has been reached. Each generation $\Psi: \rho \to \rho'$ consists of four main operators: individual repair, parent selection, crossover, and mutation. Thus, Ψ can be decomposed into:

$$\Psi = \Psi_{rep} \circ \Psi_{sel} \circ \Psi_{x} \circ \Psi_{mu} \qquad (7)$$

3.2 Repair Operator and the Fitness Function

Before evaluating each individual, infeasible set cover solutions should be transformed into feasible ones by means of a problem-specific repair operator. Infeasible solutions are those which suffer from either the existence of coverage-holes or let the targets to be over-covered by more than need sensors. The main idea of the proposed repair operator is to make hole-free targets coverage with as less number of sensors as possible. The process of the proposed repair operator Ψ_{rep} is presented next (see Algorithm 1).

$$\Psi_{rep}: \mathbb{P}_k \to \mathbb{P}_k{}' \qquad (8)$$

It takes as input the individual $\mathbb{P}_k, 1 \le k \le K$ and the set of unassigned sensors $\mathcal{S}_{sleep}$. First, it check whether the active sensors set $\mathcal{S}_k$ selected by $\mathbb{P}_k$ (i.e., $\mathcal{S}_k = \{s_i | \mathbb{P}_{ki} = 1\}$) forms coverage-hole or dense-coverage under the user-specified reliability threshold c_{th}. In case of coverage-hole, Ψ_{rep} will randomly draw from $\mathcal{S}_{sleep}$ set one sensor at a time and collect it with $\mathcal{S}_k$ (i.e., $\mathcal{S}_k = \mathcal{S}_k \cup s | s \in \mathcal{S}_{sleep}$ and $\mathcal{S}_{sleep} = \mathcal{S}_{sleep} - s$) until the new set form hole-free set cover. On the other hand, if $\mathcal{S}_k$forms dense-coverage, Ψ_{rep} will randomly deactivate one sensor at a time (i.e., $\mathcal{S}_k = \mathcal{S}_k - s | s \in \mathcal{S}_k$ and $\mathcal{S}_{sleep} = \mathcal{S}_{sleep} + s$) until it can form complete coverage with less number of sensors.

Algorithm 1: Repair Operator ($\mathbb{P}_k$, $\mathcal{S}_{sleep}$, $\mathbb{P}_k{}'$)

set S to the active sensors selected by $\mathbb{P}_k$
$\forall i \in [1, m]: S \leftarrow \{\mathbb{P}_{k,i} | \mathbb{P}_{k,i} = 1\}$
if $Cover(S, \mathcal{T}) == 1$ /* no coverage hole */
  while $Cover(S, \mathcal{T}) == 1$
    select a random sensor $s \in S$
    set $S \leftarrow S - s$
    set $\mathcal{S}_{sleep} \leftarrow \mathcal{S}_{sleep} \cup s$
  end while
  set $S \leftarrow S \cup s$ /* now $Cover(S, \mathcal{T})$ =1 */
else /* coverage holes exist after using the sensors in S */
  while $Cover(S, \mathcal{T})$ ==0
    select a random sensor $s \in \mathcal{S}_{sleep}$
    set $S \leftarrow S \cup s$
    set $\mathcal{S}_{sleep} \leftarrow \mathcal{S}_{sleep} - s$
  end while
end if

Then, to evaluate each individual solution$\mathbb{P}_k$, the fitness function Φ simply sums the number of active sensors being selected by the corresponding solution.

$$\forall k \in [1, K]$$
$$\Phi(\mathbb{P}_k) = \sum_{i=1}^{m} \mathbb{P}_{k,i} \qquad (9)$$

3.3 Selection, Crossover, and Mutation Operators

The remaining genetic operators follow the de-facto standard operators found in the simple genetic algorithms. The binary tournament selection operator is used to choose one of two random individuals $\mathbb{P}_1$ and $\mathbb{P}_2$. A proportion $p_c = 0.6$ of pairs of parents are then selected for crossover. Two cut points $r1, r2 \sim \{1, \dots, m-1\}$, are randomly selected, and the participating parent individuals, $\mathbb{P}_1$ and $\mathbb{P}_2$ are then swapped at $gene_i | r1 \le i \le r2$. Each $gene_i | 1 \le i \le m$ in the new individuals is then mutated with small probability $p_m = 0.1$.

$$\forall k \in \{1, \dots, K\}$$
$$\Psi_{sel}: \{\mathbb{P}_{k1}, \mathbb{P}_{k2}\} \rightarrow \mathbb{P}_k \quad (10)$$
$$\forall k \in \{1, \dots, K/2\}$$
$$\Psi_x : \{\mathbb{P}_{k1}, \mathbb{P}_{k2}\} \rightarrow \{\mathbb{P}_{k1}', \mathbb{P}_{k2}'\}$$
$$\mathbb{P}_{k1}' = \left(\mathbb{P}_{k1,1}, \dots, \mathbb{P}_{k1,r1}, \mathbb{P}_{k2,r1+1}, \dots, \mathbb{P}_{k2,r2}, \mathbb{P}_{k1,r2+1}, \dots, \mathbb{P}_{k1,m^i}\right) \quad (11)$$
$$\mathbb{P}_{k2}' = \left(\mathbb{P}_{k2,1}, \dots, \mathbb{P}_{k2,r1}, \mathbb{P}_{k1,r1+1}, \dots, \mathbb{P}_{k1,r2}, \mathbb{P}_{k2,r2+1}, \dots, \mathbb{P}_{k2,m^i}\right)$$

$$\forall k \in \{1, \dots, K\} \wedge \forall j \in \{1, \dots, m\}$$
$$\Psi_{mu}: \mathbb{P}_k \rightarrow \mathbb{P}_k'$$
$$\mathbb{P}_{k,i} = \begin{cases} \mathbb{P}_{k,i} \; if \; r > p_m \\ 1 - \mathbb{P}_{k,i} \; if \; r \le p_m \end{cases} \quad (12)$$

The mechanisms of the genetic operators being defined by repair, fitness evaluation, selection, crossover and mutation transform a complete population of solutions into another complete population and after a specified number of generations max_{gen}, the best individual solution (in terms of minimum Φ) will be produced. The definition of the best individual can be formulated as:

$$Best\mathbb{P}: \Leftrightarrow \nexists \mathbb{P} \in \rho_{max_{gen}} | \; \Phi(\mathbb{P} \;\;) < \Phi(Best\mathbb{P} \;\;) \quad (13)$$

3.4 Post-heuristic Operator

The best solution $Best\mathbb{P}$ provided by the GA can further be improved in terms of coverage reliability by forwarding it to a post-heuristic operator dedicated for this purpose. Algorithm 2 presents the steps of this heuristic. It operates by exploiting the existing unassigned sensors (being gathered in $\mathcal{S}_{sleep}$) and/or replacing the existing active sensors (being gathered in $Best\mathbb{P}$).

Algorithm 2: Post-Heuristic ($Best\mathbb{P}, \mathcal{S}_{sleep}$)

```
1:   /* a new cover S_active will be formed */
     set S_active = ∅
2:   set S to the sensors from both BestP and S_sleep
     S ← {BestP_j | BestP_j = 1} ∪ {S_sleep}
3:   set contributed sensors set S' ← ∅
4:   set Target ← T
5:   while Target ≠ ∅
6:      select a sensor s ∈ S that contributes to the
7:      most reliable coverage to a target t ∈ T
8:      set Target ← Target − t; S' ← S' ∪ s
9:   end while
10:  remove S' → S_active
11:  remove {S − S'} → S_sleep
```

4 Performance Evaluations

In this section we will measure the performance of the proposed GA for solving MSCP in WSNs. The evaluation is presented in terms of number of active sensors obtained (i.e., sensors cost), and coverage reliability. The results are obtained after setting WSNs and algorithm parameters into the following. The simulation area is square-shaped with side length $X_{max} = 1000m$. The simulation is divided according to:

1. Five different settings for the number of targets $n = \{10,20,30,40,50\}$.
2. Three different settings of sensor density: $m = \{100,125,150\}$.
3. For each test instance group composed from 1 and 2, we will vary the sensing range of the sensor nodes R_s to eight different values $\{100,200, \ldots,800\}$.

Thus, we have a total of 120 different test instances (composed from 1, 2 and 3). Each test instance $TI^i, i = 1, \ldots,120$ includes 10 random WSNs with different configurations. Thus the overall simulation examines a total of 1200 random networks. Uncertainty level R_u is set to $R_s * 0.5$ units, both λ and β 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. As studying the impact of varying these parameters is out of the scope of this paper, we fixed these parameters to one setting. Population size is set to 50 and will be allowed to evolve 500 times.

4.1 Impact of WSN's n, m and R_s Parameters on Solution Quality

First, figures 2-6 depict the sensors cost percentage $SC\%$ while varying number of targets n and sensing range R_s for the three different settings of sensor density. $SC\%$ is defined as the ratio between the number of active sensors used in the generated best individual solution $\mathbb{B}\text{est}\mathbb{P}$ and the total number of sensor nodes m, i.e.,:

$$SC = \frac{|\cup_{s \in \mathbb{B}\text{est}\mathbb{P}} s = 1|}{m} * 100\% \qquad (14)$$

For further qualitative presentation, figures 6 and 7 depicts $SC\%$ for the three different settings of sensor density, i.e., $m = 100,125$ and 150 and the eight settings of sensing range, while fixing target size to its extreme values, i.e., 10 and 50. Figures 9 – 11 qualitatively depict the whole performance of the proposed GA for 1200 WSNs, where results are projected in 3D space.

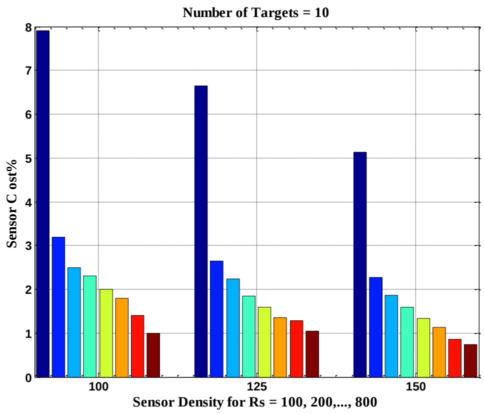

Figure 2- Percentage of sensors cost for 240 WSNs, where $number\ of\ sensors\ m = \{100,125,150\}$, $R_s = \{100,200, \ldots,800\}$ and $number\ of\ targets\ n = 10$.

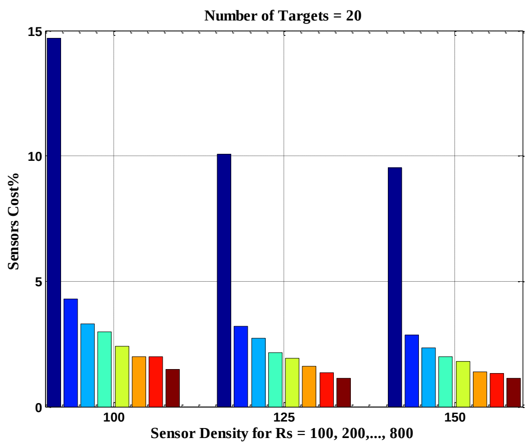

Figure 3- Percentage of sensors cost for 240 WSNs, where $number\ of\ sensors\ m = \{100,125,150\}$, $R_s = \{100,200,\ldots,800\}$ and $number\ of\ targets\ n = 20$.

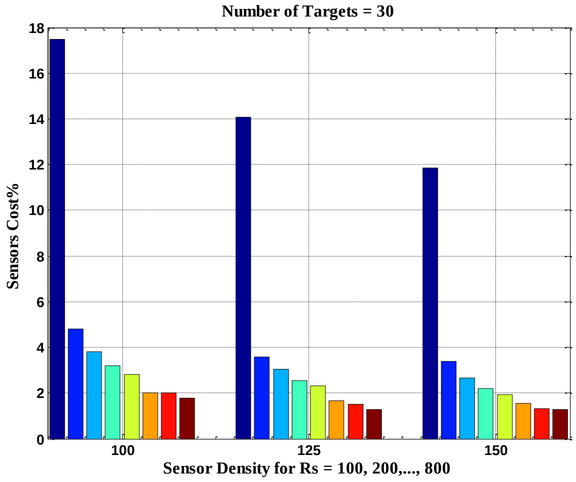

Figure 4- Percentage of sensors cost for 240 WSNs, where $number\ of\ sensors\ m = \{100,125,150\}$, $R_s = \{100,200,\ldots,800\}$ and $number\ of\ targets\ n = 30$.

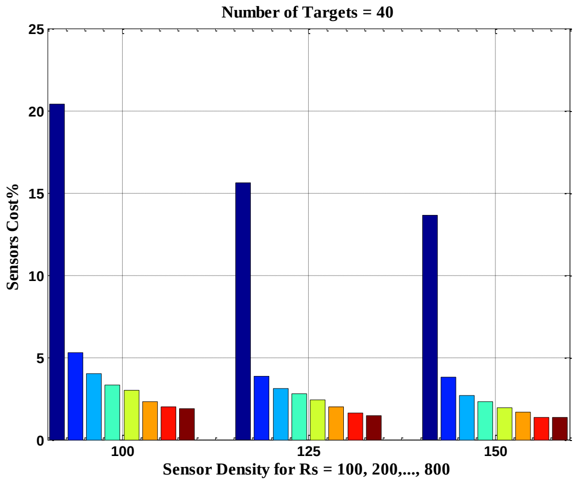

Figure 5-Percentage of sensors cost for 240 WSNs, where $number\ of\ sensors\ m = \{100,125,150\}$, $R_s = \{100,200,\ldots,800\}$ and $number\ of\ targets\ n = 40$.

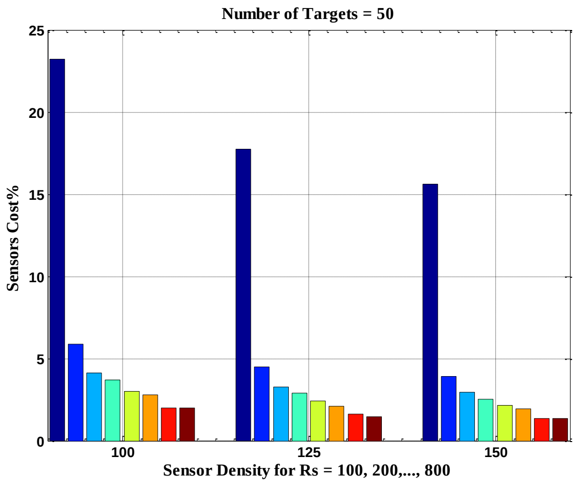

Figure 6- Percentage of sensors cost for 240 WSNs, where $number\ of\ sensors\ m = \{100,125,150\}$, $R_s = \{100,200,\ldots,800\}$ and $number\ of\ targets\ n = 50$.

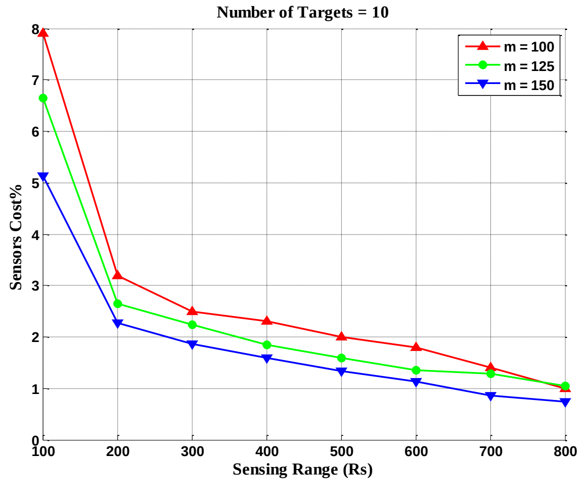

Figure 7- Percentage of sensors cost for 240 WSNs, where $number\ of\ sensors\ m = \{100,125,150\}$, $R_s = \{100,200,\dots,800\}$ and $number\ of\ targets\ n = 10$.

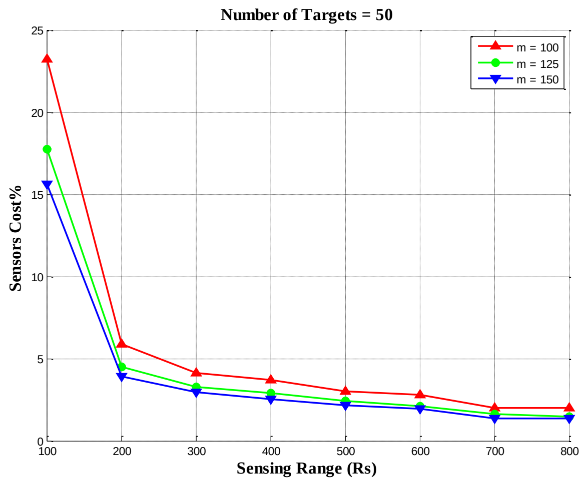

Figure 8- Percentage of sensors cost for 240 WSNs, where $number\ of\ sensors\ m = \{100,125,150\}$, $R_s = \{100,200,\dots,800\}$ and $number\ of\ targets\ n = 50$.

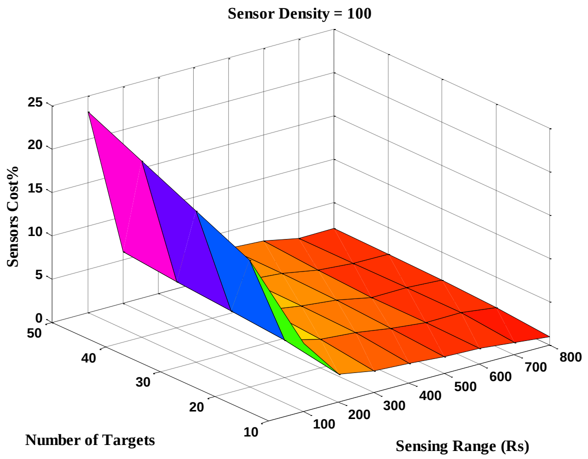

Figure 9- 3D projection of GA results. Coordinate (x: sensing range, y: number of targets, z: SC%). The simulation is experimented under 400 networks with $number\ of\ sensors = 100$.

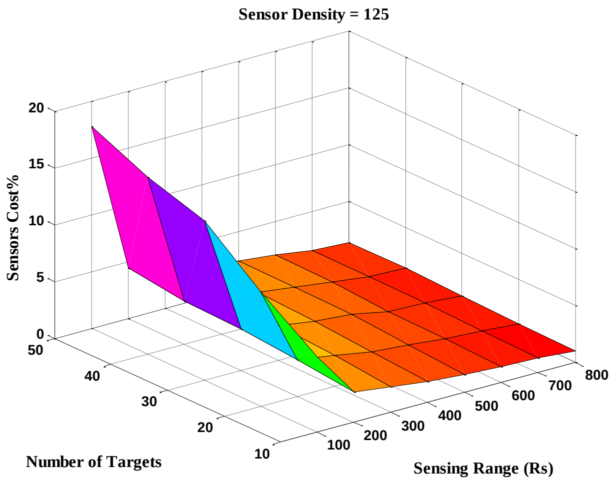

Figure 10- 3D projection of GA results. Coordinate (x: sensing range, y: number of targets, z: SC%). The simulation is experimented under 400 networks with $number\ of\ sensors$ = 125.

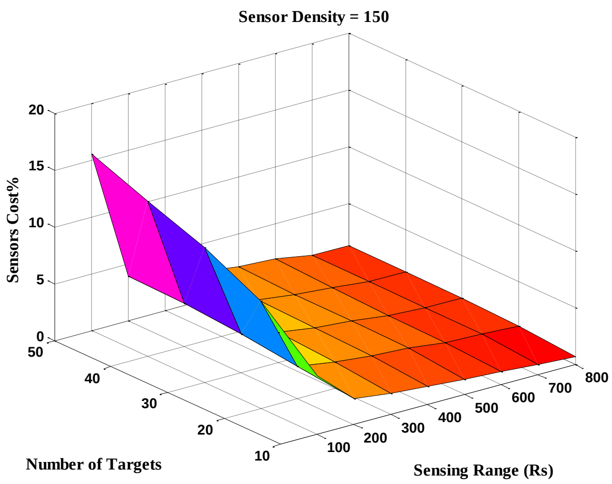

Figure 11- 3D projection of GA results. Coordinate (x: sensing range, y: number of targets, z: SC%). The simulation is experimented under 400 networks with $number\ of\ sensors$ = 150.

Results in the previous figures reveal that the proposed GA can effectively find a minimum number of active sensors from the whole sensors set, and thus configuring an energy-efficient WSN. The characteristic components of the proposed GA (being specified mainly by the chromosome binary representation and the proposed repair operator) is found to be very suitable for solving MRSCP. As expected from the qualitative results depicted in figures 7 - 10, increasing sensor density and/or sensing range provides algorithms with more alternatives for constructing complete and reliable set cover with minimum number of active sensors. On the other hand, increasing target size lets the algorithm to consume more sensors.

4.2 At Any-Time Performance of GA

Figures 12-14 depicts the any-time performance of the proposed GA while fixing target size and sensor density to their extremes, 50 and 150, respectively and varying sensing range R_s to three different values, $R_s = \{100,200,300\}$.

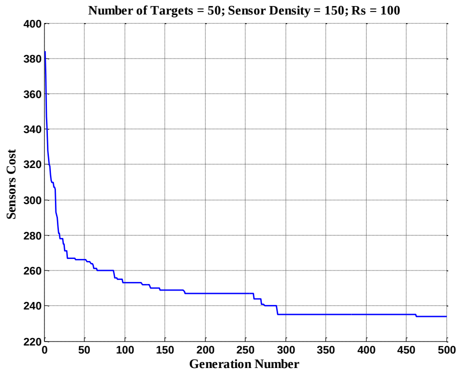

Figure 12 Any-time evolution of sensors cost for 10 WSNs, where $number\ of\ sensors\ m = 150$, $number\ of\ targets\ n = 50$ and $R_s = 100$.

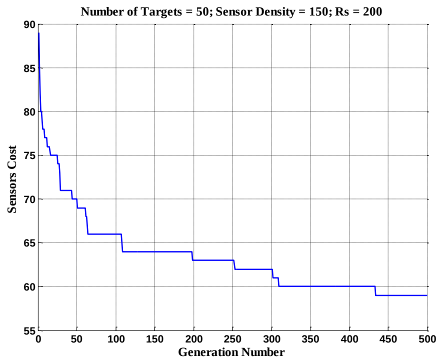

Figure 13 Any-time evolution of sensors cost for 10 WSNs, where $number\ of\ sensors\ m = 150$, $number\ of\ targets\ n = 50$ and $R_s = 200$.

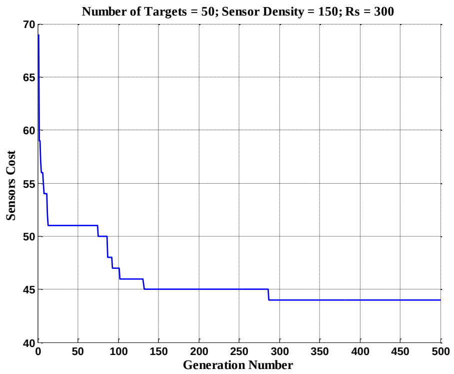

Figure 14- Any-time evolution of sensors cost for 10 WSNs, where $number\ of\ sensors\ m = 150$, $number\ of\ targets\ n = 50$ and $R_s = 300$.

The results in figures 11-13 show up that sensing radius increases, the convergence time towards the minimum number of sensors is also increased. When $R_s = 100$, we can see that the convergence occurs after 465 generations. Increasing R_s to 200, helps the algorithm to converge to the required solution after 440 generations. Moreover, letting $R_s = 300$, further increases the convergence time to less than 390 generations.

4.3 Impact of Post-heuristic Operator

The performance of the proposed GA is presented, here, *before* and *after* performing the post-heuristic operator. Figures 15 –22 depict coverage reliability and sensors cost percentage $SC\%$ being resulted from the proposed GA before and after applying post-heuristic. As we stated before that probabilistic coverage parameters are fixed throughout all simulations, we evaluate, as reference values, the average coverage reliability $\bar{r}$ (i.e., the average signal strength being detected from all the targets) of all network configurations in each test instance.

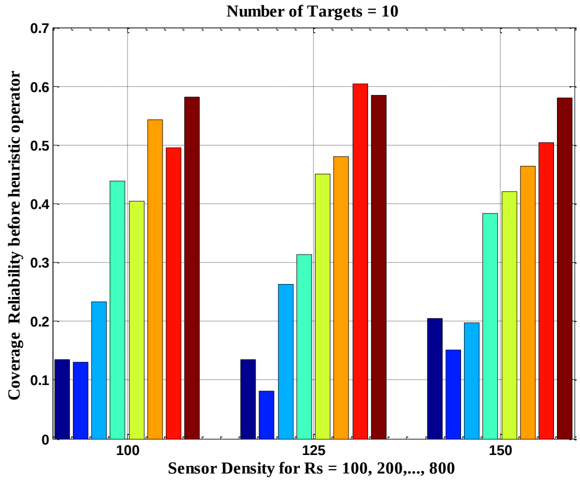

Figure 15- Coverage reliability before applying heuristic operator for 240 WSNs: $number\ of\ sensors\ m = \{100,125,150\}$, $number\ of\ targets\ n = 10$ and $R_s = \{100,200,\dots,800\}$.

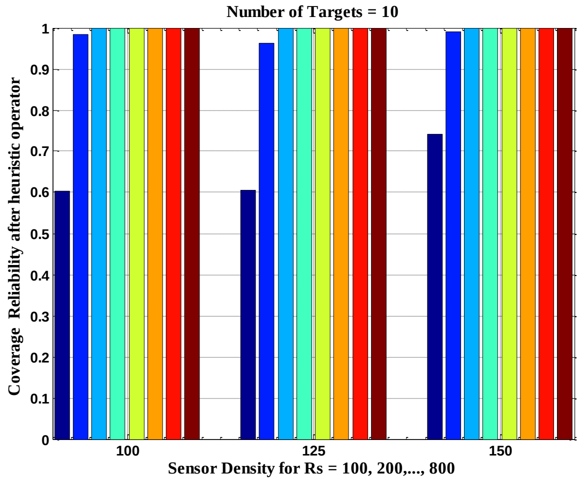

Figure 16- Coverage reliability after applying heuristic operator for 240 WSNs: $number\ of\ sensors\ m = \{100,125,150\}$, $number\ of\ targets\ n = 10$ and $R_s = \{100,200,\dots,800\}$.

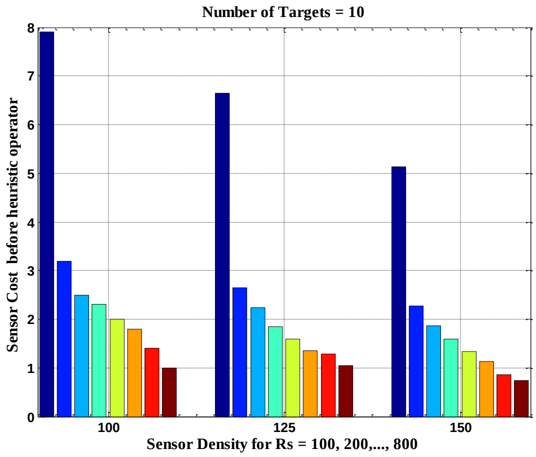

Figure 17- Sensor cost before applying heuristic operator for 240 WSNs: $number\ of\ sensors\ m = \{100,125,150\}$, $number\ of\ targets\ n = 10$ and $R_s = \{100,200,\dots,800\}$.

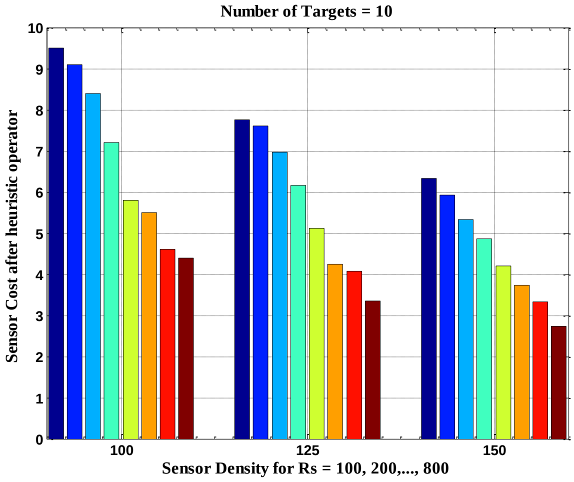

Figure 18- Sensor cost after applying heuristic operator for 240 WSNs: $number\ of\ sensors\ m = \{100,125,150\}$, $number\ of\ targets\ n = 10$ and $R_s = \{100,200, \ldots ,800\}$.

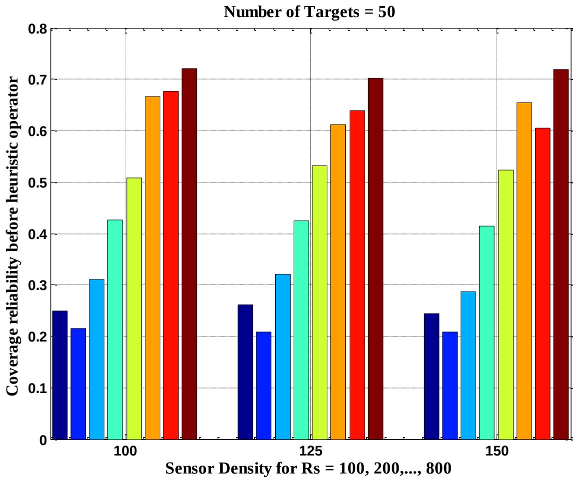

Figure 19- Coverage reliability before applying heuristic operator for 240 WSNs: $number\ of\ sensors\ m = \{100,125,150\}$, $number\ of\ targets\ n = 50$ and $R_s = \{100,200, \ldots ,800\}$.

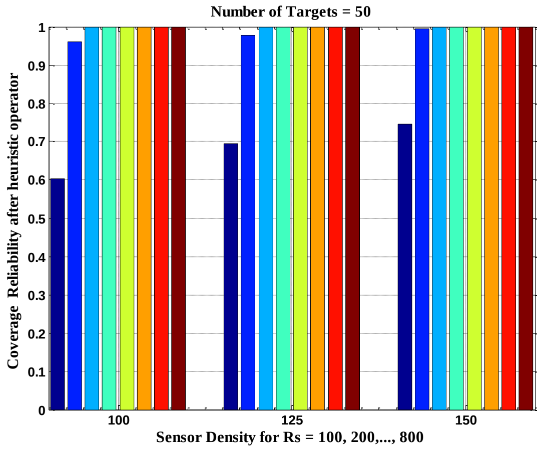

Figure 20- Coverage reliability after applying heuristic operator for 240 WSNs: $number\ of\ sensors\ m = \{100,125,150\}$, $number\ of\ targets\ n = 50$ and $R_s = \{100,200, \ldots ,800\}$.

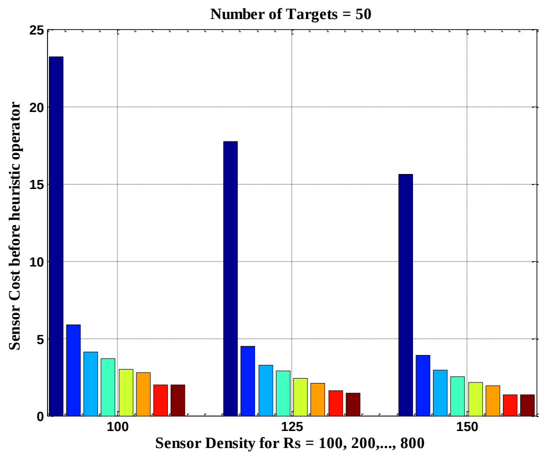

Figure 21- Sensor cost before applying heuristic operator for 240 WSNs: $number\ of\ sensors\ m = \{100,125,150\}$, $number\ of\ targets\ n = 50$ and $R_s = \{100,200, \ldots ,800\}$.

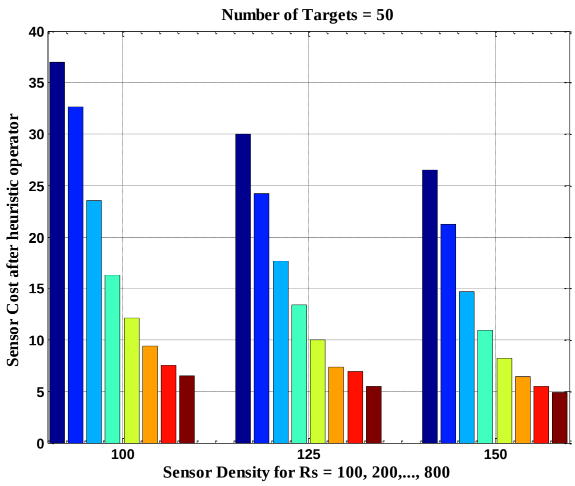

Figure 22- Sensor cost after applying heuristic operator for 240 WSNs: $number\ of\ sensors\ m = \{100,125,150\}$, $number\ of\ targets\ n = 50$ and $R_s = \{100,200, \ldots ,800\}$.

We can see that the proposed post-heuristic operator can exploit the existence of the redundant sensors to improve its covers reliability. In almost all of the test instances, the proposed post-heuristic operator attains reliable coverage of 100% while consuming very few extra sensors. For example, in figures 16 and 20, we see that the coverage reliability reaches its full certainty when $R_s \geq 300$. Also, for $R_s = 200$, the post-heuristic operator achieves high coverage reliability with more than 95%. Tables 1 and 2 quantitatively present the results of figures 16 and 20, respectively. The positive impact of the proposed post-heuristic operator can be returned back to its ability to select those sensors which lie near the targets such that their distances from the targets do not exceed $R_s - R_s * 0.5$. However, when lessening sensor density and/or sensing range to their low extremes (i.e., $m = 100$ and $R_s = 100$), the chance of finding sensors nearby targets or finding sensing areas that cover targets also decreases. The results also demonstrate that there should be a tradeoff between the two contradictory criteria of getting minimum number of active sensors and high covers reliability.

Table 1-Comparison of coverage reliability and sensors cost before and after applying post-heuristic operator. Results are presented for 24 test instances ($TI^i, i = 97, \dots, 120$) for 10 WSNs in each test instance with $m = \{100,125,150\}$, $R_s = \{400,500,\dots,800\}$ and $n = 10$.

TI	m	R_s	$\bar{r}$	Before post-heuristic operator		After post-heuristic operator	
				Coverage reliability	$SC\%$	Coverage reliability	$SC\%$
TI^1	100	100	0.1628	0.1350	7.9	0.6037	9.5
TI^2		200	0.1688	0.1297	3.2	0.9834	9.1
TI^3		300	0.2539	0.2333	2.5	1.0	8.4
TI^4		400	0.3317	0.4379	2.3	1.0	7.2
TI^5		500	0.4102	0.4042	2.0	1.0	5.8
TI^6		600	0.4794	0.5429	1.8	1.0	5.5
TI^7		700	0.5396	0.4957	1.4	1.0	4.6
TI^8		800	0.5917	0.5812	1.0	1.0	4.4
TI^9	125	100	0.1539	0.1355	6.64	0.6051	7.76
TI^{10}		200	0.1590	0.0807	2.64	0.9633	7.6
TI^{11}		300	0.2510	0.2635	2.24	1.0	6.96
TI^{12}		400	0.3374	0.3139	1.84	1.0	6.16
TI^{13}		500	0.4199	0.4509	1.60	1.0	5.12
TI^{14}		600	0.4816	0.4802	1.36	1.0	4.24
TI^{15}		700	0.5358	0.6045	1.28	1.0	4.08
TI^{16}		800	0.5863	0.5845	1.04	1.0	3.36
TI^{17}	150	100	0.1582	0.2046	5.1333	0.7415	6.3333
TI^{18}		200	0.1721	0.1513	2.2667	0.9908	5.9333
TI^{19}		300	0.2687	0.1974	1.8667	1.0	5.3333
TI^{20}		400	0.3532	0.3836	1.6000	1.0	4.8667
TI^{21}		500	0.4163	0.4211	1.3333	1.0	4.2
TI^{22}		600	0.4705	0.4636	1.1333	1.0	3.7333
TI^{23}		700	0.5244	0.5048	0.8667	1.0	3.3333
TI^{24}		800	0.5742	0.5800	0.7333	1.0	2.7333

Table 2- Comparison of coverage reliability and sensors cost before and after applying post-heuristic operator. Results are presented for 24 test instances ($TI^i, i = 97, \dots, 120$) for 10 WSNs in each test instance with $m = \{100,125,150\}$, $R_s = \{400,500,\dots,800\}$ and $n = 50$.

TI	m	R_s	$\bar{r}$	Before post-heuristic operator		After post-heuristic operator	
				Coverage reliability	$SC\%$	Coverage reliability	$SC\%$
TI^{97}	100	100	0.1741	0.2496	23.2	0.6024	37.0
TI^{98}		200	0.1682	0.2155	5.9	0.9612	32.6
TI^{99}		300	0.2577	0.3099	4.1	0.9981	23.5
TI^{100}		400	0.3447	0.4261	3.7	1.0	16.3
TI^{101}		500	0.4226	0.5081	3.0	1.0	12.1
TI^{102}		600	0.4869	0.6659	2.8	1.0	9.4
TI^{103}		700	0.5384	0.6769	2.0	1.0	7.5
TI^{104}		800	0.5832	0.7201	2.0	1.0	6.5
TI^{105}	125	100	0.1722	0.2613	17.76	0.6957	30.0
TI^{106}		200	0.1685	0.2084	4.48	0.9774	24.24
TI^{107}		300	0.2595	0.3213	3.28	1.0	17.68
TI^{108}		400	0.3472	0.4244	2.88	1.0	13.44
TI^{109}		500	0.4261	0.5317	2.4	1.0	10.0
TI^{110}		600	0.4887	0.6124	2.08	1.0	7.36
TI^{111}		700	0.5437	0.6383	1.6	1.0	6.96
TI^{112}		800	0.5862	0.7017	1.44	1.0	5.52

TI^{113}	150	100	0.1717	0.2442	15.6	0.7467	26.5333
TI^{114}		200	0.1765	0.2086	3.9333	0.9942	21.2
TI^{115}		300	0.2660	0.2869	2.9333	1.0	14.6667
TI^{116}		400	0.3466	0.4144	2.5333	1.0	10.9333
TI^{117}		500	0.4130	0.5240	2.1333	1.0	8.2
TI^{118}		600	0.4713	0.6550	1.9333	1.0	6.4
TI^{119}		700	0.5255	0.6047	1.3333	1.0	5.4667
TI^{120}		800	0.5743	0.7189	1.3333	1.0	4.9333

Also from the tables, one can see that the collaboration between the proposed GA and post-heuristic operator consumes very few sensors from the whole sensor set S (in the worst case no more than 37% in test instance TI^{97}) to achieve reliable coverage higher than the average reliability $\bar{r}$.

4.4 Worst Case Time Complexity

Here we will compute the worst-case computational complexity of the proposed GA for solving minimum reliable set cover problem (MRSCP). Without loss of generality, the computational complexity of any single-objective genetic algorithms is $O(max_i \times K)$ where K is the size of solutions to be evolved, via certain evolution operators, for max_i iterations.

Now, let us consider the computation time needed for the most critical parts of the proposed algorithm when applied for solving MRSCP. The most critical part which represents the computation time bottleneck in the proposed GA is the proposed repair operator (Algorithm 1) which costs the most computation time and in the worst-case time it linearly related to the maximum number of sensors m and targets n. Thus, the worst-case time complexity of the proposed GA can formally be described as:

$$O(GA) = max_i \times K \times |n| \times |m| \qquad (15)$$

5 Conclusions

In this paper, we have addressed designing energy-efficient WSNs with reliable coverage of the target area. The problem is modeled a minimum reliable set cover problem (MRSCP), which introduce the concept of probabilistic coverage as a more realistic coverage model for constructing the set cover. A genetic algorithm is proposed to construct a reliable set cover, where the total number of sensors used in the set is to be minimized. The result of the GA is then forwarded to a post-heuristic operator to improve the coverage reliability of the set cover. The performance of the proposed genetic algorithm is investigated in this paper under different simulation setting. The results of the simulations reveal that the number of active sensors used in the constructed set cover affected by the WSN design parameters including target size, sensor density, and sensor range. The results of this paper currently motivate us to investigate the possibility of applying multi-objective evolutionary algorithms (like MOEA/D, NSGA-II, and MOPSO [21]-[24]) to the combined optimization problem of both minimum set cover problem and cover reliability problem and handling both objective functions simultaneously instead of applying consecutive optimization mechanisms. Also, as a scope of further work, another quality of service (QoS) merit could be used to constraint the defined MRSCP. For example, different targets may need different priority of sensing quality and thus the optimization problem should reflect this diversified QoS coverage constraint in its formulation.

References

1. Garey M. R. and Johnson D. S. **1979**, *Computers and Intractability. A Guide to the Theory of NP-Completeness*. Freeman, New York.
2. Slijepcevic S. and Potkonjak M. **2001**, Power efficient organization of wireless sensor networks, in Proceeding. IEEE Int. Conference Commun, (2), pp: 472–476.
3. Cardei M., Thai M., Li Y., and Wu W. **2005**, Energy-Efficient Target Coverage in Wireless Sensor Networks, *IEEE INFOCOM*, pp: 1976-1984.
4. Tian D. and Georganas N. D. **2002**, A coverage-preserving node scheduling scheme for large wireless sensor networks, in Proceeding. 1st Assoc. Comput. Machinery Int. Workshop Wireless Sensor Netw. Appl, pp: 32–41.
5. Wang X., Xing G., Zhang Y., Lu C., Pless R., and Gill C. **2003**, Integrated coverage and connectivity configuration in wireless sensor networks, in Proceeding. 1st Int. Conference. Embedded Netw. Sensor Syst., Los Angeles, CA, pp: 28–39.

6. Liu Y. and Liang W. **2005**, Approximate coverage in wireless sensor networks," in Proceeding. IEEE Conference. Local Comput. Netw, pp: 68–75.

7. Gupta H., Zhou Z., Das S. R., and Gu Q. **2006**, Connected sensor cover: Self-organization of sensor networks for efficient query execution, *IEEE/Assoc. Comput. Machinery Trans. Netw*., 14(1), pp:55-67.

8. Ai J. and Abouzeid A. A. **2006**, Coverage by directional sensors in randomly deployed wireless sensor networks, *J. Comb. Optim*., 11(1), pp: 21–41.

9. Funke S., Kesselman A., Kuhn F., Lotker Z., and Segal M. **2007**., Improved approximation algorithms for connected sensor cover, Wireless Netw., 13(2), pp: 153–164.

10. Zhou Z., Das S. R., and Gupta H. **2009**, Variable radii connected sensor cover in sensor networks,*Assoc. Comput. Machinery Trans. Sensor Netw.*, 5(1), article 8, pp: 1–36.

11. Martins F. V. C., Carrano E. G., Wanner E. F., Takahashi R. H. C., and Mateus G. R. **2009**, A dynamic multiobjective hybrid approach for designing wireless sensor networks, *IEEE Congr. Evol. Comput*, pp: 1145–1152.

12. Cardei M. and Du D.-Z. **2005**, Improving wireless sensor network lifetime through power aware organization, *Wireless Netw*., 1193), pp: 333–340.

13. Cardei I. and Cardei M. **2008**, Energy-Efficient Connected-Coverage in Wireless Sensor Networks, *International Journal on Sensor Networks*, 3.

14. He J. , Xiong N. , Xiao Y. and Pan Y. **2010,** A Reliable Energy Efficient Algorithm for Target Coverage inWireless Sensor Networks, in:·Proceeding ICDCSW '10 Proceedings of the IEEE 30th International Conference on Distributed Computing Systems Workshops, pp: 180-188.

15. Lai C., Ting C., and Ko R. 2007, An Effective Genetic Algorithm to Improve Wireless Sensor Network Lifetime for Large-Scale Surveillance Applications, *IEEE Congr. Evol. Comput*, pp: 3531–3538.

16. Hu X. , Zhang J. , Yu Y. , Chung H. S. , Li Y. , Shi Y. , Luo X. 2010, Hybrid Genetic Algorithm Using a Forward Encoding Scheme for Lifetime Maximization of Wireless Sensor Networks, *IEEE Trans. Evol. Comp.*, 14(5), pp: 766-781.

17. Gil J.M., and Han Y.H. **2011**, A Target Coverage Scheduling Scheme Based on Genetic Algorithms in Directional Sensor Networks, *Sensors*, *11(2)*, 1888-1906;

18. Elfes A. 1987, Sonar-based real-world mapping and navigation, *IEEE Journal of Robotics and Automation*, 3(3), pp: 249–265.

19. Ghosh A. and Das S. K. **2006**, Coverage and Connectivity Issues in Wireless Sensor Networks, Chap. 6, *Mobile, Wireless, and Sensor Networks: Technology, Applications, and Future Directions*, R. Shorey, A. L. Ananda, M. C. Chan, and W. T., John Wiley & Sons, Inc.

20. Hossain A., Biswas P. K., Chakrabarti S. **2008**, Sensing Models and its Impact on Network Coverage in Wireless Sensor Network, In the Proceedings of *3rd International Conference on Industrial and Information Systems(ICIIS 2008),* IIT Kharagpur., India.

21. Huang C.F. and Tseng Y.C. **2005**, The coverage problem in a wireless sensor network, *Mobile Netw. Appl.*, 10(4), pp: 519–528.

22. Deb K. **2002**, *Multi-Objective Optimization using Evolutionary Algorithms*. Wiley, Chichester, 2001.

23. C.A. Coello Coello, D.A. Van Veldhuizen, and G.B. Lamont, *Evolutionary Algorithms for Solving Multi-Objective Problems*. Kluwer, New York.

24. Zhang Q. and Li H. **2007**, MOEA/D: A multi-objective evolutionary algorithm based on decomposition, *IEEE Transactions on Evolutionary Computation*, 11(6), pp: 712–731,.

25. Coello C. A., Toscano-Pulido G., Salazar-Lechunga M. **2004**., Handling multiple objectives with particle swarm optimization, *IEEE Trans on Evol Comput*, 8(3), pp:256–279,.