Iraqi Journal of Science
ISSN: 0067-2904
GIF: 0.851

Multi-Objective Set Cover Problem for Reliable and Efficient Wireless Sensor Networks

Bara'a Ali Attea*

Computer Science Dept. College of Science , University of Baghdad, Baghdad,Iraq .

Abstract

Achieving energy-efficient Wireless Sensor Network (WSN) that monitors all targets at all times is an essential challenge facing many large-scale surveillance applications.Single-objective set cover problem (SCP) is a well-known NP-hard optimization problem used to set a minimum set of active sensors that efficiently cover all the targeted area. Realizing that designing energy-efficient WSN and providing reliable coverage are in conflict with each other, a multi-objective optimization tool is a strong choice for providing a set of approximate Pareto optimal solutions (i.e., Pareto Front) that come up with tradeoff between these two objectives. Thus, in the context of WSNs design problem, our main contribution is to turn the definition of single-objective (SCP) into a multi-objective problem by adopting an additional conflicting objective to be optimized. To the best of our knowledge, improving coverage reliability of WSNs has not been explored while simultaneously solving SCP problem. This paper addresses the problem of improving coverage reliability of WSNsusing a realistic sensing model to handle coverage uncertainty. To this end, this paper formulates the so-called multi-objective SCP with the goal of selecting the minimum number of sensors so that the selected set reliably covers all the targets.To cope with two optimization objectives rather than one objective, this paperinvestigates the use of a multi-objective evolutionary algorithm, the so-called non-dominated sorting genetic algorithm for tackling the formulated problem. Moreover, it adopts a heuristic crossover operator designed specifically to improve the performance of the algorithm.The effectiveness of the algorithm is verified in terms of sensors cost and coverage reliability under extensive simulations.

Keywords:non-dominated solution; NSGA-II;Pareto Front; probabilistic coverage; reliability; SCP; WSNs.

مشكلة تغطية المجموعة متعددة الأهداف لموثوقية وكفاءة شبكات الأستشعار اللاسلكية

براء علي عطية*

قسم الحاسبات,كلية العلوم ,جامعة بغداد ، بغداد،العراق.

*Email:-baraali@scibaghdad.edu.com

المستخلص

تحقيق شبكة الاستشعار اللاسلكية (WSN) الموفرة للطاقة التي تراقب جميع الأهداف في جميع الأوقات هو التحدي الأساسي الذي يواجه العديد من تطبيقات المراقبة وعلى نطاق واسع. تعتبر مشكلة تغطية المجموعة (SCP) وحيدة الهدف مشكلة أمثلية NP-hard وتستخدم لتحديد مجموعة من الحد الأدنى من أجهزة الاستشعار النشطة لتغطية جميع المنطقة المستهدفة وبكفاءة. إذا أدركنا أن تصميم WSN كفوء في استخدام الطاقة وبنفس الوقت توفير تغطية موثوقة هي في المنطق متناقضة مع بعضها البعض، فأن الأمثلية متعددة الأهداف تعتبر خيار قوي لتوفير مجموعة من الحلول المثلى والتي تسمى باريتو التقريبية (أي جبهة باريتو) والتي تأتي بالمفاضلة بين هذين الهدفين. في سياق مشكلة تصميم WSNs، ذا البحث يهدف (وعلى حد علمنا لأول مرة) التحويل لتعريف مشكلة تغطية الجموعة من مشكلة وحيدة الهدف (SCP) المشكلة متعددة الأهداف من خلال الاعتماد طبيعة الأهداف المتضاربة.

ويتناول هذا البحث مشكلة تحسين موثوقية تغطية الشبكة باستخدام نموذج أستشعار واقعي التعامل مع حالة عدم اليقين في التغطية. ولهذه الغاية، تم صياغة المشكلة بنموذج يسمى متعدد الأهداف وذلك بهدف اختيار أقل عدد من أجهزة الاستشعار وبشكل موثوق تقوم بتغطية جميع الأهداف. في هذا البحث تم استخدام خوارزمية تطورية متعددة الأهداف، والمعروفة بأسم الخوارزمية الجينية ذات الترتيب غير المهيمن لحل المشكلة. وعلاوة على ذلك، تم أقتراح مشغل خلط توجيهي مصمم خصيصا لتحسين أداء الخوارزمية. يتم التحقق من فعالية الخوارزمية من حيث التكلفة والموثوقية في التغطية تحت محاكاة واسعة النطاق.

1 Introduction

Wireless sensor network (WSN) is an emerging and fast growing technological platform in myriad working environments. To name just a few, WSNs can be used in defensive and agriculture applications, environment monitoring, emergency search-and-rescue operations. For efficient and successful operation of such networks, an enough and minimum number of sensors (called *sensor cover* or *set cover*) should be activated to guarantee the required coverage while keeping the network to operate for a long period of time. It has been shown in the literature that this problem can be formulated as the minimum set cover problem (MSCP), or shortly set cover problem (SCP)[1] and proved its NP-completeness in [2, 3]. Like many other sensor network design problems, maintaining reliable coverage while activating low number of sensors 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) host several interesting characteristics for tackling MOPs, the literature recently provides many such approaches for solving different sensor design problems [4-15]. Defining a multi-objective optimization formula for the set cover problem in WSNs, however, has not been explored yet.

In this paper, we pretend to formulate a multi-objective set cover problem (MO-SCP) in WSNs and exploit a multi-objective evolutionary algorithm (MOEA) for optimizingtwo contradictory WSN's design objectives: high coverage and low sensors cost. The evolutionary mechanism is based on the well-known non-dominated sorting genetic algorithm (NSGA-II)and a heuristic crossover operator is proposed to provide the decision maker with a set of non-dominated solutions that effectively render the tradeoff between the two conflicting objectives. The main contributions of this paper areas follows:

1. The paper turns the *de-facto* definition of single-objective SCP in WSN design problems to a multi-objective SCP. The goal of which becomes to cope with contradictory objectives. The formulated optimization problem will be directed to effectively maintaining reliable coverage with a minimum number of sensors as active cover set in WSNs.
2. The paper presents NSGA-II with a proposed heuristic crossover 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 thateffectively render the tradeoff between the two conflicting objectives (i.e., sensors cost and coverage reliability).

The rest of this paper is outlined as follows. Section 2 describes the traditional formulation of SCP in WSNs as well as the new formulation of multi-objective SCP by adding probabilistic coverage means to the original problem. Section 3 provides the preliminaries to the concept of multi-objective optimization and NSGA-II, and introduces the formulation of the proposed crossover operator. The results of multi-objective optimizations are then presented in Section 4. Finally, Section 5 concludes the current work and hints some further ramifications.

2 Set Cover Optimization Problem (SCP): Definition and Formulation

1.1 Single-Objective SCP

Single-objective set cover problem (SCP) is NP-hard and can be described as follows: given several sets that share some common elements, the goal is to select the minimum number of these sets so that the selected sets contain all the elements that are contained in any of the input sets. More formally, Given a universe $\mathcal{U}$of$n$elements and a family $\mathcal{S}$of m subsets of $\mathcal{U}$, a *cover* is a subfamily$\mathcal{C} \subseteq \mathcal{U}$ whose union is equal to $\mathcal{U}$. The problem is to find a cover of $\mathcal{U}$that uses the *fewest* sets [16].

2.2 System Model and Assumptions

In order to model the system, let $\mathcal{A}$ be a square monitoring field with knowndimensions, $\mathcal{S}_{\Theta_S} = \{s_1, \dots, s_m\}$ be a set of $m$sensor nodes with set $\Theta_S = \{(x_{s_1}, y_{s_1}, r_{s_1}), \dots, (x_{s_m}, y_{s_m}, r_{s_m})\}$ of parameters concerning the spatiallocations and coverage radii. Also, let$\mathcal{T}$ be a set of n targets with known locations, i.e., $\mathcal{T} == \{(x_{t1}, y_{t1}), (x_{t2}, y_{t2}), \dots, (x_{tn}, y_{tn})\}$.All the sensors are dropped randomly in $\mathcal{A}(1 \leq \forall i \leq m \mid (x_{si}, y_{si}) = ([0, X_{max}], [0, Y_{max}]))$. 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 [17], [18] to define the notion of the probabilistic coverage of a target $\mathcal{T}_j = (x_{tj}, y_{tj})$ by a sensor s_i.

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

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 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.
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 $s_i \in S$ 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 $s_i \in S$ 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. Formally speaking:

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

2.3Multi-objective SCP(MO-SCP) Formulation

In this section, we formulate the problem of getting high coverage percentage with SCP as a *multi-objectiveSCP*:
Given:

$\mathcal{A}$: 2-D plane areawith 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 β: uncertainty-parameters adjusted according to the physical properties of the sensor.

Decision (Design) variables of a solution:
The selection of active sensors in $\mathcal{S}$, i.e.,$\{\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_m\} | \mathcal{S}_i \in \{0,1\}$.

Objective:

Two objectives are considered as optimization functions. The first objective is to minimize the number of active sensorsselected from the sensors set$\mathcal{S}$, such that it can form a complete reliable sensor cover. 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:

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

Detection of coverage holes is formulated as:

$$Cover(\mathcal{S}_i, \mathcal{T}) = \begin{cases} 1 \; if \; \forall t \in \mathcal{T} \rightarrow \exists s \in \mathcal{S}_i | Coverage(s,t) \geq c_{th} \\ 0 \; otherwise \end{cases} \quad (4)$$

The second objective is to maximize the reliability of the sensor cover. Formally speaking:

$$max \quad Coverage(\mathcal{S}_i, \mathcal{T}) = \frac{\sum_{t \in \mathcal{T}} \max_{s \in \mathcal{S}_i} Coverage(s,t)}{n} \quad (5)$$

3. Multi-objective Evolutionary Algorithm for MO-SCP

In many real world applications there may be several, normally conflicting, objectives to be systematically and simultaneously optimized to solve a given problem. In contrast to single-objective optimization, the (MOO) problem has a rather different perspective. While in the single-objective optimization there is only single global solution, but in multi-objective optimization there is a set of points, called the Pareto-optimal (PO) set, that all fit a predetermined definition for an optimum. Several MOEAs have been suggested in the literature [19] [20]. The main reason for the popularity of evolutionary algorithms (EAs) for solving multi-objective optimization is their population-based nature and ability to find multiple optima simultaneously.

3.1 Preliminaries

A general MOO problem can be formally stated as: finding the vector $\mathbb{x}^* = [x_1^*, x_2^*, \dots, x_m^*]^T$ which optimizes the objective function vector$F(\mathbb{x}) = (f_1(\mathbb{x}), \dots, f_n(\mathbb{x}))^T$, where $\mathbb{x} \in \Omega$ is the decision variable vector, $F: \Omega \longrightarrow \mathbb{R}^n$ consists of n real-valued objective functions, the objective space is $\mathbb{R}^n$ and Ω is the search space. In general, $f_1, \dots, f_n$ are in conflict with each other, thus,finding the optimal solution can be interpreted as finding a good trade-off between all $f_1, \dots, f_n$ of F. In general, the multi-objective approach to solving MOO problems generates "partial orders" of solutions leading to possible multitudes of trade-off solutions in objective space. A decision vector $\mathbb{x}^*$ is called Pareto-optimal if and only if there is no$\mathbb{x}$ that dominates $\mathbb{x}^*$, i.e., there is no $\mathbb{x}$ such that (considering minimization) $f_i(\mathbb{x}) \leq f_i(\mathbb{x}^*) \forall i \in \{1,2,\dots,n\}$ and $\exists j \in \{1,2,\dots,n\}$: $f_j(\mathbb{x}) < f_j(\mathbb{x}^*)$. However, two decision vectors $\mathbb{x}_1^*$ and $\mathbb{x}_2^*$ are mutually non-dominating if $\mathbb{x}_1^* \nsucc \mathbb{x}_2^*$ and $\mathbb{x}_2^* \nsucc \mathbb{x}_1^*$ hold. Given the space Ω of candidate solution vectors, $\mathbb{x}$, the non-dominated set $N(\Omega) = \{\mathbb{x}_1^*, \dots, \mathbb{x}_n^*\} \subset \Omega : \nexists \mathbb{x}_i \in \{\Omega - N(\Omega)\}$ and $\mathbb{x}_i \succ \mathbb{x}_j^*$.

Without loss of generality, we assume throughout this paper that the optimization functions are expressed as minimization problems. One candidate solution,$\mathbb{x}_1$, may be better than another solution, $\mathbb{x}_2$, with respect to f_i(i.e. $f_i(\mathbb{x}_1) < f_i(\mathbb{x}_2)$, but, worse with respect to f_j (i.e.$f_j(\mathbb{x}_1) > f_j(\mathbb{x}_2)$). Thus, in MOO, the candidate solution vectors can be compared following the concept of dominance [20]: a solution vector $\mathbb{x}_1$ dominates $\mathbb{x}_2$ (or similarly $\mathbb{x}_2$ is dominated by $\mathbb{x}_1$), denoted by $\mathbb{x}_1 \succ \mathbb{x}_2$, if and only if the following two conditions hold:

- $\forall i \in \{1,2,\dots,m\} | f_i(\mathbb{x}_1) \leq f_i(\mathbb{x}_2)$; and

- $\exists j \in \{1,2,\dots,m\}$: $f_j(\mathbb{x}_1) < f_j(\mathbb{x}_2)$.

However, $\mathbb{x}_1$ and $\mathbb{x}_2$ are said to be mutually non-dominating pair if both $\mathbb{x}_1 \nsucc \mathbb{x}_2$ and $\mathbb{x}_2 \nsucc \mathbb{x}_1$ hold. Then, given the whole space Ω of candidate solution vectors, $\mathbb{x}$, the non-dominated solution set $N(\Omega) = \{\mathbb{x}_1^*, \dots, \mathbb{x}_n^*\} \subset \Omega$: $\nexists \mathbb{x}_i \in \{\Omega - N(\Omega)\}$ and $\mathbb{x}_i \succ \mathbb{x}_j^*$.

3.2 Non-dominated Sorting Genetic Algorithm (NSGA-II)

There are several variants of MOEAs but with the common aim of how to retain the non-dominated solutions generated during the search. Deb et al. proposed NSGA-II to use GA with the traditional crossover and mutation operators. The basic idea of NSGA-II is to divide the created population into several fronts corresponding to different ranks. The ranks are assigned by means of non-domination [21], [22]. A crowding distance (based on either fitness values or parameters values) is added to each individual and the population is classified into fronts starting from the first front of solutions having a maximum dummy fitness. Then, the classified group of solutions is ignored and another front of non-dominated individuals is considered. The process continues until last n^{th} front containing inferior region of non-dominated solutions. During selection, the individuals in the first front will always get more copies than the rest of the population. By the mechanism of NSGA-II, no explicit archive is used to store the non-dominated solutions set$N(\Omega)$.The general framework of NSGA-II is shown in Algorithm 1**.**

Algorithm 1. Framework of NSGA-II

Input:
- Multi-objective problem $f(\mathbb{x}) = (f_1(\mathbb{x}), f_2(\mathbb{x}), \dots, f_m(\mathbb{x}))$
- Population size, N
- Maximum number of generations, gen_{max}

Output:
- Non-domination fronts

Step 1 – Setup
- $gen = 0$.
- Initialize random population of solutions, $\mathbb{P} = \{\mathbb{P}_1, \dots, \mathbb{P}_N\}$
- Evaluate objectives vector $f(\mathbb{P}_i), \forall i \in \{1, \dots, N\}$.
- Assign rank $r(\mathbb{P}_i), \forall i \in \{1, \dots, N\}$ based on Pareto dominance.

Step 2 – Update:
- **For**$i = 1, \dots, N$: Generate Child $\mathbb{P}'_i$ using binary tournament selection, recombination, and mutation.
- **For** each Parent in $\mathbb{P}$ and Child in $\mathbb{P}'$
  i) Assign rank based on Pareto.
  ii) Generate sets of non-dominated vectors.
  iii) Add solutions to the next generation starting from the 1^{st} front until N individuals.
  iv) Determine crowding distance between points on each front.
- Elitist selection on the lower front and are outside a crowding distance.

Step 3 – Stopping criteria
- If $gen = gen_{max}$, then stop and output non-dominated fronts,
  otherwise $gen = gen + 1$, go to Step 2.

3.3 NSGA-II for MO-SCP

As NSGA-II is population-based optimization algorithms, let us consider a population ρ 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 $\mathbb{P}_{1\le i\le K} \in \rho$is represented as a fixed-length vector of size$m$, where each element (i.e., gene) value controls the *active/sleep setting* of the corresponding j^{th} sensor. Thus, $\forall k \in \{1,\dots,K\}$ $and\forall j \in \{1,\dots,m\}$: $\mathbb{P}_k = \left(\mathbb{P}_{k,1}, \mathbb{P}_{k,2}, \dots, \mathbb{P}_{k,m}\right)$ $s.t.$:

$$\mathbb{P}_{k,j} = \begin{Bmatrix} 1, & if\ s_j\ is\ active \\ 0, & if\ s_j\ is\ sleep \end{Bmatrix} \tag{6}$$

where0 means inactive (i.e., unassigned) sensor, while 1 means active (i.e., assigned) sensor. Then, the whole configuration space δ for the NSGA-II can be created by the Cartesian product of activation/inactivation of all m sensors:

$$\delta = \prod_{i=1}^{m}(\{0,1\}) = 2^m \tag{7}$$

Let us to consider that NSGA-II handles only $K \ll \delta$ different individual solutions at a time. It starts with an initial random population $\rho_0 \subset \delta$, $|\rho_0| = K$ and continues until a maximum number of generations max_{gen}has been reached.NSGA-II can be described as a process formulated in an iterative evolution function $\Psi: \{\rho, EP\} \to \{\rho', EP'\}$with $\Psi(\rho_i) = \rho_{i+1}$, where i is the iteration index and ρ_i is the population in iteration i. EPisthe first front of non-dominated solutions. The population starts with an initial random population ρ_0 and continues until a maximum number of iterations max_ihas been reached. Each iteration i consists of a five main operations: individual repair, fitness vector evaluation, parents selection, traditional/heuristic perturbation, and EP update to be discussed next.

Based on the chromosome representation, we adopted an intuitive and fast random initialization mechanism, where each gene can take a random $0/1$value. Formally speaking:

$$(\forall i \in \{1,\dots,N\}, \forall j \in [1,m]): \mathbb{P}_{ij} \sim \{0,1\} \tag{8}$$

where$\sim$ means a uniform bias.

However, 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. Formally speaking:

$$\Psi_{rep}: \mathbb{P}_k \to \mathbb{P}_k{}' \tag{9}$$

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 thresholdc_{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.

Then, the fitness vector $\overrightarrow{\Phi}$of each individual solution $\mathbb{P}_k$ is evaluated using Eqn. 3 and Eqn. 5, i.e.,:

$\vec{\Phi}(\mathbb{P}_k) = \{|\mathcal{S}_k|, Coverage(\mathcal{S}_k, \mathcal{T})\}$(10)

For selection operator, we imitate similar functions to that found in the implementation of the traditional NSGA-II. For the heuristic perturbation function, we proposed one evolution operator Ψ that provides heuristic for improving the solution coverage's reliability or sensors cost. The process of the proposed heuristic crossover operator is presented next (see Algorithm 2). It takes as input two parent individuals $\mathbb{P}_1$, $\mathbb{P}_2$being selected from ρ by the selection operator and perturbs them with probability $p_{per} = 1.0$ to return one child $\mathbb{C}$. For each child, the proposed heuristic will be directed (with equal probability) either towards maximizing the reliability of the generated set covers or towards minimizing the number of active sensors. Thus, for each child $\mathbb{C}_i, 1 \le i \le K$, the heuristic procedure also takes as input a flag F indicating whether to prefer maximizing reliability or minimizing sensors covers.

Algorithm 2: Heuristic-Crossover ($\mathbb{P}_1$,$\mathbb{P}_2$, $\mathbb{C}$, F)	
1:	set $\mathbb{C} \leftarrow \emptyset$
2:	set S to the sensors from both $\mathbb{P}_1$ and $\mathbb{P}_2$
3:	set $\mathcal{Target} \leftarrow \mathcal{T}$
4:	**while** $\mathcal{Target} \neq \emptyset$
5:	**if**F == reliability-heuristic **then**
6:	select a sensor $s \in S$ that contributes to the most *reliable coverage* to a target $t \in \mathcal{T}$
7:	**else** /* heuristic towards minimizing sensors cost */
8:	select a sensor $s \in S$ that contributes to *cover more* number of targets $t \in \mathcal{T}$.
9:	**end if**
10:	set the index of s in $\mathbb{C}$ to 1
11:	set $\mathcal{Target} \leftarrow \mathcal{Target} - t$;
12:	**end while**

The proposed heuristic recursively selects sensors in the child. The set S (in line 2) collects sensors from both parents that have an index value equal to 1. The sensors in S can then participate in creating the sensor cover in the child. According to the flag F, the procedure will select from S those sensors having the highest contribution to the heuristic (line 5 to 9). 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

The evaluation of NSGA-II performance for solving MO-SCP is presented in this section. The results are obtained after setting WSNs and algorithm parameters into the following.

1. The simulation area is square-shaped with side length $X_{max} = 1000m$.
2. Five different settings for the number of targets $n = \{10,20,30,40,50\}$.
3. Four different settings of sensor density: $m = \{100,125,150,175\}$.
4. For each test instance group composed from 2 and 3, the sensing range of the sensor nodes$R_s$is varied to eight different values$\{100,200,\ldots,800\}$.

Thus, we have a total of 160 different test instances (composed from 2, 3 and 4). Each test instance $TI^i, i = 1, \ldots, 160$ includes 10 random WSNs with different configurations, and the results are averaged over each 10 simulations. Thus the overall simulation examines a total of 1600 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.The probability for heuristic crossover operator is 1.0.

Figures-1 and Figures-2 (a and b) depict the 3D projection of NSGA-II results on the two extremes of m parameter (i.e., number of sensors). Part a of the figures presents the performance of NSGA-II on the first objective function, i.e., sensors cost. On the other hand, part b presents the performance of NSGA-II on the second objective function, i.e., coverage reliability. The two parts of the figures reflect the contradictory nature of the two objectives, i.e., sensors cost and coverage reliability. For example, while increasing sensors cost to 20 or more, the coverage reliability of the generated set cover improved to reach more than 90%.

Comparing the projection of the results on the 3D figures of part a, one can see that increasing number of sensors positively effects towards minimizing number of active sensors selected in the set cover. Moreover, increasing sensing radius and/or decreasing numberof targets can result in activating less active sensors in the final set cover. Also, for part b of the figures, we can realize that the total set cover reliability can be improved but at the expense of increasing sensing radius, and/or increasing number of sensors, and/or decreasing number of targets.

Figures-(3 – 6) give more details for the NSGA-II performance on sensors cost of the generated set cover. Form the whole near Pareto-Optimal solutions set, we select the two extreme solutions that correspond to the best and worst solutions in terms of sensors cost.

5 Conclusions

Minimum set cover problem is a well-known NP-hard problem used in WSNs design problem, seeking to minimize number of active sensor nodes in the generated set cover. Unlike all of the existing work that rely on the traditional single-objective definition, this paper introduces a *multi-objectiveSet Cover Problem* (MO-SCP), adding an additional but contradictory objective function (i.e., coverage reliability). The formulated problem is then tackled by the well-known multi-objective evolutionary algorithms, NSGA-II. Instead of the traditional use of two perturbation operators (i.e., crossover and mutation), a heuristic crossover operator is proposed for MO-SCP. The results over 160 WSNs reveal that NSGA-II can provide a set of near Pareto-Optimal solutions that trade-off the contradictory nature of the two optimized objectives. An extension to this work can be recommended to explore the ability of NSGA-II for the generalized version of minimum set cover problem.

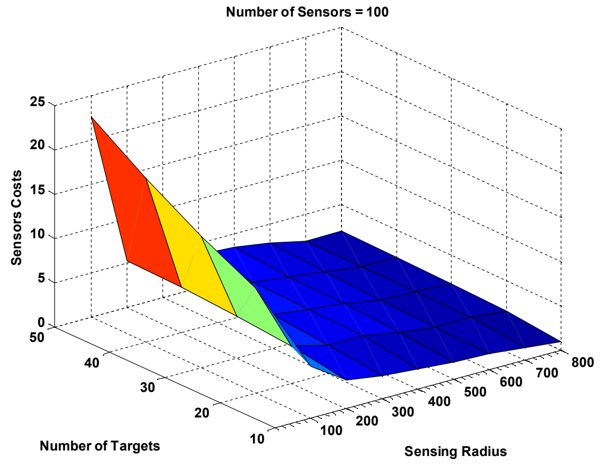

Figure 1-(a) Sensors cost results of NSGA-II as 3D space. (x: sensing radius with incremental step of 100, y: number of targets with incremental step of 10, z: active sensors cost out of 100 sensors). The simulation is experimented under 400WSNs.

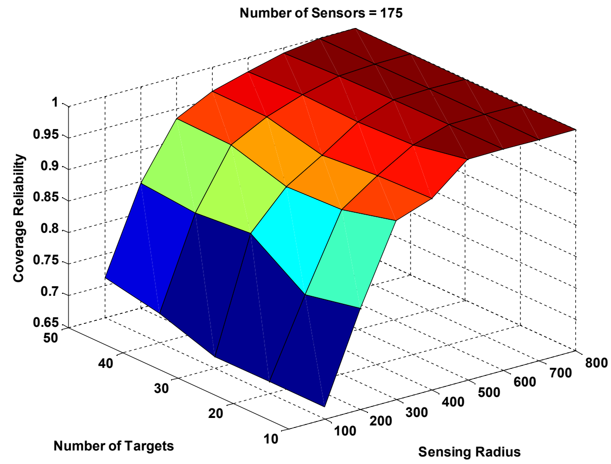

Figure 1-(b) Coverage reliability results of NSGA-II as 3D space. (x: sensing radius with incremental step of 100, y: number of targets with incremental step of 10, z: coverage reliability). The simulation is experimented under 400 WSNs.

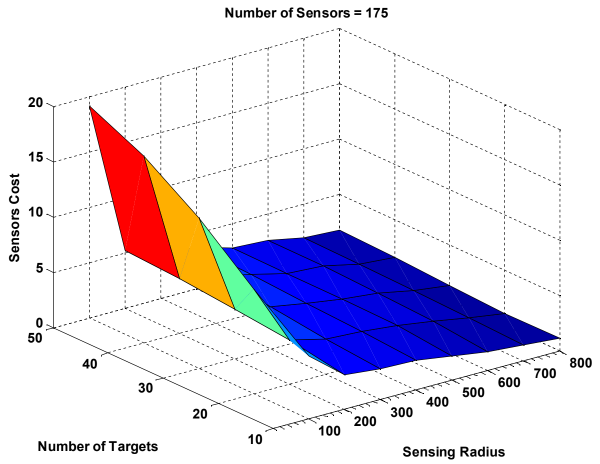

Figure 2-(a) Sensors cost results of NSGA-II as 3D space. (x: sensing radius with incremental step of 100, y: number of targets with incremental step of 10, z: active sensors cost out of 175 sensors). The simulation is experimented under 400 WSNs.

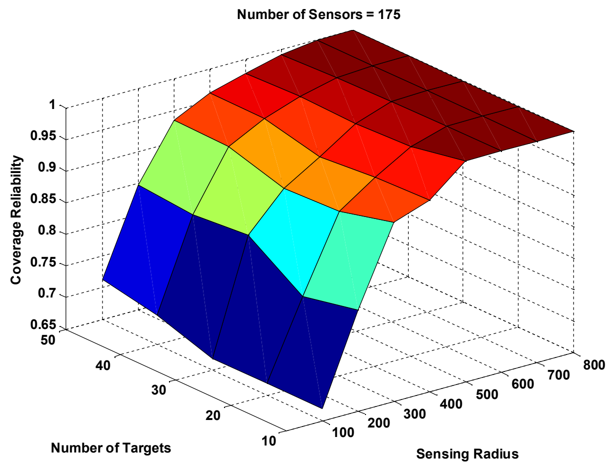

Figure 2-(a) Coverage reliability results of NSGA-II as 3D space. (x: sensing radius with incremental step of 100, y: number of targets with incremental step of 10, z: coverage reliability). The simulation is experimented under 400 WSNs.

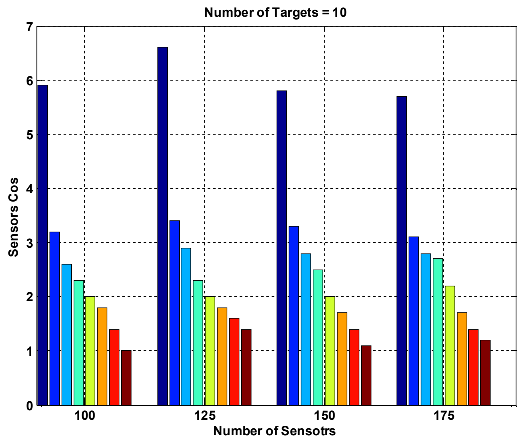

Figure 3-(a) Minimum sensors cost found in the generated near Pareto-Optimal solutions set. Results are experimented on 320 WSNs, where $number\ of\ sensors\ m = \{100,125,150,175\}$, $R_s = \{100,200,\ldots,800\}$ and $number\ of\ targets\ n = 10$.

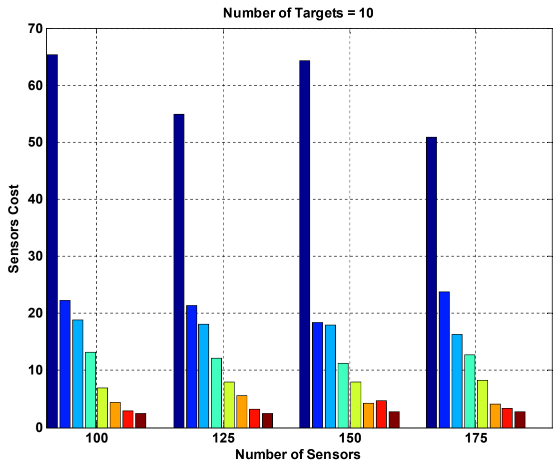

Figure 3-(b) Maximum sensors cost found in the generated near Pareto-Optimal solutions set. Results are experimented on 320 WSNs, where $number\ of\ sensors\ m = \{100,125,150,175\}$, $R_s = \{100,200,\ldots,800\}$ and $number\ of\ targets\ n = 10$.

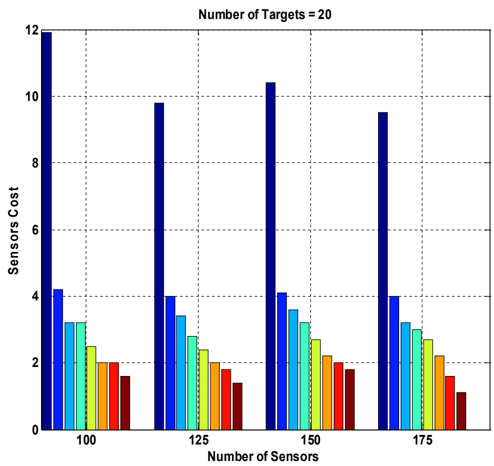

Figure 4-(a) Minimum sensors cost found in the generated near Pareto-Optimal solutions set. Results are experimented on 320 WSNs, where $number\ of\ sensors\ m = \{100,125,150,175\}$, $R_s = \{100,200,\ldots,800\}$ and $number\ of\ targets\ n = 20$.

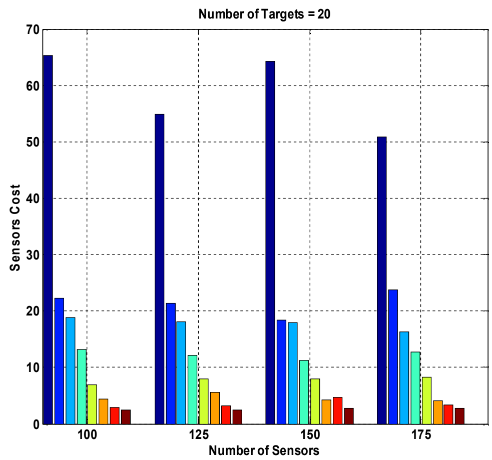

Figure 4-(b) Maximum sensors cost found in the generated near Pareto-Optimal solutions set. Results are experimented on 320 WSNs, where $number\ of\ sensors\ m = \{100,125,150,175\}$, $R_s = \{100,200,\ldots,800\}$ and $number\ of\ targets\ n = 20$.

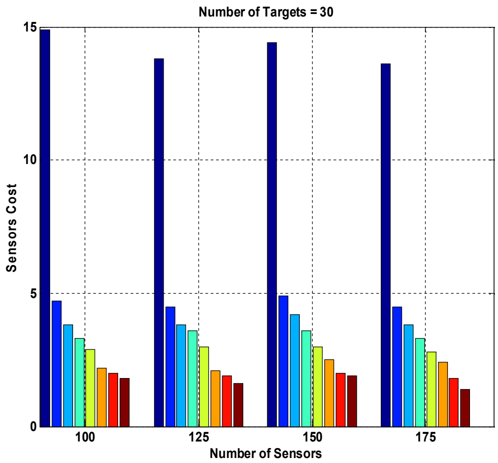

Figure 5-(a) Minimum sensors cost found in the generated near Pareto-Optimal solutions set. Results are experimented on 320 WSNs, where $number\ of\ sensors\ m = \{100,125,150,175\}$, $R_s = \{100,200,\ldots,800\}$ and $number\ of\ targets\ n = 30$.

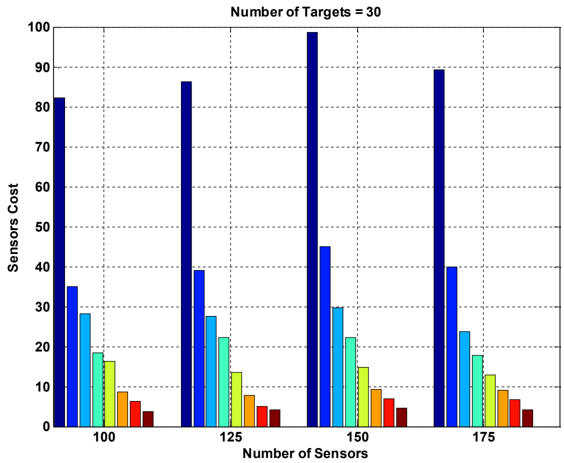

Figure 5-(b) Maximum sensors cost found in the generated near Pareto-Optimal solutions set. Results are experimented on 320 WSNs, where $number\ of\ sensors\ m = \{100,125,150,175\}$, $R_s = \{100,200,\ldots,800\}$ and $number\ of\ targets\ n = 30$.

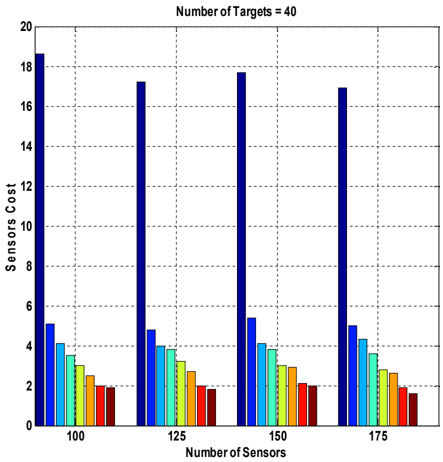

Figure 6-(a) Minimum sensors cost found in the generated near Pareto-Optimal solutions set. Results are experimented on 320 WSNs, where $number\ of\ sensors\ m = \{100,125,150,175\}$, $R_s = \{100,200,\ldots,800\}$ and $number\ of\ targets\ n = 40$.

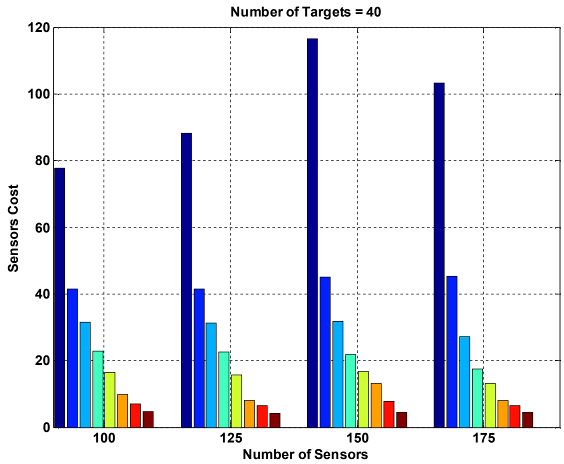

Figure 6-(b) Maximum sensors cost found in the generated near Pareto-Optimal solutions set. Results are experimented on 320 WSNs, where $number\ of\ sensors\ m = \{100,125,150,175\}$, $R_s = \{100,200,\ldots,800\}$ and $number\ of\ targets\ n = 40$.

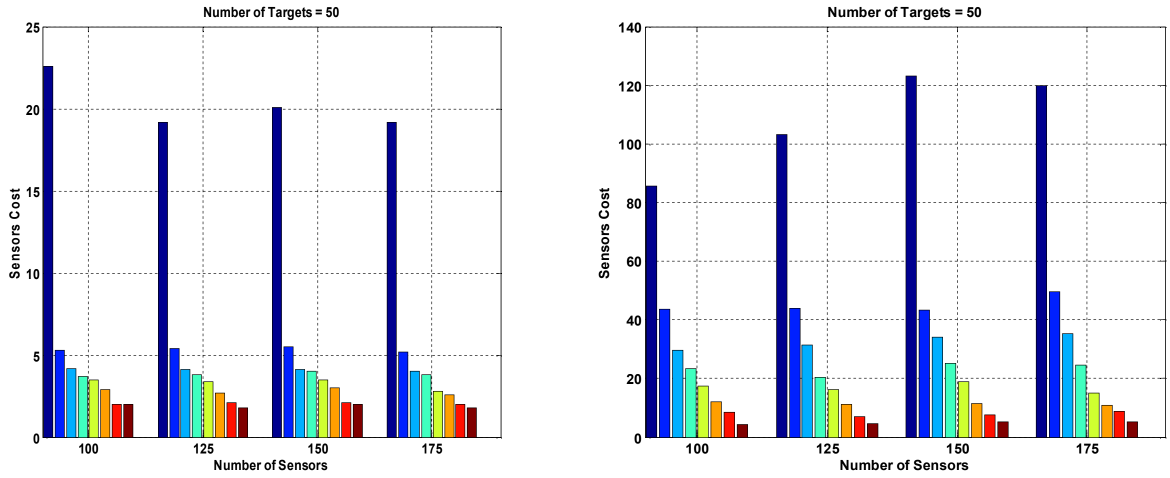

Figure 7- (left) Minimum sensors cost found in the generated near Pareto-Optimal solutions set. Results are experimented on 320 WSNs, where $number\ of\ sensors\ m = \{100,125,150,175\}$, $R_s = \{100,200,\dots,800\}$ and $number\ of\ targets\ n = 50$.**(right)**Maximum sensors cost found in the generated near Pareto-Optimal solutions set. Results are experimented on 320 WSNs, where $number\ of\ sensors\ m = \{100,125,150,175\}$, $R_s = \{100,200,\dots,800\}$ and $number\ of\ targets\ n = 50$.

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 Proc. IEEE Int. Conf. Commun., vol. 2. Finland*, pp. 472–476.

3.Cardei,M., Thai,M., Li Y., and Wu W.,**2005**, “Energy-Efficient Target Coverage in Wireless Sensor Networks*”, IEEE INFOCOM 2005, Mar*. pp. 1976-1984.

4.Jourdan,D. B. andde Weck, O. L.,**2004**, “Multi-objective genetic algorithm for the automated planning of a wireless sensor network to monitor a critical facility”, in *SPIE Proc*., vol. 5403, pp. 565–575.

5.Jourdan,D. B. and deWeckO. L.,**2004**, “Layout optimization for a wireless sensor network using a multi-objective genetic algorithm”, in *IEEEVeh. Technol. Conf. P.*, vol. 5, pp. 2466–2470.

6.Rajagopalan,R., **2010**, "A Multi-objective Optimization Approach for Data Fusion in Mobile Agent Based Distributed Sensor Networks", *IEEE Instrumentation and Measurement Technology Conference*, pp. 208-212.

7.Molina,G., Alba E., and TalbiE.-G.,**2008**, “Optimal sensor network layout using multi-objective metaheuristics”, *J. Univer.Comput.Sci.*, vol. 14, no. 15, pp. 2549–2565, 2008.

8.Rajagopalan,R., Mohan, C. K., Varshney,P., MehrotraK.,**2005**, "Multi-objective Mobile Agent Routing in Wireless Sensor Networks", *The 2005 IEEE Congress on Evolutionary Computation*, Vol. 2, pp. 1730 - 1737

9.Woehrle,M., Brockhoff,D., HohmT., and BleulerS.,**2008**, “Investigating coverage and connectivity tradeoffs in wireless sensor networks: The benefits of moeas,” in *19th Int. Conference on Multiple CriteriaDecision Making*, pp. 7–12.

10.Konstantinidis, A.,**2009**,"Multiobjective Deployment and Power Assignment in Wireless Sensor Networks using Metaheuristics".

11.Kang,C. and Chen,J.,**2009**, “An evolutionary approach for multi-objective 3d differentiated sensor network deployment”, in *Proc. IEEE Int.Conf. CSE’09*, Vancouver, Canada, vol. 1, pp. 187–193.

12.Jia,J., Chen J., Chang,G., Wen,Y., and Song J.,**2009**, “Multi-objective optimization for coverage control in wireless sensor network with adjustable sensing radius,” *Computers and Mathematics with Applications*, vol. 57, no. 11–12, pp. 1767–1775.

13.Jia,J., Chen J., Chang,G., and Tan,Z., **2009**, “Energy efficient coverage control in wireless sensor networks based on multi-objective genetic algorithm”, *Comput. Math. Appl.*, vol. 57, no. 11–12, pp. 1756–1766.

14.Konstantinidis,A., Yang,K., Zhang,Q., and Zeinalipour-YaztiD.,**2009**, “A multi-objective evolutionary algorithm for the deployment and power assignment problem in wireless sensor networks”, *Comput.Netw*.

15.Özdemir,S., Attea B. A., and Khalil,Ö. A.,**2012**,Multi-objective Evolutionary Algorithm Based on Decomposition for Energy Efficient Coverage in Wireless Sensor Networks, *Wireless Personal Communications*, Springer, 2012, DOI 10.1007/s11277-012-0811-3.

16.Yoo,S.,**2010**, A Novel mask-coding representation for set cover problems with applications in test suite minimization, 2nd International Symposium on Search Based Software Engineering, pp.19-28.

17.Ghosh,A. and Das,S. K.,**2006**, " Coverage and Connectivity Issues in Wireless Sensor Networks", *Mobile, Wireless, and Sensor Networks: Technology, Applications, and Future Directions*, pp. 221-256.

18.Hossain,A., Biswas,P. K., andChakrabarti, S. **2008**, "Sensing Models and its Impact on Network Coverage in Wireless Sensor Network", *In Industrial and Information Systems, 2008. ICIIS 2008.IEEE Region 10 and the Third international Conferenceon* (pp. 1-5).

19. Deb,K., **2001**, *Multiobjective Optimization Using Evolutionary Algorithms*. New York: Wiley.

20.Coello,C.,Veldhuizen, D. V., and Lamont G., **2002**, *Evolutionary Algorithms for Solving Multi-Objective Problems*. Norwell, MA: Kluwer.

21.Srinivas, N. and Deb, K.,**1994**, “Multi-objective function optimization using non-dominated sorting genetic algorithms,” *Evolutionary Computation*, vol. 2, no. 3, pp. 221-248.

22.Deb,K., A., Pratap,S. Agarwal, and Meyarivan,T.,**2000**, “A fast and elitist multi-objective genetic algorithm: NSGA-II,” In *Proceedings Parallel Problem Solving from NatureVI*, pp. 849-858.