Research Article

Dynamic Deployment for Hybrid Sensor Networks Based on Potential Field-Directed Particle Swarm Optimization

Ying Zhang, Yunlong Qiao, Wei Zhao, Wei Chen, and Jinde Cao

1College of Information Engineering, Shanghai Maritime University, Shanghai 201306, China
2Department of Computer Science, Tennessee State University, Nashville, TN 37209, USA
3Department of Mathematics, Research Center for Complex Systems and Network Sciences, Southeast University, Nanjing 210096, China
4Department of Mathematics, Faculty of Science, King Abdulaziz University, Jeddah 21589, Saudi Arabia

Correspondence should be addressed to Yunlong Qiao; qiaoyunlong501@163.com

Received 7 November 2014; Revised 21 March 2015; Accepted 22 March 2015

Abstract

For a hybrid sensor network, the effective coverage rate can be optimized by adjusting the location of the mobile nodes. For many deployments by APF (artificial potential field), due to the common problem of barrier effect, it is difficult for mobile nodes to diffuse by the weaker attraction when the nodes initially distribute densely in some places. The proposed deployment algorithm PFPSO (Potential Field-Directed Particle Swarm Optimization) can overcome this problem and guide the mobile nodes to the optimal positions. Normally the requirement is different for the effective coverage rate between the hotspot area and the ordinary area. On the basis of PFPSO, NPFPSO (Nonuniform PFPSO) algorithm was also proposed to implement nonuniform coverage according to the importance degree of the monitoring area. Simulation result illustrates that PFPSO algorithm can effectively improve the effective coverage rate of the network, and NPFPSO algorithm can obtain a balanced result of effective coverage rate for both hotspot area and ordinary area.

1. Introduction

The wireless sensor network becomes a research hotspot because of its great application value in the military and environmental monitoring. In most applications, manual deployment is nearly impossible because it is hard for human to approach the target field. In traditional deployment strategies, a plenty of fixed nodes including many superfluous nodes are dropped dispersedly on the target area by airplane to guarantee the effective coverage rate; these nodes can be used for targets monitoring and events tracing. In most cases these redundant nodes are involved in the communication of the whole tasks, and it can guarantee the service completed. Apparently this deployment manner will generate a lot of redundant nodes to guarantee effective coverage rate, and it is not an optimal way to complete the tasks. Although sometimes this manner can enhance the service quality of the network, it causes a lot of waste of devices and leads to conflict of communication, energy waste, and shortening network lifetime. As we know, the network with fixed sensor nodes cannot repair the coverage of network by itself due to its poor environmental adaptability. The mobile sensor networks only composed of mobile nodes can increase the coverage rate and improve network service quality by guiding the nodes moving. But, unfortunately, mobile node is a bit too expensive to be suitable for large-scale deployment. The hybrid sensor networks consist of both fixed nodes and mobile nodes, and the mobile nodes can repair the coverage holes where the fixed nodes cannot cover them by adjusting the positions of mobile nodes. It can achieve a better balance between the economy and network service quality.

For wireless sensor networks, a reasonable deployment can save more time and money to set up a network and quickly cover the target area. It can extend the network lifetime by coordination control and adapt to the change of network topology. The network's coverage rate directly affects
the accuracy and comprehensiveness of the monitoring result. Deployment optimization problem of hybrid sensor network has been proved to be an NP-hard problem [1]. The deployment for hybrid sensor networks is more complex than the pure mobile sensor networks due to the existence of fixed nodes. Considering the characteristics of the hybrid sensor networks, we proposed the PFPSO (Potential Field-Directed Particle Swarm Optimization) algorithm to optimize the deployment when mobile nodes distribute densely in some places. It can overcome the common defect of APF approach, enhance the convergence speed, and improve the effective coverage rate.

In more realistic environments, such as detecting coastal water quality in drain outlet or harbor areas and searching and rescuing in some sea areas, high surveillance accuracy is required for these hotspot regions and low accuracy for less focal regions. The region which is more important and is with higher event occurrence probability can be regarded as hotspot region. Therefore, according to the importance of the target area, we proposed the NPFPSO (Nonuniform PFPSO) algorithm for nonuniform coverage by modifying the fitness function of PFPSO and velocity components of APF. It can improve the detection probability and effective coverage rate of hotspot areas and get a balanced result of effective coverage rate for both hotspot area and ordinary area.

The rest of this paper is structured as follows: Section 2 introduces some related research work of deployment issues of WSN. The generalized problem of uniform coverage and nonuniform coverage is stated in Section 3. PFPSO and NPFPSO deployment algorithms are stated in detail in Sections 4 and 5. PFPSO performance evaluation and the virtual force acceleration parameter $c_v$ of this algorithm are discussed in Section 6. A set of Pareto solutions in two different situations used to evaluate NPFPSO is shown in Section 7. Finally, Section 8 concludes this paper.

2. Related Works

2.1. Uniform Coverage. There are many deployment strategies in the mobile sensor networks, which can be divided into virtual force, swarm intelligence algorithms, and computational geometry. However, in these common strategies, there is less attention on hybrid sensor networks, and many of the algorithms applied in mobile networks are not applicable for hybrid sensor networks.

Zou and Chakrabarty [2] proposed a Virtual Force Algorithm (VFA) which assumes that repulsion and attraction exist among the sensor nodes. When the distance between two nodes is less than the threshold $d_{th}$, the nodes will repel each other outward to reduce the redundant coverage area. On the contrary, when the distance between two nodes is greater than the threshold $d_{th}$, the nodes will aggregate to repair coverage gaps by attractive force. However, the things will be deteriorated by using VFA in hybrid sensor networks. In this case, fixed nodes and mobile nodes have consistency on the coverage effect, and fixed nodes will also have virtual force on mobile nodes, so the mobile nodes are often limited in the region surrounded by fixed nodes to be difficult to diffuse. In addition, many parameters in VFA which have great impact on optimization result of network are all needed to be set up by experiences.

Huang [3] proposed a self-deployment method named as Ion-6. The mobile nodes are modeled as ions which are linked by ionic bonds. The mobile nodes achieve even distribution by forming a regular hexagonal topology. Wang et al. used Voronoi diagrams to detect coverage holes and proposed three protocols: VEC, VOR, and Minimax to control the movement of mobile nodes [4]. Because all these methods do not consider the influence of fixed nodes, these geometry-based deployment strategies cannot be applied in hybrid networks. Wang et al. [1] proposed the bidding protocol based on Voronoi diagram and greedy algorithm which can be used in hybrid sensor network. In this algorithm, the coverage holes will be detected by Voronoi diagram, and then the surrounding mobile nodes will be hidden. Mobile nodes with the highest bidding will be chosen to move to the target point and repair the coverage holes. In this process, mobile nodes can get rid of the constraint from fixed nodes. But obtaining Voronoi diagram needs the global location information of all other nodes in the network. Moreover, it is not flexible enough to find the target position of mobile nodes.

Wu et al. [5] used PSO in ad hoc network deployment optimization to improve network coverage rate. However, the convergence speed of coverage rate goes up slowly due to the velocity of particles in PSO updating randomly, and the computational complexity will rise exponentially as the number of mobile nodes increases. In order to improve the convergence speed and reduce the computational complexity of the algorithm, Wang et al. proposed parallel PSO (PPSO) algorithm [6] and virtual force directed PSO optimization (VFPSO) algorithm [7], and CVFPSO [8]. The algorithms can perform global searching by PSO to overcome the barrier effect of fixed nodes and perform particle acceleration by using the virtual force between the nodes. The VFPSO and CVFPSO algorithms based on virtual force all need to set up the threshold distance $d_{th}$, attraction coefficient $w_a$, and the repulsion coefficient $w_r$, which can adjust the density of nodes indirectly. The adjusting parameters are too complicated to be implemented in practice. Moreover, they cannot directly reflect the detection probability of every point in the monitoring area. So these algorithms can only be used for solving problems of uniform coverage.

2.2. Nonuniform Coverage. When we consider the importance of monitoring area and the factors of event probability of occurrence for nonuniform coverage, these algorithms above are not applicable for the problems. Currently there are few researches on nonuniform coverage. Nonuniform coverage solutions could be divided into two categories, the first achieves different detection probability for the points in the region, and the second makes the node density consistent with the event probability of occurrence in the target area.

Zou and Chakrabarty [9] proposed the Max-Avg-Cov algorithm to maximize the monitoring points of average coverage and proposed the Max-Min-Cov algorithm to maximize the monitoring points of minimum monitoring.
probability based on the probabilistic detection model of sensor. Aitsaadi et al. [10] proposed PFDA algorithm and MODA algorithm based on artificial potential field method, which combines artificial potential field method and Tabu search algorithm to perform complete coverage for the target area. However, the nodes deployment strategies above in initialization stage need to be deployed manually, so it is not applicable for the sensor networks with random deployment [11]. The deployment goals of these algorithms above are to have the points in monitoring area achieve the expected measuring probability to meet the requirement conditions.

Koutsougeras et al. [12] used SOM (Self-Organizing Maps) method to distribute sensors for events coverage by utilizing the attraction from events to nodes and the phenomenon of nodes trending to move to the area with high density of events. Xia et al. [13] defined the conception of entropy to evaluate the balance of coverage and effectiveness of fish swarm. The method can drive the sensors to cover almost all the events and make the density of sensors match the density of the events. All the algorithms above adopt event-driven coverage model and let nodes cover events to optimize detection quality as far as possible. But they do not consider the network connectivity problems, and the events density cannot always correctly express the difference between hotspot area and the ordinary area in practice.

3. Concepts Statement and Precondition Assumption

3.1. Hybrid Sensor Network. Hybrid network is a network which is composed of fixed nodes and mobile nodes. At the beginning of deployment, all nodes are deployed in the region randomly, the fixed nodes will be fixed at a position and never move again, and on the contrary the mobile nodes could even move on. Compared with mobile network, the hybrid network could save more energy efficiently. Compared with fixed network, the hybrid network could save more resource of devices and has more flexibility to adapt to different environment. So it has more significance to be used in reality. But, sometimes, when we optimize hybrid networks by moving mobile nodes with some potential field algorithm (such as APF), it will face the problem of barrier effect which will weaken the effectiveness of the optimization.

3.2. Sensor Detection Model. \( S_{ov} \) represents the set of sensor nodes deployed on the target area \( A \), and suppose the sensors have the same detection radius \( r \). Considering a sensor node \( s_i \) deployed at point \((x_i, y_i)\), for point \((x_p, y_p)\) on \( A \), the Euclidean distance between \( s_i \) and \( p \) can be denoted as

\[
d(s_i, p) = \sqrt{(x_i - x_p)^2 + (y_i - y_p)^2}.
\]

In the binary detection model, if the target point \( p \) locates in detection radius of sensor \( s_i \), point \( p \) will be detected with probability 1 by sensor \( s_i \). Otherwise, the detection probability \( c_p(s_i) \) will be equal to 0, which means point \( p \) cannot be detected by sensor \( s_i \). Its analytical model can be expressed as the following formula:

\[
c_p(s_i) = \begin{cases} 
1 & \text{if } d(s_i, p) \leq r \\
0 & \text{other.}
\end{cases}
\]  

(1)

In practical application, due to the terrain, obstacles, and noises, the binary model will not be applicable to describe the situation. In fact, the perception model of sensor nodes is illustrated as a certain probability distribution model. The monitoring probability will decrease with the distance \( d(s_i, p) \) elongating. Considering the factor of electromagnetism and white noises in real situation, the detection probability \( c_p(s_i) \) of sensor \( s_i \) on point \( p \) can be presented as the following formula [14]:

\[
c_p(s_i) = \begin{cases} 
1 & \text{if } d(s_i, p) \leq r - r_e \\
ed^{-\alpha(s_i, p)/\lambda_1 - \beta_2} & \text{if } r - r_e < d(s_i, p) \leq r + r_e \\
0 & \text{other},
\end{cases}
\]  

(2)

where \( r_e \) is the parameter of detection reliability related to the characters of sensor nodes, \( \alpha \), \( \lambda_1 \), and \( \beta_2 \) are the parameters related with detection probability of sensors, and \( \lambda_1 \) and \( \lambda_2 \) are the input parameters, and they can be defined as the following formula:

\[
\lambda_1 = r_e - r + d(s_i, p),
\]

\[
\lambda_2 = r_e + r - d(s_i, p).
\]  

(3)

Figure 1 illustrates the curve of detection probability of sensor nodes changing with the distances.

3.3. Uniform Coverage Problem in Hybrid Sensor Networks. Suppose that all the nodes are randomly scattered in a two-dimensional monitoring area \( A \) while initializing. If point \( p \) is within the probabilistic perception of the sensor \( s_i \), the perception range can be expressed as follows: \( d(s_i, p) \leq (r + r_e) \), and the sensor \( s_i \) is the neighbor sensor of point \( p \). The detection probability of point \( p \) is the union detection probability of all its neighbor sensors within the monitoring area. Considering a grid point \( p \) lying in the overlap region sensed by a set of neighbor sensors \( S_{ne} \), the detection probability of the point which can be effectively detected by at least one sensor node is denoted as \( c_p(S_{ne}) \). It can be calculated by the following formula:

\[
c_p(S_{ne}) = 1 - \prod_{s_i \in S_{ne}} \left(1 - c_p(s_i)\right),
\]  

(4)

where \( S_{ne} \) is the set of neighbor sensors of point \( p \) and \( c_p(s_i) \) is the detection probability of sensor \( s_i \) at point \( p \). Suppose \( c_{th} \) is the threshold of predefined effective detection probability; the condition that point \( p \) can be effectively covered can be expressed as the following formula:

\[
c_p(S_{ne}) \geq c_{th}.
\]  

(5)
The deployment of a sensor network should both meet the condition of coverage and meet the condition of connectivity. When the node communication radius is more than twice the radius of perception, the coverage problem should only be considered for a sensor network, since this condition can guarantee the connectivity of the network [15–17]. Therefore, in this paper, we assume that there is a sink node which can collect location information of all nodes in the target area, and the communication radius of ordinary nodes meets the following condition: \( R_c \geq 2(r + r_e) \).

To summarize, the uniform deployment of hybrid sensor networks can be described as follows: \( m \) fixed nodes and \( n \) mobile nodes are scattered randomly in the target area \( A \). We can adjust the position of mobile nodes to maximize the effective coverage of target area. In particular, when \( m = 0 \), the hybrid sensor network coverage problem is equivalent to the coverage problems of mobile sensor networks.

### 3.4. Nonuniform Coverage Problem of Hybrid Networks

According to the monitoring requirements for different regions in target environment, sometimes we need to perform the nonuniform coverage in target area. It can pay more attention to some hotspot areas with more nodes and simplify sensor resources for some ordinary areas. Suppose \( A \) represents the 2-D target area, \( A_{\text{hot}} \) represents the hotspot area in \( A \), and \( A_{\text{ordinary}} \) represents ordinary surveillance area. The relationship of \( A_{\text{hot}} \) and \( A_{\text{ordinary}} \) meets the following formula:

\[
A_{\text{hot}} \cup A_{\text{ordinary}} = A
\]

\[
A_{\text{hot}} \cap A_{\text{ordinary}} = \emptyset.
\]

\( p_{\text{hot}} \) and \( p_{\text{ordinary}} \) represent the monitoring points in \( A_{\text{hot}} \) and \( A_{\text{ordinary}} \), respectively, and these areas can be covered effectively if the condition meets the following formula:

\[
c_{p_{\text{hot}}}(S_{\text{ov}}) \geq c_{th_{\text{hot}}}
\]

\[
c_{p_{\text{ordinary}}}(S_{\text{ov}}) \geq c_{th_{\text{ordinary}}}
\]

where \( c_{th_{\text{hot}}} \) and \( c_{th_{\text{ordinary}}} \) are the predefined effective detection thresholds of coverage probability for \( A_{\text{hot}} \) and \( A_{\text{ordinary}} \) according to their important levels, normally \( c_{th_{\text{hot}}} > c_{th_{\text{ordinary}}} \).

Thus the nonuniform deployment of hybrid sensor networks can be described as follows: \( m \) fixed nodes and \( n \) mobile nodes are scattered randomly in the target area \( A \). We can adjust the positions of mobile nodes to maximize the effective coverage rate of hotspot region and ordinary surveillance region. In particular, when \( A_{\text{hot}} = \emptyset \), the nonuniform coverage problem is equivalent to uniform coverage problem.

### 4. PFPSSO Algorithm for Uniform Coverage

For PSO algorithm, the initial position and velocity of the particles are generated randomly. It cannot guide the particles moving effectively only depending on these parameters: the particle velocity, the individual best position, and the global colony best position. On the other hand, by APF algorithm, the mobile nodes can bypass the fixed nodes and repair the coverage holes by the attractive force generated by uncovered grid points. However, sometimes it will meet the problem of barrier effect: when the fixed nodes distribute in some region with high density, the coverage probability will be high in their neighboring regions, and the nearby mobile nodes will be difficult to spread due to the diminishing attraction by the coverage hole. That is to say we cannot obtain an ideal optimal result only depending on any single one of these two algorithms.

The proposed PFPSSO algorithm integrates their advantages of artificial potential field and particle swarm optimization organically. It can own two functions: (1) With the direction of the virtual force, the mobile nodes could move towards the better position to heal the coverage holes. (2) Implementing global optimization to make the mobile nodes overcome the barrier effect from the fixed nodes.

#### 4.1. Some Basic Concepts in PFPSSO Algorithm

The virtual force derives from artificial potential field, which was originally introduced to robot path planning to avoid obstacles and find the optimal path [10, 18, 19]. Here every node can move under the resultant force by the attractive force from targets and the repulsive force from obstacles. This kind of resultant force \( \vec{F} \) can be given by the following formula:

\[
\vec{F} = -\vec{V}U = \left( \begin{array}{c}
-\frac{\partial U}{\partial x_c} \\
-\frac{\partial U}{\partial y_c}
\end{array} \right),
\]

where \( U \) is the potential field function and \( (x_c, y_c) \) is the coordinates of the robot.
The deployment optimization problem of hybrid sensor network can be treated as healing coverage holes. The coverage hole will produce an attractive force to the mobile nodes. So, in this paper, we assume the point whose detection probability does not meet the condition of detection will generate an attractive potential field to the mobile nodes. The potential field function can be given by the following formula [10]:

\[ U^a(s) = \sum_{p \in A} \left( c_{th} - c_{p/s} \right) \cdot 1_{i f (c_{p/s} < c_{th})}, \]  

(9)

where \( A \) is the monitoring target area and \( c_{p/s} \) is the union detection probability at point \( p \) when a sensor is deployed at point \( s \) [10]:

\[ c_{p/s} = 1 - \left( 1 - c_p \left( s \right) \prod_{s' \in S_{m\backslash l}\{s\}} \left( 1 - c_p \left( s' \right) \right) \right). \]  

(10)

The virtual force \( \vec{F}_s \) of a sensor node received can be deduced as the following formula:

\[ \vec{F}_s = -\vec{V}U^a(s) = \left( \begin{array}{c} F_x^a \\ F_y^a \end{array} \right) = \left( \begin{array}{c} -\frac{\partial U^a(s)}{\partial x_s} \\ -\frac{\partial U^a(s)}{\partial y_s} \end{array} \right), \]  

(11)

where \( F_x^a \) and \( F_y^a \) represent the acting forces from the \( x \)-axis and \( y \)-axis directions, respectively. We can put the variables of formulas (2), (9), and (10) into (11) and get the results as formula (12) after differentiation. Consider

\[
\frac{\partial U^a}{\partial x_s} = \sum_{p \in A} \left[ \left( x_s - x_p \right) \frac{\alpha_1 \beta_x \lambda_x^{\beta_x-1} + \alpha_2 \beta_x \lambda_x^{\beta_x+1}}{\lambda_x^{\beta_x+1}} \right] e^{-\alpha_x \lambda_x^{\beta_x+1} + \alpha_2 \lambda_x^{\beta_x+1}} \prod_{s' \in S_{m\backslash l}\{s\}} \left( 1 - c_p \left( s' \right) \right),
\]

\[
\frac{\partial U^a}{\partial y_s} = \sum_{p \in A} \left[ \left( y_s - y_p \right) \frac{\alpha_1 \beta_x \lambda_x^{\beta_x-1} + \alpha_2 \beta_x \lambda_x^{\beta_x+1}}{\lambda_x^{\beta_x+1}} \right] e^{-\alpha_x \lambda_x^{\beta_x+1} + \alpha_2 \lambda_x^{\beta_x+1}} \prod_{s' \in S_{m\backslash l}\{s\}} \left( 1 - c_p \left( s' \right) \right),
\]

where \( A_s \) is the set of the points which do not meet the detection probability in the sensor’s detection range from \( r - r_c \) to \( r + r_c \):

\[ A_s = \{ p \in A \mid c_p \left( S_m \right) < c_{th}, \left| d \left( s_j, p \right) - r \right| \leq r_c \}. \]  

(13)

Then the sensor nodes will update the original location \((x_{old}, y_{old})\) to new location \((x_{new}, y_{new})\) according to the orientation and magnitude of the total virtual force \( \vec{F}_s \), and it can be calculated by the following formula:

\[ x_{new} = \begin{cases} x_{old}, & \text{if } |F_x^a| = 0 \\ x_{old} + \frac{F_x^a}{|F_x^a|} \times MaxStep \times e^{-1/|F_x^a|}, & \text{if } |F_x^a| \neq 0 \end{cases}, \]

\[ y_{new} = \begin{cases} y_{old}, & \text{if } |F_y^a| = 0 \\ y_{old} + \frac{F_y^a}{|F_y^a|} \times MaxStep \times e^{-1/|F_y^a|}, & \text{if } |F_y^a| \neq 0, \end{cases} \]  

(14)

where \( MaxStep \) is the predefined maximum step size of mobile nodes, the values of \( F_x^a, F_y^a \) can be got by formulas (11) and (12).

Particle swarm optimization is a stochastic global optimization algorithm for seeking the solution by imitating the flocks of bird [20, 21]. Due to fast convergence and less parameters for setting up, it is widely used in solving problems of multidimensional space. Suppose \( X_i = (x_{i1}, x_{i2}, \ldots, x_{in}) \) is the current position vector of particle \( i \), and \( V_i = (v_{i1}, v_{i2}, \ldots, v_{in}) \) is the current velocity vector of particle \( i \). \( f(X) \) is the fitness function used to evaluate whether the position of particle is good or not. For the minimized objective function \( f(X) \), the optimal position \( P_i = (p_{i1}, p_{i2}, \ldots, p_{in}) \) of particle \( i \) can be obtained by the following formula:

\[ P_i \left( t + 1 \right) = \begin{cases} P_i \left( t \right), & \text{if } f \left( X_i \left( t + 1 \right) \right) \geq f \left( P_i \left( t \right) \right) \\ X_i \left( t + 1 \right), & \text{if } f \left( X_i \left( t + 1 \right) \right) < f \left( P_i \left( t \right) \right). \end{cases} \]  

(15)

The best position of the colony \( P_g(t) = \min \{ f(P_0(t)), f(P_1(t)), \ldots, f(P_n(t)) \} \). The velocity and position of every generation particle can be updated as formulas (16) and (17), respectively. Consider

\[ v_{ij} \left( t + 1 \right) = w \left( t \right) v_{ij} \left( t \right) + c_1 r_{ij} \left( t \right) \left( p_{iy} \left( t \right) - x_{ij} \left( t \right) \right) + c_2 r_{ij} \left( t \right) \left( p_{gy} \left( t \right) - x_{ij} \left( t \right) \right) \]  

(16)

\[ x_{ij} \left( t + 1 \right) = x_{ij} \left( t \right) + v_{ij} \left( t + 1 \right), \]  

(17)

where \( c_1 \) and \( c_2 \) are the weighting factors of local optimization and global optimization, respectively. They can be used to adjust the evolutionary step of local optimization and global optimization of the particle. \( r_{ij} \) and \( r_{ij} \) are the independent random number in the range of \([0, 1]\). Subscript \( i \) is corresponding to the number \( i \) particle, and the subscript \( j \) is corresponding to the \( j \) dimension of the particle. \( w \) is the impact inertial factor indicating the influence of the past value to current value, and usually it can take a value between 0.4 and 0.9. It will gradually decrease with the iteration updating.
4.2. The Principle of PFPSO. In order to improve the effectiveness of coverage of the network, a kind of central scheduling deployment strategy was proposed, which integrates APF and PSO algorithm and overcomes the disadvantages of the two algorithms in hybrid networks deployment. Suppose a sink node in the target area can collect the information of coordinates for all the nodes; we can perform PFPSO algorithm and inform mobile nodes to move to the designated location directly according to the calculation result. This can reduce the moving distances of the nodes in the process of optimization.

Assuming $n$ mobile nodes and $m$ fixed nodes are randomly deployed in the target area. The position coordinates of $n$ mobile nodes can be regarded as one solution of PFPSO algorithm. The dimension of the searching space $N = 2n$ and the position vector of particle $i$ can be expressed as $X_i = (x_{i1}, x_{i2}, \ldots, x_{in}, y_{i1}, y_{i2}, \ldots, y_{in})$, where $x_{ij}$ and $y_{ij}$ are the coordinates of mobile node $j$. The effective coverage of monitoring area can be regarded as the fitness function, and it is expressed as the following formula:

$$f(X_i(t)) = \left\{ \forall p \in A \left| c_p(S_{ov}) \geq c_{th,ov} \right. \right\}.$$  

(18)

For PFPSO algorithm, the velocity of each particle is updated according to not only the historical optimal solutions of local and global positions, but also the attractive force exerting on mobile nodes. The influence of attractive force is reflected in the last item of formula (19). So the traditional velocity updating formula of PSO has been improved in PFPSO:

$$v_{ij}(t+1) = \omega(t)v_{ij}(t) + c_1r_{i1}(t)(p_{ij}(t) - x_{ij}(t)) + c_2r_{i2}(t)(g_{ij}(t) - x_{ij}(t))$$

\[ + c_3r_{i3}(t)g_{ij}(t), \]

(19)

where $c_3$ is the acceleration factor of attractive force and $r_{i3}$ is also an independent random number in the range of $[0, 1]$ just like $r_{i1}$ and $r_{i2}$. $g_{ij}$ is the distance effected by the potential field force derived from $j$th element in the position vector of $i$th particle, which can be expressed by the following formula:

$$g_{ij}(t) = \begin{cases} \frac{P(1)}{S_1} \times \text{MaxStep} \times e^{1/(P(1)/S_1)}, & j = 1, 2, \ldots, n \\ \frac{P(j)}{S_{j+1}} \times \text{MaxStep} \times e^{1/(P(j)/S_{j+1})}, & j = n + 1, n + 2, \ldots, 2n. \end{cases}$$

(20)

The PFPSO algorithm’s flow chart is illustrated as Figure 2.

The pseudo code of PFPSO is as shown in Algorithm 1.

5. NPFPSSO Algorithm for Nonuniform Coverage

Nonuniform coverage of hybrid sensor network can be regarded as a multiobjective optimization problem. Assume that $f_{hot}(X_i(t))$ and $f_{ordinary}(X_i(t))$ represent the effective coverage of the hotspot region and the ordinary region, respectively, which can be expressed as formula (21) and formula (22). Consider

$$f_{hot}(X_i(t)) = \left\{ \forall p \in A_{hot} \left| c_p(S_{ov}) \geq c_{th,hot} \right. \right\}$$

\[ \frac{A_{hot}}{A_{total}} \]

(21)

$$f_{ordinary}(X_i(t)) = \left\{ \forall p \in A_{ordinary} \left| c_p(S_{ov}) \geq c_{th,ordinary} \right. \right\}$$

\[ \frac{A_{ordinary}}{A_{total}} \]

(22)

What the nonuniform coverage needs to do is to maximize the objective functions $f_{hot}(X_i(t))$ and $f_{ordinary}(X_i(t))$ by optimizing the mobile nodes’ location. Ordinarily the number of mobile nodes is limited, so there is some conflict between the objective functions $f_{hot}(X_i(t))$ and $f_{ordinary}(X_i(t))$, and the solution of this kind of nonuniform coverage is a group of Pareto optimal solutions [22]. In this paper, the target weighting method is adopted by the proposed NPFPSSO algorithm [23]; it can integrate $f_{hot}(X_i(t))$ and $f_{ordinary}(X_i(t))$ with linear fusion, so the problem can be solved by converting into a single objective function. The fitness function of NPFPSSO can be calculated as the following formula:

$$f(X_i(t)) = \alpha f_{hot}(X_i(t)) + (1 - \alpha) f_{ordinary}(X_i(t)),$$  

(23)

where the weighting coefficient $\alpha \in [0, 1]$ and it can be used to make a balanced control for the effective coverage of the hotspot region and ordinary region.

Normally different target areas have different detection thresholds in nonuniform coverage of target environment. In order to make potential field force guide the mobile nodes moving more effectively, we improved formula (13) as the following formula:

$$A_s = A_{s,hot} \cup A_{s,ordinary},$$  

(24)

where $A_{s,hot}$ and $A_{s,ordinary}$ are the set of the points whose detection probability does not meet the effective detection thresholds in the sensing range of $[r - r_c, r + r_c]$, and they can be expressed as the following formula:

$$A_{s,hot} = \left\{ \forall p \in A_{hot} \left| c_p(S_{ov}) < c_{th,hot}, |d(s_i, p) - r| \leq r_c \right. \right\}$$

\[ \leq r_c \]

\[ \leq r_c \]

$$A_{s,ordinary} = \left\{ \forall p \in A_{ordinary} \left| c_p(S_{ov}) < c_{th,ordinary}, |d(s_i, p) - r| \leq r_c \right. \right\}.$$  

(25)

The NPFPSSO algorithm’s flow chart is illustrated as Figure 3.

The pseudo code of NPFPSSO algorithm is as shown in Algorithm 2.
6. The Simulation Testing of PFPSO and Performance Analysis

To investigate the performance of PFPSO in the deployment optimization, we simulate the algorithms in the simulation platform MATLAB2009. In this simulation, the problems of energy consumption and routings of the network are not involved. Supposing the monitoring target area is a 2-D square region with area of 200 m \times 200 m, 100 ordinary nodes are scattered randomly on this square target region which are composed of \( m \) fixed nodes and \( n \) mobile nodes. In order to expediently evaluate the effective coverage of wireless sensor network, the target region is divided into 40000 grids with each area of 1 m \times 1 m. We can calculate the detection probability of each grid, and the proportion of these grids which meet formula (5) is the effective coverage rate of the network. Some simulation results indicated that the absolute deviation between calculated value and the precise value of the coverage measure will be in the range of 0.1%–0.5% when the granularity (the distance between grid points) is as the scale of 0.25%–4% relative to the size of the target monitoring area [13]. The parameters of probabilistic detection model are set as \( \alpha_1 = 1, \alpha_2 = 0, \beta_1 = 1, \) and \( \beta_2 = 1.5. \) The detection radius of each node is \( r = 14 \) m, the reliability parameter of measurement is \( r_e = 7 \) m, and the communication radius is \( R_c = 2(r + r_e) = 42 \) m. The effective detection threshold is \( c_{th} = 0.9 \) and the maximum movement step of mobile nodes is \( MaxStep = 1 \) m.
### Algorithm 1: PFPSO.

\begin{algorithm}
\begin{algorithmic}[1]
\Require
Mobile nodes original coordinates \((X_m, Y_m)\)
Mobile nodes number \(n\)
Particle number \(M\)
Detection parameters of sensor \(r_s, r_t\)
The target field \(A\)
Effective detection probability threshold \(c_{th}\)
Maximal iteration of PFPSO \(MaxIter\)
\Ensure
The new coordinates of mobile nodes \(pg\)\_\(best\)
The effective coverage of the target field \(coverage\)
\end{algorithmic}
\begin{algorithmic}
\State \(x(0) \leftarrow (X_m, Y_m)\)
\For {\(i \leftarrow 1\) to \(M\)}
\State \(v_i(0) \leftarrow \) Randomly initializing the velocity of every particle
\State \(x_i(1) \leftarrow x(0) + v_i(0)\)
\State Calculate the \(f(x_i(1))\) of particle \(i\)
\EndFor
\State \(pg\)\_\(best \leftarrow \max\{f(x_i(1)), f(x_s(1)), \ldots, f(x_M(1))\}\)
\State \(coverage \leftarrow f(pg\)\_\(best)\)
\For {\(t \leftarrow 2\) to \(MaxIter\)}
\For {\(i \leftarrow 1\) to \(M\)}
\For {\(j \leftarrow 1\) to \(n\)}
\State \(A_j \leftarrow \{\forall p \in A | c_p < c_{th}\text{ and } |d(s_j, p) − r| \leq r_s\}\)
\State Calculate attractive virtual force \(F^a\)
\EndFor
\State Compute the velocity \(v_i(t)\) of particle \(i\)
\State \(x_i(t) \leftarrow x_i(t−1) + v_i(t)\)
\State Calculate the \(f(x_i(t))\) of particle \(i\)
\EndFor
\State \(pg\)\_\(best \leftarrow \max\{f(x_1(t)), f(x_s(t)), \ldots, f(x_M(t))\}\)
\If {\(f(pg\)\_\(best) < f(pg\)\_\(best)\)}
\State \(pg\)\_\(best \leftarrow g\)\_\(best\)
\EndIf
\State \(coverage \leftarrow f(pg\)\_\(best)\)
\EndFor
\end{algorithmic}
\end{algorithm}

6.1. The Effectiveness of PFPSO Compared with Other Typical Algorithms. Suppose in some cases the mobile nodes are initially distributed in dense state, and we want to make them scatter out in the target area for an optimal coverage. For example, 60 fixed nodes are randomly deployed uniformly in the region of 200 m \(\times\) 200 m 2-D area and 40 mobile nodes randomly distributed in the central square region with the area of 100 m \(\times\) 100 m. The initial coverage diagram is shown as Figure 4. In this figure, the red regions represent the detection probability of 100%, the blue regions mean the detection probability does not reach the threshold, and the transitional color shows that the detection probability of the point is higher than the threshold but lower than 100%.

Figures 5(a) and 5(b) show the coverage diagram by APF algorithm with running 50 iterations and the corresponding trajectories of mobile nodes, respectively. Apparently there are more uncovered regions with blue color in Figure 5(a), and the mobile nodes are almost bounded in the central area as shown in Figure 5(b). Figure 5(b) illustrates the moving status of mobile nodes from the original position “•” to the terminal position “o,” and “*” represents the fixed node distributed randomly. Apparently APF algorithm cannot achieve the desired optimization of coverage when mobile nodes are influenced by the barrier effect due to the smaller attractive force generated by uncovered grid points acting on the mobile nodes.

In order to verify the effectiveness of PFPSO, suppose the local and global optimum values in PSO and the virtual force exerting on mobile nodes generate the same impact on the position optimization of mobile nodes, which is to set the acceleration factors as follows: \(c_1 = c_2 = c_3 = 1\). The maximum iterations \(MaxIter\) is taken as 100 times, the number of particles is \(M = 20\), and the inertia coefficient is taken as \(w = 0.9 − 0.5 \times (1 − t/MaxIter)\) according to experiences. In the same initial coverage state as Figure 3, the coverage result by PFPSO after 100 iterations is shown as Figure 6.

Investigating the two figures, Figures 5(a) and 6(a), we can find the coverage rate by performing PFPSO has been improved a lot comparing with APF method. We also test the pure PSO algorithm under the same initial coverage state, and the whole testing results of coverage rate for these
Initialize velocity and position of mobile particles; setting hotspot threshold and ordinary region threshold

Calculate the virtual forces from the set points of hotspots and ordinary area

Update the velocity of the mobile particles

Calculate the combined fitness function which consists of $f_{\text{hot}}(x_i(t))$ and $f_{\text{ordinary}}(x_i(t))$ for the target area

$f(P_i) > f(P_g)$?

Yes

$P_g = P_i$

No

Reach MaxIter?

Yes

Output $P_g$ of hotspots region and ordinary region

No

Figure 3: NPFPSO algorithm’s flow chart.

three methods are shown in Table 1. Figure 6(b) illustrates the displacement of mobile nodes from their original position “∗” to the terminal position “o” and the “∗” represents the fixed node distributed randomly. As Figure 6(b) illustrates, apparently the mobile nodes can diffuse from the original central area to all the other areas uniformly for repairing the coverage holes. Due to adding an item of velocity component caused by virtual force in the velocity update calculating as formula (19), the velocity of the mobile nodes in PFPSO is higher than the velocity in APF and PSO algorithm. It has been advantageous to make the nodes spread rapidly, and it is more useful to overcome the barrier effect of fixed nodes encapsulated.

Figure 7 and Table 1 illustrate the raising curve of effective coverage and the final coverage rate after 100 iterations for APF, PSO, and PFPSO, respectively. In Figure 7 we can find that the two coverage curves of PSO and PFPSO rise rapidly and almost overlap in the first 10 iterations. It indicates that the mobile nodes are mainly guided by the velocity component of PSO in the preliminary stage of optimization of PFPSO. With increasing of iterations, the curve of PSO increases slowly in later period, and it is exceeded by APF at the 28th rounds of iteration. During 10 to 50 rounds of iteration, the slopes of the curve of PFPSO and APF are roughly similar which indicates that the mobile nodes are mainly guided to optimal position by the attractive force.
In addition, as shown in Figure 7 and Table 1, we can find that the convergence speed of PFPPO algorithm is obviously higher than APF and PSO. So PFPPO can effectively overcome the barrier effect from the fixed nodes as using potential field method, and it can achieve a higher coverage rate for performing optimization.

6.2 Discussion Regarding Parameter \( c_3 \) Evaluation. For formula (19), when \( c_3 = 0 \), PFPPO and PSO algorithm are the same, and when \( c_3 = c_2 = 0 \), PFPPO and APF algorithm are the same. So PSO and APF algorithm can be regarded as the special forms of the proposed PFPPO algorithm. The parameter \( c_3 \) directly controls the weight of position optimization for mobile nodes by the virtual force in the performing period, and it influences the convergence speed and searching result of PFPPO. We analyzed the movement of the mobile nodes during the algorithm execution. At the beginning, the mobile nodes are gathered in the center, and only fewer points which do not meet the detection threshold will generate attractive force to the mobile nodes. By the initial velocity generated randomly by PSO in the early stage, the mobile nodes can diffuse from the density state in the original centre position. In the middle and later periods, with the mobile nodes diffusing, the virtual force components in the velocity updating of particles become the main driving factor for mobile nodes guiding. According to the analysis above, considering the change rule of inertial coefficient \( w \), three changing circumstances will be validated for the parameter.

Algorithm 2: NPFPSO.
The first is the constant, and it means $c_3$ will be the same value during all the process. The second means $c_3$ will rise monotonically, and it indicates the influence of potential field force rises up gradually. The third one means $c_3$ will rise up in the first half stage and falls off in the later half stage. The three lines $l_1$, $l_2$, and $l_3$ represent the typical three kinds of changing circumstances of $c_3$:

\begin{align*}
\text{l}_1: \ & y = 1 \\
\text{l}_2: \ & y = \frac{2t}{\text{Maxiter}} \\
\text{l}_3: \ & y = 4 \times \left(\frac{1}{2} - \left|\frac{t}{\text{Maxiter}} - \frac{1}{2}\right\right).
\end{align*}

The three lines are shown in Figure 8 where Maxiter is the maximum iterations, $t$ is the current iteration, and $y$ is the value of $c_3$. The regions of enclosed shape surrounded by line $l_i$ $(i = 1, 2, 3)$, x-axis, and the line of $x = \text{Maxiter}$ have the same area. It means the integrals to x-axis in the range of $[0, \text{Maxiter}]$ for these three lines are equivalent, and these integral areas have the same size: Maxiter. It means that the virtual forces generated by APF make the same working capacity for the mobile nodes during the execution. When $c_3$ takes the changing value with the curves of $l_1$, $l_2$, and $l_3$, respectively, and the other conditions remain unchanged, the effective coverage of performing PFPSO is shown in Figure 8 accordingly ($\text{Maxiter} = 50$).

In Figure 9, effective coverage rates corresponding to $l_1$, $l_2$, and $l_3$ are very close to each other during the early process of 1st to 10th iteration. During the interval of 15th to 38th iteration, the effective coverage rate of $l_3$ is a little higher than the one of $l_2$ and $l_1$. After 40 iterations, the search result of PFPSO algorithm approaches the optimal solution. During the whole execution of algorithm, when the work done by the calculated virtual forces on the mobile nodes is the same, the final optimal results of coverage rate by PFPSO are roughly the same. Their rising speeds of the curves of effective coverage rate are slightly different. For these three types of changing lines, the rising of $l_1$ is the slowest, the fastest is the one of $l_3$, and the middle is the one of $l_2$.

This phenomenon can be explained that in the early state it is difficult to diffuse for the mobile nodes densely gathered in the centre region due to the fixed nodes blocking as a barrier. At this time the coverage probability will be high in their neighboring regions, so the mobile nodes will be difficult to spread due to the lower attraction by the coverage holes. In this state the mobile nodes mainly depend on the local and global optimization factors of the particles to update their positions. With the mobile nodes no longer gathering in a dense state, the parameter $c_3$ also increases in larger way; the weight of potential field force effect on the mobile nodes becomes larger and larger. So in the next stage the mobile nodes will be mainly guided by the virtual force of potential field. Testing results indicate that if we start to weaken the effect of potential field in the later half stage gradually, a better convergence speed will be achieved. So the better strategy of $c_3$ changing is just like the variation trend of $l_3$. Apparently, in the early half stage of the algorithm implementing, with the mobile nodes scattering out gradually, the effect of the potential field force acting on the mobile nodes should increase accordingly and the value of parameter $c_3$ should be improved to enhance the influence of the velocity component of potential field force. This can accelerate the convergence speed of PFPSO algorithm effectively. In order to get a steady convergence, in the late half stage of performing, the value of parameter $c_3$ should be decreased gradually.

So two conclusions could be achieved: (i) The convergence speed will be different if the changing trend of $c_3$ is different. (ii) During the PFPSO performing, in order to get rid of the barrier effect and get a better convergence speed, the local and the global optimization factors of the particles should take active role to guide the mobile nodes to update their location at the beginning stage, and the virtual force generated by attractive field of the coverage holes should take a main role to guide the mobile nodes to update location in the middle stage. In the late stage, the effect of potential field should be restrained gradually.

7. NPFPSS Performance Analysis

Assuming the target environment is a square region with the area of 200 m $\times$ 200 m the hotspot region $A_{hot}$ is a square area with 100 m $\times$ 100 m, and the rest is the ordinary region. At the initial stage, 100 sensor nodes consisting of 60 fixed nodes and 40 mobile nodes are deployed randomly in the target area. The detection radius of each node is $r = 14$ m, the parameter of detection reliability of sensor nodes is $r_e = 7$ m, and communication radius is $R_c = 2(r + r_e) = 42$ m. The maximum moving step of velocity component of potential field in formula (20) is MaxStep = 1 m. The other parameters of probabilistic detection model are set as follows: $\alpha_1 = 1$, $\alpha_2 = 0$, $\beta_1 = 1$, and $\beta_2 = 1.5$. The effective detection thresholds of hotspot region and ordinary region are $c_{th,hot} = 0.95$ and $c_{th,ordinary} = 0.75$, respectively. The coverage diagram of initial state is shown as Figure 10.

The acceleration factors in NPFPSS are set as follows: $c_1 = c_2 = c_3 = 1$, the maximum iterations $\text{Maxiter} = 50$, the number of particles is $M = 20$, and the inertia coefficient...
The region of dotted frame in Figure 11(b) is the hotspot area. The mobile nodes moved from the original “∗” to the terminal “○.” We can find that originally all the nodes distribute randomly in the target area. With NPFPSO performing, the number of mobile nodes in dotted frame increases. After optimization, the number of increased mobile nodes in the center frame is 7, and the effective coverage rate of hotspot region increases 38.61% compared to its initial state. The red line in Figure 11(c) represents the effective coverage rate of hotspot areas and the blue line represents the effective coverage rate of ordinary area.

A set of Pareto optimal solutions with different weight parameter $\alpha$ is given in Table 2. The effective coverage of hotspot area and ordinary area is listed inside.

Comparing the two circumstances above, adjusting the value of $\alpha$ can change the effective coverage when the number of mobile nodes is 40. When $\alpha \in [0.6, 0.7]$, the effective coverage of the two regions will approach a high level. When the number of mobile nodes is 50, the regulation effect of

**Table 1: Coverage comparison for these three algorithms.**

| Coverage rate         | Initialization | PSO   | APF   | PFPSO |
|-----------------------|----------------|-------|-------|-------|
| After running 50 iterations | 60.17%         | 79.93% | 80.97% | 89.97% |
| After running 100 iterations | 60.17%         | 79.57% | 84.67% | 91.03% |

is set as $w = 0.9 - 0.5 \times (1 - t/\text{Maxiter})$ according to the experiences.

We can verify the nonuniform deployment algorithm with two types of hotspot region distribution: (i) The hotspot region is in the center of target area. (ii) The hotspot region is in the left bottom corner of target area.

**7.1. Hotspot Region in the Center.** Figure 11 illustrates the coverage state, displacement of mobile nodes, and coverage curves when NPFPSO algorithm is performed for 50 iterations with weight parameter $\alpha = 0.6$ in formula (23).
Table 2: Comparison table of effective coverage with different weight parameter $\alpha$ when hotspot region is in the center of the target area.

| Fixed nodes : mobile nodes | Coverage of hotspot region | Coverage of ordinary region |
|----------------------------|----------------------------|----------------------------|
|                            | $\alpha = 0.5$ | $\alpha = 0.6$ | $\alpha = 0.7$ | $\alpha = 0.5$ | $\alpha = 0.6$ | $\alpha = 0.7$ |
| 60 : 40                    | 90.65%          | 99.22%          | 97.11%          | 96.89%          | 97.31%          | 96.44%          |
| 50 : 50                    | 98.20%          | 99.59%          | 98.90%          | 97.97%          | 97.53%          | 98.23%          |

Figure 7: Effective coverage curves of the three algorithms.

Figure 8: Three changing lines of parameter $c_3$.

Figure 9: Effective coverage curves by performing PFPSO based on three changing lines of $c_3$.

Figure 10: Initial coverage state for ununiform coverage testing.

Parameter $\alpha$ is no longer obvious and the coverage of two regions will all approach a high level.

7.2. Hotspot Region in the Left Bottom Corner. Figure 12 illustrates the coverage state, displacement of mobile nodes, and coverage curves when NPFPSO algorithm is performed for 50 iterations with weight parameter $\alpha = 0.4$ in formula (23).

As Figure 12(b) is illustrating, with NPFPSO performed, the number of mobile nodes in left bottom dotted frame increases, and the increased number of mobile nodes is 9. The effective coverage rate of hotspot region increases 43.84%
### Table 3: Comparison table of effective coverage with different weight parameter $\alpha$ when hotspot region is in the left bottom of the target area.

| Fixed nodes: Mobile nodes | Coverage of hotspot region | Coverage of ordinary region |
|---------------------------|---------------------------|-----------------------------|
|                           | $\alpha = 0.3$ | $\alpha = 0.4$ | $\alpha = 0.5$ | $\alpha = 0.3$ | $\alpha = 0.4$ | $\alpha = 0.5$ |
| 60:40                     | 76.29%            | 99.95%             | 99.93%         | 97.31%            | 96.78%             | 94.58%         |
| 50:50                     | 87.79%            | 99.91%             | 99.93%         | 98.40%            | 94.35%             | 96.25%         |

![Nonuniform coverage diagram of NPFPSO with hotspot region in the center](image1)

![Displacement of mobile nodes in nonuniform coverage by NPFPSO with hotspot region in the center](image2)

![Effective coverage curves by NPFPSO with hotspot region in the center](image3)

**Figure 11:** Nonuniform coverage by NPFPSO with hotspot region in the center ($\alpha = 0.6$, fixed nodes: mobile nodes = 60:40).

compared to its initial state. The red line in Figure 12(c) represents the effective coverage rate of hotspot areas and the blue line represents the effective coverage rate of ordinary area.

A set of Pareto optimal solutions with weight parameter $\alpha$ taking 3 different values is listed in Table 3.

Comparing the circumstances above, when the hotspot region is at the left bottom corner, mobile nodes will concentrate more in this hotspot region. Effective coverage of two kinds of regions will all approach a better state while the weight parameter $\alpha \in [0.4, 0.5]$.

Since the algorithm presented in the paper is a centralized algorithm, the sensor nodes do not need to move with each iteration of the algorithm. The positions of sensor nodes are only updated at the end of the whole iteration process. So the movement trajectory of a node is nearly a straight line. As a result, compared with the distributed algorithms in which the locations of the nodes will be updated with each iteration, the
calculation of energy consumption of sensor nodes here is not so crucial in this case. We just suppose all the mobile nodes have the ability to move to the optimal positions for normal operation within the target area. Regarding the problem of energy efficiency, we had presented other papers to study and analyze it particularly.

8. Conclusion

Deployment algorithm of network is one of the core technologies of wireless sensor networks. Position optimization for mobile nodes can improve the effective coverage rate of hybrid sensor networks. This paper proposed the PFPSO algorithm which improves the typical PSO with the potential field to enhance the convergence speed of optimization, and it can overcome the barrier effect from fixed nodes for the deployment of hybrid sensor networks. The method can drive mobile nodes with dense state to spread to the coverage holes rapidly under the performing of global optimization in the early stage, and then the virtual force generated by potential field can direct the particles updating towards the optimal position. The proposed method can improve the network coverage rate significantly in hybrid sensor networks. On the basis of PFPSO, NPFPSO algorithm was proposed to solve the nonuniform coverage problem by converting the multiobjective optimization problem into a single objective optimization problem. Attractive force generated by potential field can accelerate the convergence speed of the algorithm. The fitness function can be regulated by adjusting the weighting factor $\alpha$ to direct the mobile nodes gathering to the hotspot region. NPFPSO can achieve an optimal result to effective coverage for both hotspot region and ordinary region in hybrid sensor networks.
In more marine realistic situations, the sensor-carried buoys or shallow buoys could be regarded as the fixed detecting nodes, and the AUVs or other boats with sensor-carried could be regarded as the mobile detecting nodes in marine environment. If we just need to gather information for a target region on the sea, and some of the detecting nodes are mobile nodes, the FPFSO algorithm is applicable to be used for optimizing the deployment of the sensor network. However, sometimes we need to do the key monitoring for some key regions. For marine application, we need to emphatically detect the coastal water quality in drain outlet or harbor areas and search and rescue people in some key areas of ocean in which the accidents happen frequently, and also we need to focus on the region with high event occurrence probability in the ocean. For all these circumstances above, the NPFPSO algorithm is suitable to be used for optimizing the effective coverage for hotspot region and ordinary region. Clearly accurate surveillance is required for these hotspot regions and low accuracy surveillance for less focal regions. Anyway, the proposed algorithms could be of more significance to solve these kinds of optimal problems of deployment of sensor networks in practice.

Conflict of Interests

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

Acknowledgments

This work was supported by National Nature Science Foundation of China (no. 61273068), Nature Science Foundation of Shanghai (no. 12ZR142600), and Scientific Research Innovation Project of Shanghai Education Committee (no. 13YZ084).

References

[1] G. Wang, X. Cao, P. Berman, and T. F. La Porta, “Bidding protocols for deploying mobile sensors,” IEEE Transactions on Mobile Computing, vol. 6, no. 5, pp. 515–528, 2007.

[2] X. Wang, S. Wang, and J.-J. Ma, “Dynamic sensor deployment strategy based on virtual force-directed particle swarm optimization in wireless sensor networks,” Acta Electronica Sinica, vol. 35, no. 11, pp. 2038–2042, 2007.

[3] Y. Zou and K. Chakrabarty, “Uncertainty-aware and coverage-oriented deployment for wireless sensor networks,” Journal of Parallel and Distributed Computing, vol. 64, no. 7, pp. 788–798, 2004.

[4] N. Aitsaadi, N. Achir, K. Boussetta, and G. Pujolle, “Artificial potential field approach in WSN deployment: cost, QoM, connectivity, and lifetime constraints,” Computer Networks, vol. 55, no. 1, pp. 84–105, 2011.

[5] N. Aitsaadi, N. Achir, K. Boussetta, and G. Pujolle, “Potential field approach to ensure connectivity and differentiated detection in WSN deployment,” in Proceedings of the IEEE International Conference on Communications (ICC ’09), pp. 1–6, IEEE, June 2009.

[6] C. Koutsougeras, Y. Liu, and R. Zheng, “Event-driven sensor deployment using self-organizing maps,” International Journal of Sensor Networks, vol. 3, no. 3, pp. 142–151, 2008.

[7] X. Wang, S. Wang, and J.-J. Ma, “Dynamic sensor deployment strategy based on virtual force-directed particle swarm optimization in wireless sensor networks,” Acta Electronica Sinica, vol. 35, no. 11, pp. 2038–2042, 2007.

[8] X. Wang, S. Wang, and J.-J. Ma, “An improved co-evolutionary particle swarm optimization for wireless sensor networks with dynamic deployment,” Sensors, vol. 7, no. 3, pp. 354–370, 2007.

[9] S. Prakash and N. Patidar, “Area based coverage for three deployment distributions and their connectivity in wireless sensor networks,” in Proceedings of the 4th International Conference on Communication Systems and Network Technologies (CSNT ’14), pp. 128–132, IEEE, Bhopal, India, April 2014.

[10] Z.-Z. Yu, J.-H. Yan, J. Zhao, Z.-F. Chen, and Y.-H. Zhu, “Mobile robot path planning based on improved artificial potential field method,” ISRN Robotics, vol. 2012, Article ID 940920, 10 pages, 2012.

[11] H. Adeli, M. Tabrizi, N. A. Mazloomian et al., “Path planning for mobile robots using iterative artificial potential field method,” International Journal of Computer Science Issues, vol. 8, no. 4, pp. 28–32, 2011.

[12] 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.
[22] E. Huang and M. Goetschalckx, "Strategic robust supply chain design based on the Pareto-optimal tradeoff between efficiency and risk," *European Journal of Operational Research*, vol. 237, no. 2, pp. 508–518, 2014.

[23] R. K. Tripathi, Y. N. Singh, and N. K. Verma, "Clustering algorithm for non-uniformly distributed nodes in wireless sensor network," *Electronics Letters*, vol. 49, no. 4, pp. 299–300, 2013.