Research Article

Multiobjective Optimization for Topology and Coverage Control in Wireless Sensor Networks

Seyed Mahdi Jameii, 1 Karim Faez, 2 and Mehdi Dehghan 3

1Department of Computer Engineering and Information Technology, Islamic Azad University, Qazvin Branch, Qazvin, Iran
2Department of Electrical Engineering, Amirkabir University of Technology, Tehran, Iran
3Department of Computer Engineering, Amirkabir University of Technology, Tehran, Iran

Correspondence should be addressed to Seyed Mahdi Jameii; jameii@qiau.ac.ir

Received 10 June 2014; Accepted 8 September 2014

Academic Editor: Ilker Demirkol

Copyright © 2015 Seyed Mahdi Jameii et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Coverage, connectivity, and network lifetime are important issues in wireless sensor networks (WSNs). Balancing the energy consumption in the network, reducing the transmission range of nodes, and density control of active nodes are approaches to extend the network lifetime. However, transmission range reduction and smaller number of active nodes can affect the network topology and may cause the network to be disconnected. So, there exist conflicts among lifetime, coverage, and connectivity. In this paper, these conflicting issues are considered and an evolutionary multiobjective optimization approach based on nondominated sorting genetic algorithm-II (NSGA-II) is proposed to optimize them. Simulation results demonstrate that the proposed algorithm can improve the network lifetime and coverage while maintaining the network connectivity.

1. Introduction

Wireless sensor networks (WSNs) are kind of wireless networks including a group of sensors which can be deployed rapidly and cheaply, and thereby they can be used for different purposes such as environment monitoring and object tracking [1]. WSNs play an important role in the improvement of the quality of people’s lives [2, 3]. Energy is an extremely critical resource in this kind of networks, thus prolonging the lifetime of network and making energy-efficient protocols are major design challenges in WSNs [4, 5]. Area coverage, point coverage [6], and barrier coverage [7] are three types of coverage in WSNs. Usually, in the case of random deployment, the number of deployed sensors is more than what is required [8]. It is not necessary to keep all sensor nodes in active mode simultaneously. We need a scheduling scheme which remains some sensors in active state to provide sensing services and puts other sensors to sleep state to conserve energy [9]. Sensor scheduling decreases the number of active sensors, avoids the channel collision, reduces the energy consumption in the network, and prolongs the network lifetime [10]. In addition, for ensuring that each sensor is able to communicate with the sink node, network connectivity should also be considered in WSNs [11, 12]. Balanced energy consumption in the network and efficient transmission ranges adjustment of the sensors are other approaches for energy conservation and prolonging the network lifetime [13]. Maximizing covered area, balanced energy consumption, minimizing the transmission range of nodes, minimizing the number of active nodes, and maintaining the connectivity of active nodes are critical issues in WSNs which are usually conflicting. The present paper aims to consider these conflicting issues simultaneously and proposes an evolutionary multiobjective optimization approach based on nondominated sorting genetic algorithm-II (NSGA-II) for optimizing them.

The remainder of this paper is organized as follows. Section 2 reviews some of the previous related works. Section 3 describes the related definitions and concepts. Section 4 gives network model and assumptions made in this paper. The proposed solution is presented in Section 5.
In Section 6, the simulation results and discussions are provided and Section 7 concludes the paper.

2. Related Works

Network coverage, connectivity, and lifetime are important issues in WSNs. A lot of works have been done in recent years by the researchers for addressing these issues. Open geographic density control (OGDC) is one of the popular coverage algorithms [14]. In this approach, by controlling the density of the active sensors, the energy consumption can be reduced. In this algorithm, deterministic placement of the nodes is considered, but this kind of placement is only possible for the small-scale networks. The Coverage Configuration Protocol (CCP) is another coverage control protocol proposed in [15]. In this protocol, every node maintains the information of its neighbors and periodically broadcasts a hello message which contains current location of node and its status. The trade-off between the energy consumption and network coverage is not considered in this protocol. For example, in the case that only one intersecting point of a node is not covered by its neighbor, the protocol will activate this node. So, this protocol is not energy efficient in the case of the large-scale networks.

The authors of [16] proposed approximation algorithms for both maximum lifetime and minimum cost coverage problems, but connectivity is not their concern. In [17], for modeling the area coverage in WSNs, a degree-constrained minimum-weight extension of the CDS (connected dominating set) problem called DCDS is provided and an approach based on learning automata called LAEEC is proposed for finding a near-optimal solution. The proposed solution balances the network load on the active sensors and improves the network coverage and lifetime. The authors of [18] used improved NSGA-II to optimize the network coverage and nodes utilization simultaneously. In [19] for solving the conflict among coverage and equilibrium of energy consumption in WSN, a multiobjective optimization strategy is proposed. The proposed strategy is based on a new evolutionary algorithm called multiobjective free search algorithm (MOFS). The problem of network coverage and connectivity is addressed in [20] and an efficient solution is proposed for maintaining coverage and preserving the connectivity of the network while minimizing the count of the active sensor nodes.

The sensor deployment problem for WSNs is studied in [21, 22]. However, their objective is finding optimal locations for placing the sensors and maximizing the coverage rather than selecting covering sensor nodes from a randomly deployed sensor network. Additionally, connectivity has never been a requirement in these papers. The concept of multiobjective evolutionary algorithm based on decomposition (MOEA/D) is used in [23] for relocating the mobile nodes in a WSN with the goal of providing maximum sensing coverage area and minimizing the energy consumption. The goal of research presented in [24] is maintaining sensing coverage by activating a small number of sensors and consuming a small amount of energy in a WSN. The authors of [12] formulated maximum disjoint sets for maintaining coverage and connectivity problem in WSN and proposed a heuristic algorithm for solving it. A flexible algorithm presented in [25] uses an evolutionary computational approach to optimize connectivity and energy cost of WSNs in addition to coverage. The problem of positioning sensor nodes in WSN considering coverage with minimum energy consumption is studied in [26] and a multiobjective optimization approach is proposed for optimizing the energy consumption and coverage. The research presented in [27] used the characteristics of Voronoi diagram and direction-adjustable directional sensors and proposed a distributed greedy algorithm, which can improve the effective field coverage of directional sensor networks. The sensor field is divided into Voronoi cells and the sensor working direction is evaluated based on Voronoi vertices. Considering the coverage contribution of convex polygonal cell of sensors and the coverage overlap of direction select between neighbor sensors, the working direction is adjusted and controlled, so as to improve the overall sensing field coverage ratio in the sensor network environment without global information. The authors of [28] proposed a multiobjective optimization algorithm to efficiently schedule the nodes of a wireless sensor network and to optimize energy consumption, lifetime, and coverage.

The authors of [29] formulated the sensor node deployment task as a constrained multiobjective optimization problem with the aims of maximizing the area of coverage and minimizing the number of deployed sensor nodes. They used a decomposition approach for converting the problem of approximation of the Pareto fronts into a number of single-objective optimization problems. The authors of [30] proposed a multiobjective hybrid optimization algorithm to solve the dynamic coverage and connectivity problem (DCCP) in flat WSNs. In [31], an algorithm based on Euclidean distance called CWSC is proposed to k-cover the sensing region while minimizing the number of working sensors. The goal of the research presented in [32] is addressing the network lifetime maximization problem with connectivity and coverage constraints by scheduling the activity of sensors.

Although all of the above-mentioned researches have achieved the coverage requirement, some of them have not considered the network lifetime [12, 20, 22, 27] and the network connectivity [16, 21–23]. Some works such as [14, 15, 17–20, 24] require that the transmission range is at least twice the size of the sensing range; otherwise, the connectivity may not be guaranteed.

All of the above studies except [25] have not considered the transmission range adjustment for sensor nodes to maximize the network lifetime. These works assumed the same transmission range for all sensors. Among the above-mentioned works, only [19] considered the balanced energy consumption in the network.

In this paper, we propose a multiobjective optimization approach to optimize the area coverage, network lifetime, network connectivity, transmission range adjustment of nodes, and balanced energy consumption simultaneously in WSN without any sensing range or transmission range restriction.
3. Related Definitions and Concepts

3.1. Multiobjective Optimization Problems. A multiobjective optimization problem (MOP) is stated as follows [33]:

Minimize or maximize \( F(\mathbf{x}) = (F_1(\mathbf{x}), \ldots, F_m(\mathbf{x})) \) \( (1) \)

subjected to \( \mathbf{x} \in \Omega \), where \( \Omega \subset \mathbb{R}^n \) is the decision space, \( F: \Omega \rightarrow \mathbb{R}^m \) consists of \( m \) real-valued objective functions, and \( \mathbb{R}^m \) is objective space. In some cases, a MOP has various equality and inequality constraints as follows:

\[
g_i(\mathbf{x}) \leq 0, \quad i = 1, \ldots, p,
\]
\[
h_j(\mathbf{x}) = 0, \quad j = p + 1, \ldots, q.
\]

In the case that optimization problems involve constraints, the search space \( X \) can be defined as follows:

\[
\Omega = \{ \mathbf{x} \mid \mathbf{x} \in X, \mathbf{x} \text{ meets constrains } C_1, \ldots, C_q \}, \quad (3)
\]

where \( X \) is the base search space and \( \mathbf{x} \) is a solution of the feasible region \( \Omega \) that satisfies all \( q \) constraints, that is, \( C_1, \ldots, C_q \). The solutions that satisfy all constraints are called feasible and those that do not are infeasible. In a multiobjective optimization problem where there are \( m \) objectives \( F_1, \ldots, F_m \) for minimization, we have the following definitions:

**Definition 1.** Assume that \( x \) and \( y \) are two decision variables; \( x \) is said to dominate \( y \), denoted by \( x < y \), if and only if \( F_i(x) \leq F_i(y) \) for every \( i \in \{1, 2, \ldots, m\} \) and \( F_i(x) < F_i(y) \) for at least one index \( j \in \{1, 2, \ldots, m\} \).

**Definition 2.** \( x^* \in \Omega \) is said to be Pareto-optimal (or non-dominated) if there is no other \( x \in \Omega \) so that \( x \) dominates \( x^* \). The set of all Pareto-optimal solutions in the decision space is called the Pareto-optimal set (PS). The image of the PS in the objective space is called the Pareto-optimal front (PF).

Figure 1(a) illustrates a PF with \( m = 2 \). The Pareto-optimal solutions in the PF (marked as asterisk) provide better values for objective functions than any other solution in the objective space. The ideal solution (marked as rectangle) provides the minimum objective values and it is often considered unreachable. The remaining solutions (marked as solid circles) are all dominated by at least one solution of the PF. In Figure 1(b), the bold curve indicates the PF of a constrained MOP. The solid circles are the feasible solutions in the feasible region and the remaining solutions outside the feasible region (marked as triangle) are considered infeasible.

3.2. NSGA-II Algorithm. There are several well-known multiobjective genetic algorithms (MOGAs). NSGA-II [34] is one of the most popular algorithms. By introducing the fast nondominated sorting approach, binary crowding tournament selection, and elitism, the Pareto-optimal front can be searched effectively. NSGA-II has a diversity-preserving mechanism which assures convergence towards the Pareto-optimal front without losing solution diversity. The standard NSGA-II algorithm works as follows [34].

Initially, a random population \( D_t \) \((t = 0)\) is created and is sorted into different nondominance levels. For each solution a fitness value equal to its nondominance level is assigned.

For creating an offspring population \( E_t \) of size \( N \), some operators such as binary tournament selection, mutation, and recombination are used. Then the following steps will be repeated until the number of generations reaches the maximum number of generations.

**Step 1.** Combine parent and offspring populations to create \( P_t = D_t \cup E_t \). Perform a fast nondominated sorting in \( P_t \) according to the algorithm shown in Algorithm 1 and identify different fronts \( F_i, i = 1, 2, \ldots \).

**Step 2.** Set a new population \( D_{t+1} = \emptyset \). Set counter \( i = 1 \).

While \( |D_{t+1}| + |F_i| < N \), do \( D_{t+1} = D_{t+1} \cup F_i \) and \( i = i + 1 \).
Fast non-dominated sorting algorithm [34]

**Input:** population \( P \)

**Output:** The non-dominated fronts \( (F_{r_1}, F_{r_2}, \ldots) \)

For each \( p \in P \)
  For each \( q \in P \)
    If \( (p < q) \) then
      \( S_p = S_p \cup \{q\} \)
    else if \( (q < p) \) then
      \( n_p = n_p + 1 \)
      if \( n_p = 0 \) then
        \( F_i = F_i \cup \{p\} \)
    \( i = 1 \)
  while \( F_i \neq \emptyset \)
  \( H = \emptyset \)
  For each \( p \in F_i \)
    For each \( q \in S_p \)
      \( n_q = n_q - 1 \)
      if \( n_q = 0 \) then
        \( H = H \cup \{q\} \)
    \( i = i + 1 \)
  \( F_i = H \)

**Algorithm 1:** The pseudocode of the fast nondominated sorting algorithm.

**Step 3.** Select \( (N - |D_{t+1}|) \) most widely spread solutions from \( F_{r_i} \) using the binary crowding tournament selection operator and insert them into \( D_{t+1} \).

**Step 4.** Create an offspring population \( E_{t+1} \) from \( D_{t+1} \) by using the binary crowding tournament selection, crossover, and mutation operators. Set \( t = t + 1 \).

The process of nondominated sorting and filling the population \( D_{t+1} \) steps can be performed together. In Step 3, the crowding-sorting of the solutions in front \( F_{r_i} \), which is the last front that could not be completely accommodated, is performed by using a crowded-distance metric.

The crowding comparison operator compares two solutions and returns the winner of the tournament. The winner is selected based on two attributes: the nondominance ranking \( r_i \) and the local crowding distance \( d_i \) in the population. This crowding distance attribute of a solution \( i \) is a measure of the search space around \( i \), which is not occupied by any other solution in the population. Based on \( r_i \) and \( d_i \), the binary crowding tournament selection operator works as follows.

A solution \( i \) wins a tournament over another solution \( j \) if any of the following conditions is true.

1. \( r_i < r_j \) (this assures that the selected solution lies on a better nondominated front).
2. \( r_i < r_j \) and \( d_i < d_j \) (this is applied when both solutions lie on the same front and the above condition cannot be applied; in this case, the solution residing in a less crowded area with a larger \( d_i \) wins).

#### 3.3. Dynamic Crowding Distance (DCD) Algorithm

Horizontal diversity of Pareto-front is very important in MOEAs and it is realized by removing excess individuals in the nondominated set. For removing the excess individuals, NSGA-II uses crowding distance (CD) measure. The individuals with lower value of CD have a high priority to be removed in the removal process. The value of CD is calculated using

\[
CD_i = \frac{1}{N_{\text{obj}}} \sum_{z=1}^{N_{\text{obj}}} |F_{i+1}^z - F_{i-1}^z|, 
\]

where \( N_{\text{obj}} \) is the number of objectives, \( F_{i+1}^z \) is the \( z \)th objective of the \( i+1 \)th individual, and \( F_{i-1}^z \) is the \( z \)th objective of the \( i-1 \)th individual after sorting the population according to CD value. The major drawback of crowding distance is the lack of uniform diversity in obtained nondominated solutions. To overcome this problem, DCD method is suggested in [35]. The DCD of individuals is calculated based on

\[
CD_{D_i} = \frac{CD_i}{\log(1/\text{Var}_i)},
\]

where \( CD_i \) is calculated by (4) and \( \text{Var}_i \) is calculated based on

\[
\text{Var}_i = \frac{1}{N_{\text{obj}}} \sum_{z=1}^{N_{\text{obj}}} |F_{i+1}^z - F_{i-1}^z|^2 - CD_i, 
\]

\( \text{Var}_i \) is the variance of CDs of individuals which are neighbors of the \( i \)th individual.

#### 4. Network Model and Assumptions

The sensor network consists of a static resource-rich sink node and \( N \) stationary resource-constrained sensor nodes deployed randomly with uniform distribution over a finite, two-dimensional region \( A \). The transmission range of these nodes is also limited such that the entire network cannot be traversed in a single hop. Sensor nodes are assumed to be heterogeneous and each sensor node can adjust its radio transmission range \( R_i \) to vary from 0 to the maximum
transmission range $R$. We assumed eight predefined transmission range levels for each sensor. We also assumed that all sensor nodes have the same sensing range $R_s$ and each node knows its position and also the positions of its neighbors within its transmission range.

The sensing range and the transmission ranges of sensor nodes are assumed as perfect disks which means that a node can sense the area points which are within a distance of $R_s$ from the node.

Region $A$ is divided into some uniform consecutive grids to make the coverage problem more computationally manageable. We have assumed a cluster-based architecture for WSN and each cluster is modeled as a mesh of grids, one cluster head, and NC member of nodes with different transmission range. This model is depicted in Figure 2.

We assume a continuous monitoring application and all sensor nodes send their sensed data to its cluster head periodically. All cluster heads transmit their data to a base station periodically. The cluster formation and routing are not considered in this paper. We considered area coverage in each cluster and modeled the monitoring status of a grid which centered at $(x', y')$ as follows:

$$\text{grid}(x', y') = \begin{cases} 1 & \text{if } \exists j \in \{1, \ldots, N\}, d(x_j, y_j, (x', y')) \leq R_s, \\ 0 & \text{otherwise,} \end{cases}$$

(7)

where 1 indicates that this grid is covered and 0 otherwise.

For ensuring that there are no collisions during data communication, SMAC [36] is used as medium access control protocol. Also, we adopt the path loss communication model as in [37]. All the nodes in the network have the same initial energy. Equation (8) calculates the residual energy of sensor $i$, at time $t$ [13]:

$$E_i(t) = E_i(t-1) - \left[ E_{tx}^i(t) + E_{rx}^i(t) + E_s \right],$$

(8)

where $E_{tx}^i(t)$ and $E_{rx}^i(t)$ are transmission and reception energy consumed by sensor $i$, respectively, and are calculated based on

$$E_{tx}^i(t) = \left[ (r_i(t) + k) \times p \times \text{amp} \right],$$

(9)

$$E_{rx}^i(t) = r_i(t) \times E_{ct}.$$ 

$E_s$ is the total energy consumed by sensor $i$ for sensing and processing with a fixed sensing range $R_s$ in one time slot, $r_i(t)$ is the amount of bits that $i$ receives and relays towards the cluster head at time $t$, $k$ is the amount of data generated by $i$ which should be forwarded to the cluster head, amp is the energy consumption of the power amplifier, and $E_{ct}$ is the required energy for receiving a bit of data.

5. Multiobjective Optimization Approach

This section describes the proposed multiobjective optimization algorithm for transmission range adjustment, scheduling, coverage and connectivity control called TASCC. TASCC is proposed based on NSGA-II to optimize coverage, number of active nodes, and balanced energy consumption while maintaining the connectivity in WSNs. As mentioned earlier, we assumed a cluster-based architecture and the network lifetime is scheduled into rounds.

Since the cluster heads consume more energy, for balancing the residual energy of nodes, new cluster heads are chosen in each round and then, the new clusters will be formed. Each cluster head is responsible for calculating the transmission range and determining the scheduling status of its member nodes by executing the TASCC algorithm. The final results calculated by the cluster heads are sent to their member nodes and each member node will adjust its transmission range and scheduling status.

5.1. Multiobjective Formulation of TASCC. The TASCC is formulated as a multiobjective optimization problem as follows.

Decision Variable 1. Scheduling status of sensor $j$ is defined as follows:

$$\text{Status}_j = \begin{cases} 1, & \text{if sensor } j \text{ is active} \\ 0, & \text{otherwise.} \end{cases}$$

(10)

Decision Variable 2. Transmission range of sensor $j$ ($R_j^i$), $0 \leq R_j^i \leq R_s$(max).

Objectives. Maximizing the coverage rate, minimizing the percentage of active sensor nodes, and minimizing the unbalanced energy consumption.

Constraint. Network connectivity: the network connectivity means that there is at least one path between each sensor node and the sink node.
**Objective 1.** Minimizing the rate of the uncovered area.

At first, the rate of area coverage which is the percentage of the covered grids over the total grids is formulated as follows:

\[
F_1 = \text{Coverage Rate} = \frac{\sum_{x' \in X} \sum_{y' \in Y} \text{grid}(x', y')}{G},
\]

(11)

where \((X, Y)\) is the center of grids coordinates in the cluster, \(G\) is the total number of grids in the cluster, and grid\((x', y')\) is calculated according to (7).

To translate this objective to a minimization objective, the rate of the uncovered area is defined as follows:

\[
F_1 = 1 - F_1'.
\]

(12)

**Objective 2.** Minimizing the unbalanced energy consumption which is evaluated as follows:

\[
F_2 = E_{\text{unbalanced}} = \frac{\text{Max}(E_i) - \text{Min}(E_i)}{\text{Max}(E_i)},
\]

for each node \(i\) in the cluster,

(13)

where \(E_i\) is the residual energy of sensor \(i\) and is calculated according to (8). The higher value of \(E_{\text{unbalanced}}\) means the more unbalanced energy consumption.

**Objective 3.** Minimizing the number of active sensors is evaluated according to

\[
F_3 = \sum_{j \in NC} \text{Status}_j,
\]

(14)

where NC is the total number of sensors in the cluster.

The optimization problem is defined as follows:

\[
F = \text{Argmin}\{F_1, F_2, F_3\}.
\]

(15)

**5.2. Genetic Representation of the Solutions.** Each solution \(X\) consists of NC items. Each item \(j\) represents an individual sensor node \(j\) and has two parts: status \(j\) and \(R_j^t\), which represent the scheduling status of sensor node \(j\) and the transmission range of sensor \(j\), respectively. So, we need four bits for each item: the first bit is used to describe the scheduling status of the sensor node and the three remaining bits describe one of the eight predefined values of transmission range. This genetic representation of solution is depicted in Figure 3.

For a sensor node, the possibilities of four-bit encoding scheme are represented in Table 1.

**5.3. TASCC Algorithm.** At first, the algorithm generates an initial population of size \(M\). In each solution in the initial population, \(G\) sensor nodes are activated uniformly and randomly \((G\) is the number of grids in the cluster). To every activated sensor \(j\) in the initial solution, a transmission range level \(R_j^t\) is assigned as (16) such that it reaches the sensor \(k\) which is its closest neighbor and is closer to cluster head compared to \(j\):

\[
R_j^t = \begin{cases} 
(d_{j, \text{cluster head}}) & \text{cluster head is } j's \text{ closest neighbor and } \\
& d_{j, \text{cluster head}} \leq R_{\text{max}} \\
(d_{j,k}) & (X_k, Y_k) \text{ is } j's \text{ closest neighbor, and } \\
& \text{distance } (k, \text{clusterhead}) < \text{distance } (j, \text{clusterhead}), R_k^t \neq 0, \text{ and } d_{j,k} \leq R_{\text{max}} \\
0 & \text{otherwise.}
\end{cases}
\]

(16)

Each solution in the initial population is evaluated using the objective functions \(F_1, F_2, F_3\) and connectivity constraint mentioned earlier. Then, offspring population will be created from the existing (parent) population using the crowded tournament selection, crossover, and mutation operator.

We use two-point crossover operator with randomly selected crossing points in our solution. Figure 4 illustrates the results of applying two-point crossover on two sample solutions with six parts.

The mutation type used in this paper depends on the selected bit in the solution to be muted. If the selected bit is the first bit of an item in the solution (represents the scheduling status of node), the mutation operator works as follows.

At first, the minimum number of required active sensors \(S_{\text{min}}\) for satisfying the requested coverage rate is calculated using

\[
S_{\text{min}} = \left\lceil \frac{A_c \cdot d_{\text{cov}}}{A_s} \right\rceil,
\]

(17)

where \(A_c\) is the area of the cluster which will be monitored, \(A_s\) is the circular sensing area of a sensor, and \(d_{\text{cov}}\) is the requested coverage rate by the designer. Since circular coverage is assumed for each node, we have

\[
A_s = \pi (R_s)^2.
\]

(18)
Table 1: The possibilities of four-bit encoding scheme.

| Binary encoding | Description |
|-----------------|-------------|
| 0000            | Being an inactive sensor |
| 0100            | Being an active sensor operating in a first (lowest) level of transmission range |
| 1001            | Being an active sensor operating in a second level of transmission range |
| ...             | ... |
| 1111            | Being an active sensor operating in a last (highest) level of transmission range |

Figure 3: Genetic representation of solution.

Before applying mutation operator on the solution \( S \), the number of active sensors in \( S \) (\( S_{\text{active}} \)) is calculated and then, based on \( S_{\text{active}} \) and \( S_{\text{min}} \), the mutation operator will apply on it by considering the following rules.

If \( S_{\text{active}} < S_{\text{min}} \) and the selected bit is 0, then it will be changed to 1;

else if \( S_{\text{active}} > S_{\text{min}} \) and the selected bit is 1, then it will be changed to 0;

else the mutation operator should select another bit for mutation.

Regarding these rules, it is obvious that this type of mutation employs problem-based knowledge and will reduce the energy consumption of the network and improve the network coverage.

If the selected bit for mutation is not the first bit of an item on the solution, we use the classical mutation which swaps the selected bit (0 becomes 1 and vice versa).

After creating the offspring population, it combined with the parent population and the new population will be created. Then, the fast nondominated sorting algorithm is performed on this combined population to identify different fronts \( F_i \), \( i = 1, 2, \ldots \). Then the new population is created and step by step contains the solutions of different nondominated fronts. At first, it contains the solutions of the best nondominated front \( F_1 \) and continues with solutions of the second nondominated and so on.

The new population of size \( M \) may not accommodate all fronts of the combined population with size \( 2M \). So, if all the solutions of the current front cannot be accommodated in the new population, the DCD values of the solutions in this front are calculated based on (5) discussed in Section 3. Then, the solutions will be sorted in descending order according to DCD values and the required solutions for new population will be selected from the first of this front. The above-mentioned process continues until the number of generations reaches the maximum number of generations.

The pseudocode of the TASCC algorithm is outlined in Algorithm 2.

6. Simulation Results and Discussions

In this section, we evaluate the performance of the TASCC algorithm via simulations. Simulations were performed using the NS-2 simulator. The simulations are conducted for different number of sensor nodes in the network and the averages of the results are depicted.

6.1. Simulation Settings. Experiments are conducted on a 100 * 100 square meter region. We consider the grids of size 2 m * 2 m, and if centers of all grids are covered, we assume that the entire region is covered. The simulation parameters are summarized in Table 2.

6.2. Results. We consider the following metrics for evaluation and compare the TASCC algorithm with the OGDC [14], CCP [15], LAEEC [17], and ECCA [18]:

(1) coverage rate of the network;
(2) connectivity of the network;
(3) number of active sensors;
(4) balanced energy consumption;
(5) network lifetime.

Experiment 1. In this experiment, 100 sensor nodes are deployed in the region and the coverage rate of the network is evaluated. A grid is said to be covered, if the center of that grid is covered at least by a sensor node. The coverage rate of the monitoring region versus the number of active
**TASCC Algorithm**

**Input:** Network parameters \((A_c, NC, d_{cov}, G, R_s, R_t(\text{max}), \ldots)\)  

**Output:** Non-dominated solutions.  

1. Choose population size \(M\), crossover probability \((P_c)\) and mutation probability \((P_m)\), crossover and mutation index, maximum number of generations \((\text{gen}_{\text{max}})\).
2. Set the generation counter \(t = 0\).
3. Generate an initial population \(D_t\) as follows:
   - For \(i = 1, \ldots, M\) do
     - Step 2.1. Activate \(G\) sensor nodes uniformly and randomly in solution of population \(D_t\);
     - Step 2.2. For every activated sensor \(j\) in solution \(i\), assign transmission range level \((R_j)\) according to (16)
4. Evaluate each solution in \(D_t\) using objective functions and connectivity constrain;
5. Apply the crowded tournament selection, mutation and crossover operators to create offspring population \(E_t\) from \(D_t\);
6. Do fast non-dominated sorting on \(P_t\) according to Algorithm 1, resulting non-dominated fronts \(F_{t1}, F_{t2}, \ldots\);
7. Set \(D_{t+1} = \emptyset; i = 1\);
   - While \(|D_{t+1}| + |F_i| < M\) do
     - \((D_{t+1} = D_{t+1} \cup F_i; i = i + 1)\);
     - If \(|D_{t+1}| < M\) then
       - Calculate DCD values of the solutions in \(F_i\) and sort the solutions in descending order according to DCD values;
       - Add the first \(M - |D_{t+1}|\) solution from \(F_i\) to \(D_{t+1}\);
     - If \(t < \text{gen}_{\text{max}}\) then set \(t = t + 1\) and go to Step 3.

**Algorithm 2:** The pseudocode of the TASCC algorithm.

**Table 2:** Simulation parameters.  

| Parameter                  | Value                  |
|----------------------------|------------------------|
| Field dimensions \((100 \times 100)\) m² |                        |
| \(R_s\)                     | 10 m                   |
| Max \(R_t\)                 | 35 m                   |
| Levels of transmission ranges (8 levels) | 0, 5, 10, 15, \ldots, 35 |
| \(\text{gen}_{\text{max}}\)   | 200                    |
| Amp                        | 100 pJ/bit/m²          |
| \(E_s\)                     | 5 nJ/bit               |
| \(E_{\text{rx}}\)            | 5 nJ/bit               |
| \(E_{\text{tx}}\)            | 50 nJ/bit              |
| \(E_{\text{init}}\)          | 2 J                    |
| Population size             | 100                    |
| Mutation probability        | 0.2                    |
| Crossover probability       | 0.9                    |

**Figure 5:** The coverage rate versus the number of active sensors at various generations.

**Figure 6:** Plots the coverage rate versus the number of active sensors for different algorithms. As shown in this figure, with the same number of active sensors, TASCC can obtain a higher coverage rate than other algorithms because TASCC considers the number of active sensors and the coverage rate simultaneously. The coverage rate of CCP decreases with increase in the number of working sensors. This is because a large number of message exchanges are required in CCP to maintain neighborhood information. In the case of low density of active sensors, OGDC provides less coverage rate rather than the other protocols. This is due to the fact that OGDC suffers from many redundant active sensors.
Experiment 2. In this experiment, we consider different number of deployed nodes and the coverage rate of 0.9 and measure the required number of active nodes for satisfying this coverage rate for different algorithms. The results of this experiment are presented in Figure 7. According to the results shown in this figure, it can be seen that TASCC has the minimum number of active nodes as compared to the other protocols. With the increasing of number of deployed sensors, TASCC maintains the same number of active sensors (about 25 sensors). This is because the number of required active sensors is independent of the density of the deployed sensors while for other algorithms, the number of required active nodes increases as the number of deployed sensors increases.

Experiment 3. In this experiment, we consider different number of deployed nodes and the coverage rate of 0.9 and compare the lifetime of the proposed algorithm with other algorithms. The network lifetime is defined as the round when the first node depletes its energy. The changes in the network lifetime are shown in Figure 8 where the number of sensors changes from 50 to 200. It can be seen that for all algorithms, as the network size increases, the network lifetime increases too. It is observed that the proposed algorithm has the longest lifetime. This is because in the proposed algorithm, in addition to density control of active nodes, balancing the energy consumption in the network and adjusting the transmission range of nodes result in extending the network lifetime. As mentioned earlier, in other algorithms, the transmission range should be at least twice the size of the sensing range and this may cause more energy consumption and shortening of the network lifetime.

The network lifetime in CCP is much shorter than in other algorithms. This is because CCP periodically broadcasts hello messages, and thus more energy will be consumed.

Experiment 4. In this experiment, 200 sensor nodes are deployed in the region and the residual energy of these nodes is measured after 200 generations in order to calculate
the unbalanced energy consumption of different algorithms according to (13). The value of this parameter varies between 0 and 1. At the ideal case, the maximum and the minimum residual energy of the nodes will be the same and the value of the unbalanced energy consumption parameter will be 0. On the other hand, in the worst case, the energy of at least one sensor will be depleted while some other nodes are full of energy and according to (13), the value of this parameter will be 1. Figure 9 shows the unbalanced energy consumptions for different algorithms. Based on the results, we find that the unbalanced energy consumption in the proposed algorithm is lower than the other algorithms. This is due to the fact that the proposed algorithm, unlike the other algorithms, considers the balanced energy consumption as an objective and avoids the rapid depletion of the energy and periodically changes the active sensors to balance the residual energy of nodes.

Experiment 5. In this experiment, different network sizes are considered and the number of unconnected sensors in the proposed algorithm is calculated and compared with other algorithms. Figure 10 shows the result of this experiment. As shown in this figure, TASCC outperforms other algorithms in terms of connectivity. This is because the proposed algorithm unlike the other algorithms can dynamically adjust the transmission range of the sensors to maintain the connectivity and reduce the unconnected sensors.

7. Conclusions

Maximizing covered area, balancing energy consumption, minimizing the transmission range of nodes, minimizing the number of active nodes, and maintaining the connectivity of nodes are critical issues in WSNs. These issues are usually conflicting. In this paper, we proposed a multiobjective optimization algorithm called TASCC to optimize these conflicting issues simultaneously. We executed different simulation experiments to show the performance of the proposed algorithm. The results obtained from the simulations showed that with the same number of active sensors, TASCC can obtain a higher coverage rate than other algorithms. Moreover, due to balanced energy consumption and dynamically adjusting the transmission range of the sensors in TASCC, the network lifetime was prolonged.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] H. Huang, G. Hu, and F. Yu, “Energy-aware geographic routing in wireless sensor networks with anchor nodes,” International Journal of Communication Systems, vol. 26, no. 1, pp. 100–113, 2013.
[2] G. Y. Liu, B. G. Xu, and H. B. Chen, “Decentralized estimation over noisy channels in cluster-based wireless sensor networks,” International Journal of Communication Systems, vol. 25, no. 10, pp. 1313–1329, 2012.
[3] C.-X. Liu, Y. Liu, Z.-J. Zhang, and Z.-Y. Cheng, “High energy-efficient and privacy-preserving secure data aggregation for wireless sensor networks,” International Journal of Communication Systems, vol. 26, no. 3, pp. 380–394, 2013.
[4] I. S. Alshawi, L. Yan, W. Pan, and B. Luo, “Fuzzy chessboard clustering and artificial bee colony routing method for energy-efficient heterogeneous wireless sensor networks,” International Journal of Communication Systems, vol. 26, no. 6, pp. 114–129, 2013.
[5] F.-C. Jiang, D.-C. Huang, C.-T. Yang, C.-H. Lin, and K.-H. Wang, “Design strategy for optimizing power consumption of sensor node with Min(N,T) policy M/G/1 queuing models,” International Journal of Communication Systems, vol. 25, no. 5, pp. 652–671, 2012.
[6] Z. Fang and J. Wang, “Convex combination approximation for the min-cost WSN point coverage problem,” in Proceedings of the 3rd International Conference on Wireless Algorithms, Systems, and Applications, Dallas, Tex, USA, 2008.
[7] M. K. Watfa and S. Commuri, “Boundary coverage and coverage boundary problems in wireless sensor networks,” International Journal of Sensor Networks, vol. 2, no. 3–4, pp. 273–283, 2007.
[8] B. Wang, H. B. Lim, and D. Ma, “A survey of movement strategies for improving network coverage in wireless sensor networks,” Computer Communications, vol. 32, no. 13-14, pp. 1427–1436, 2009.

[9] J. Li, K. Yue, W. Liu, and Q. Liu, “Game-theoretic based distributed scheduling algorithms for minimum coverage breach in directional sensor networks,” International Journal of Distributed Sensor Networks, vol. 2014, Article ID 341309, 12 pages, 2014.

[10] J. Yu, S. Ren, S. Wan, D. Yu, and G. Wang, “A stochastic k-coverage scheduling algorithm in wireless sensor networks,” International Journal of Distributed Sensor Networks, vol. 2012, Article ID 746501, 11 pages, 2012.

[11] H. M. Ammari and S. K. Das, “Centralized and clustered k-coverage protocols for wireless sensor networks,” IEEE Transactions on Computers, vol. 61, no. 1, pp. 118–133, 2012.

[12] L. Liu, B. Hu, and L. Li, “Energy conservation algorithms for maintaining coverage and connectivity in wireless sensor networks,” IET Communications, vol. 4, no. 7, pp. 786–800, 2010.

[13] A. Konstantinidis, K. Yang, Q. Zhang, and D. Zinealipour-Yazti, “A multi-objective evolutionary algorithm for the deployment and power assignment problem in wireless sensor networks,” Computer Networks, vol. 54, no. 6, pp. 960–976, 2010.

[14] H. Zhang and J. Hou, “Maintaining sensing coverage and connectivity in large sensor networks,” Ad Hoc and Sensor Wireless Networks, vol. 1, no. 2, pp. 89–124, 2005.

[15] X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, and C. Gill, “Integrated coverage and connectivity configuration in wireless sensor networks,” ACM Transactions on Sensor Networks, vol. 1, no. 1, pp. 36–72, 2005.

[16] P.-J. Wan, X. Xu, and Z. Wang, “Wireless coverage with disparate ranges,” in Proceedings of the 12th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc ’11), New York, NY, USA, May 2011.

[17] J. Akbari Torkestani, “An adaptive energy-efficient area coverage algorithm for wireless sensor networks,” Ad Hoc Networks, vol. 11, no. 1, pp. 1655–1666, 2013.

[18] J. Jia, J. Chen, G. Chang, and Z. Tan, “Energy efficient coverage control in wireless sensor networks based on multi-objective genetic algorithm,” Computers & Mathematics with Applications, vol. 57, no. 11-12, pp. 1756–1766, 2009.

[19] H. Zhou, T. Liang, C. Xu, and J. Xie, “Multiobjective coverage control strategy for energy-efficient wireless sensor networks,” International Journal of Distributed Sensor Networks, vol. 2012, Article ID 720734, 10 pages, 2012.

[20] S. Misra, M. Pavan Kumar, and M. S. Obaidat, “Connectivity preserving localized coverage algorithm for area monitoring using wireless sensor networks,” Computer Communications, vol. 34, no. 12, pp. 1484–1496, 2011.

[21] X. Han, X. Cao, E. L. Lloyd, and C.-C. Shen, “Deploying directional sensor networks with guaranteed connectivity and coverage,” in Proceedings of the 5th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON ’08), pp. 153–160, San Francisco, Calif, USA, June 2008.

[22] Y. E. Osais, M. St-Hilaire, and F. R. Yu, “Directional sensor placement with optimal sensing range, field of view and orientation,” Mobile Networks and Applications, vol. 15, no. 2, pp. 216–225, 2010.

[23] B. A. Attea, F. Y. Okay, S. Ozdemir, and M. A. Akayol, “Multi-objective evolutionary algorithm based on decomposition for efficient coverage control in mobile sensor networks,” in Proceedings of the 6th International Conference on Application of Information and Communication Technologies (AICT ’12), Tbilisi, Ga, USA, October 2012.

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

[25] S. B. Chaudhry, V. C. Hung, R. K. Guha, and K. O. Stanley, “Pareto-based evolutionary computational approach for wireless sensor placement,” Engineering Applications of Artificial Intelligence, vol. 24, no. 3, pp. 409–425, 2011.

[26] H. Z. Abidin, N. M. Din, and Y. E. Jalil, “Multi-objective Optimization (MOO) approach for sensor node placement in WSN,” in Proceedings of the 7th International Conference on Signal Processing and Communication Systems (ICSPCS ’13), pp. 1–5, Carrara, Australia, December 2013.

[27] T. W. Sung and C. S. Yang, “Voronoi-based coverage improvement approach for wireless directional sensor networks,” Journal of Network and Computer Applications, vol. 39, no. 1, pp. 202–213, 2014.

[28] 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 Transactions on Systems, Man and Cybernetics C: Applications and Reviews, vol. 42, no. 6, pp. 1093–1102, 2012.

[29] S. Sengupta, S. Das, M. D. Nasir, and B. K. Panigrahi, “Multi-objective node deployment in WSNs: in search of an optimal trade-off among coverage, lifetime, energy consumption, and connectivity,” Engineering Applications of Artificial Intelligence, vol. 26, no. 1, pp. 405–416, 2013.

[30] F. V. C. Martins, E. G. Carrano, E. F. Wanner, R. H. C. Taka-hashi, and G. R. Mateus, “A hybrid multiobjective evolutionary approach for improving the performance of wireless sensor networks,” IEEE Sensors Journal, vol. 11, no. 3, pp. 545–554, 2011.

[31] J. Yu, X. Deng, D. Yu, G. Wang, and X. Gu, “CWSC: connected k-coverage working sets construction algorithm in wireless sensor networks,” International Journal of Electronics and Communications, vol. 67, no. 11, pp. 937–946, 2013.

[32] F. Castaño, A. Rossi, M. Sevaux, and N. Velasco, “A column generation approach to extend lifetime in wireless sensor networks with coverage and connectivity constraints,” Computers and Operations Research, 2013.

[33] K. Deb, Multi-Objective Optimization Using Evolutionary Algorithms, John Wiley & Sons, New York, NY, USA, 2002.

[34] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, “A fast and elitist multiobjective genetic algorithm: NSGA-II,” IEEE Transactions on Evolutionary Computation, vol. 6, no. 2, pp. 182–197, 2002.

[35] B. Luo, J. Zheng, J. Xie, and J. Wu, “Dynamic crowding distance—a new diversity maintenance strategy for MOEAs,” in Proceedings of the 4th International Conference on Natural Computation, pp. 580–585, Jinan, China, October 2008.

[36] W. Ye, J. Heidemann, and D. Estrin, “An energy-efficient MAC protocol for wireless sensor networks,” in Proceedings of the 21th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM ’02), pp. 1567–1576, New York, NY, USA, June 2002.

[37] X. Liu and P. Mohapatra, “On the deployment of wireless data backhaul networks,” IEEE Transactions on Wireless Communications, vol. 6, no. 4, pp. 1426–1435, 2007.