Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 985410, 14 pages
http://dx.doi.org/10.1155/2013/985410

Research Article

A PSO-Optimized Minimum Spanning Tree-Based Topology Control Scheme for Wireless Sensor Networks

Wenzhong Guo,[1,2] Bin Zhang,[1] Guolong Chen,[1] Xiaofeng Wang,[2] and Naixue Xiong[3]

[1] *College of Mathematics and Computer Science, Fuzhou University, Fuzhou 350108, China*
[2] *College of Computer, National University of Defense Technology, Changsha 410073, China*
[3] *School of Computer Science, Colorado Technical University, Colorado Spring, CO 80907, USA*

Correspondence should be addressed to Guolong Chen; fzucgl@163.com

Received 6 January 2013; Accepted 15 March 2013

Academic Editor: Hongju Cheng

Wireless sensor networks (WSNs) are networks of autonomous nodes used for monitoring an environment. Topology control is one of the most fundamental problems in WSNs. To overcome high connectivity redundancy and low structure robustness in traditional methods, a PSO-optimized minimum spanning tree-based topology control scheme is proposed in this paper. In the proposed scheme, we transform the problem into a model of multicriteria degree constrained minimum spanning tree (mcd-MST) and design a nondominated discrete particle swarm optimization (NDPSO) to deal with this problem. To obtain a better approximation of true Pareto front, the multiobjective strategy with a fitness function based on niche and phenotype sharing function is applied in NDPSO. Furthermore, a topology control scheme based on NDPSO is proposed. Simulation results show that NDPSO can converge to the non-dominated front quite evenly, and the topology derived under the proposed topology control scheme has lower total power consumption, higher robust structure, and lower contention among nodes.

1. Introduction

A wireless sensor network (WSN) is a system of spatially distributed sensor nodes to collect important information in the target environment. WSNs have been envisioned for a wide range of applications, such as battlefield intelligence, environmental tracking, and emergency response. Each sensor node has limited computational capacity, battery supply, and communication capability [1]. Topology control and management—how to determine the transmission power of each node so as to maintain network connectivity while consuming the minimum possible power—are one of the most important issues in WSNs [2, 3]. Without proper topology control algorithms in placing a randomly connected multihop wireless sensor network may suffer from poor network utilization, high end-to-end delays, and short network lifetime. Topology control in WSNs is NP-hard [4], therefore approximate methods can be used to tackle it efficiently.

Generally, the topology control technologies can be divided into three types. One is to reduce the node redundancy in a given network that usually periodically let selected nodes enter energy-saving mode. Chen et al. [5] proposed a distributed algorithm in which each node makes its local decision on whether to sleep or not. Ding et al. [6] presented an adaptive partition scheme for node scheduling and topology control with the aim of reducing energy consumption. The second type of topology control is to use the clustering strategy such as the well-known LEACH [7]. It used rotation of the cluster head in order to evenly distribute the energy consumption. Cluster heads collected and aggregated all signals and then transmitted the fused information to the base station. Lindsay and Raghavendra [8] proposed PEGASIS which formed a chain including all nodes in the network. In PEGASIS, each node communicated only with a close neighbor and took turns transmitting to the base station. It was better than LEACH because it performed data aggregation at each chain node. The third type of topology control focuses on reducing link redundancy, and several topology control algorithms have been proposed. For example, Li et al. [9] introduced a cone-based topology control algorithm called CBTC which required only the availability of directional information. In this algorithm, each node tried to find the minimum power p to ensure that the transmitting with p could reach some node in every cone of degree α. The algorithm had been analytically shown to be able to preserve the network connectivity if $\alpha < 5\pi/6$. Rodoplu and Meng [10] proposed a relay-region and enclosure-based approach (R and M) that introduced the notion of relay region and enclosures for the purpose of power control. It could guarantee connectivity of the entire network, and the resulted topology was a minimum power topology. However, the network topology given by this algorithm had a high degree of link redundancy, which would reduce the efficiency of the upper network protocols. To solve this problem, Li and Halpern [11] proposed an improved protocol called SMECN, which had lower link maintenance costs than R and M and could achieve a significant saving in energy consumption. Based on the local minimum spanning tree (MST), Li et al. [12] proposed a network topology control algorithm called LMST which could effectively reduce the power consumption. The topology derived under this algorithm could also preserve the network connectivity, and the degree of each node in the resulted topology was bounded by 6. However, this algorithm ignored the structure robustness and communication interference, so some network performance under LMST may be weaken.

In a word, all the algorithms mentioned previously have taken different approaches to study the topology control and management in WSNs. However, the structure robustness and the radio contention are often ignored. It is a challenge to design a novel topology control scheme which can overcome the defects (i.e., high redundancy of connectivity and low robustness of structure) of traditional methods. This challenge motivates us to integrate the MST method and particle swarm optimization (PSO) by developing a PSO-optimized minimum spanning tree-based topology control scheme in wireless sensor networks to prolong the network lifetime. In our previous work [13], we have transformed the problem of topology control into a model of multicriteria degree constrained minimum spanning tree (mcd-MST) and designed a MST-based topology control scheme with NDPSO called MCMST. Compared with [13], the major contributions of this study are summarized as follows.

(1) Similar to [13], the topology control problem is also transformed into the model of mcd-MST with low power consumption, high structure robustness, and low node degree and the constraint of the node degree corresponds with the low ratio interference.

(2) We do performance analysis of multiobjective PSO (MOPSO) in detailed and use three standard test functions (ZDT1, ZDT2, and ZDT6) to compare MOPSO with another two classical multiobjective evolutionary algorithms, NSGA and SPEA.

(3) In this paper, we adopt NDPSO, which is a nondominated discrete particle swarm optimization, to solve this mcd-MST problem. It aims at obtaining a better approximation of true Pareto front by applying the multiobjective strategy with a fitness function based on niche and phenotype sharing function. Moreover, we further do some extended experiments to analyze the performance of NDPSO by varying the number of objectives, number of vertexes, the correlation, and maximum permissible degree constrained.

(4) We introduce the MCMST topology control scheme in WSNs [13] and compare it with another two topology control schemes, SMECN and LMST, in different performance factors which include average link length, average radius, average link strong, average physical degree, and average logic degree.

The remainder of this paper is organized as follows. Section 2 describes the problem. In Section 3, we present the details of the proposed NDPSO algorithm for the mcd-MST problem. Section 4 introduces the proposed MCMST scheme based on NDPSO. In Section 5, we analyze the proposed NDPSO algorithm under a variety of scenarios. And the comparison of our proposed MCMST scheme and other existing schemes is carried out in the simulation. Finally, we make conclusions and discuss the future work in Section 6.

2. Problem Description

2.1. mc-MST. MST problem is to find a least-cost spanning tree in an edge-weighted graph, and multicriteria minimum spanning tree (mc-MST) problem is one of the extended formulations of the MST problem. In mc-MST, a vector of weights is defined for each edge, and the problem is to find all Pareto optimal spanning trees. Consider a connected and undirected graph $G = (V, E)$, where $V = \{v_1, v_2, \ldots, v_n\}$ is a finite set of vertices representing terminals or telecommunication stations, and so forth, and $E = \{e_{1,2}, e_{1,3}, \ldots, e_{i,j}, \ldots e_{n-1,n}\}$ which is defined as follows:

$$e_{i,j} = \begin{cases} 1, & \text{if } v_i, v_j \text{ have edge} \\ 0, & \text{otherwise,} \end{cases} \qquad (1)$$
$$(i = 1, 2, \ldots, n-1;\ j = i+1, i+2, \ldots, n).$$

Each edge has m positive real numbers, representing m attributes defined on it and denoted with $w_{i,j} = \{w_{i,j}^1, w_{i,j}^2, \ldots, w_{i,j}^m\}$. In practice, $w_{i,j}^k$ $(k = 1, 2, \ldots, m)$ may represent the distance, cost, and so on.

Let $x = x_{1,2}, x_{1,3}, \ldots, x_{i,j}, \ldots, x_{n-1,n}$ be defined as follows:

$$x_{i,j} = \begin{cases} 1, & \text{if } e_{i,j} = 1 \text{ and is selected} \\ 0, & \text{otherwise,} \end{cases} \qquad (2)$$
$$(i = 1, 2, \ldots, n-1;\ j = i+1, i+2, \ldots, n).$$

Then a spanning tree of graph G can be expressed as the vector x. Let X be the set of all such vectors corresponding to spanning trees in graph G, and the mc-MST problem can be formulated as follows:

$$\begin{aligned} \min \quad & f_1(x) = \sum w_{i,j}^1 x_{i,j}, \\ \min \quad & f_2(x) = \sum w_{i,j}^2 x_{i,j}, \\ & \cdots \\ \min \quad & f_m(x) = \sum w_{i,j}^m x_{i,j}, \end{aligned}$$

$$\left(i = 1, 2, \ldots, n-1;\; j = i+1, i+2, \ldots, n\right),$$
$$\text{s.t.} \quad x \in X, \tag{3}$$

where $f_i(x)$ is the ith objective to be minimized for the problem

$$1 \le \sum x_{i,j} \le d; \quad (i = 1, 2, \ldots, n), \tag{4}$$

where d denotes the maximum permissible degree.

If the degree constraint condition of formula (4) is further considered for the formula (3), the mc-MST problem will be transformed into a mcd-MST problem, thus the mcd-MST is a typical mc-MST.

Compared with the traditional MST problem, the mcd-MST problem has more than one objective. Because these multiple objectives exist and usually conflict with each other, we cannot determine which edge has the least weight and span one to form a spanning tree like the process of vertex growth or edge growth. If we transform the multiple objectives into a single objective according to some criteria, the problem can be easily solved by using any MST algorithm, but only one single solution can be obtained. Actually, this transformation is not an easy work both for the decision maker and the system analyzer in practice.

2.2. Model Formulation. A wireless sensor network can be described as a directed graph $G(V, E)$ in a two-dimensional plane, where V is the set of nodes and E is the set of edges. The relative neighbor graph $G_i(V_i, E_i)$ of node i is denoted as the subgraph located in the disk centered at node i with a radius $r_{\max}$, where $|V| = n$ and $|E| = m$. In this paper, a minimum spanning tree with low power consumption, high structure robustness, and low node degree in $G_i(V_i, E_i)$ is constructed. The constraint of node degree corresponds with the low ratio interference. The low power consumption and high structure robustness can be described as follows.

Objective 1 (low power consumption). One has

$$\text{Min}\left\{y_1(i) = \sum p_k \mid i \in V, k \in V_i\right\}, \tag{5}$$

where $y_1(i) \in (0, (n_i - 1)p_{\max})$ and p_k stands for the power consumption of node k.

Objective 2 (high structure robustness). One has

$$\text{Max}\left\{y_2(i) = \sum R_{jk} \mid (j,k) \in T_i, T_i \subset G_i(V_i, E_i)\right\}, \tag{6}$$

where R_{jk} stands for the robustness value between node j and node k.

Therefore, the mcd-MST problem with low power consumption and high structure robustness can be described as searching for sensor nodes with abundant surplus energy to communicate with each other, and avoiding too much interference among them. We denote an m-dimensional vector to describe the states of links. If an edge belongs to T_i, then $x_j = 1$, otherwise $x_j = 0$. Additionally, the original model must be consistent with the features of MST, so the mathematical model can be described as follows:

$$\begin{aligned} \min z_1(i) &= W_1 X = \sum_{j=1}^{m} w_{1j} x_j, \\ \min z_2(i) &= W_2 X = \sum_{j=1}^{m} w_{2j} x_j, \end{aligned} \tag{7}$$

where W_1, W_2 correspond to the weight of the vectors implying low power consumption and high structure robustness, respectively. At the same time, the objectives must satisfy the node degree constraint in order to form a degreeconstrained tree structure.

3. Proposed Algorithm

The MST problem has been well studied, and many efficient polynomial-time algorithms [14] have been developed by Dijkstra, Kruskal, Prim. Although the basic MST problem is in polynomial time, the addition of one or more constraints often transforms it into a multiobjective problem (MOP), and so approximate methods must be used if one is to tackle it efficiently. In [15], Zhou and Gen described a Prufer-encoded genetic algorithm (GA), which used Srinivas and Deb's Nondominated Sorting method and a Prufer based encoding [16, 17]. And an algorithm for enumerating all Pareto optimal spanning trees was put forward to evaluate their proposed GA. However, Knowles [18] pointed out that the proposed enumeration algorithm was not correct. In our previous work [19], we put forward a nongenerational multiobjective GA (MOGA) to deal with the mc-MST and presented an improved enumeration algorithm to evaluate our proposed MOGA. In [19], the improved enumeration algorithm was proved to be able to find out all true Pareto optimal solutions, so it can be used to replace the algorithm of Zhou and Gen for evaluating the quality of mc-MST solutions generated by an approximate method such as the GA in [15] or [19].

Definition 1 (dominance [20]). A vector $U = (u_1, u_2, \ldots, u_n)$ is said to dominate $V = (v_1, v_2, \ldots, v_n)$ if and only if U is partially less than V, that is, for all $i \in \{1, 2, \ldots, n\}, u_i \le v_i \wedge \exists\, i \in \{1, 2, \ldots, n\}, u_i \prec v_i$.

Definition 2 (pareto optimal [20]). A solution $X_u \in U$ is said to be Pareto optimal if and only if there is no $x_v \in V$ for which $V = f(X_v) = (v_1, v_2, \ldots, v_n)$ dominates $U = f(X_u) = (u_1, u_2, \ldots, u_n)$.

PSO is a relatively recent heuristic optimization technique developed by Kennedy and Eberhart [21]. Ease of implementation, high quality of solutions, computational efficiency, and speed of convergence are strengths of the PSO. In the past several years, PSO has been successfully applied in many researches and applications such as [22–27]. WSN issues such as node deployment, localization, energy-aware clustering, data aggregation, and topology control are often formulated as optimization problems, and PSO has been applied to address these WSN issues. Kulkarni and Venayagamoorthy [28] introduce PSO and discuss its suitability for WSN applications. They also present a brief survey of how PSO is tailored to address these issues.

3.1. Basic PSO. PSO is a population based search problem where individuals, referred to as particles, are grouped into a swarm. In PSO, each particle is defined as a potential solution to a problem in a D-dimensional space, with the ith particle represented as $X_i = (X_{i1}, X_{i2}, \ldots, X_{iD})$. The particle adjusts its position in search space according to its own experience and that of neighboring particles. Each particle also maintains a memory (pbest) of its previous best position represented as $p_i = (p_{i1}, p_{i2}, \ldots, p_{iD})$ and a velocity along each dimension represented as $V_i = (V_{i1}, V_{i2}, \ldots, V_{iD})$. In each generation, the *pbest* vector of the particle with the best fitness in the local neighborhood designated p_{gd}.

In each generation of the early versions of PSO, the particles were manipulated according to the following equation:

$$\begin{gathered} v_{id} = w * v_{id} + c_1 r_1 \left(p_{id} - x_{id}\right) + c_2 r_2 \left(p_{gd} - x_{id}\right), \\ x_{id} = x_{id} + v_{id}, \end{gathered} \tag{8}$$

where d is the number of dimensions (variables), w is the inertia weight, c_1 and c_2 are two positive constants, called acceleration constants, and r_1 and r_2 are two random numbers within the range $[0, 1]$. A constant $V_{\max}$ is often used to limit the velocities of the particles and improve the resolutions of the search space.

3.2. NDPSO. As (8) mentioned in the previous section, it is obvious that the basic PSO cannot be directly used to generate a discrete combinatorial solution of the MST problem. Since the PSO algorithm was proposed by Kennedy and Eberhart in 1995, attempts have been made to apply the PSO algorithm to discrete combinatorial problems lately [29–35]. In this section, inspired by the [32] and our previous work [29], NDPSO is applied to deal with mc-MST problem. In the proposed algorithm, the phenotype sharing function of the objective space is applied in the definition of fitness function to obtain a better approximation of true Pareto front.

3.2.1. Representation of Particles. Here we also adopt the method of [15, 19] to build the representation scheme of particles.

Procedure: Encoding

Step 1. Let vertex j be the smallest labeled leaf vertex in a labeled tree T.
Step 2. Set k to be the first digit in the permutation if vertex k is incident to vertex j.
Step 3. Remove vertex j and the edge from j to k; we have a tree with $n - 1$ vertices.
Step 4. Repeat the previously steps until one edge is left, and produce the Prufer number or permutation with $n - 2$ digits in order.

Procedure: Decoding

Step 1. Let P be the original Prufer number, and let P be the set of all vertices not included in P.
Step 2. Let j be the vertex with the smallest label in P, and let k be the leftmost digit of P. Add the edge from j to k into the tree. Remove j from P and k from P. If k does not occur anywhere in the remainder of P, put it into P.
Step 3. Repeat the process until no digits left in P.
Step 4. If no digits remain in P, there are exactly two vertices, r and s, in P. Add edge from r to s into the tree and form a tree with n-1edges.

3.2.2. Discrete Procedure of PSO. Here, the position of the ith particle at iteration t can be updated as follows:

$$X_i^t = c_2 \oplus F_3 \left(c_1 \oplus F_2 \left(w \oplus F_1 \left(X_i^{t-1}\right), P_i^{t-1}\right), G_i^{t-1}\right). \tag{9}$$

The update equation consists of three components as follows, where r_1, r_2, and r_3 are uniform random numbers generated between 0 and 1.

(a) λ_i^t represents the velocity of the particle,

$$\lambda_i^t = w \oplus F_1 \left(X_i^{t-1}\right) = \begin{cases} F_1 \left(X_i^{t-1}\right), & r_1 < w, \\ X_i^{t-1}, & \text{else}, \end{cases} \tag{10}$$

where F_1 indicates the insert (mutation) operator with the probability of w.

(b) δ_i^t is the “cognition” part of the particle for the private thinking of the particle itself,

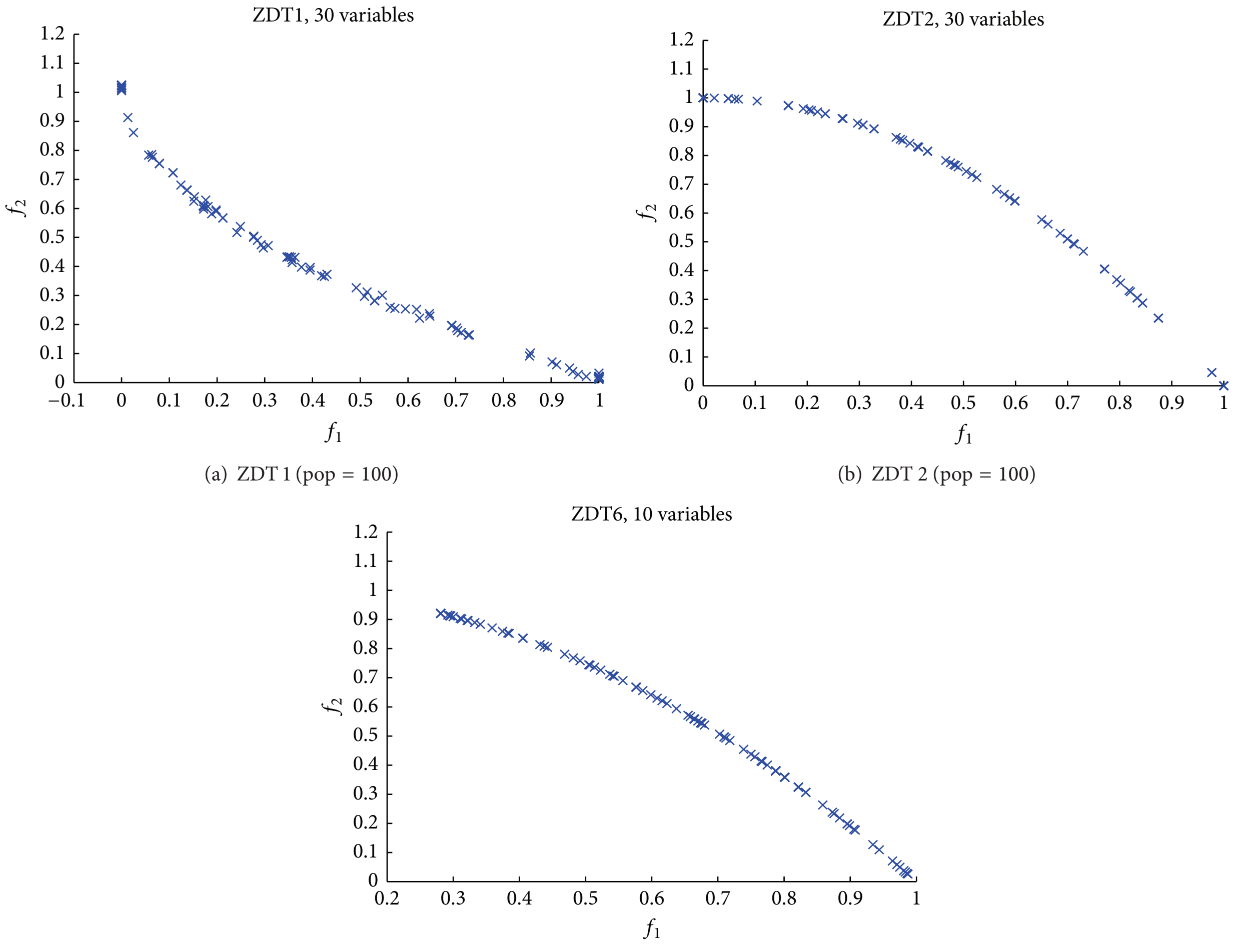
$$\delta_i^t = c_1 \oplus F_2 \left(\lambda_i^t, P_i^{t-1}\right) = \begin{cases} F_2 \left(\lambda_i^t, P_i^{t-1}\right), & r_2 < c_1, \\ \lambda_i^t, & \text{else}, \end{cases} \tag{11}$$

where F_2 represents the crossover operator with the probability of c_1.

(c) X_i^t is the “social” part of the particle representing the collaboration among particles,

$$X_i^t = c_2 \oplus F_3 \left(\delta_i^t, G_i^{t-1}\right) = \begin{cases} F_3 \left(\delta_i^t, G_i^{t-1}\right), & r_3 < c_2, \\ \delta_i^t, & \text{else}, \end{cases} \tag{12}$$

where F_3 represents the crossover operator with the probability of c_2.



(a) ZDT 1 (pop = 100)

(b) ZDT 2 (pop = 100)

(c) ZDT 6 (pop = 100)

Figure 1: The results of MOPSO in ZDT1, ZDT2, and ZDT6.

3.2.3. Fitness Value Function

Definition 3. Target distance fd_{ij}: fd_{ij} is the distance between the two particles i and j. Supposed that the distance has m dimensions which are noted as $f_1 d_{ij}, f_2 d_{ij}, \ldots, f_m d_{ij}$, respectively, and

$$fd_{ij} = f_1 d_{ij} + f_2 d_{ij} + \cdots + f_m d_{ij} = \left| f_1 \left(x^i\right) - f_1 \left(x^j\right) \right| + \left| f_2 \left(x^i\right) - f_2 \left(x^j\right) \right| + \cdots + \left| f_m \left(x^i\right) - f_m \left(x^j\right) \right|, \tag{13}$$

where $i \neq j$.

Definition 4. Dominance measure $D(i)$: $D(i)$ expresses the state of domination the ith particle with respect to the current population, and

$$D(i) = \sum_{j=1}^{p} nd(i, j), \tag{14}$$

where $nd(i, j)$ is one if particle j dominate particle i, and zero otherwise.

Definition 5. Sharing function $sh(fd_{ij})$:

$$sh\left(fd_{ij}\right) = \begin{cases} 1, & \text{if } fd_{ij} \leq \sigma_s, \\ 0, & \text{otherwise,} \end{cases} \tag{15}$$

where σ_s is a sharing parameter.

Definition 6. The neighbor density measure $N(i)$: $N(i)$ associated with particle i is defined as

$$N(i) = \sum_{j=1}^{p} sh\left(fd_{ij}\right). \tag{16}$$

Definition 7. The fitness of A given particle $F(i)$: $F(i)$ is then defined as

$$F(i) = (1 + D(i)) \times (1 + N(i)). \tag{17}$$

Compared with the single-object PSO, during the search process of multiobjective PSO (MOPSO), particles often have more than one personal best and global best value. We save these information into an external archive [36]. A proper mechanism of choosing leader particles can help to find more

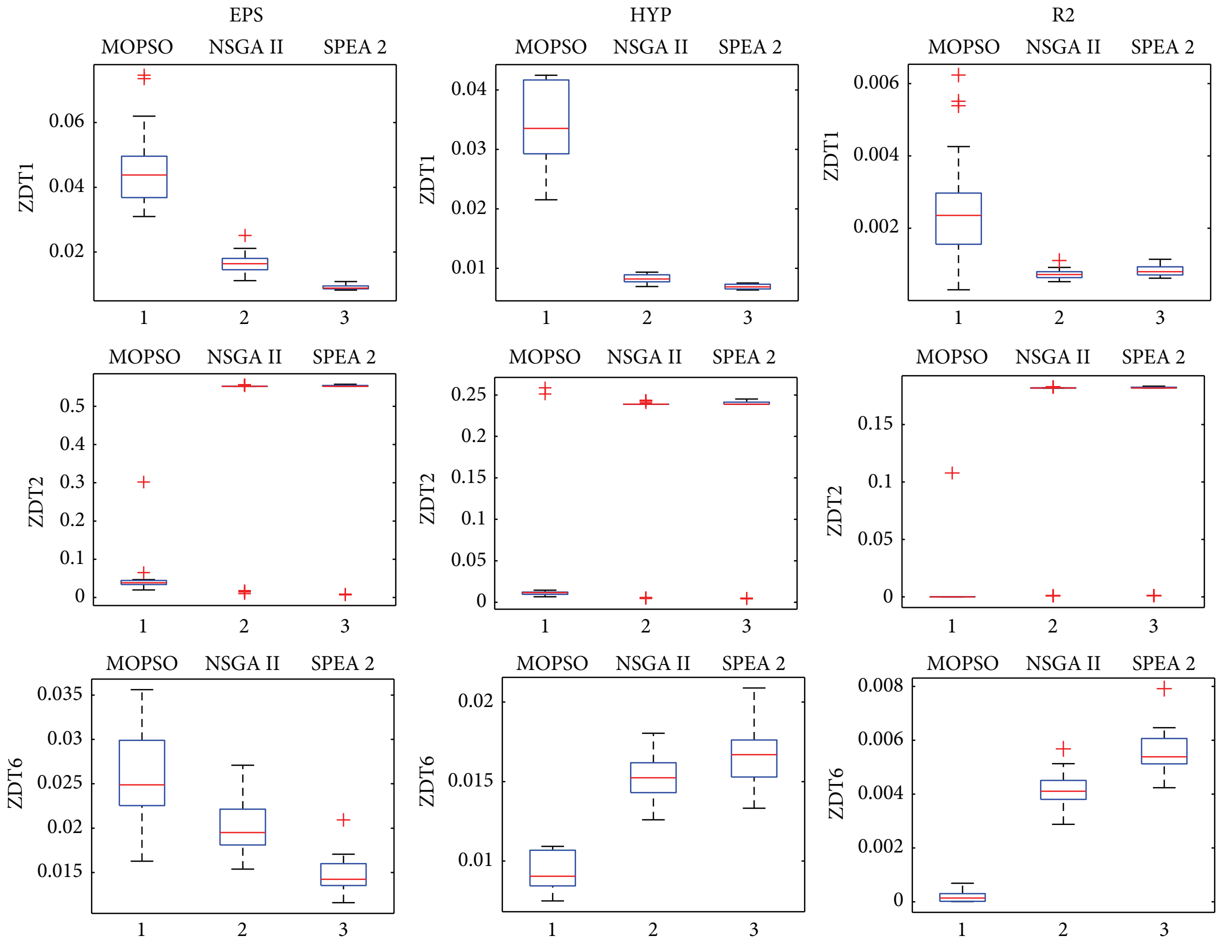
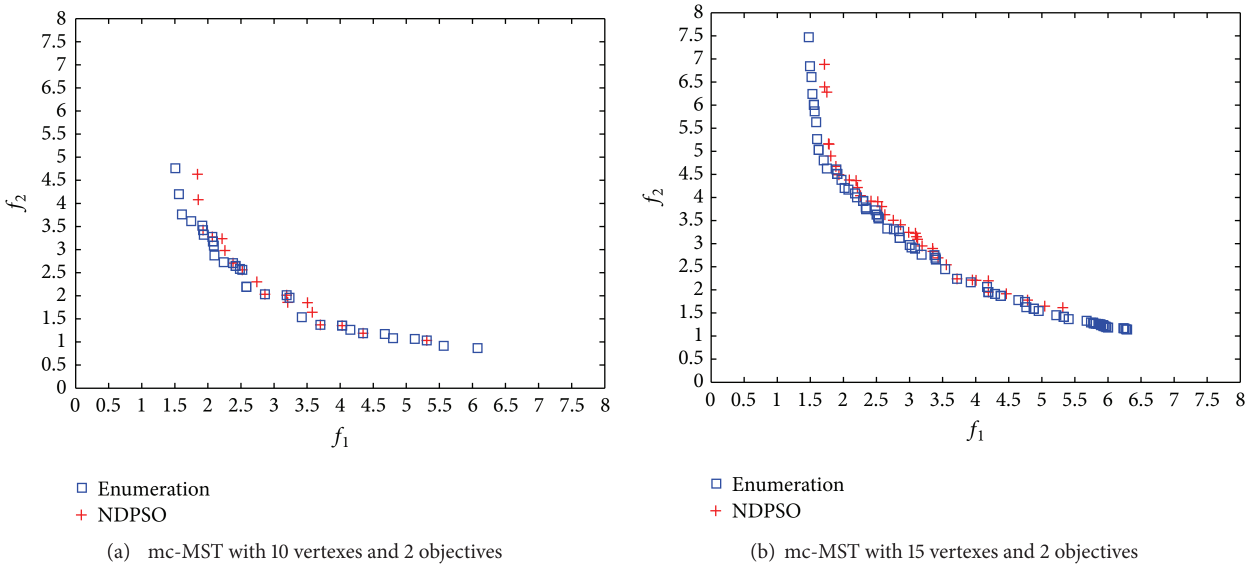


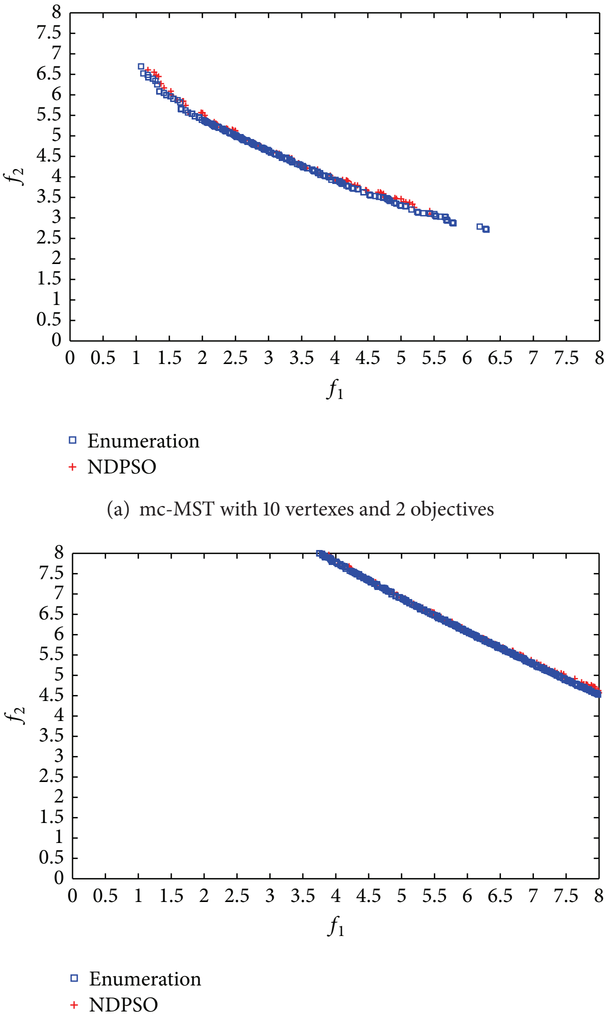
Figure 2: Boxplots of MOPSO, NSGA II, and SPEA 2 algorithm performance indicator values.



(a) mc-MST with 10 vertexes and 2 objectives

(b) mc-MST with 15 vertexes and 2 objectives

Figure 3: The effect of NDPSO with correlation = 0 on the different problem.



(a) mc-MST with 10 vertexes and 2 objectives

(b) mc-MST with 15 vertexes and 2 objectives

Figure 4: The effect of NDPSO with correlation = −0.7 on the different problem.

Pareto solutions in a shorter time. So it is very important to decide how to choose the proper leader particles to direct the movement of particles. In order to avoid the external archive growing too big, we adopt ε-dominance [37] to reduce the external archive.

3.2.4. Algorithm Overview. The details of NDPSO algorithm can be described as follows.

NDPSO Algorithm for mcd-MST

```
Step 1. Initialize swarm.
Step 2. Modify degree if not satisfied.
Step 3. Calculate the fitness value.
Step 4. Initialize leaders in an external archive.
Step 5. Filter (leaders).
Step 6. Iteration = 0.
Step 7. For each particle: mutation; select leader;
    SelfCross; SocialCross; modify degree;
    update Position; update pbest.
Step 8. Update leaders in the external archive.
Step 9. Filter (leaders).
Step 10. Iteration++.
Step 11. If iteration<iterationMax, go to Step 7.
Step 12. Output results.
```

4. Topology Control Scheme

As mentioned in the previous sections, we constructed the multiobjective minimum spanning tree model according to characteristics of problem and designed a NDPSO algorithm to deal with the model. Sensor nodes are usually presented in the intersection of many node adjacency coverage graphs, because the local topology is needed to adjust overall after deployed. Then the node selects the one whose requirement for coverage is maximum from a number of the spanning trees. Inspired by the idea of local minimum spanning tree structure and topology adjustment strategy of [12], a topology control scheme based on NDPSO called MCMST is proposed in this section, and the details of the MCMST are described as follows.

Topology Control Scheme Based on (MCMST)

Step 1. Each wireless sensor node u collects the local information of its own reachable neighborhood graph $G_u(V_u, E_u)$ through broadcasting a HELLO message using its maximal transmission power periodically.

Step 2. According to the information collected in the previously step and the performance evaluation function, each node u computes the minimum spanning tree with degree constraints in the graph $G_u(V_u, E_u)$ using the NDPSO algorithm independently in the previously section.

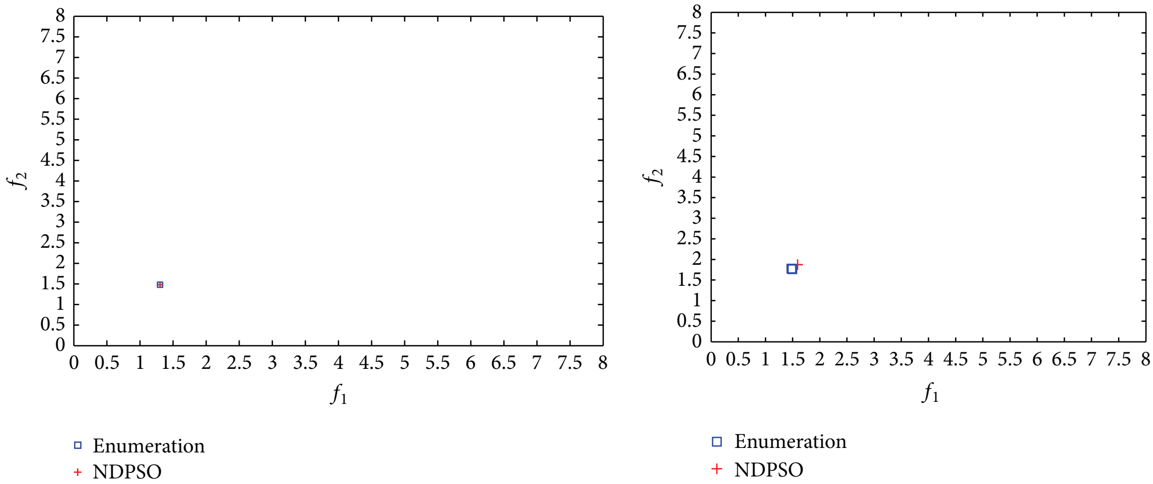
Step 3. Each node u has to adjust its transmission power according to its local minimum spanning tree $T_u(V(T), E(T))$, where $V(T) = V_u$, $E(T) \subset E_u$. At this time, node $u$requires a power lever that can reach the farthest one-hop neighbor in T_u.

5. Simulation Results

In this section, we first conduct simulations to compare the performance of PSO under our multiobjective strategy (MOPSO) with two classic multiobjective evolutionary algorithms in three standard test functions. Furthermore, the solution sets generated by NDPSO are also compared with solution sets from the enumerating algorithm [19] which is proved to find all Pareto optimal spanning trees. Finally, we conduct extensive simulations to compare the performance of our proposed MCMST scheme with other existing schemes.

5.1. Performance Analysis of MOPSO

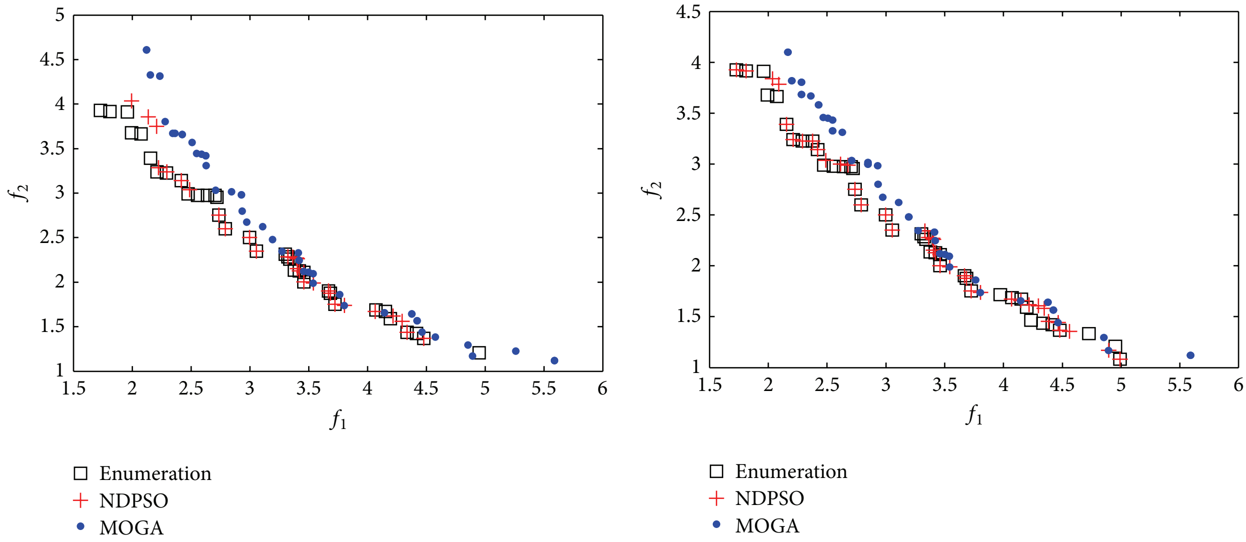
5.1.1. Experimental Setup. We use three standard test functions (ZDT1, ZDT2, and ZDT6) [38] and compare MOPSO with two classic multiobjective evolutionary algorithms



(a) mc-MST with 10 vertexes and 2 objectives

(b) mc-MST with 15 vertexes and 2 objectives

FIGURE 5: The effect of NDPSO with correlation = 0.95 on the different problem.



(a) mcd-MST with $d = 3$

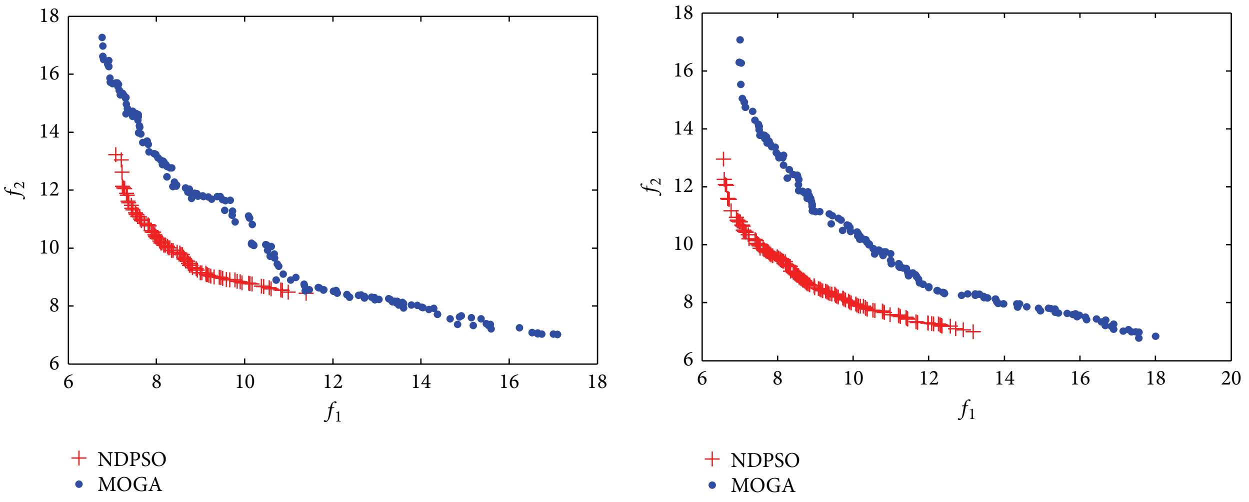
(b) mcd-MST with $d = 4$

FIGURE 6: Comparison on 15 vertices with $d = 3$ and $d = 4$.

(NSGA [39] and SPEA [40]). The three test functions represent some typical function types of noninferior front in multiobjective optimization problem, respectively. ZDT1 is a convex function, ZDT2 is a nonconvex function, and ZDT6 is noncontiguous function. In order to avoid algorithm randomness of a singular run, we independently run each algorithm for 20 times and compare the results of 20 experiments based on the statistical methods of hypothesis testing [41]. The experimental parameters are set as $c_1 = c_2 = 2.0$, $\sigma_s^1 = 0.01$, and $\sigma_s^2 = 1$. In addition, pop $= 100$ denotes the scale of population of all algorithms, and the maximum iteration times is 500 in each run.

To evaluate the performance of those multiobjective algorithms, Zitzler et al. developed several indicators [42]. Here, we adopt three quantitative indicators (unary additive EPS indicator $I_{\varepsilon+}^1$, HYP indicator I_H^-, and unary R2 indicator I_{R2}^1) in this paper.

5.1.2. Results and Analysis. As to the three test functions ZDT1, ZDT2, and ZDT6, in most cases, the proposed algorithm in this paper (MOPSO) can converge to the Pareto front within 100 iterations and have a small computational cost. Figure 1 shows the distribution situation of Pareto solution at a certain run. From the figure, we can infer that for each function, algorithm can obtain a well distribution of Pareto front. The results of detailed performance parameters are shown in Figure 2, and the Kruskal Wallis test results of statistical analysis are shown in Table 1.



(a) mcd-MST with $d = 3$

(b) mcd-MST with $d = 4$

FIGURE 7: Comparison on 50 vertices with $d = 3$ and $d = 4$.

From Figure 2 and Table 1, we can see that as to ZDT2 issue, MOPSO proposed in this paper is better than other multiobjective evolutionary algorithms in three indicators under the circumstance of same dominance rank distribution. So MOPSO algorithm is good at solving nonconvex Pareto front function. For ZDT6, MOPSO algorithm is superior to HSGA II and SPEA 2 on both R2 and HYP indicator, while MOPSO is little inferior to them on EPS indicator. According to [42], it also shows that if the results from multiple Pareto compliant indicators are different, it is hard to know which algorithm is better. It is obvious that MOPSO algorithm can obtain a better result in the front nonuniform ZDT6. As to ZDT1, although MOPSO is inferior to NSGA II and SPEA 2 on three parameters, the value of overall difference is small. Considering that it is unable to find manifest difference from other evolutionary algorithm on dominance rank distribution of solution and MOPSO can converge to Pareto front within few number of iteration times, MOPSO algorithm can obtain a satisfying solution in convex Pareto font optimization function. According to the results of three test function, we can see that MOPSO algorithm proposed in this paper has strong global search ability, and it can converge to the Pareto front by less computing cost. Furthermore, its distribution of solution is quite good. What we have mentioned previously demonstrates that MOPSO is worth being studied in the field of multiobjective optimization problem.

5.2. Performance Analysis of NDPSO

5.2.1. Experimental Setup. Let us consider a complete graph $G = (V, E)$, where $n = |V|$ and each edge has m weights, the problem's goal is to find all Pareto optimal solutions of G. The coordinate of vertexes in G and the weights of each edge are generated randomly. Suppose each edge in the graph has two weights, we propose generations for the following types:

(a) random: random (uncorrelated) real number weighted graphs;

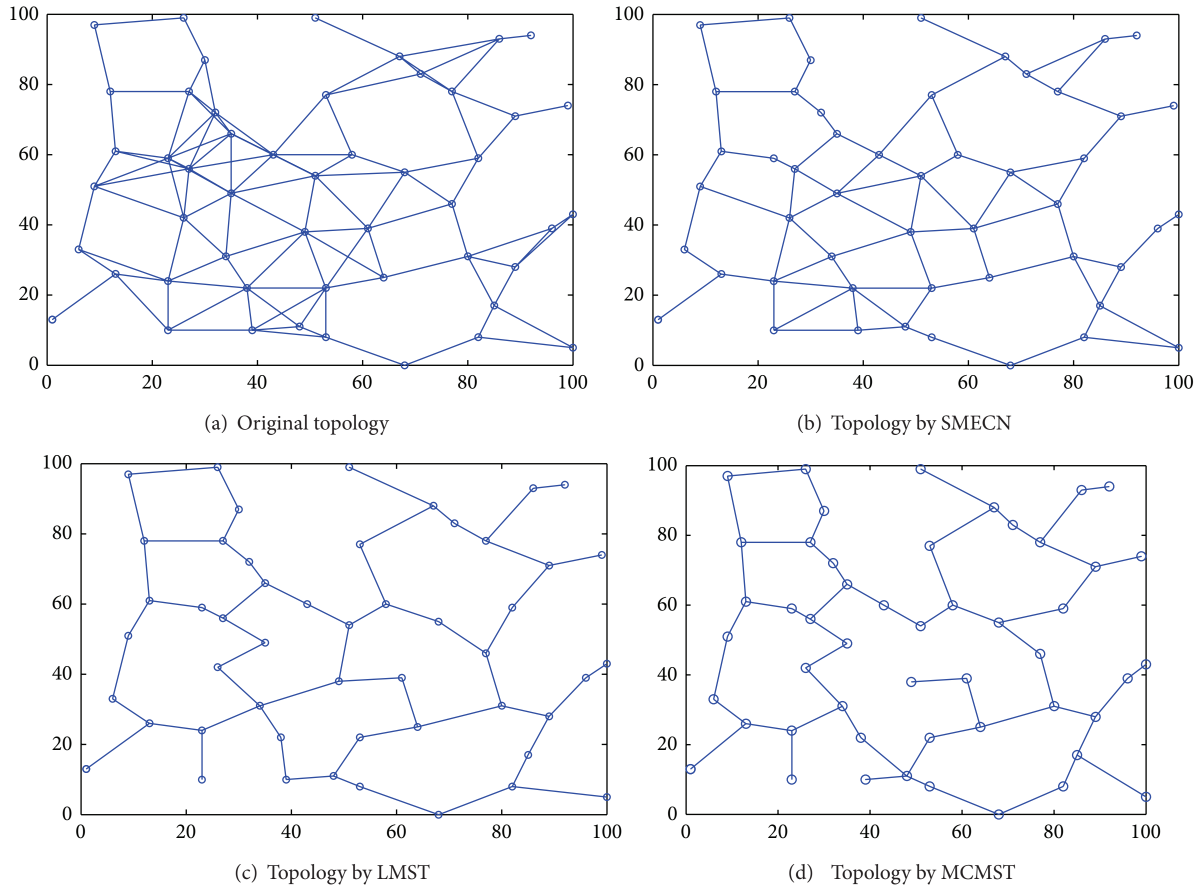
(b) correlated: random correlated real number weighted graphs;

(c) anticorrelated: random anticorrelated real number weighted graphs.

In [18], Knowles proposed an algorithm to generate correlated (and anticorrelated) graphs. Knowles declared that all subsequent weights are either positively or negatively correlated with respect to the first component, and the values of the weights are within a same range. All the weights are of course not negative, but we find that the algorithm will generate negative value when $\alpha \leq 0$, and we presented an improved algorithm in [19] to generate correlated (and anticorrelated) graphs. The results of the improved algorithm can be seen in [19].

5.2.2. Results and Analysis. In [19], we proposed an improved enumeration algorithm which is proved to be able to find all the optimal solutions. So it can be a tool for evaluating the performances of other heuristic algorithms instead of the enumeration algorithm of Zhou and Gen [15]. In the first simulation, to evaluate the NDPSO algorithm, the solution sets generated by it are compared with solution sets from the improved enumeration [19], and the results are displayed from Figures 3, 4, and 5. Here, we consider 2-objective optimal problem. And the parameters are set as follows: the number of vertices is 10 and 15, respectively, population size is 50, and maximum generation is 1000.

From the results, we can draw a conclusion that the NDPSO algorithm has similar performance with increased size compared with the MOGA in [19]. However, the population size and the maximum generation in MOGA are set as 400 and 20000, respectively. Thus, the NDPSO is superior to MOGA in space and time complexity, and it is really effective to deal with the mc-MST problem.



(a) Original topology

(b) Topology by SMECN

(c) Topology by LMST

(d) Topology by MCMST

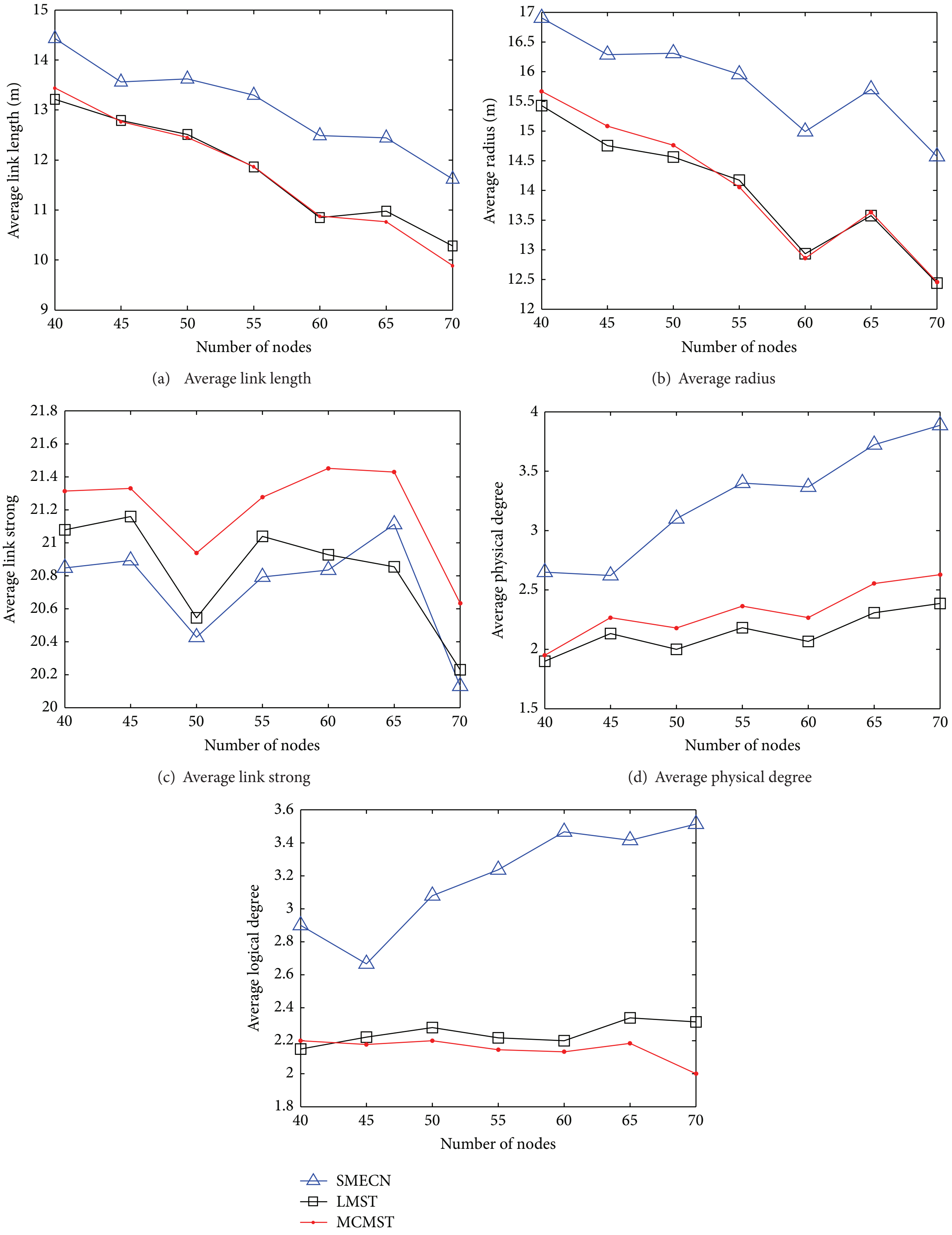
Figure 8: Network topologies derived under different topology control scheme.

In real network environmental, to satisfy the requirement of lifetime objective in wireless sensor networks and aim at the defect that high redundancy of connectivity or low robust of structure in traditional schemes, the model of topology control in WSNs can be transformed into a problem of mcd-MST. So in the second simulation, the problem's goal is to find mcd-MST solutions of G. The coordinate of vertexes in G and the weights of each edge are generated randomly. For simplicity, however, we suppose each edge in the graph has only two weights, which can be easily extended. We display the results from Figures 6 and 7. Figure 6 shows that the NDPSO algorithm converges to the real Pareto optimal solutions better than the MOGA in [19] under the same parameters setting no matter $d = 3$ or $d = 4$. From Figure 7, as a direct representation, we can find that the proposed NDPSO algorithm dominate the MOGA algorithm.

5.3. Performance Analysis of MCMST

5.3.1. Experimental Setup. Our simulation is done for a network of n nodes that are placed uniformly at random in a rectangular region of 100 by 100 meters. We assume that all nodes have the same maximum emission radius R, and the Euclidean distance between node i and node j is $d(i, j)$. The transmission power between node i and node j is defined as $C_{ij} = k * d(i, j)^2$, $k = 0.005$, the robustness of an edge between node i and node j is defined as $R_{ij} = e(i, j)/(e_i^2 + e_j^2)^{1/2}$, the maximal transmission range is $R_{\max} = 20$ m for all the nodes, and the initial energy of each node is equivalent to a random value between 20 and 40 Joule.

5.3.2. Result and Analysis. In the first simulation, 50 nodes are uniformly distributed in the region. Figure 8(a) shows the initial topology generated using the maximum transmission power, and the topology generated by different topology control schemes is shown in Figures 8(b), 8(c), and 8(d), respectively. As shown in Figure 8, SMECN, LMST and MCMST all dramatically reduce the average node degree while maintaining network connectivity. Moreover, LMST, and MCMST outperform SMECN in the sense that few edges formed in the final network topology, because they take the demand for low-power consumed into account and try to reduce nodes' transmission powers. However, the network topology generated by MCMST is slightly different form the one derived by LMST because LMST has not considered the robustness of topology structures, and at the same time MCMST has taken the edge coverage and node degree constrained into consideration.



(a) Average link length

(b) Average radius

(c) Average link strong

(d) Average physical degree

(e) Average logic degree

Figure 9: Performance comparisons among different topology control scheme.

Table 1: The Kruskal Wallis test results of EPS, HYP, and R2 indicator.

		MOPSO	NSGA II	SPEA 2
ZDT 1				
$I_{\varepsilon+}^{1}$	MOPSO	—	1	1
	NSGA II	$4.16519E-17$	—	1
	SPEA 2	$3.48666E-34$	$4.16519E-17$	—
I_{H}^{-}	MOPSO	—	1	1
	NSGA II	$2.53616E-16$	—	1
	SPEA 2	$4.00793E-32$	$4.88893E-15$	—
I_{R2}^{1}	MOPSO	—	1	0.99996
	NSGA II	$7.6029E-08$	—	0.0551455
	SPEA 2	$3.99655E-05$	0.944854	—
ZDT 2				
$I_{\varepsilon+}^{1}$	MOPSO	—	$2.20462E-06$	$2.16472E-05$
	NSGA II	0.999998	—	0.72699
	SPEA 2	0.999978	0.27301	—
I_{H}^{-}	MOPSO	—	0.000234484	0.00112884
	NSGA II	0.999766	—	0.689058
	SPEA 2	0.998871	0.310942	—
I_{R2}^{1}	MOPSO	—	$1.18542E-14$	$7.16123E-14$
	NSGA II	1	—	0.658011
	SPEA 2	1	0.341989	—
ZDT 6				
$I_{\varepsilon+}^{1}$	MOPSO	—	1	1
	NSGA II	$3.27923E-07$	—	1
	SPEA 2	$1.28526E-20$	$3.94864E-11$	—
I_{H}^{-}	MOPSO	—	$3.10337E-14$	$1.21161E-20$
	NSGA II	1	—	0.000456755
	SPEA 2	1	0.999543	—
I_{R2}^{1}	MOPSO	—	$8.72107E-20$	$4.22686E-29$
	NSGA II	1	—	$1.43107E-07$
	SPEA 2	1	1	—

This table shows each pair of the value of P which is composed of algorithm O_R (row) and algorithm O_C (column), and the corresponding alternative hypothesis is that algorithm O_R is superior to algorithm O_C in indicator. The value of significance level is 0.05.

In the second simulation, we vary the number of nodes in the region from 40 to 70. The average radius and the average link length for the topologies generated using the SMECN, LMST, and MCMST with link removal are shown, respectively, in Figures 9(a) and 9(b). MCMST and LMST outperform SMECN in the both two cases, and MCMST scheme has similar performance with LMST scheme. However, with the size of nodes increasing, MCMST has better performances than LMST in terms of the average link length. The results imply that our proposed scheme can provide a better spatial reuse and is power-efficient. As shown in Figure 9(c), the average link strong for the topologies generated using our scheme outperforms than that of SMECN and LMST because the structure robustness of network is considered in our proposed MCMST scheme. The average physical degree and the average logic degree for the topologies using SMECN, LMST, and MCMST are shown, respectively, in Figures 9(d) and 9(e). From Figure 9(d), we can know that MCMST, and LMST outperform SMECN and MCMST is little inferior to LMST. As shown in Figure 9(e), the average logical degree for the topologies generated using the MCMST has better performance than LMST and SMECN with the size of nodes increasing. It implies that our proposed scheme can get a better network topology with low radio interference.

In a word, MCMST scheme proposed in this paper has made a tradeoff among the energy consumption, radio interference, and structure robustness, and the performance of the resulting network topology has been greatly improved.

6. Conclusions

Due to the demand for the optimization of the network lifetime, this paper adopts the idea of a local minimum spanning tree, transforms the model into a multicriteria degreeconstrained minimum spanning tree, and proposes a NDPSO algorithm. The phenotype sharing function of the objective space is applied in the definition of fitness function to obtain a better approximation of true Pareto front, and the global convergence of the algorithm is proved using the theorem of Markov chain. Then, a topology control scheme based on NDPSO is put forward. Simulation results show that the obtained topology in this paper has low overall power consumption, high structural robustness, and controllable internode communication interference characteristics and can effectively prolong the lifetime of WSNs. And our algorithm can obtain a better approximation of true Pareto front.

Future work will cover the following problem of the proposed MCMST scheme: (1) consider the reliability in wireless sensor networks effectively; (2) how to reduce the complexity and improve the convergence efficiency of our NDPSO.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant no. 61103175 and no. 61103194, the Key Project of Chinese Ministry of Education under Grant no. 212086, the Technology Innovation Platform Project of Fujian Province under Grant no. 2009J1007, the Key Project Development Foundation of Education Committee of Fujian province under Grant no. JA11011, and the Fujian Province High School Science Fund for Distinguished Young Scholars under Grant no. JA12016.

References

[1] T. Laukkarinen, J. Suhonen, and M. Hannikainen, "A survey of wireless sensor network abstraction for application development," *International Journal of Distributed Sensor Networks*, vol. 2012, Article ID 740268, 12 pages, 2012.

[2] N. Ababneh, "Performance evaluation of a topology control algorithm for wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2010, Article ID 671385, 16 pages, 2010.

[3] L. Lobello and E. Toscano, "An adaptive approach to topology management in large and dense real-time wireless sensor networks," *IEEE Transactions on Industrial Informatics*, vol. 5, no. 3, pp. 314–324, 2009.

[4] S. Zarifzadeh, N. Yazdani, and A. Nayyeri, "Energy-efficient topology control in wireless ad hoc networks with selfish nodes," *Computer Networks*, vol. 56, no. 2, pp. 902–914, 2012.

[5] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: an energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks," *Wireless Networks*, vol. 8, no. 5, pp. 481–494, 2002.

[6] Y. Ding, C. Wang, and L. Xiao, "An adaptive partitioning scheme for sleep scheduling and topology control in wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 20, no. 9, pp. 1352–1365, 2009.

[7] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proceedings of the 33rd Annual Hawaii International Conference on System Siences (HICSS-33 '00)*, pp. 1–10, January 2000.

[8] S. Lindsay and C. Raghavendra, "PEGASIS: power-efficient gathering in sensor information systems," in *Proceedings of IEEE Aerospace Conference*, pp. 1125–1130, 2002.

[9] L. Li, J. Y. Halpern, P. Bahl, Y. M. Wang, and R. Wattenhofer, "Analysis of a cone-based distributed topology control algorithm for wireless multi-hop networks," in *Proceedings of the 20th Annual ACM Symposium on Principles of Distributed Computing*, pp. 264–273, August 2001.

[10] V. Rodoplu and T. H. Meng, "Minimum energy mobile wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, pp. 1333–1344, 1999.

[11] L. Li and J. Y. Halpern, "A minimum-energy path-preserving topology-control algorithm," *IEEE Transactions on Wireless Communications*, vol. 3, no. 3, pp. 910–921, 2004.

[12] N. Li, J. C. Hou, and L. Sha, "Design and analysis of an mst-based topology control algorithm," in *Proceedings of the 22nd Annual Joint Conference of the IEEE Computer and Communication Scocieties*, pp. 1702–1712, 2003.

[13] W. Z. Guo, J. H. Park, L. T. Yang, A. V. Vasilakos, N. X. Xiong, and G. L. Chen, "Design and analysis of a MST-based topology control scheme with PSO for wireless sensor networks," in *Proceedings of IEEE Asia-Pacific Services Computing Conference*, pp. 360–367, December 2011.

[14] Y. H. Chen, "Polynomial time approximation schemes for the constrained minimum spanning tree problem," *Journal of Applied Mathematics*, vol. 2012, Article ID 394721, 8 pages, 2012.

[15] G. Zhou and M. Gen, "Genetic algorithm approach on multi-criteria minimum spanning tree problem," *European Journal of Operational Research*, vol. 114, no. 1, pp. 141–152, 1999.

[16] J. Gottlieb, B. A. Julstrom, G. R. Raidl, and F. Rothlauf, "Prufer numbers: a poor representation of spanning trees for evolutionary search," in *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 343–350, 2001.

[17] N. Srinivas and K. Deb, "Multi-objective optimization using nondominated sorting in genetic algorithms," *Evolutional Computation*, vol. 2, no. 3, pp. 221–248, 1994.

[18] J. D. Knowles, *Local-search and hybrid evolutionary algorithms for Pareto optimization [thesis]*, University of Reading, West Berkshire, UK, 2002.

[19] G. Chen, S. Chen, W. Guo, and H. Chen, "The multi-criteria minimum spanning tree problem based genetic algorithm," *Information Sciences*, vol. 177, no. 22, pp. 5050–5063, 2007.

[20] D. A. van Veldhuizen and G. B. Lamont, "Multiobjective evolutionary algorithm test suites," in *Proceedings of the 14th ACM Symposium on Applied Computing (SAC '99)*, pp. 351–357, March 1999.

[21] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948, December 1995.

[22] X. Li and X. Yao, "Cooperatively coevolving particle swarms for large scale optimization," *IEEE Transactions on Evolutionary Computation*, vol. 16, no. 2, pp. 210–224, 2012.

[23] S. S. Jiang, Z. W. Zhao, S. Mou, Z. S. Wu, and Y. Luo, "Linear decision fusion under the control of constrained PSO for WSNs," *International Journal of Distributed Sensor Networks*, vol. 2012, Article ID 871596, 11 pages, 2012.

[24] Y. Morsly, N. Aouf, M. S. Djouadi, and M. Richardson, "Particle swarm optimization inspired probability algorithm for optimal camera network placement," *IEEE Sensors Journal*, vol. 12, no. 5, pp. 1402–1412, 2012.

[25] Y. Shen, G. Wang, and C. Tao, "Particle swarm optimization with novel processing strategy and its application," *International Journal of Computational Intelligence Systems*, vol. 4, no. 1, pp. 100–111, 2011.

[26] D. Caputo, F. Grimaccia, M. Mussetta, and R. E. Zich, "Genetical swarm optimization of multihop routes in wireless sensor networks," *Applied Computational Intelligence and Soft Computing*, vol. 2010, Article ID 523943, 14 pages, 2010.

[27] K. Zielinski, P. Weitkemper, R. Laur, and K. D. Kammeyer, "Optimization of power allocation for interference cancellation with particle swarm optimization," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 1, pp. 128–150, 2009.

[28] R. V. Kulkarni and G. K. Venayagamoorthy, "Particle swarm optimization in wireless-sensor networks: a brief survey," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 41, no. 2, pp. 262–267, 2011.

[29] W. Z. Guo, H. L. Gao, G. L. Chen, and L. Yu, "Particle swarm optimization for the degree-constrained MST problem in WSN topology control," in *Proceedings of International Conference on Machine Learning and Cybernetics*, vol. 3, pp. 1793–1798, July 2009.

[30] W. N. Chen, J. Zhang, H. S. H. Chung, W. L. Zhong, W. G. Wu, and Y. H. Shi, "A novel set-based particle swarm optimization method for discrete optimization problems," *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 2, pp. 278–300, 2010.

[31] G. Lapizco-Encinas, C. Kingsford, and J. Reggia, "Particle swarm optimization for multimodal combinatorial problems and its application to protein design," in *Proceedings of IEEE Congress on Evolutionary Computation*, pp. 1–8, July 2010.

[32] Q. K. Pan, M. F. Tasgetiren, and Y. C. Liang, "A discrete particle swarm optimization algorithm for the permutation flowshop sequecing problem with Makespan criteria," in *Proceedings of the 26th SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence*, pp. 19–31, 2006.

[33] S. H. Ling, F. Jiang, H. T. Nguyen, and K. Y. Chan, "Hybrid fuzzy logic-based particle swarm optimization for flow shop scheduling problem," *International Journal of Computational Intelligence and Applications*, vol. 10, no. 3, pp. 335–356, 2011.

[34] R. M. Aliguliyev, "Clustering techniques and discrete particle swarm optimization algorithm for multi-document summarization," *Computational Intelligence*, vol. 26, no. 4, pp. 420–448, 2010.

[35] W. Z. Guo, N. X. Xiong, A. V. Vasilakos, G. L. Chen, and C. L. Yu, "Distributed k-connected fault-tolerant topology control algorithms with PSO in future autonomic sensor systems," *International Journal of Sensor Networks*, vol. 12, no. 1, pp. 53–62, 2012.

[36] R. Balling, "The maximin fitness function: multi-objective city and regional planning," in *Proceedings of the 2nd International Conference on Evolutionary Multi-Criterion Optimization*, vol. 2632, pp. 1–15, 2003.

[37] F. Neumann and M. Laumanns, "Speeding up approximation algorithms for NP-hard spanning forest problems by multi-objective optimization," in *Electronic Colloquium on Computational Complexity*, Report no. 29, 2005.

[38] E. Zitzler, *Evolutionary Algorithms for Multi-Objective Optimization: Methods and Applications*, Swiss Federal Institute of Technology, Zurich, Switzerland, 1999.

[39] R. E. Steuer, *Multiple Criteria Optimization: Theory, Computation, and Application*, John & Wiley Sons, New York, NY, USA, 1986.

[40] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: improving the strength pareto evolutionary algorithm," in *Proceedings of Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems (EUROGEN '01)*, September 2001.

[41] W. J. Conover, *Practical Nonparametric Statistics*, John Wiley & Sons, New York, NY, USA, 3rd edition, 1999.

[42] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. da Fonseca, "Performance assessment of multiobjective optimizers: an analysis and review," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 2, pp. 117–132, 2003.

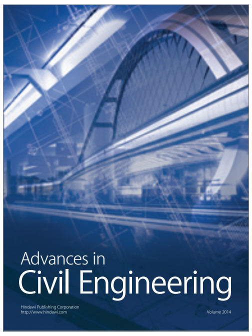
Journal of
Engineering

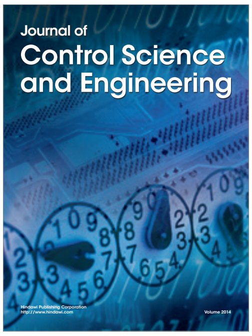
The Scientific World Journal

International Journal of
Rotating Machinery

Journal of
Sensors

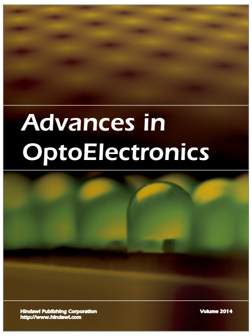
International Journal of
Distributed Sensor Networks


Advances in
Civil Engineering


Journal of
Control Science and Engineering

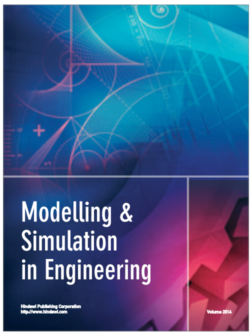
Journal of
Robotics

Journal of
Electrical and Computer Engineering

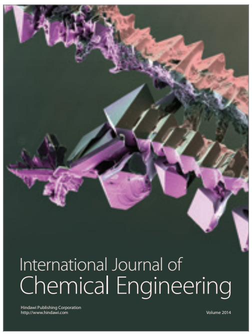

Advances in
OptoElectronics

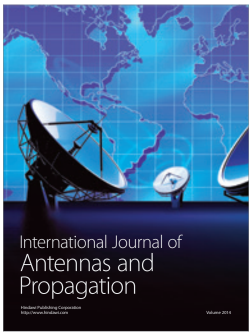
VLSI Design

International Journal of
Navigation and Observation

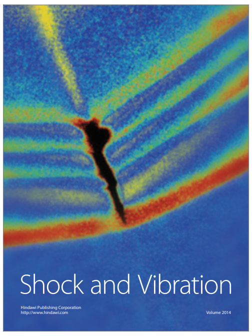

Modelling & Simulation in Engineering

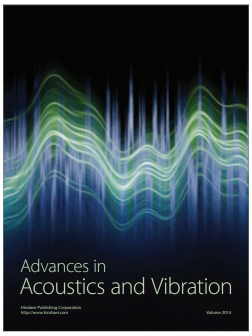
International Journal of
Aerospace Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and Propagation

Active and Passive Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration