

Problem Specific MOEA/D for Barrier Coverage with Wireless Sensors

Xiao Zhang, Yu Zhou, *Student Member, IEEE*, Qingfu Zhang, *Senior Member, IEEE*, Victor C. S. Lee, *Member, IEEE*, and Minming Li, *Senior Member, IEEE*

Abstract—Barrier coverage with wireless sensors aims at detecting intruders who attempt to cross a specific area, where wireless sensors are distributed remotely at random. This paper considers limited-power sensors with adjustable ranges deployed along a linear domain to form a barrier to detect intruding incidents. We introduce three objectives to minimize: 1) total power consumption while satisfying full coverage; 2) the number of active sensors to improve the reliability; and 3) the active sensor nodes' maximum sensing range to maintain fairness. We refer to the problem as the tradeoff barrier coverage (TBC) problem. With the aim of obtaining a better trade-off among the three objectives, we present a multiobjective optimization framework based on multiobjective evolutionary algorithm (MOEA)/D, which is called problem specific MOEA/D (PS-MOEA/D). Specifically, we define a 2-tuple encoding scheme and introduce a cover-shrink algorithm to produce feasible and relatively optimal solutions. Subsequently, we incorporate problem-specific knowledge into local search, which allows search procedures for neighboring subproblems collaborate each other. By considering the problem characteristics, we analyze the complexity and incorporate a strategy of computational resource allocation into our algorithm. We validate our approach by comparing with four competitors through several most-used metrics. The experimental results demonstrate that PS-MOEA/D is effective and outperforms the four competitors in all the cases, which indicates that our approach is promising in dealing with TBC.

Index Terms—Barrier coverage, evolutionary algorithms, multiobjective optimization, wireless sensor networks (WSNs).

I. INTRODUCTION

IN LAST several years, there are a growing number of applications in the domain of wireless sensor networks (WSNs). One of the most significant applications is intrusion detection.

Manuscript received December 15, 2015; revised April 3, 2016; accepted June 15, 2016. This work was supported in part by the National Natural Science Foundation of China under Grant 61473241, and in part by the Research Grants Council of Hong Kong under Project CityU 117913. A preliminary version of this paper appeared in the Proceedings of the 8th International Conference on Evolutionary Multi-Criterion Optimization. This paper was recommended by Associate Editor G. G. Yen. (*Corresponding author: Yu Zhou.*)

X. Zhang, Y. Zhou, V. C. S. Lee, and M. Li are with the Department of Computer Science, City University of Hong Kong, Hong Kong (e-mail: xiao.zhang@my.cityu.edu.hk; yzhou57-c@my.cityu.edu.hk; csvlee@cityu.edu.hk; minming.li@cityu.edu.hk).

Q. Zhang is with the Department of Computer Science, City University of Hong Kong, Hong Kong, and also with the School of Computer Science and Electronic Engineering, University of Essex, Colchester CO4 3SQ, U.K. (e-mail: qingfu.zhang@cityu.edu.hk).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCYB.2016.2585745

Specifically, wireless sensors are deployed to detect intruders that attempt to penetrate a country border or boundary of battlefield. These surveillance applications using WSNs solution, which are referred to as barrier coverage [1], have been extensively studied and discussed in the past few years [2], [3]. In most of the applications, the deployment region of wireless sensors can be abstracted as a line segment. The linear WSNs own the advantages of high efficiency, easy implementation, and extension [4], [5]. With respect to deterministic deployment of sensors, the high efficiency of WSNs can be achieved by comprehensive and detailed analysis [6], [7]. However, surveillance tasks may need to be carried out in hard-to-reach areas, where limited-power sensors have to be distributed remotely in random. For example, wireless sensors, which are dropped from aircraft, have to wake up and operate independently in an unattended and hostile environment, organize themselves as a network to detect the intrusion.

Power conservation is a crucial issue for extending lifetime of wireless barrier. Because the power of sensor nodes is limited and can hardly be recharged after deployment, energy-efficient mechanism is needed to reduce power consumption of nodes while meeting the coverage requirement. With regard to deploying sensor nodes in random, redundant sensor nodes are needed to satisfy full coverage and extend the lifetime of the barrier. Hence, it is desirable to make some sensors in active state and keep other sensors in sleeping state. Inspired by this technique, the problem of min-cost linear coverage (MCLC) problem [8] is to minimize the total power consumption while each point on the barrier is covered. However, this problem cannot be solved optimally by efficient algorithms since it has been proved to be NP-hard [8]. In addition, minimizing total power consumption of the network cannot ensure minimizing individual sensor's power consumption. Sensor nodes can consume different amount of power. Sensor nodes' power will get depleted quickly by using large sensing ranges, which then lead to the decrease of network lifetime. Thus, minimizing active sensor nodes' maximum sensing range is significant, which is to measure the sensor networks performance in a fairness perspective. Moreover, sensors are failure-prone and each single sensor fails independently with a certain probability [9]. Given the requirement of full coverage, the fewer sensors are activated, the higher reliability will be achieved. Thus, it is important to minimize the number of active sensors to improve the reliability without impacting the coverage.

In this paper, instead of solving the single objective problems with various constraints found in previous works, we formulate a multiobjective optimization algorithm to obtain a better tradeoff among the following three objectives.

- 1) *Objective 1 (Power)*: Minimizing the total power consumption.
- 2) *Objective 2 (Reliability)*: Minimizing the number of active sensors.
- 3) *Objective 3 (Fairness)*: Minimizing the active sensor nodes' maximum sensing range.

However, these three objectives are conflicting in nature and the detailed illustration can refer to Section III-B.

This problem can be regarded as a multiobjective optimization problem (MOP), named tradeoff barrier coverage (TBC) problem. Traditional algorithms might not be applicable and only a few techniques address these objectives simultaneously. Multiobjective evolutionary algorithm (MOEA), which is a population based algorithm that naturally leads to Pareto optimal solutions, has been successfully applied to deal with MOPs in WSNs [10]–[12]. In recent years, several MOEAs have been proposed to give approximations on the Pareto optimal solutions (Pos). However, a high-efficiency MOEA should attain solutions while maintaining high diversity and convergence and be in close proximity to the Pos. Among the most-used approaches, strength Pareto evolutionary algorithm 2 (SPEA2) [13] and nondominated sorting genetic algorithm II (NSGA-II) [14] consider a MOP as a whole and utilize the Pareto dominance relation to sort the solutions. However, evenly distributed Pareto solutions may not always be easily attained. MOEA/D treats a MOP as a group of subproblems with initial weights before searching, and optimizes these subproblems in a collaborative manner. The decomposition framework cannot only help to guarantee the fast convergence of the approximated solutions but also maintain a considerable diversity of the Pareto solutions. Moreover, it is reported in [15] that, MOEA/D with objective normalization is able to handle objectives with disparate scales efficiently for complex problems. Inspired by the benefits of the framework of MOEA/D, a novel and MOEA/D-based algorithm using problem specific knowledge is applied to TBC in this paper.

A. Overview of Our Contributions

In this paper, we redefine the barrier coverage problem to a MOP with three objectives, which is called TBC. Objective 1 is related to the power efficiency issue, since sensors are equipped with a limited-power battery. To this end, the network as a whole must minimize the total power consumption for the sake of power conservation. Objective 2 focuses on minimizing the number of active sensors. This objective is motivated by the facts that the sensors are failure-prone and there is an inverse correlation between the number of active sensors and reliability [9]. In addition, to put it crudely, minimizing the active sensor nodes' maximum sensing range is to make sure that the sensing ranges of the activated sensors are distributed fairly, which is referred to as objective 3. This objective is suitable to measure the sensor network

performance from a fairness perspective. Below are the main contributions of this paper.

- 1) With the aim of optimizing the TBC, we present a multiobjective optimization algorithm based on MOEA/D [16]. Specifically, we define a 2-tuple encoding scheme. Then, we discover an interesting observation on the geometric structure of the problem and introduce a cover-shrink algorithm to repair our solutions, which allows us to have feasible and relatively optimal solutions.
- 2) Subsequently, we incorporate problem-specific knowledge into local search, which allows search procedures for neighboring subproblems collaborate each other. By considering the problem characteristics, we analyze the complexity of the problem and find out the distribution of the hardness. It is desirable to incorporate a strategy of computational resource allocation to our problem [17]. Each subproblem is allocated a certain amount of computational resource according to the distribution.
- 3) We validate our approach by comparing with four representative MOEAs through three most-used metrics. The experimental results show that PS-MOEA/D outperforms the four competitors in almost all the cases. In addition, we analyze the convergence of our algorithm and show the benefits of computational resource allocation strategy.

The remainder of this paper is organized as follows. A literature review is conducted to get an overview on the most influential studies in Section II. Section III gives the network model and assumptions, and describes a multiobjective formulation on TBC. The framework of the proposed PS-MOEA/D for TBC is presented in Section III-C. The details of PS-MOEA/D, including the specification and implementation, are given in Section IV. The experiment results and performance comparisons are provided in Section V. Finally, this paper is concluded in Section VI.

II. RELATED WORKS

Barrier coverage with wireless sensors concentrates on detecting intruders that attempt to penetrate a specific area [18], [19], where a chain of sensor nodes are deployed on or along the virtual border to form a network [20]. With the requirement of protecting sensitive facilities and national borders increasing, barrier coverage has a growing number of applications.

Power scarcity is a crucial issue for wireless barrier coverage due to the restrictions on sensors' battery size. A number of barrier coverage problems with respect to power issue have been explored in recent years [21]–[23]. Sensor scheduling [24], [25] is a widely used method to prolong the lifetime of networks. It helps to extend the lifetime of networks by inactivating some sensors while satisfying the coverage requirement. However, minimizing the instant energy is more practical than maximizing coverage lifetime, because the unanticipated failure of sensors invalidates the obtained scheduling. Fan *et al.* [8] studied MCLC to minimize the instant power

consumption, which has been prove to be NP-hard. In addition to these studies, some approximation algorithms have been proposed [8], [26].

Network outage or service unavailability, partial or whole, may not only be due to power exhaustion of the sensor nodes. Some sensors may lose its function due to mechanical issues 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 the sensor networks for coverage. To improve the reliability of WSN, Shakkottai *et al.* [27] considered an unreliable wireless sensor grid network for coverage, where the sensors are placed in a square area. In this model, all sensors are failure-prone, i.e., each single sensor node fails independently with a certain probability.

A sensor can maintain its functioning until its battery power is fully exhausted. The function of the virtual barrier keeps on working until power exhaustion of the first active sensor. Any dead sensor can result in gaps or holes of the barrier, which may lead to incalculable consequences [28]. If one active sensor's sensing range in the coverage is set to be too large, its battery power will be depleted quickly [29].

Giving a proposal for wireless barrier coverage taking into account so many aspects simultaneously is a challenging and significant issue. To this end, it is desirable to use MOEAs to solve such WSNs optimization problems. Rajagopalan *et al.* [30] presented MOEA approaches to deal with distributed detection for area coverage or monitoring. The target of the problem is to minimize the power consumption and path loss while the energy of detected signal is maximized. Lanza-Gutierrez *et al.* [31] tried to solve the relay node placement problem under SPEA2 [13] and NSGA-II frameworks. However, it is difficult to address the complex instances since the running time is very long. While both coverage and power issues have been extensively explored in the past few years, little effort however, has been made to tackle the coverage, power consumption and reliability simultaneously or explicitly. Martins *et al.* [32] give a multiobjective hybrid optimization algorithm to maintain the network connectivity and a certain degree of coverage, while prolonging the network lifetime. They also considered that sensors are failure-prone. This is a good attempt to address WSN optimization problem in the multiobjective perspective.

As reported in [10], fuzzy dominance is incorporated into the original MOEA/D to solve the differentiated coverage problem. They aimed to obtain a better tradeoff among the coverage, energy consumption, and the network lifetime. MOEA/D is an easily-extensible but efficient evolutionary multiobjective algorithm framework. Several applications and extensions on MOEA/D have been made recently [33]–[35]. Zhang *et al.* [36] considered a barrier coverage problem in multiobjective perspective. In addition to sensor nodes for sensing the intruders, the sink nodes are used to gather data in [36]. Except for the objectives of power consumption and reliability, they study the efficiency of data gathering between sensors and sink nodes without consideration for the power consumption of sink nodes. For this MOP, they present an algorithm based on MOEA/D, which was compared with a weighted sum algorithm experimentally.

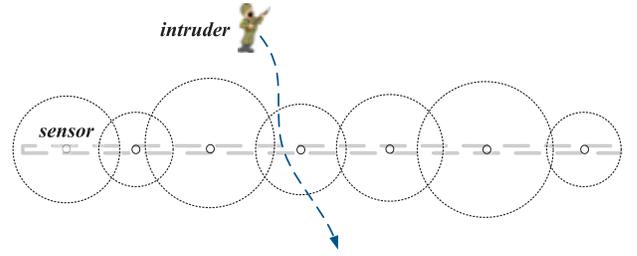


Fig. 1. Wireless barrier coverage model.

III. MODEL AND PROBLEM FORMULATION

A. Model and Assumptions

1) *Barrier Model*: We view the barrier as a long, narrow region. It can be abstracted to a line segment, where a set of sensors $\{s_1, s_2, \dots\}$ from left to right are distributed in random. Fig. 1 gives an example scenario to illustrate the barrier model. An intruder can be detected by the WSN as it attempts to penetrate the linear barrier.

Assumption 1: The sensor nodes with limited power are randomly deployed on a linear domain.

Assumption 2 (Disc Sensing): Each active sensor has an adjustable sensing range of r ; any moving object within the sensing range can be reliably detected by the sensor.

Assumption 3: The power consumption for each sensor s_i is $P_i = \rho \cdot r_i^\kappa$ where $\kappa \in [2, 6]$ is path-loss exponent and ρ is a proportionality constant.

B. Multiobjective Formulation

We define the TBC as a MOP in the following way. Let B be a line segment on the x -axis, called a barrier. With a little abuse of notation, for any point x on L , we also use x to denote its coordinated on L . We say the coordinates of left and right endpoint of B is 0 and β , respectively. Let $\mathbf{S} = \{s_1, s_2, \dots, s_n\}$ be a set of n sensors and randomly located on the line segment B . Each sensor s_i with coordinate x_i has a sensing range r_i . The problem is to activate a subset $\mathbf{S}^* \subseteq \mathbf{S}$ and assign them sensing ranges such that the barrier B is fully covered by the sensors in \mathbf{S}^* . TBC is formulated as a MOP.

Decision Variable 1: Working status of sensor s_i

$$\mu_i = \begin{cases} 1, & \text{if sensor } s_i \text{ is active} \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

Decision Variable 2: Sensing range of sensor s_i : r_i , $0 \leq r_i \leq r_{\max}$.

Constraint: Every point p of the barrier B is covered by at least one active sensor s_i : $p \in [x_i - r_i, x_i + r_i]$.

Objective 1 (Power): Minimizing the total power consumption: $f_1 = \sum_{s_i \in \mathbf{S}^*} \rho \cdot r_i^\kappa$.

Objective 2 (Reliability): Minimizing the number of active sensors nodes: $f_2 = \sum_{s_i \in \mathbf{S}^*} \mu_i$.

Objective 3 (Fairness): Minimizing the active sensor nodes' maximum sensing range: $f_3 = \min \max_{s_i \in \mathbf{S}^*} r_i$.

Next, we take a simple instance to illustrate the conflict among the objectives. Fig. 2 shows three feasible solutions for the coverage problem, where the length of the barrier is assumed to be 1 and $\kappa = 2$.

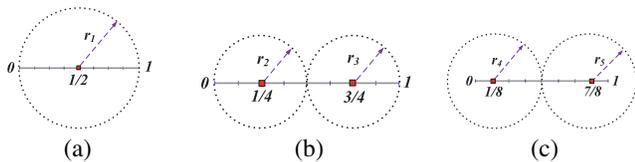


Fig. 2. Example to illustrate of conflict among the objectives. (a) $P_1 = r_1^2 = (1/4)$. (b) $P_2 = r_2^2 + r_3^2 = (1/8)$. (c) $P_3 = r_4^2 + r_5^2 = (9/32)$.

- 1) *Power Versus Reliability*: In the first solution [Fig. 2(a)], the power consumption is $P_1 = r_1^2 = (1/4)$, the number of active sensors R_1 is 1, and the active sensor nodes' maximum sensing range $F_1 = (1/2)$. In the second solution [Fig. 2(b)], the power consumption is calculated by $P_2 = r_2^2 + r_3^2 = (1/8)$, and the number of active sensors R_2 is 2. Thus, we have that $P_1 > P_2$ yet $R_1 < R_2$.
- 2) *Power Versus Fairness*: Considering Fig. 2(c), $P_3 = r_4^2 + r_5^2 = (9/32)$, and the active sensor nodes' maximum sensing range $F_3 = (3/8)$. We have $P_1 < P_3$ yet $F_1 > F_3$.
- 3) *Reliability Versus Fairness*: In Fig. 2(c), the number of active sensors R_3 is 2. We have $F_1 > F_3$ yet $R_1 < R_3$.

In summary, 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, reducing the maximum sensing range of sensor nodes, may increase the number of active sensors and/or power consumption. However, to meet the coverage requirement, either more sensors, if available, are activated to cover the region or larger sensing ranges are assigned, leading to higher power consumption. Thus, finding the tradeoff among them is worth exploring.

C. PS-MOEA/D for TBC

TBC is generally formulated as follows:

$$\begin{aligned} & \text{minimize } F(\tilde{\delta}) = (f_1(\tilde{\delta}), f_2(\tilde{\delta}), f_3(\tilde{\delta})) \\ & \text{subject to } \tilde{\delta} \in \tilde{\Omega} \end{aligned} \quad (2)$$

where $\tilde{\delta} = \{(\mu_1, r_1), (\mu_2, r_2), \dots, (\mu_i, r_i), \dots\}$ is the decision variable, and $\tilde{\Omega}$ is the decision space. f_1, f_2 , and f_3 correspond to objective 1 (power), objective 2 (reliability), and objective 3 (fairness) in Section III-B, respectively.

We have shown that the objectives in problem (2) may conflict with each other and an improvement on one objective may lead to the deterioration of another. Pos are optimal tradeoff candidates among all objectives. The details of Pareto optimum terminology are given in [37], in which Pareto dominance, Pareto optimal, Pareto set (PS), and Pareto front (PF) are explained explicitly. The true PF is difficult to find, so an approximation of the true PF is required to give a good perception to the practical problems.

Tchebycheff approach [38] is employed to decompose the MOP into a number of subproblems. Let $\lambda^1, \lambda^2, \dots, \lambda^j, \dots$, be a set of uniformly spread weighted vectors and \tilde{z} be an ideal point. The problem can be decomposed into scalar

optimization subproblems as follows:

$$\text{minimize } g^{te}(x|\lambda^j, \tilde{z}) = \max_{1 \leq i \leq 3} \left\{ \frac{1}{\lambda_i^j} \cdot |f_i(x) - \tilde{z}_i| \right\}. \quad (3)$$

Using the Tchebycheff approach with different weight vectors, different Pos can be obtained. MOEA/D minimizes all the three objective functions simultaneously. Neighborhood relations among these decomposed subproblems are determined according to the weight vectors. The original MOEA/D was first presented in [16], in which all the subproblems received equal treatment and were allocated the same amount of computational resource (i.e., evolution process). The idea of computational resource allocation was first given in [39], in which it had been observed that specific portions of the PF in an MOP would need more efforts to be computed than others. Thereby, it is reasonable that more computational resource should be assigned to the subproblems with higher complexity. The proposed PS-MOEA/D adopts the high-level idea of the computational resource strategies in MOEA/D-GRA [17] and adapts this technique to be applied to TBC. As shown in Algorithm 1, each subproblem with index j (j th subproblem), which refers to $g^{te}(x|\lambda^j, \tilde{z})$ in (3), is associated with a different probability prob_i .

First of all, we give the general framework with initial settings of PS-MOEA/D. The details of each algorithm will be specified in Section IV. At each iteration, they pass the population as parameters, and maintain and update the following data, in which ($i = 1, \dots, n$), n is the number of subproblems.

- 1) λ^i is to lead the operators for adjusting different objective preference for i th subproblem.
- 2) $\tilde{N}_i \subset \{1, \dots, n\}$: The neighborhood index set of i th subproblem.
- 3) \tilde{x}_i : The current solution to i th subproblem, which is defined as a struct. \tilde{x}_i is with domains of decision variables and three objectives values of i th subproblem.
- 4) $F(\tilde{x}_i)$: The objective function value of \tilde{x}_i , which is computed by the λ^i and the three objective values in \tilde{x}_i .
- 5) prob_i : The probability that i th subproblem should be invested. If prob_i is higher than some others, more evolution process will be conducted on the i th subproblem.
- 6) $\tilde{z} = \tilde{z}_1, \dots, \tilde{z}_m$: An invalid ideal solution, in which $m = 3$ is the number of objectives. It is a reference point of the optimal values in terms of the three objectives.

In Algorithm 1, TBC is decomposed into n scalar optimization subproblems. As indicated in line 13, for TBC, as one of the objectives, the number of active sensors, is discrete. It is natural to decompose the MOP based on the number of active sensors. In this end, the hardness for all subproblems are different and the hardness of each subproblem \tilde{H}_i follows a binomial distribution $\tilde{H}_i = (C_N^i/2^N)$ as shown in Fig. 3 where we set the problem scale $N = 100$. Thus, the proposed PS-MOEA/D is based on the MOEA/D-GRA, in which each i th subproblem is associated with a different probability prob_i . Remark that prob_i corresponds to \tilde{H}_i . The probability set p decides how

Algorithm 1 PS-MOEA/D Framework for TBC**Input:**

- N : Total number of sensors
- x_i : Coordinate for each sensor s_i for $1 \leq i \leq n$
- f_1, f_2, f_3 : Objective functions
- N_P : Population size
- N_N : Size of neighborhood

Output:

Pop : Final solutions

- 1: Initialize $\lambda^i, \tilde{N}_i, \tilde{x}^i$ and $prob_i$ for each i -th subproblem, $1 \leq i \leq n$
- 2: Initialize $\tilde{z} = \{\tilde{z}_1, \tilde{z}_2, \tilde{z}_3\}$ and Pop .
- 3: **while** Stopping criteria not satisfied **do**
- 4: **for** $i = 1$ to n **do**
- 5: **if** $rand() \leq prob_i$ **then**
- 6: Produce the sensors status \tilde{y} for i -th subproblem.
/* Offspring Reproduction and Replacement */
- 7: Conduct a Cover-Shrink algorithm on \tilde{y} for i -th subproblem to obtain the trial solution \tilde{y} .
- 8: Local Search Procedure on i -th subproblem and generate two trial solutions \tilde{y}_a and \tilde{y}_b .
- 9: Update \tilde{z}_j ($j=1,2,3$) by $\min\{\tilde{z}_j, f_j(\tilde{y})\}$.
- 10: Update the population Pop by \tilde{y}, \tilde{y}_a and \tilde{y}_b in terms of F .
- 11: **end if**
- 12: **end for**
- 13: Update the probability set $p = \{prob_1, \dots, prob_n\}$.
- 14: **end while**
- 15: **return** Pop

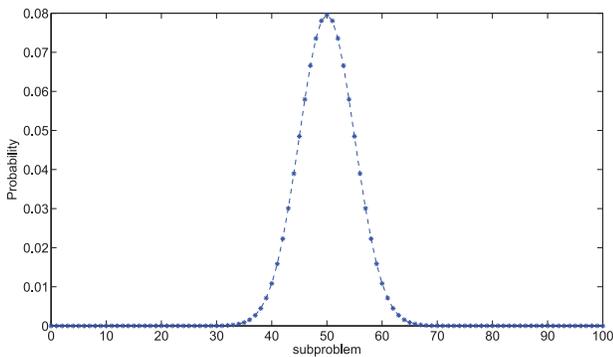


Fig. 3. Hardness distribution for subproblems.

computational resources are to be assigned among subproblems. If probability $prob_i \geq rand()$, i th subproblem is picked to be computed in lines 6–10.

IV. ALGORITHM SPECIFICATION AND IMPLEMENTATION

A. Solution Encoding

The solution is represented by a 2-tuple coding structure $C = \{C_s, C_r\} = \{(\mu_1, r_1), (\mu_2, r_2), \dots, (\mu_i, r_i), \dots, (\mu_N, r_N)\}$. The boolean μ_i describes the working status of the sensor node and r_i denotes the value of its radius. This encoding scheme is associated with the population initialization as line 6 of Algorithm 1. As shown in Fig. 4 for an example,

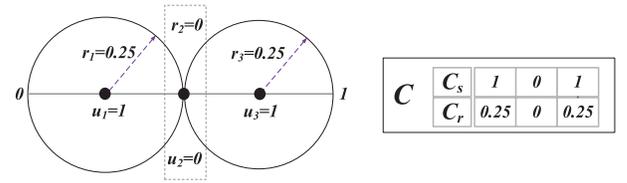


Fig. 4. Example of 2-tuple coding structure and the corresponding solution.

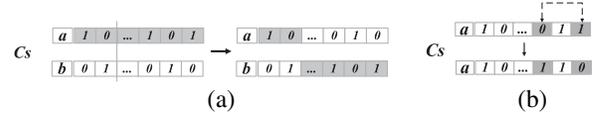


Fig. 5. Offspring reproduction. (a) Crossover. (b) Mutation.

we assume that the length of the line segment is 1 and three sensors are deployed on the line segment with coordinates (0.25, 0.5, 0.75). The solution picks two sensors $\{(\mu_1 = 1, r_1 = 0.25), (\mu_3 = 1, r_3 = 0.25)\}$ with the coordinates (0.25, 0.75), excluding the sensor with the coordinate (0.5) which is encoded as $\{(\mu_2 = 0, r_2 = 0)\}$.

B. Reproduction and Replacement

1) *Crossover and Mutation*: The sensors status C_{s^i} of the solution \tilde{y} is produced in line 6 in Algorithm 1. We set one parent a to the current solution for i th subproblem, and randomly select a solution b as the other parent from $\{\tilde{x}^t | t \in \tilde{N}_i\}$. Then, one could produce two new solutions by an one-point crossover on solutions a and b [40], as shown in Fig. 5(a). We choose the better solution as y_1 , then replace the solution \tilde{x}^t of corresponding t th subproblem if improved. We can use mutation operator, i.e., interchange as shown in Fig. 5(b), to make the number of gene “1” or “0” be equal with \tilde{x}^a . The mutation operator selects two nonidentical genes within a relatively small interval in random, in order to be further improved by fine-tuning the solution. For the replacement, we can use the newly generated solution \tilde{x} to replace \tilde{x}^t if improved. The details are shown in Algorithm 2.

2) *Repair Operator*: The repair operator is to handle the constraint of full coverage, i.e., to ensure the feasibility of the solutions. We design a cover-shrink algorithm, which consists of two parts (cover and shrink), is to ensure the requirement and functionality of barrier coverage with the smallest possible deterioration on the objectives. We set x_0 as the leftmost point and β as the rightmost point of the line segment B . We use $d_{(i-1)}$ to denote the distance $x_i - x_{i-1}$ for $1 < i < n$, $d_1 = x_1$ and $d_n = \beta - x_n$. The cover part is based on an algorithm in [8]. Though it is an approximation algorithm, and there could be several overlaps between sensors, we can improve it by the shrink part. The detail of cover-shrink algorithm is shown in Algorithm 3, whose overall time complexity is $O(N)$. After this, we finalize the initialization and generate the trial solutions as shown in line 6 of Algorithm 1.

Consider the example in Fig. 6, we process the cover and shrink procedures as follows. First, we set $r_1 = \max(d_1, (d_{12}/2))$, $r_2 = \max((d_{12}/2), (d_{23}/2))$, and $r_3 = \max((d_{23}/2), d_3)$ according to the cover procedure. We

Algorithm 2 Offspring Generation for i th Subproblem

1: Set population

$$Pool = \begin{cases} \tilde{N}_i & \text{if } rand() < prob_i \\ \{1, \dots, N\} & \text{otherwise} \end{cases}$$

- 2: $\tilde{x}^a \leftarrow \tilde{x}^i$, and select $\tilde{x}^b \in \tilde{N}_i$ in random
- 3: Generate a new solution y_1 by an one-point crossover operator on \tilde{x}^a and \tilde{x}^b /* With the probability of crossover P_{Cr} */
- 4: **if** $g^{te}(\tilde{x}^i | \lambda^j, z^*) > g^{te}(y_1 | \lambda^j, z^*)$ **then**
- 5: Set $\tilde{x}^i = y_1$
- 6: **end if**
- 7: Generate a new solution y_2 by randomly interchange two non-identical genes (0 - 1) on \tilde{x}^i /* With the probability of mutation P_{Mu} */
- 8: **if** $g^{te}(\tilde{x}^i | \lambda^j, z^*) > g^{te}(y_2 | \lambda^j, z^*)$ **then**
- 9: Set $\tilde{x}^i = y_2$
- 10: **end if**

Algorithm 3 Cover-Shrink Algorithm

- 1: Initialize $k = 0$ and a solution \tilde{x}
- 2: **for** $i = 1$ to N **do**
- 3: **if** $u_i = 1$ **then**
- 4: $k = k + 1$, $v_k = 1$, coordinate $x_k = x_i$
- 5: **end if**
- 6: **end for**
- 7: **for** $j = 1$ to k **do**
- 8: **if** $j == 1$ **then**
- 9: $r_1 = \max(d_1, \frac{d_{(2)(1)}}{2})$ /* Cover part */
- 10: **else if** $j == k$ **then**
- 11: $r_k = \max(\frac{d_{(k)(k-1)}}{2}, d_k)$
- 12: **else**
- 13: $r_j = \max(\frac{d_{(j)(j-1)}}{2}, \frac{d_{(j+1)(j)}}{2})$
- 14: **end if**
- 15: **end for**
- 16: **for** $i = 2$ to $k - 1$ **do**
- 17: **if** $(x_i - r_i < x_{i-1} + r_{i-1}) \& \& (x_i + r_i > x_{i+1} - r_{i+1})$ **then**
- 18: $r_i = \max(0, \min(x_i - x_{i-1} - r_{i-1}, x_{i+1} - x_i - r_{i+1}))$
/* Shrink part */
- 19: **end if**
- 20: **end for**

can see that there are two overlaps o_1 and o_2 for sensor s_2 . Then, we can shrink the sensing range of s_2 by $\min(o_1, o_2)$ to make it tangent with s_1 or s_3 .

C. Local Search Procedure

In each iteration as shown line 8 in Algorithm 1, when the best-so-far solutions \tilde{x}_{is} are updated, a local search procedure is performed to refine \tilde{x}_{is} . The idea of problem-specific local search strategies is inspired by workload balancing. After we randomly choose the i th subproblem with solution \tilde{x}^i , we conduct our problem-specific local search strategies on it, specifically forward operator and backward operator. With respect to the search direction, the original point can be chosen

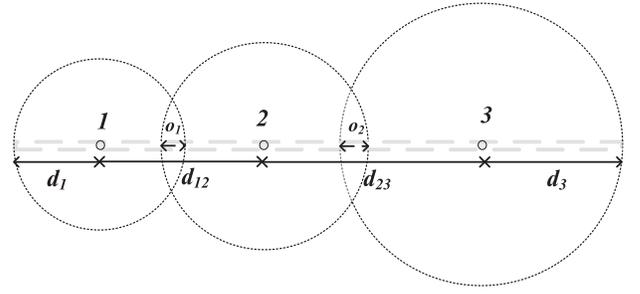


Fig. 6. Illustration of cover-shrink algorithm.

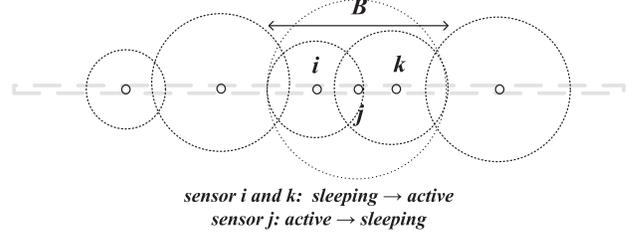


Fig. 7. Forward local search operator.

in random, and the destination is its 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 4. After the local search procedure, we update the solutions of $(i - 1)$ th subproblem and $(i + 1)$ th subproblem. Take the forward operator in Fig. 7 as example, the active sensor j with a large sensing range 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 and there is an improvement on objectives, the status of sensor i is set from active to sleeping, and the status of sensors i and k is set from sleeping to active with corresponding radii. Similarly, backward operator is the inverse process of forward operator. Noted that the solutions produced by Algorithm 4 have been repaired to be feasible.

D. Complexity Analysis

1) *Space Complexity*: During the search, PS-MOEA/D uses an internal population $O(N)$ to store nondominated solutions and external population to store current solutions $O(N)$ for each iteration. The size of population is determined after the initialization. It will not increase with the iterations.

2) *Time Complexity*: The major computational cost in each iteration of PS-MOEA/D is involved in lines 6–8 of Algorithm 1. Specifically, after the initialization, Algorithm 3 needs $O(N)$ time. In each generation, two individuals are chosen for later processing, which can be done in constant time. In relation to the local search, as shown in Algorithm 4, line 2 for searching the gene g with maximum radii takes $O(N)$ time. Then, for the lines 2–7, they take $O(1)$ time. Line 8 for finding the minimum d takes $O(N)$ time, and the other lines take $O(1)$ time. Overall, Algorithm 4 runs in $O(N)$ time. Thus, the overall time complexity for each generation is $O(N^2)$.

Algorithm 4 Local Search Strategies

-
- 1: Pick the i -th subproblem with solution \tilde{x}^i
 - 2: Search the gene g with maximum radii r_g and two nearest genes $g_1 = g_2 = 0$ /* forward operator */
 - 3: Assign the radii to genes g_1 and g_2 to replace the cover range of the gene g
 - 4: Set $g = 0, g_1 = g_2 = 1$ and obtain \tilde{x}_{ls}^{i+1}
 - 5: **if** \tilde{x}_{ls}^{i+1} is better than \tilde{x}^{i+1} **then**
 - 6: $\tilde{x}^{i+1} \leftarrow \tilde{x}_{ls}^{i+1}$
 - 7: **end if**
 - 8: Find two disjoint genes g_3 and g_4 with the minimum d /* backward operator */
 - 9: Find the nearest gene $g_0 = 0$ between g_3 and g_4 if exists
 - 10: Assign the radii to g_0 to replace the cover range of the gene g_3 and g_4
 - 11: Set $g_0 = 1, g_3 = g_4 = 0$ and obtain \tilde{x}_{ls}^{i-1}
 - 12: **if** \tilde{x}_{ls}^{i-1} is better than \tilde{x}^i **then**
 - 13: $\tilde{x}^{i-1} \leftarrow \tilde{x}_{ls}^{i-1}$
 - 14: **end if**
-

V. EXPERIMENTS AND DISCUSSION

In this section, we assess the performance of the proposed PS-MOEA/D for TBC. The proposed PS-MOEA/D is compared with MOEA/D-GRA and other three well-known MOEAs to prove the effectiveness and validate the performance. The details of this experimentation are introduced as follows.

A. Experimentation

This section is devoted to give the techniques and parameters adopted in the experiments. Then, three performance measures are given for the evaluation on MOEAs.

1) *Compared Techniques*: With the aim of studying whether the solutions of PS-MOEA/D is comparable to the widely used MOEAs, we compare the implemented PS-MOEA/D with the following algorithms.

- 1) MOEA/D-GRA is the baseline algorithm, in which a generalized resource allocation strategy based on utility function is incorporated into the original MOEA/D. Computational efforts are distributed to the subproblems based on their utilities [36]. The utility for each subproblem is computed by the relative decrease of the objective in a number of times of evaluation. Tchebycheff decomposition is employed in MOEA/D-GRA.
- 2) NSGA-II [14] selects individuals according to Pareto dominance relation and reproduces offsprings iteratively. Specifically, the new population is sorted by ranking according to the relationship of dominance, and the solutions are widespread by using crowding distance.
- 3) SPEA2 is an elitist MOEA. This algorithm adopts a density estimation to discriminate between individuals and an enhanced archive truncation for diversity preservation. SPEA2 is widely used to treat theory and practice problems in recent year, such that it becomes a reference algorithm.

TABLE I

PARAMETER CONFIGURATION FOR MOEAS, WHERE S_PO REFERS TO THE SIZE OF POPULATION, P_CR REFERS TO THE PROBABILITY OF CROSSOVER, P_MU REFERS TO THE PROBABILITY OF MUTATION, AND S_NE REFERS TO THE SIZE OF NEIGHBORHOOD

	PS-MOEA/D	MOEA/D-GRA	NSGA-II	SPEA2	IBEA
S_Po	100	100	100	100	100
P_Cr	0.8	0.8	0.8	0.7	0.8
P_Mu	0.2	0.2	0.1	0.2	0.2
S_Ne	5	5	-	-	-

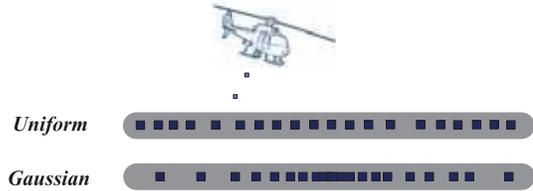


Fig. 8. Wireless sensor deployments following uniform and Gaussian distribution.

- 4) IBEA [41] defines the optimization goal in the light of a performance measure (hypervolume indicator in this paper) and then directly uses this hypervolume indicator in the selection process to control the new population.

Note that we are using the same reproduction mechanism (crossover, mutation, and solution repair procedure) for the four competitors and PS-MOEA/D.

2) *Parameters Setup and Test Data*: The experiments were conducted on a 3.4 GHz Intel PC with 4 GB RAM. The programming language is MATLAB (R2013a). Following the practice in [11], the parameters for each algorithm have been tuned slightly according to the preliminary experiments on the quality of the solutions produced. With the purpose of making a fair comparison, for all MOEAs, we set the parameters based on the best performance, as shown in Table I. Unless stated otherwise, the experiments are conducted under these settings.

According to different deployment methods and purposes, the coordinates of the sensor positions may obey different distributions. More specifically, when the sensors are dropped off from an aircraft that flies over the middle of a field to protect an important target, most sensors are expected to fall somewhere close to the central line, and a few sensors are likely to end up further out. Thus, the resulting sensor deployment obeys a Gaussian distribution. One could then argue that if all of the sensors are deployed randomly and uniformly along the axis of flight path, the resulting sensor deployment obeys a uniform distribution. For these reasons and suggestions given in [42], both uniform distribution and Gaussian distribution as shown in Fig. 8 are studied. In the experiments, the length of barrier is set as 1000 units. And we set $\rho = 1$ and $\kappa = 2$. With the aim of evaluating the performance of our algorithm, we will conduct comparative experiments in different scales (the number of randomly deployed sensors). Specifically, we name the test instance of 100 randomly deployed sensors under uniform (Gaussian) distribution as U100 (G100), and so on.

3) *Performance Metrics*: In this paper, we use the metrics of hypervolume $HV(X, R)$ [43], inverted generational

distance (IGD) [44] and set coverage [43]. HV and IGD can measure the quality of the obtained PF from the perspectives of both convergence and diversity.

The hypervolume metric, as shown in (4), gives the volume that is calculated by the solution set S and reference set r . We calculate the volume by using normalized objective values. The large value of hypervolume indicates that the algorithm is able to produce good-quality solutions.

The IGD is defined as follows. Let P^* denote the true PF in an MOP. Let S be an obtained solution set. The mean distance from P^* to S is formulated as (5), where $\text{dis}(s, P^*)$ computes the smallest Euclidean distance from p to all points in S . In order to attain a smaller value of IGD, one should be closer to and lose less part of the complete true PF. However, except for test benchmark problems, the real PFs of real world optimization problems are usually unknown. As suggested in [45], in the absence of the true PF in our problem, the unknown PF is approximated by selecting nondominated solutions among all solutions obtained by different MOEAs (i.e., PS-MOEA/D, MOEA/D-GRA, NSGA-II, SPEA2, and IBEA)

$$\text{HV}(S, r) = \text{Volume} \left(\bigcup_{i=1}^{|S|} v_i \right) \quad (4)$$

$$\text{IGD}(S, P^*) = \frac{\sum_{p \in P^*} \text{dis}(p, S)}{|P^*|}. \quad (5)$$

The set coverage $C(\tilde{A}, \tilde{B})$ metric computes the pairwise domination relation between solution sets \tilde{A} and \tilde{B} . As shown in (6), it concentrates on the overlaps between two solution sets, a (b) denotes the solution in set \tilde{A} (\tilde{B}). The higher the value of $C(\tilde{A}, \tilde{B})$ obtained, more diversely and evenly the solution set \tilde{A} distributed. Remark that set coverage does not depend on true PFs or reference set. It is applicable to make the comparison among solutions obtained by MOEAs

$$C(\tilde{A}, \tilde{B}) = \frac{|b \in \tilde{B} | \exists a \in \tilde{A} : a < b|}{|\tilde{B}|}. \quad (6)$$

B. Performance Evaluation

In this section, we give the performance comparisons of the proposed algorithm with MOEA/D-GRA, as well as three representative approaches in the multiobjective domain, i.e., NSGA-II, SPEA2, and IBEA.

First, we compare above MOEAs with the proposed algorithm by using hypervolume and IGD metrics. All the results presented are obtained by executing 20 independent runs of each algorithm on each instance. From the numerical point of view, Tables II and III collect the average and standard deviation values (mean_{std}) of hypervolume and IGD obtained by PS-MOEA/D, MOEA/D-GRA, NSGA-II, and SPEA2. We point out those instances in which there exists differences statistically nonsignificant between our proposed algorithm and the competitors by S -indicator. A significance level is set to 5% as in [11].

The notation ++ reflects that our algorithm obtains a significant improvement over all the four competitors, while N+ indicates an insignificant improvement of PS-MOEA/D over

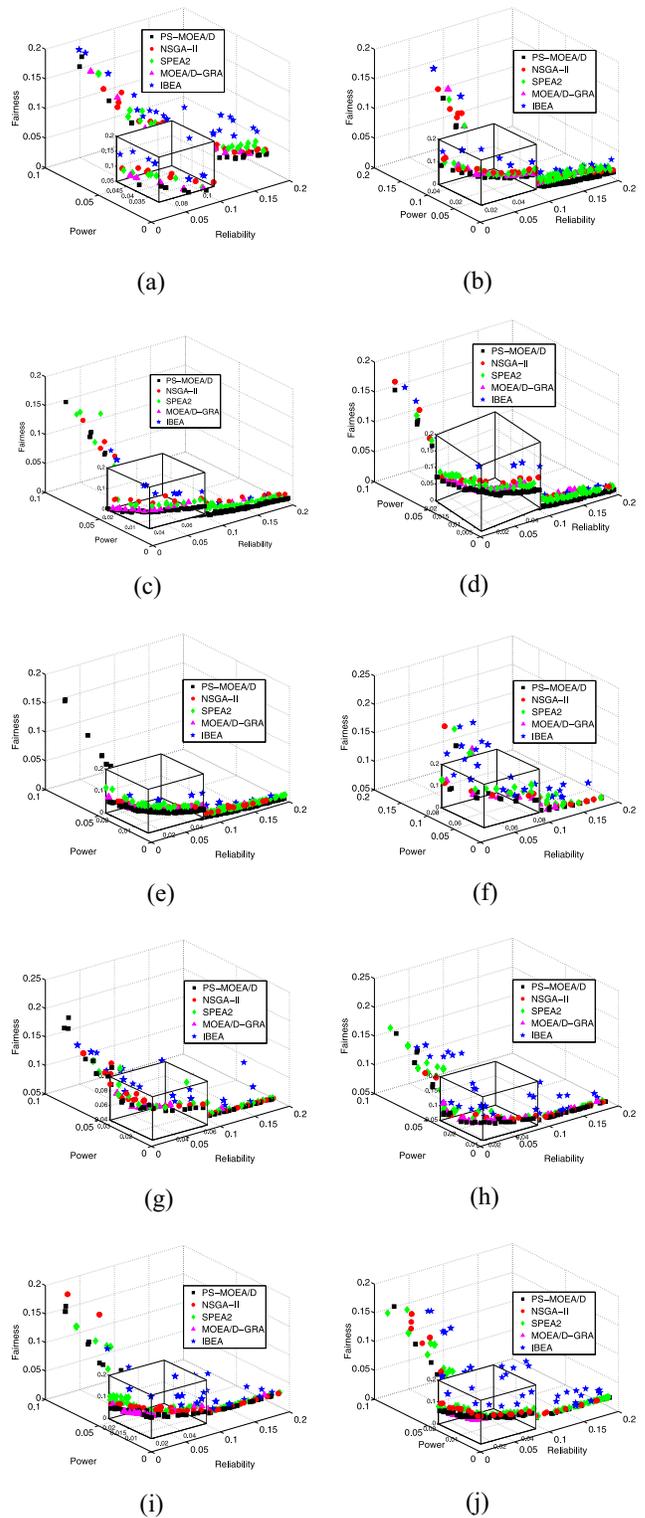


Fig. 9. Nondominated solutions of a single run of all the algorithms for each instance. (a) U100. (b) U300. (c) U500. (d) U700. (e) U900. (f) G100. (g) G300. (h) G500. (i) G700. (j) G900.

NSGA-II. Overall, PS-MOEA/D generally outperforms the other approaches in all the instances. Furthermore, with the aim of highlighting the superiority of PS-MOEA/D, we have selected the nondominated solutions of a single run of all the algorithms for each instance, and present graphic representations of their obtained nondominated solutions in Fig. 9 and zoom in

TABLE II
PERFORMANCE COMPARISON OF MOEA/D, NSGA-II, SPEA2, AND IBEA WITH
PS-MOEA/D FROM THE VIEWPOINT OF HYPERVOLUME (mean_{std})

	MOEA/D-GRA	NSGA-II	SPEA2	IBEA	PS-MOEA/D	S-indicator
<i>U</i> 100	0.3750 _{0.0319}	0.4351 _{0.0408}	0.1832 _{0.0570}	0.3348 _{0.0130}	0.4955 _{0.0178}	++
<i>U</i> 300	0.3537 _{0.0193}	0.4400 _{0.0649}	0.2205 _{0.0234}	0.3937 _{0.0124}	0.5944 _{0.0113}	++
<i>U</i> 500	0.3786 _{0.0904}	0.4552 _{0.0741}	0.3168 _{0.0904}	0.4683 _{0.0767}	0.6090 _{0.0119}	++
<i>U</i> 700	0.3245 _{0.0163}	0.4598 _{0.0751}	0.3137 _{0.0344}	0.4054 _{0.0901}	0.6304 _{0.0110}	++
<i>U</i> 900	0.3193 _{0.0915}	0.4529 _{0.0750}	0.3239 _{0.1251}	0.3940 _{0.0614}	0.6335 _{0.0113}	++
<i>G</i> 100	0.2435 _{0.0691}	0.4083 _{0.0295}	0.1430 _{0.0836}	0.3582 _{0.0317}	0.4266 _{0.0005}	N+
<i>G</i> 300	0.2002 _{0.0112}	0.4060 _{0.0683}	0.1885 _{0.0437}	0.3919 _{0.0786}	0.4251 _{0.0001}	N+
<i>G</i> 500	0.1257 _{0.0571}	0.4182 _{0.0967}	0.3189 _{0.0408}	0.3593 _{0.0624}	0.4321 _{0.0001}	N+
<i>G</i> 700	0.1682 _{0.0689}	0.4408 _{0.0310}	0.3406 _{0.0450}	0.4339 _{0.0622}	0.4773 _{0.0005}	++
<i>G</i> 900	0.1740 _{0.0963}	0.4521 _{0.0618}	0.3672 _{0.0400}	0.4373 _{0.0655}	0.5195 _{0.0000}	++

TABLE III
PERFORMANCE COMPARISON OF MOEA/D, NSGA-II, SPEA2, AND IBEA
WITH PS-MOEA/D FROM THE VIEWPOINT OF IGD (mean_{std})

	MOEA/D-GRA	NSGA-II	SPEA2	IBEA	PS-MOEA/D	S-indicator
<i>U</i> 100	0.0129 _{0.0058}	0.0029 _{0.0015}	0.0024 _{0.0004}	0.0094 _{0.0009}	0.0008 _{0.0001}	++
<i>U</i> 300	0.0076 _{0.0006}	0.0029 _{0.0013}	0.0027 _{0.0005}	0.0049 _{0.0008}	0.0004 _{0.0000}	++
<i>U</i> 500	0.0055 _{0.0004}	0.0022 _{0.0011}	0.0023 _{0.0011}	0.0030 _{0.0006}	0.0003 _{0.0000}	++
<i>U</i> 700	0.0057 _{0.0008}	0.0026 _{0.0009}	0.0024 _{0.0012}	0.0037 _{0.0007}	0.0004 _{0.0000}	++
<i>U</i> 900	0.0062 _{0.0005}	0.0021 _{0.0008}	0.0025 _{0.0008}	0.0039 _{0.0005}	0.0007 _{0.0000}	++
<i>G</i> 100	0.0231 _{0.0005}	0.0069 _{0.0006}	0.0063 _{0.0003}	0.0110 _{0.0005}	0.0028 _{0.0001}	++
<i>G</i> 300	0.0247 _{0.0004}	0.0050 _{0.0003}	0.0068 _{0.0032}	0.0084 _{0.0002}	0.0019 _{0.0001}	++
<i>G</i> 500	0.0212 _{0.0030}	0.0059 _{0.0003}	0.0060 _{0.0028}	0.0091 _{0.0017}	0.0012 _{0.0000}	++
<i>G</i> 700	0.0220 _{0.0020}	0.0060 _{0.0008}	0.0050 _{0.0018}	0.0088 _{0.0007}	0.0013 _{0.0000}	++
<i>G</i> 900	0.0231 _{0.0045}	0.0047 _{0.0008}	0.0037 _{0.0011}	0.0079 _{0.0008}	0.0020 _{0.0000}	++

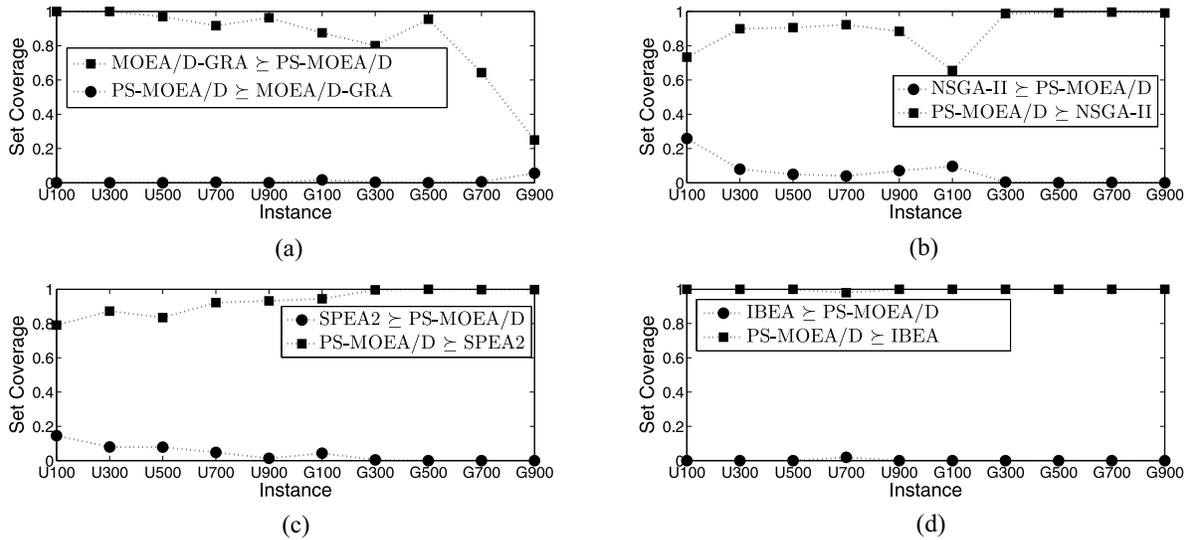


Fig. 10. Pairwise comparison between PS-MOEA/D and competitors in terms of set coverage. PS-MOEA/D versus (a) MOEA/D-GRA, (b) NSGA-II, (c) SPEA2, and (d) IBEA.

some area for better visualization. It can be observed that PS-MOEA/D obtains higher quality solutions from the distribution of the nondominated solutions in the figures.

With regard to the IGD values in Table III, PS-MOEA/D perform the best, having a clear advantage over the other four competitors on all the test cases. Specifically, all the MOEAs obtain better results on uniform test cases rather than Gaussian ones. MOEA/D-GRA has the worst IGD results on all the test cases compared to other approaches. This is because poor IGD results are likely to be given to solution sets with good convergence but small diversity when reference solutions have large diversity.

Second, we compare the solutions of the MOEAs by using the set coverage metric, which allows comparison of two algorithms in terms of Pareto dominance. In Fig. 10, a coverage relation in terms of pairs is presented. As shown in Fig. 10(a), the solutions obtained by PS-MOEA/D generally dominate above 60% of those obtained by MOEA/D-GRA, and even over 80% on uniform instances. PS-MOEA/D works fairly well on *G*700 and *G*900 over MOEA/D-GRA. This is probably because MOEA/D-GRA could produce good but nonuniformly distributed solutions by assigning large amounts of computational resources in some specific region. It can be observed from Fig. 10(b) and (c) that the PS-MOEA/D dominates above 65%

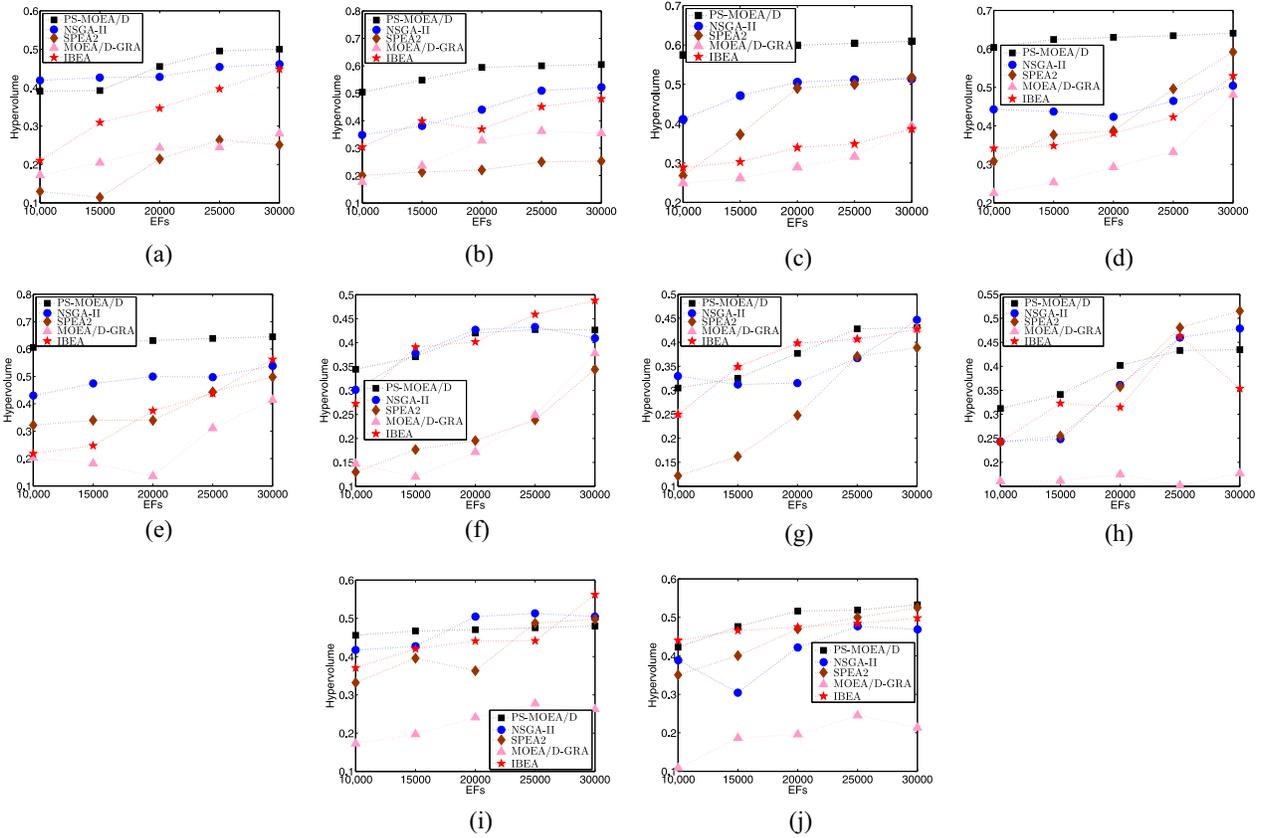


Fig. 11. Convergence analysis on five MOEAs for each instance. (a) U100. (b) U300. (c) U500. (d) U700. (e) U900. (f) G100. (g) G300. (h) G500. (i) G700. (j) G900.

of the nondominated solutions obtained by the NSGA-II of a run, and more than 80% of the nondominated solutions obtained by the SPEA2. As we can see from Fig. 10(d), PS-MOEA/D attains 100% approximately over IBEA in nearly all the instances. Moreover, compared with NSGA-II, SPEA2, and IBEA [Fig. 10(b)–(d)], we observe very good dominance results obtained by our algorithm when the instance scale becomes large.

Moreover, we show an example of the convergence analysis on a single run of each instance in Fig. 11. We use the hypervolume as the indicator. Obviously, we can see that all MOEAs typically have good convergence in terms of hypervolume, but fall into slight degradation occasionally. This is most probably because it is affected by failure of maintaining the diversity of solutions. Nevertheless, it is clearly shown that PS-MOEA/D achieves not only high-quality PFs, but also a good convergence.

Furthermore, simulation results of convergence analysis on PS-MOEA/D are also shown in Fig. 12 to investigate the influence of different evaluations on the obtained PFs. We only show a portion for each PF for clearer presentation. Obviously, we can observe that the solutions on large number of evaluations is better than the others. It can be graphically observed that the superiority of the solutions obtained on 30 000 evaluations. Moreover, in the case that the number of active sensor nodes is either very large or small, we can hardly obtain good solutions when the evaluation time is very low. This is because

the assigned computational resource follows binomial distribution. When the evaluation time is low, very little computational resource are assigned to the subproblems with very small or large number of active sensor nodes. Thus, few solutions can be found during the low number of evaluation times. Besides, comparing to the Gaussian instances, the PFs obtained spread more evenly in the uniform ones. This is because a large number of sensors are deployed closely to the middle of the barrier by following the Gaussian deployment. Since the constraint for TBC is to fully cover the whole barrier instead of a specific target, densely-deployed sensors in a region reduce the diversities of the PFs to some degree.

Summarizing, we have evaluated the performance of four multiobjective algorithms in two different simulation scenarios, i.e., uniform and Gaussian. Three quantitative metrics have been adopted for the evaluation of solutions, which are hypervolume, IGD, and set coverage. Regarding to the hypervolume and IGD, PS-MOEA/D outperforms the MOEA/D-GRA, NSGA-II, and SPEA2 in all the cases. Also, PS-MOEA/D shows good Pareto dominance over the other competitors. We can say that our approach attains, in all cases, comparable performance to the most-used approaches, obtaining the best results in TBC. With respect to the convergence of our algorithm, it clearly shows that PS-MOEA/D attains a fast convergence with high-quality solutions, and validate the effectiveness of the computational resource allocation strategy.

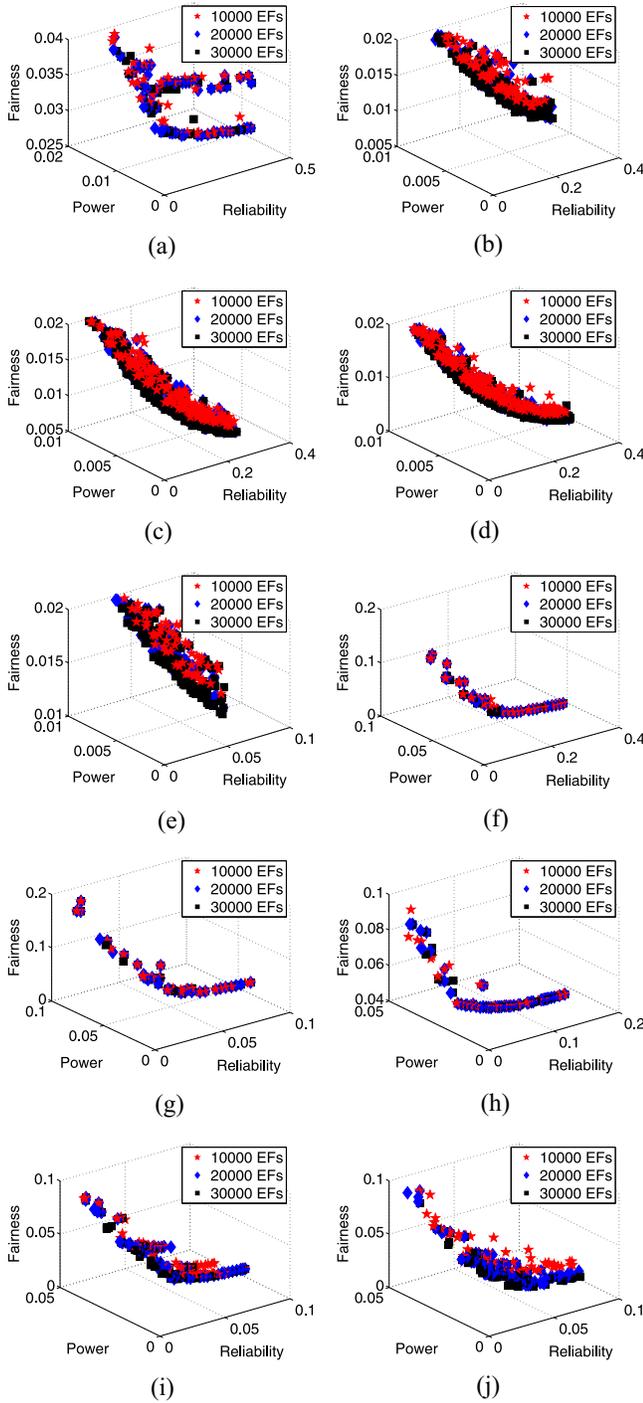


Fig. 12. PFs obtained by PS-MOEA/D convergence along the evaluations for each instance. (a) U100. (b) U300. (c) U500. (d) U700. (e) U900. (f) G100. (g) G300. (h) G500. (i) G700. (j) G900.

VI. CONCLUSION

In this paper, we study the multiobjective problem TBC, which is refined from the barrier coverage problem. A PS-MOEA/D is proposed for finding optimal tradeoff solutions for TBC. More concretely, we define a 2-tuple encoding scheme. Then, a cover-shrink algorithm is proposed to produce feasible and relatively optimal solutions. Subsequently, we incorporate problem-specific knowledge into local search,

which allows search procedures for neighboring subproblems collaborate each other. Experimental study is provided, which explores the tradeoff among power consumption, reliability and fairness. We perform diverse comparisons to validate our approach. The results indicate that PS-MOEA/D is very competitive in dealing with TBC. PS-MOEA/D may be further improved by exploring more problem-specific knowledge in future and would be significant to undertake a real-world border surveillance task.

REFERENCES

- [1] S. Kumar, T. Lai, and A. Arora, "Barrier coverage with wireless sensors," *Wireless Netw.*, vol. 13, no. 6, 2007, pp. 817–834.
- [2] Y. Yoon and Y.-H. Kim, "An efficient genetic algorithm for maximum coverage deployment in wireless sensor networks," *IEEE Trans. Cybern.*, vol. 43, no. 5, pp. 1473–1483, Oct. 2013.
- [3] X. Liu, "A deployment strategy for multiple types of requirements in wireless sensor networks," *IEEE Trans. Cybern.*, vol. 45, no. 10, pp. 2364–2376, Oct. 2015.
- [4] M. Noori and M. Ardakani, "Characterizing the traffic distribution in linear wireless sensor networks," *IEEE Commun. Lett.*, vol. 12, no. 8, pp. 554–556, Aug. 2008.
- [5] A. Arora *et al.*, "A line in the sand: A wireless sensor network for target detection, classification, and tracking," *Comput. Netw.*, vol. 46, no. 5, pp. 605–634, 2004.
- [6] Z. Zhou, S. R. Das, and H. Gupta, "Variable radii connected sensor cover in sensor networks," *ACM Trans. Sensor Netw.*, vol. 5, no. 1, 2009, Art. no. 8.
- [7] B. Wang, H. Xu, W. Liu, and H. Liang, "A novel node placement for long belt coverage in wireless networks," *IEEE Trans. Comput.*, vol. 62, no. 12, pp. 2341–2353, Dec. 2013.
- [8] H. Fan, M. Li, X. Sun, P.-J. Wan, and Y. Zhao, "Barrier coverage by sensors with adjustable ranges," *ACM Trans. Sensor Netw.*, vol. 11, no. 1, 2014, Art. no. 14.
- [9] S. Sastry, T. Radeva, J. Chen, and J. L. Welch, "Reliable networks with unreliable sensors," *Pervasive Mobile Comput.*, vol. 9, no. 2, pp. 311–323, 2013.
- [10] S. Sengupta, S. Das, M. Nasir, A. V. Vasilakos, and W. Pedrycz, "An evolutionary multiobjective sleep-scheduling scheme for differentiated coverage in wireless sensor networks," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 42, no. 6, pp. 1093–1102, Nov. 2012.
- [11] A. Rubio-Largo, M. A. Vega-Rodriguez, J. A. Gomez-Pulido, and J. M. Sanchez-Perez, "Multiobjective metaheuristics for traffic grooming in optical networks," *IEEE Trans. Evol. Comput.*, vol. 17, no. 4, pp. 457–473, Aug. 2013.
- [12] C.-P. Chen *et al.*, "A hybrid memetic framework for coverage optimization in wireless sensor networks," *IEEE Trans. Cybern.*, vol. 45, no. 10, pp. 2309–2322, Oct. 2015.
- [13] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm," in *Proc. Evol. Methods Design Optim. Control Appl. Ind. Probl. (EUROGEN)*, vol. 3242. Athens, Greece, 2001, pp. 95–100.
- [14] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.
- [15] S. Jiang and S. Yang, "An improved multiobjective optimization evolutionary algorithm based on decomposition for complex Pareto fronts," *IEEE Trans. Cybern.*, vol. 46, no. 2, pp. 421–437, Feb. 2016.
- [16] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, Dec. 2007.
- [17] A. Zhou and Q. Zhang, "Are all the subproblems equally important? Resource allocation in decomposition-based multiobjective evolutionary algorithms," *IEEE Trans. Evol. Comput.*, vol. 20, no. 1, pp. 52–64, Feb. 2016.
- [18] E. Onur, C. Ersoy, H. Deliç, and L. Akarun, "Surveillance wireless sensor networks: Deployment quality analysis," *IEEE Netw.*, vol. 21, no. 6, pp. 48–53, Nov./Dec. 2007.
- [19] Z. Sun *et al.*, "Bordersense: Border patrol through advanced wireless sensor networks," *Ad Hoc Netw.*, vol. 9, no. 3, pp. 468–477, 2011.
- [20] S. Kumar, T. H. Lai, and A. Arora, "Barrier coverage with wireless sensors," in *Proc. 11th Int. Conf. Mobile Comput. Netw.*, Cologne, Germany, 2005, pp. 284–298.

- [21] P.-J. Wan and C.-W. Yi, "Coverage by randomly deployed wireless sensor networks," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2658–2669, Jun. 2006.
- [22] A. Saipulla, C. Westphal, B. Liu, and J. Wang, "Barrier coverage of line-based deployed wireless sensor networks," in *Proc. 28th IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Rio de Janeiro, Brazil, 2009, pp. 127–135.
- [23] Z. Wang, H. Chen, Q. Cao, H. Qi, and Z. Wang, "Fault tolerant barrier coverage for wireless sensor networks," in *Proc. 33rd Int. Conf. Comput. Commun.*, vol. 2, Toronto, ON, Canada, 2014, pp. 1869–1877.
- [24] S. Kumar, T. H. Lai, M. E. Posner, and P. Sinha, "Maximizing the lifetime of a barrier of wireless sensors," *IEEE Trans. Mobile Comput.*, vol. 9, no. 8, pp. 1161–1172, Aug. 2010.
- [25] A. Bar-Noy and B. Baumer, "Maximizing network lifetime on the line with adjustable sensing ranges," in *Proc. 7th ALGOSENSORS*, Saarbrücken, Germany, 2011, pp. 28–41.
- [26] J. Chen *et al.*, "On energy-efficient trap coverage in wireless sensor networks," *ACM Trans. Sensor Netw.*, vol. 10, no. 1, 2013, Art. no. 2.
- [27] S. Shakkottai, R. Srikant, and N. B. Shroff, "Unreliable sensor grids: Coverage, connectivity and diameter," *Ad Hoc Netw.*, vol. 3, no. 6, pp. 702–716, 2005.
- [28] A. Bar-Noy, D. Rawitz, and P. Terlecky, "Maximizing barrier coverage lifetime with mobile sensors," in *Proc. 21st Eur. Symp. Algorithms*, Sophia Antipolis, France, 2013, pp. 97–108.
- [29] H. Luo, H. Du, H. Huang, Q. Ye, and J. Zhang, "Barrier coverage with discrete levels of sensing and transmission power in wireless sensor networks," in *Proc. 8th China Conf. Wireless Sensor Netw.*, Xi'an, China, 2014, pp. 14–23.
- [30] R. Rajagopalan, C. K. Mohan, P. Varshney, and K. Mehrotra, "Multi-objective mobile agent routing in wireless sensor networks," in *Proc. IEEE Congr. Evol. Comput.*, vol. 2, Edinburgh, U.K., 2005, pp. 1730–1737.
- [31] J. M. Lanza-Gutierrez, J. A. Gomez-Pulido, M. A. Vega-Rodriguez, and J. M. Sanchez-Perez, "A parallel evolutionary approach to solve the relay node placement problem in wireless sensor networks," in *Proc. 15th Annu. Conf. Genet. Evol. Comput.*, Amsterdam, The Netherlands, 2013, pp. 1157–1164.
- [32] F. V. C. Martins, E. G. Carrano, E. F. Wanner, R. H. C. Takahashi, and G. R. Mateus, "A hybrid multiobjective evolutionary approach for improving the performance of wireless sensor networks," *IEEE Sensors J.*, vol. 11, no. 3, pp. 545–554, Mar. 2011.
- [33] X. Cai, Y. Li, Z. Fan, and Q. Zhang, "An external archive guided multiobjective evolutionary algorithm based on decomposition for combinatorial optimization," *IEEE Trans. Evol. Comput.*, vol. 19, no. 4, pp. 508–523, Aug. 2015.
- [34] Z. Wang, Q. Zhang, A. Zhou, M. Gong, and L. Jiao, "Adaptive replacement strategies for MOEA/D," *IEEE Trans. Cybern.*, vol. 46, no. 2, pp. 474–486, Feb. 2016.
- [35] S. Liao and Q. Zhang, "A multiutility framework with application for studying tradeoff between utility and lifetime in wireless sensor networks," *IEEE Trans. Veh. Technol.*, vol. 64, no. 10, pp. 4701–4711, Oct. 2015.
- [36] X. Zhang, Y. Zhou, Q. Zhang, V. C. S. Lee, and M. Li, "Multi-objective optimization of barrier coverage with wireless sensors," in *Proc. 8th Int. Conf. Evol. Multi Criterion Optim.*, Guimarães, Portugal, 2015, pp. 557–572.
- [37] C. A. C. Coello, G. B. Lamont, and D. A. Van Veldhuizen, *Evolutionary Algorithms for Solving Multi-Objective Problems*. New York, NY, USA: Springer, 2007.
- [38] K. Miettinen, *Nonlinear Multiobjective Optimization*. Boston, MA, USA: Kluwer, 1999.
- [39] Q. Zhang, W. Liu, and H. Li, "The performance of a new version of MOEA/D on CEC09 unconstrained MOP test instances," in *Proc. IEEE Congr. Evol. Comput.*, Trondheim, Norway, 2009, pp. 203–208.
- [40] L. Davis, *Handbook of Genetic Algorithms*. New York, NY, USA: Van Nostrand Reinhold, 1991.
- [41] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," in *Proc. Int. Conf. Parallel Prob. Solving Nat. (PPSN VIII)*, Birmingham, U.K., 2004, pp. 832–842.
- [42] Y. Wang, W. Fu, and D. P. Agrawal, "Gaussian versus uniform distribution for intrusion detection in wireless sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 2, pp. 342–355, Feb. 2013.
- [43] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach," *IEEE Trans. Evol. Comput.*, vol. 3, no. 4, pp. 257–271, Nov. 1999.
- [44] H. Li and Q. Zhang, "Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 284–302, Apr. 2009.

- [45] H. Ishibuchi, H. Masuda, Y. Tanigaki, and Y. Nojima, "Difficulties in specifying reference points to calculate the inverted generational distance for many-objective optimization problems," in *Proc. IEEE Symp. Comput. Intell. Multi Criteria Decis. Making (MCDM)*, Orlando, FL, USA, 2014, pp. 170–177.



Xiao Zhang received the B.Eng. and M.Eng. degrees from the South-Central University for Nationalities, Wuhan, China, in 2009 and 2011, respectively. He is currently pursuing the Ph.D. degree with the Department of Computer Science, City University of Hong Kong, Hong Kong.

He was a Visiting Scholar with the Department of Computer Science, Utah State University Logan, Logan, UT, USA, in 2015. His current research interests include algorithms design and analysis, wireless networks, and computational intelligence.



Yu Zhou (S'16) received the B.S. degree in electronics and information engineering and the M.S. degree in circuits and systems from Xidian University, Xi'an, China, in 2009 and 2012, respectively. He is currently pursuing the Ph.D. degree in computer science with the City University of Hong Kong, Hong Kong.

His current research interests include compressed sensing, image processing, machine learning, and computational intelligence.



Qingfu Zhang (M'01–SM'06) received the B.Sc. degree in mathematics from Shanxi University, Taiyuan, China, in 1984, and the M.Sc. degree in applied mathematics and the Ph.D. degree in information engineering from Xidian University, Xi'an, China, in 1991 and 1994, respectively.

He is a Professor with the Department of Computer Science, City University of Hong Kong, Hong Kong, and a Changjiang Visiting Chair Professor with Xidian University. He is currently leading the Metaheuristic Optimization Research Group, City University of Hong Kong. He holds two patents and has authored several research publications. His current research interests include evolutionary computation, optimization, neural networks, data analysis, and their applications.

Dr. Zhang is an Associate Editor of the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION and the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B: CYBERNETICS. He is also an Editorial Board Member of three other international journals. He was a recipient of the Unconstrained Multiobjective Optimization Algorithm Competition at the Congress of Evolutionary Computation 2009, for a multi-objective optimization algorithm developed in his group, and the 2010 IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION Outstanding Paper Award.



Victor C. S. Lee (M'92) received the Ph.D. degree in computer science from the City University of Hong Kong, Hong Kong, in 1997.

He is currently an Assistant Professor with the Department of Computer Science, City University of Hong Kong. His current research interests include data dissemination in vehicular networks, real-time databases, and performance evaluation.

Dr. Lee was the Chairman of the IEEE Hong Kong Section Computer Chapter from 2006 to 2007. He is a member of ACM and the IEEE Computer Society.



Minming Li (SM'15) received the B.Eng., M.Eng., and Ph.D. degrees from Tsinghua University, Beijing, China.

He is currently an Associate Professor with the Department of Computer Science, City University of Hong Kong, Hong Kong. His current research interests include wireless networks, algorithms design and analysis, combinatorial optimization, scheduling, key management, and computational economics.