Multi-objective Optimization of Barrier Coverage with Wireless Sensors

Xiao Zhang(✉), Yu Zhou, Qingfu Zhang, Victor C.S. Lee, and Minming Li

Department of Computer Science,City University of Hong Kong,
83 Tat Chee Avenue, Kowloon, Hong Kong
{xiao.zhang,yzhou57-c}@my.cityu.edu.hk
{qingfu.zhang,csvlee,minming.li}@cityu.edu.hk

Abstract. Barrier coverage focuses on detecting intruders in an attempt to cross a specific region, in which limited-power sensors in these scenarios are supposed to be distributed remotely in an indeterminate way. In this paper, we consider a scenario where sensors with adjustable ranges and a few sink nodes are deployed to form a virtual sensor barrier for monitoring a belt-shaped region and gathering incidents data. The problem takes into account three relevant objectives: minimizing power consumption while meeting the barrier coverage requirement, minimizing the number of active sensors (reliability) and minimizing the transmission distances between active sensors and the nearest sink node (efficiency of data gathering). It is shown that these three objectives are conflicting in some degree. A Problem Specific MOEA/D with local search methods is proposed for finding optimal tradeoff solutions and compared with a classical algorithm. Experimental results indicate that knee regions exist, and these knee regions may provide the best possible tradeoff for decision makers.

1 Introduction

In recent years, there has been increasing development in the field of wireless sensor networks (WSNs). One of the most important applications in WSNs is border surveillance and intrusion detection, such as detecting intruders crossing country borders or boundaries of battlefields. Many recent works have addressed such surveillance applications by using WSNs to organize the network nodes as a barrier [1]. For deterministic deployment of sensors, the high performance can be achieved sufficiently by analysis. However, surveillance tasks may involve hard-to-reach areas, in which case unmanned mission way is more desirable. Specifically, limited-power sensors and several sink nodes in these scenarios are supposed to be distributed remotely, for example, dropped from aircraft; they wake up, organize themselves as a network, and start sensing the area for intrusion. When a sensor detects an intrusion, the event is reported to the sink node so that an appropriate decision is made.

Power efficient is always a critical issue in wireless barrier coverage. The single objective optimization problem, minimizing the total power consumption while the barrier is full covered, is referred to as *General Min-Cost Linear Coverage*

A. Gaspar-Cunha et al. (Eds.): EMO 2015, Part II, LNCS 9019, pp. 557–572, 2015.
DOI: 10.1007/978-3-319-15892-1_38

problem (GMCLC) [2]. However, there is no efficient way to get exact optimal solutions since it is proved to be NP-hard [3]. In addition, since sensors are vulnerable to failure, it is important to minimize the number of active sensors to improve the reliability while meeting the coverage requirement. Besides, in general wireless sensor networks, a large number of sensor nodes, which are generally compact and inexpensive, are distributed in an observation area while sink nodes with comparatively sufficient power are defined as the data gathering center. Long transmission distances between sensor nodes and sink nodes cause low efficiency of data gathering and high energy consumption.

In this paper, we take the considerations above into account simultaneously and propose an algorithm to achieve the following objectives:

- *Objective 1*: Minimizing the total power consumption via activating a subset of the sensor nodes and adjusting their sensing ranges.
- *Objective 2*: Minimizing the number of active sensors to improve the reliability of coverage.
- *Objective 3*: Minimizing the active sensors' average distance from the nearest sink node to improve the efficiency of data gathering.

However, these three objectives are conflicting in nature. The sensors are failure-prone: each sensor fails independently with a certain probability. Under the condition of fully coverage, the fewer sensors activated, the higher reliability achieved. Meanwhile, the power consumption is proportional to the radii of active sensors. Next, we take a simple instance to illustrate the conflict among objectives. Fig. 1 shows two feasible solutions for the coverage problem. In the first solution (the left one), the power consumption is $Cost_1 = r_1^{\kappa}$ and the number of active sensors $|\mathbf{S_1}^*|$ is *one*. In the second solution (the right one), the power consumption is calculated by $Cost_2 = r_2^{\kappa} + r_3^{\kappa}$, and number of active sensors $|\mathbf{S_2}^*|$ is *two*. Since $r_1 = r_2 + r_3 = \frac{m}{2}$, we have $Cost_1 > Cost_2$. That is to say, minimizing the total power consumption may increase the number of active sensors, and require the active sensors to be distributed evenly along the barrier. In addition, minimizing the active sensors average distance from the nearest sink node may result in more active sensors close to the sink node. However, to meet the coverage requirement, either more sensors, if available, are activated to cover the region far away from the sink node or a larger sensing range is assigned to the farthest sensors, leading to higher power consumption. Thus, finding the tradeoff among them is worth exploring.

This problem can be formulated as a *Multi-objective Optimization Problem (MOP)*. Classical algorithms may not be applicable and few approaches tackle these objectives simultaneously. It is reasonable to use *Multi-Objective Evolutionary Algorithms (MOEAs)*, which have been proven efficient and effective in dealing with MOPs in wireless sensor networks [4] [5].

In this paper, we refine the barrier coverage problem to an *MOP* with three objectives, which is referred to as *Tradeoff on Barrier Coverage with Adjustable Sensing Radius Problem (TBCAP)*. Solutions are obtained through a problem specific MOEA, which adopts the framework of decomposition-based multiobjective

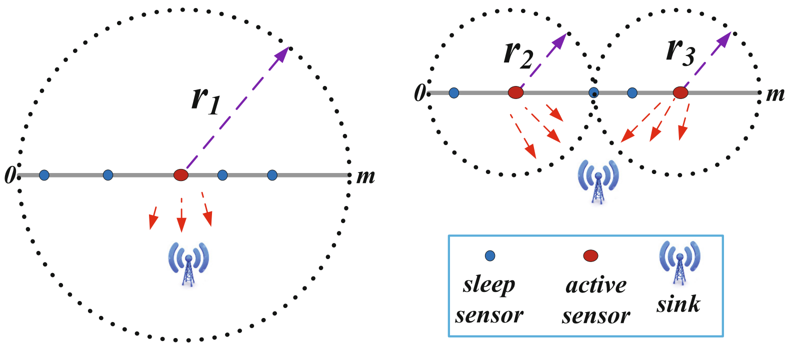


Fig. 1. Illustration of conflict among the objectives

evolutionary algorithm (MOEA/D) [6] as the baseline algorithm. We call it PS-MOEA/D. The PS-MOEA/D employs the problem-specific operators and local search methods. Besides, in order to improve the search, we incorporate a dynamic strategy of computational resource assignment. Moreover, a perturbation is involved to search for the global optimal solution.

The remainder of this paper is organized as follows. The related works is presented in Section 2. In Section 3, we define and formulate the problem, and give a naive algorithm for finding the tradeoff. Section 4 presents the details of the problem specific MOEA/D. Section 5 shows the experimental results and analysis. Finally, Section 6 outlines the conclusions and future directions.

2 Related Works

A heterogeneous WSN consists of several types of nodes with different capability, in which a large number of sensor nodes with the capabilities of sensing data, while fewer sink nodes may have larger battery and more powerful processing resource [7]. They are widely used in surveillance [8] [9]. Among them, barrier coverage problem deals with how to deploy sensor nodes to form barrier coverage for detecting intruders crossing a belt-shaped area of interest [10] [11] [12].

Optimizing the efficiency of data gathering and transmission quality between sensors and sinks have been widely studied [7] [13]. Mhatre et al. [7] studies a heterogeneous sensor network in which nodes are to be deployed over a unit area for the purpose of surveillance. They determined the optimum sensor nodes and sink nodes intensities (λ_0, λ_1).

Power consumption is always a critical issue in wireless barrier coverage. It helps to prolong the network lifetime by turning off some sensors while meeting given coverage requirements. Since it is proved to be NP-hard [3], several approximation algorithms have been proposed in recent years [2] [3] [14] .

Moreover, network failure, partial or whole, may not only be due to power exhaustion of the sensor nodes. Some sensors may stop functioning due to mechanical problems when they are working. This may result in unexpected consequences. Very few researchers focus on the reliability of the sensor networks for coverage.

To improve the reliability of coverage requirement, Sanjay et al. [15] consider an unreliable wireless sensor grid network for coverage with sensors placed in a square of unit area. In this model, all sensors are failure-prone, i.e., each node fails independently with a certain probability.

Proposing a scheme for wireless coverage considering so many aspects together is a challenging problem. To this end, MOEAs may provide a desirable model for solving such sensor network design problems. While both coverage and power consumption have been extensively studied in the past [16], few attempts however, have been made on tackling the coverage, power consumption, reliability and efficiency of data gathering simultaneously or explicitly. Martins et al. [17] presented multiobjective hybrid optimization algorithms for minimizing the power consumption and maximizing the coverage in flat WSNs subject to node failures. In [16], the problem objectives are stated as maximizing the coverage and minimizing energy consumption for maximizing the network lifetime. A sleep scheduling method is incorporated into a multiobjective optimization framework. Recently, Lanza-Gutierrez et al. [18] use MOEAs to optimize a WSN composed of a set of sensors, a sink node and relay nodes, analyzing the performance of algorithms by objectives of the average energy consumption suffered by the sensors and the average coverage provided by the network.

3 Preliminaries

3.1 Multiobjective Problem and MOEA/D

An MOP is generally formulated as follows.

$$\begin{aligned} &minimize\ F(x) = (f_1(x), \ldots, f_m(x)) \\ &subject\ to\ x \in \Omega \end{aligned} \tag{1}$$

where Ω is the decision space and $x \in \Omega$ is a decision variable. R^m consists of m objective functions $f_1, \ldots, f_m$: R^m is the objective space. The objectives in problem (1) often conflict with each other and an improvement on one objective may lead to the deterioration of another. A Pareto optimal solution is an optimal tradeoff candidates among all objectives. The Pareto optimum terminology is described in [19], in which *Pareto dominance*, *Pareto optimal*, *Pareto Set* (PS) and *Pareto Front* (PF) are defined formally. The decision makers require an approximation to the PF for a good insight to the problem and make the decision.

Tchebycheff approach [20] is employed to decompose the MOP into a number of sub-problems. Let $\lambda^1, \lambda^2, \ldots, \lambda^n$, be a set of uniformly spread weighted vectors and z^* be an ideal point. The problem can be decomposed into scalar optimization sub-problems as follows.

$$minimize\ g^{te}(x|\lambda^j, z^*) = max_{1 \le i \le m}\{{\lambda_i}^j|f_i(x) - z^*|\} \tag{2}$$

Therefore, one is able to obtain different Pareto optimal solutions by solving a set of single objective optimization problems defined by the Tchebycheff approach with different weight vectors. MOEA/D minimizes all these m objective functions simultaneously in a single run. Neighborhood relations among these single objective sub-problems are defined based on the distances among their weight vectors. Each sub-problem is optimized by using information mainly from its neighboring sub-problems. The details of MOEA/D can be found in [6].

3.2 Problem Formulation

Barrier Model. Consider a WSN consisting of a set of sensor nodes and several sink nodes, in which sensor nodes form a virtual sensor barrier for monitoring a belt-shaped region to detect and send intruding events to one of the sink nodes. Fig. 2 shows an illustration of the barrier model. Intrusion is assumed to occur from top to bottom. The assumptions are as follows.

- The sensor nodes and sink nodes are assumed to be randomly deployed and static once deployed with known positions.
- Assume that each sensor has an adjustable disk sensing range r and is equipped with limited power.
- The sink nodes with sufficient energy (comparing to sensor nodes) are not failure-prone.

Mathematical Model. We define the following notations formally, which are used in the analysis in the mathematical model:

- $\mathbf{S}$: a set $\mathbf{S}$ of N sensors $\{\mu_1, \mu_2, ..., \mu_N\}$ are randomly distributed on a belt region which needs to be monitored.
- r_i: each sensor μ_i has an adjustable sensing range r_i. The power consumption of each active sensor is proportional to r_i^κ for some positive constant $\kappa \geq 2$.
- Π: a set Π of π sink nodes $\{s_1, s_2, ..., s_\pi\}$ are distributed on a belt region, in which $\pi << N$.
- (x_i, y_i): each sensor μ_i has a coordinate to denote the location.
- (x_j^s, y_j^s): each sink node s_j has a coordinate to denote the location.
- ${d^j}_i$: the distance between each sensor $\mu_i \in \mathbf{S}^*$ to its closest sink node $s_j \in \Pi$. The distance from the sensor μ_i to the sink node s_j is $d_i^j = \sqrt{(x_j^s - x_i)^2 + (y_j^s - y_i)^2}$.

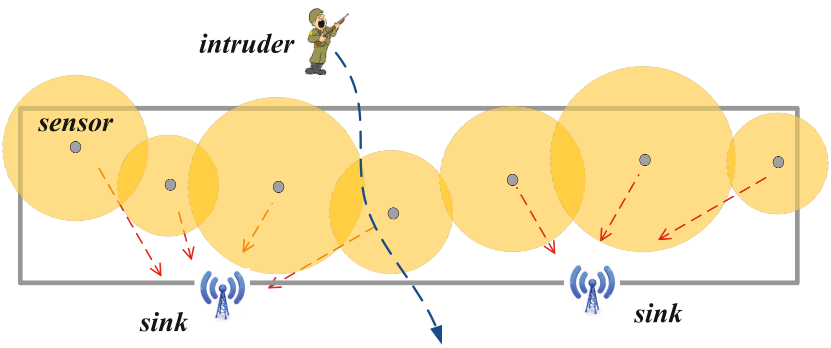


Fig. 2. Wireless barrier coverage model

- $\mathbf{S}^*$: a subset $\mathbf{S}^* \subseteq \mathbf{S}$ of sensors are activated and assigned ranges to formulate a two-layer decision variable $\Omega = \{(u_1, r_1), (u_2, r_2), ..., (u_i, r_i), ..., (u_N, r_N)\}$, $u_i \in \{0, 1\}$.

In this way, *TBCAP* can be stated as an MOP, where we minimize the power consumption (f_1), the number of active sensors (f_2), and the active sensors' average distance from the closest sink node (f_3).

$$\begin{aligned} f_1 &= \sum_{\mu_i \in \mathbf{S}^*} {r_i}^{\kappa} \\ f_2 &= |\mathbf{S}^*| \\ f_3 &= \frac{\sum_{\mu_i \in \mathbf{S}^*} d_i^j}{|\mathbf{S}^*|} \end{aligned} \tag{3}$$

3.3 Weighted-Sum Algorithm

Weighted-Sum Algorithm (WSA) as the most widely used classical method for MOP is used for comparing performance with our proposed PS-MOEA/D. It is the simplest yet efficient approach to find solutions on the entire Pareto-optimal set. The WSA in this paper is based on a genetic algorithm, which has the following procedures.

Solution Encoding. The solution is represented by a two-layer coding structure C = $\{(u_1,r_1),(u_2,r_2),...,(u_i,r_i),...,(u_N,r_N)\}$. The boolean u_i describes the working status of the sensor node and r_i indicates the value of its radii.

Repairing Initial Solutions. An approximation algorithm is necessary to guarantee the barrier coverage requirement, which can be found in [3].

Steady State Evolution. The Steady State Genetic Algorithm(SSGA) [21] based operators are adopted in WSA, which benefit from selecting two individuals and combining them to obtain two offsprings by crossover and mutation operators. Then, if these two new individuals are more adapted than the worst two individuals of the population, the former are included in the population by replacing the latter.

Shrink Process. After performing initialization, there could be several overlaps between sensors. If so, the radii of those sensors can be shrunk and repaired immediately after operations.

Evaluation. In each generation, the fitness is calculated by weighted-sum method after normalization. Specifically, $\mathit{fitness} = \omega_1 \times \sum_{\mu_i \in \mathbf{S}^*} {r_i}^{\kappa} + \omega_2 \times |\mathbf{S}^*| + \omega_3 \times \frac{\sum_{\mu_i \in \mathbf{S}^*} d_i^j}{|\mathbf{S}^*|}$, where ω_1, ω_2 and ω_3 are weights varying between *zero* and *one* and $\omega_1 + \omega_2 + \omega_3 = 1$.

4 Problem-Specific MOEA/D

In general, encoding representation, repair process and shrink process are identical to the WSA in Section 3.3. In order to improve the search ability of MOEA/D for *TBCAP*, some modification and improvement have been introduced. In the following part, we explain the procedure of Algorithm 1 in detail. The following sections are related to the main steps of the PS-MOEA/D.

Algorithm 1. PS-MOEA/D Framework for *TBCAP*

Input:
N_P: Population size and number of sub-problems
N_N: Size of neighborhood
M_E: Maximum number of evaluations
Output:
P: Final solutions
Step 1-Initialization: randomly generate an initial population and set parameters

Step 2-Repairing: repair the solutions to meet problem requirement
Step 3-Decomposition: decompose the *TBCAP* to N_P sub-problems
Step 4-Evaluation
While $e < M_E$
  Step 4.1-Selection: selection of sub-problems by using tournament selection based on μ_i as Sel_P or Per_P
  For $i = 1 : |Sel_P|$
    Step 4.2-Mutation: generate a new solution by mutation operator
    Step 4.3-Local Search: use of forward-LS and backward-LS
    Step 4.4-Update: update of current and neighboring solutions
  End-for
  Step 4.5-Perturbation: perturbation operator on Per_P
  $e \leftarrow$ e + 1
  Step 4.6-Update Utility: calculate and update the utility
End-while
return P
End

4.1 Problem Decomposition

let $\Lambda = \lambda^1, \lambda^2, \ldots, \lambda^n$, be a set of uniformly spread weighted vectors, z^* be the ideal point and values of $f_j(x)$ in problem (3) have been normalized. Thus, the objective function of i-th sub-problem can be referred to problem (2). *TBCAP* is decomposed into scalar optimization sub-problems. A neighborhood of weight vector λ^i is defined as a set of its several weight vectors in Λ. The neighborhood of i-th sub-problem consists of all the sub-problems with the weight vectors from the neighborhood of λ^i. MOEA/D provides an easy yet efficient way to take the advantage of scalarization method and solve all subproblems simultaneously with different objective preference in a single run. In this paper, Λ is used to guide the problem specific operators for adjusting the degree of power consumption, reliability and efficiency of data gathering and therefor obtaining different preference barrier coverage.

Algorithm 2. Local Search Strategies

Input:

χ_k (one individual of k-th sub-problem)

N_N^k (neighborhood size of the sub-problem)

Output:

χ_k (updated individual of k-th sub-problem)

1: Randomly choose a neighborhood j of χ_k
2: If $j < k$
3: For $i = 1 : k - j$
Forward-LS(χ_k,χ_j)
End-for
4: else
5: For $i = 1 : j - k$
Backward-LS(χ_k,χ_j)
End-for
6: **return** χ_k
7: End

4.2 Genetic Operators

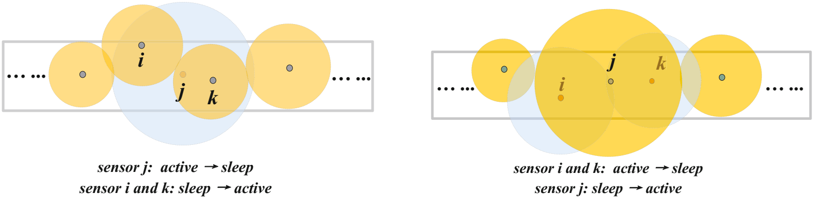
Selection operators choose the most suitable solutions to produce offspring. In this paper, we have adopted a tournament selection operator based on utility for each sub-problem, which has been tested to be fast and effective [22]. Mutation operator randomly selects two genes within a specific range (a relatively small interval), in order to be further improved by fine-tuning the solution.

4.3 Local Search: Forward-LS and Backward-LS

Two original problem-specific local search strategies, as shown in Algorithm 2, have been developed. There are two search directions, i.e., *Forward-LS* (Fig. 3(a)) and *Backward-LS* (Fig. 3(b)).

The idea of problem-specific local search strategies is inspired by workload balancing, which is to construct two possible search directions for an offspring whose performance is better. The search procedure is from starting point to ending point. We set the search direction based on the number of active sensors of starting point and ending point. The starting point can be randomly selected, and the ending point is the best individual of the neighborhood. When an offspring shows improvement in terms of the objective function, it is adopted as the solution of this subproblem. The details are given in Algorithm 3 and 4.

For example, consider the *Forward-LS* in Fig. 3(a), the active sensor j with a large sensing radius to cover a specific region B of the barrier. Then, search from the nearby sleeping sensors to check if there exists two sleeping sensors i and k, which can be assigned sensing ranges to cover B. If exists, we set sensor i from the status active to sleep, and sensor i and k from sleep to active with corresponding radii.



(a) Forward local search (b) Backward local search

Fig. 3. Local search procedures description

Algorithm 3. Forward-LS

Input: χ_k, χ_j
Output:
χ_k
1: Find the gene g with maximum radii r_g
2: Find the two nearest genes g_1 and g_2 with radii *zero* around gene g
3: Assign the radius to genes g_1 and g_2 to replace the gene g, produce χ'_k
4: If χ'_k is better than χ_k
$\chi_k \leftarrow \chi'_k$
End-if
5: **return** P
6: End

4.4 Dynamical Resource Allocation

The sub-problems may have different computational difficulties, which makes it reasonable to assign different amounts of computational effort to different problems [22]. As we can see, for the *TBCAP*, the complexity fits binomial distribution with the number of active sensors. Thereby, more computational resource based on utility will be assigned to the sub-problems with higher complexity.

4.5 Utilities Update

We define and compute a utility for each sub-problem. Computational efforts are distributed to these sub-problems based on their utilities. If evaluation times is a multiplication of a certain number, then we compute the relative decrease of the objective for each sub-problem i, Δ_i. The utility of the sub-problem can be calculated as follows.

$$\mu_i = \begin{cases} 1.00 & \text{if } \Delta_i > 0.001 \\ (0.99 + 0.01\frac{\Delta_i}{0.001}) & \text{otherwise} \end{cases}$$

4.6 Perturbation

Perturbation improves the quality of solutions found by PS-MOEA/D, thereby speeding up the search for global optimal solution. Let χ_k be the current solution

Algorithm 4. Backward-LS

Input: χ_k, χ_j
Output:
χ_k
1: Find two disjoint genes g_1 and g_2 with the minimum radius
2: Find the nearest gene g_0 with radii 0 to g_1 and g_2
3: Assign the radii to g_0 to replace the gene g_1 and g_2, produce χ'_k
4: If χ'_k is better than χ_k
$\chi_k \leftarrow \chi'_k$
End-if
5: **return** P
6: End

to the k-th sub-problem, we apply a random interchange move on χ_k to produce χ'_k. It randomly selects a number of genes within a specific range (a relatively large interval), in order to jump out of local optimum.

5 Experiments and Discussions

5.1 Experimentation

This section presents the setup of the experimentation, with the purpose of validating the performance of the implemented PS-MOEA/D. The experiments are conducted on a 3.4GHz Intel PC with 4GB RAM. The programming language is MATLAB(R2013a). The proposed algorithm runs with the following parameter values: the maximum number of evaluations $M_E = 1{,}000$, neighborhood size $niche = 20$, mutation rate $P_m = 1.0$ and the experiments for each instance are replicated for 10 independent trials. Since an analysis of the parameter sensitivity is not a major concern of this study, we have not performed any previous analysis to fix these values.

Depending on the deployment method, the coordinates of the sensor positions may follow a particular distribution. For instance, if sensors are thrown off an aircraft that flies over the middle of a field, most sensors are expected to fall somewhere close to the central line, and several sensors are likely to end up further out. One could then argue that the sensor distribution is uniform along the axis of route. Thus, the experiments fall into two major parts, i.e., Uniform distribution and Gaussian distribution. In the experiments, the length of barrier and the default number of sink node is set as 1000 units and *one*, respectively. The offsets of the sensors are assumed to be 0.

5.2 Performance Comparison

In this section, we study the effectiveness of the proposed PS-MOEA/D on *TBCAP*. To do so, we compare the proposed method with the WSA.

Performance Measure The quality of the obtained non-dominated solutions is usually evaluated from three perspectives: (i) the closeness to the true PF, (ii) diversity and (iii) uniformity. No single metric can reflect all these aspects and often a number of metrics are used. In this study, we use the *Set Coverage* C(X,Y) $(X \succeq Y)$ [23] and *distance to reference set* $D_{ref}(X,R)$ [24] metrics.

$$C(X,Y) = \frac{|y \in Y | \exists x \in X : x \prec y|}{|Y|} \tag{4}$$

The C(X,Y) metric compute the percentage of solutions in Y dominated by solutions in X, divided by the total number of solutions in Y. The higher the value of C(X,Y) obtained, more diversely and uniformly the solution set X distributed.

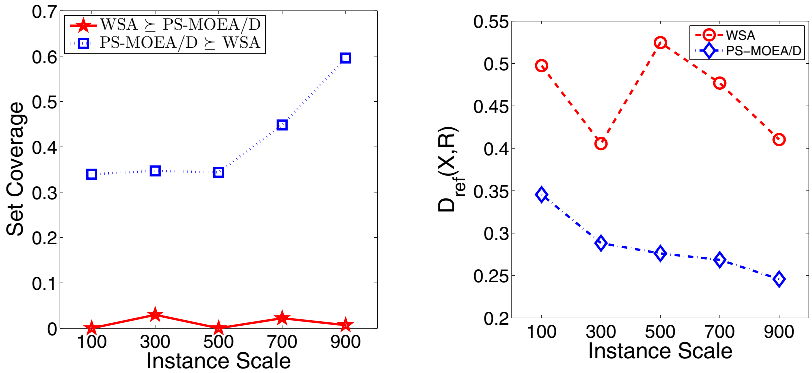
$$D_{ref}(X,R) = \frac{\sum_{r \in R}\{min_{x \in X}\{dis(x,r)\}\}}{|R|} \tag{5}$$

The distance from reference set calculates the average distance from a solution in the reference set R to the closest solution in X. The smaller the value of $D_{ref}(X,R)$, the closer the set X is to R. In the absence of the real reference set (i.e., true PF), we calculate the average distance of each single point to the nadir point since we consider minimization objectives.

Comparison with the WSA. We validate the performance of PS-MOEA/D by conducting comparison experiments in different scale (the number of randomly deployed sensors) *TBCAP*. Fig. 4 shows that the PS-MOEA/D outperforms the WSA in terms of *set coverage* and *distance to reference set* on Uniform instances, where the horizontal axis represents the number of randomly deployed sensors and the vertical axis represents the mean values of *set coverage* and *distance to reference set*. Similar results on Gaussian instances can be found in Fig. 5. From the experimental results, it is observed that PS-MOEA/D obtains better PFs than the WSA. Specifically, in both figures, the PS-MOEA/D obtains a percentage of dominance 30% to 60%; if we check the inverse coverage relation, the fraction of non-dominated solutions achieved by the WSA that dominates the Pareto sets obtained by PS-MOEA/D, in all cases, this fraction is close to 0%. Besides, PS-MOEA/D performs better on average than WSA in terms of *distance to reference set*. In addition, it can be noticed that PS-MOEA/D shows better stability of results as the instance scale increases. Summarizing that, the PS-MOEA/D has obtained more evenly distributed PFs providing a better approximation towards the nadir point than the WSA.

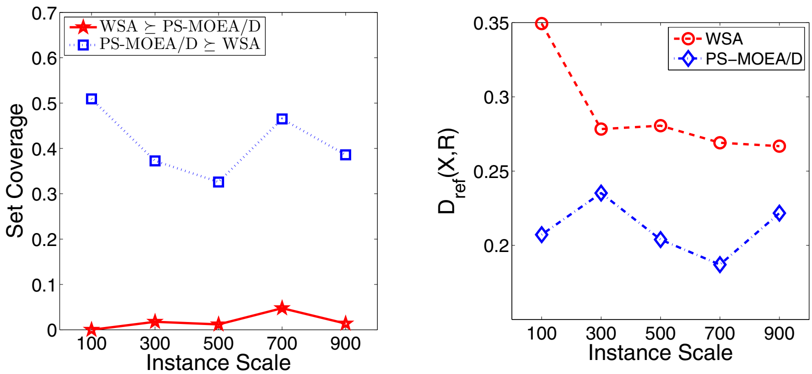
5.3 Existence of Knee Regions

Knee points are made up of Pareto-optimal solutions, which provide the best possible tradeoff among the three conflicting objectives, in other words, any improvement in one objective must outweigh the aggregated deterioration of other objectives. These are probably the most interesting solutions in many real-world problems. Faced with multiple methods for finding knee points [25] [26] [27], since it is a challenging topic to find the true extreme Pareto optimal solutions for the *TBCAP*, we propose to find the knee points based on a trade-off metric designed by Rachmawati and Srinivasan [26].



(a) Set Coverage (b) D_{ref}

Fig. 4. Comparison between PS-MOEA/D and WSA on Uniform instance



(a) Set Coverage (b) D_{ref}

Fig. 5. Comparison between PS-MOEA/D and WSA on Gaussian instance

Following this metric, we define a notation $\rho(X_i, S)$ to represent the least amount of improvement per unit deterioration by substituting any alternative solution from non-dominated solution set S with X_i. Solutions residing in convex knee regions have the highest values in terms of $\rho(X_i, S)$. It allows us to define the strong degree of knee points by setting a threshold value θ. It can be mathematically defined as follow.

$$\rho(X_i, S) = min_{X_j \in S; i \neq j} \frac{\sum_{1 \leq m \leq M} max(0, f_m(X_j) - f_m(X_i))}{\sum_{1 \leq m \leq M} max(0, f_m(X_i) - f_m(X_j))} \tag{6}$$

$$S_{knee}^{\theta} = \{X_i | \rho(X_i, S) \geq \theta, X_i \in S\} \tag{7}$$

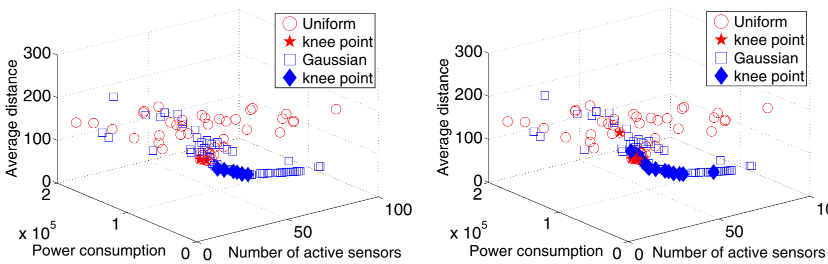
X_j corresponds to a member of the non-dominated solutions S that are non-dominated with respect to X_i; $f_m(X_i)$ corresponds to the m-th objective value of solution X_i, S^{θ}_{knee} denotes the set of knee points with the threshold value θ.

To demonstrate the existence of a knee region for this problem, two sets of experiments have been conducted with different θ values, namely 0.5 and 0.25. In Fig. 6, it shows that the obtained knee points obtained on the 500 sensors Uniform and Gaussian deployment. Note that in this part of simulation, we assume there is only one sink node, which is located in the middle of the barrier. It can be noticed that the fronts have clear knee points in which it is more reasonable to take a final decision about which solution should be adopted.

Besides, comparing with the Uniform deployment method, more knee points have been found by the Gaussian deployment method. The PF obtained by the Uniform deployments spreads more evenly than the Gaussian deployment. Except for the reason of sensors' positions, the biggest reason is the location of the sink node. Since in this part of simulation, the sink node is assumed to be located in the middle of the barrier. Following the Gaussian deployment, a large number of sensors are deployed closely to the middle of the barrier. Thus, in this case, the solutions tend to be high quality for the objective of average distance.

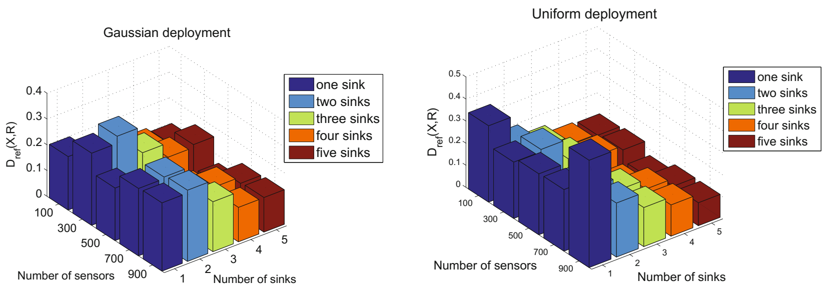
5.4 Effect of the Number of Sink Nodes

Intuitively, when more sink nodes are deployed, the estimated average distance from sensor nodes to the nearest sink node should be shorter and influence other objectives. The number of sink nodes is one of the factors that may influence the obtained PFs. We compare the obtained PFs by PS-MOEA/D with the number of sink nodes from 1 to 5. Assume that the sink nodes are uniformly located along the barrier, then we run the experiments to validate the effect of the number of sink nodes. As expected, from the results of Fig. 7, we can observe an important property that more sink nodes are uniformly deployed, better PFs can be obtained. The major reason is that the active sensors may have more sink nodes to be chosen as the nearest sink.



(a) $S^{0.5}_{knee}$ (b) $S^{0.25}_{knee}$

Fig. 6. Knee region of instance with 500 sensors



(a) Gaussian deployment (b) Uniform deployment

Fig. 7. D_{ref} values found by varying number of sinks on different scale instances

6 Conclusion

This paper has made several contributions. Firstly, *TBCAP* is defined and formulated. Secondly, a PS-MOEA/D has been proposed for finding optimal tradeoff solutions. Thirdly, an experimental investigation has been presented, which explores the tradeoff among reliability, power consumption and average distance. A comparative study is conducted to evaluate the proposed approach. Additionally, the effect of the number of sink nodes to the PF have also been studied. Our future work will enrich the model to make it closer to reality and further improve the performance of the PS-MOEA/D.

Acknowledgments. This work was supported in part by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China [Project No. CityU 115312].

References

1. Yu, Z., Teng, J., Li, X., Xuan, D.: On wireless network coverage in bounded areas. In: Proceedings of the IEEE Conference on Computer Communications, pp. 1195–1203. IEEE (2013)
2. Fan, H., Lee, V.C.S., Li, M., Zhang, X., Zhao, Y.: Barrier coverage using sensors with offsets. In: Cai, Z., Wang, C., Cheng, S., Wang, H., Gao, H. (eds.) WASA 2014. LNCS, vol. 8491, pp. 389–400. Springer, Heidelberg (2014)
3. Fan, H., LI, M., Sun, X., Wan, P.J., Zhao, Y.: Barrier coverage by sensors with adjustable ranges. ACM Transactions on Sensor Networks **11**(1), 14(1)—14(20) (2014)
4. Konstantinidis, A., Yang, K., Zhang, Q., Zeinalipour-Yazti, D.: A multi-objective evolutionary algorithm for the deployment and power assignment problem in wireless sensor networks. Computer Networks **54**(6), 960–976 (2010)
5. Yoon, Y., Kim, Y.H.: An efficient genetic algorithm for maximum coverage deployment in wireless sensor networks. IEEE Transactions on Cybernetics **43**(5), 1473–1483 (2013)

6. Zhang, Q., Li, H.: Moea/d: A multiobjective evolutionary algorithm based on decomposition. IEEE Transactions on Evolutionary Computation **11**(6), 712–731 (2007)
7. Mhatre, V.P., Rosenberg, C., Kofman, D., Mazumdar, R., Shroff, N.: A minimum cost heterogeneous sensor network with a lifetime constraint. IEEE Transactions on Mobile Computing **4**(1), 4–15 (2005)
8. Onur, E., Ersoy, C., Deliç, H., Akarun, L.: Surveillance wireless sensor networks: deployment quality analysis. IEEE Network **21**(6), 48–53 (2007)
9. Sun, Z., Wang, P., Vuran, M.C., Al-Rodhaan, M.A., Al-Dhelaan, A.M., Akyildiz, I.F.: Bordersense: Border patrol through advanced wireless sensor networks. Ad Hoc Networks **9**(3), 468–477 (2011)
10. Kumar, S., Lai, T.H., Arora, A.: Barrier coverage with wireless sensors. In: Proceedings of Annual International Conference on Mobile Computing And Networking, pp. 284–298. ACM (2005)
11. Saipulla, A., Westphal, C., Liu, B., Wang, J.: Barrier coverage with line-based deployed mobile sensors. Ad Hoc Networks **11**(4), 1381–1391 (2013)
12. He, S., Gong, X., Zhang, J., Chen, J., Sun, Y.: Curve-based deployment for barrier coverage in wireless sensor networks. IEEE Transactions on Wireless Communications **13**(2), 724–735 (February 2014)
13. Lee, E., Park, S., Lee, J., Oh, S., Kim, S.H.: Novel service protocol for supporting remote and mobile users in wireless sensor networks with multiple static sinks. Wireless Networks **17**(4), 861–875 (2011)
14. Chen, J., Li, J., He, S., He, T., Gu, Y., Sun, Y.: On energy-efficient trap coverage in wireless sensor networks. ACM Transactions on Sensor Networks **10**(1), 2–29 (2013)
15. Shakkottai, S., Srikant, R., Shroff, N.: Unreliable sensor grids: coverage, connectivity and diameter. Proceedings of the IEEE Conference on Computer Communications **2**, 1073–1083 (March 2003)
16. Sengupta, S., Das, S., Nasir, M., Vasilakos, A.V., Pedrycz, W.: An evolutionary multiobjective sleep-scheduling scheme for differentiated coverage in wireless sensor networks. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews **42**(6), 1093–1102 (2012)
17. Martins, F.V., Carrano, E.G., Wanner, E.F., Takahashi, R.H., Mateus, G.R.: A hybrid multiobjective evolutionary approach for improving the performance of wireless sensor networks. IEEE Sensors Journal **11**(3), 545–554 (2011)
18. Lanza-Gutierrez, J.M., Gomez-Pulido, J.A., Vega-Rodriguez, M.A., Sanchez-Perez, J.M.: A parallel evolutionary approach to solve the relay node placement problem in wireless sensor networks. In: Proceeding of Annual Conference on Genetic and Evolutionary Computation, pp. 1157–1164. ACM (2013)
19. Coello, C.C., Lamont, G.B., Van Veldhuizen, D.A.: Evolutionary algorithms for solving multi-objective problems. Springer (2007)
20. Miettinen, K.: Nonlinear multiobjective optimization, vol. 12. Springer (1999)
21. Smith, J., Fogarty, T.C.: Self adaptation of mutation rates in a steady state genetic algorithm. In: Proceedings of IEEE International Conference on Evolutionary Computation, pp. 318–323. IEEE (1996)
22. Zhang, Q., Liu, W., Li, H.: The performance of a new version of moea/d on cec09 unconstrained mop test instances. In: IEEE Congress on Evolutionary Computation, pp. 203–208 (2009)
23. Zitzler, E., Thiele, L.: Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. IEEE Transactions on Evolutionary Computation **3**(4), 257–271 (1999)

24. Czyzżak, P., Jaszkiewicz, A.: Pareto simulated annealing a metaheuristic technique for multiple-objective combinatorial optimization. Journal of Multi-Criteria Decision Analysis **7**(1), 34–47 (1998)
25. Deb, K., Miettinen, K., Sharma, D.: A hybrid integrated multi-objective optimization procedure for estimating nadir point. In: Ehrgott, M., Fonseca, C.M., Gandibleux, X., Hao, J.-K., Sevaux, M. (eds.) EMO 2009. LNCS, vol. 5467, pp. 569–583. Springer, Heidelberg (2009)
26. Rachmawati, L., Srinivasan, D.: Multiobjective evolutionary algorithm with controllable focus on the knees of the pareto front. IEEE Transactions on Evolutionary Computation **13**(4), 810–824 (2009)
27. Bechikh, S., Ben Said, L., Ghédira, K.: Searching for knee regions in multi-objective optimization using mobile reference points. In: Proceedings of the ACM Symposium on Applied Computing, pp. 1118–1125. ACM (2010)