Localization Algorithm in Wireless Sensor Networks Based on Multiobjective Particle Swarm Optimization

Ziwen Sun, Li Tao, Xinyu Wang, and Zhiping Zhou

School of Internet of Things Engineering, Jiangnan University, Lihu Road 1800, Wuxi 214122, China

Correspondence should be addressed to Ziwen Sun; sunziwen@jiangnan.edu.cn

Received 19 October 2014; Revised 24 January 2015; Accepted 18 February 2015

Academic Editor: Chase Wu

Copyright © 2015 Ziwen Sun 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.

Based on multiobjective particle swarm optimization, a localization algorithm named multiobjective particle swarm optimization localization algorithm (MOPSOLA) is proposed to solve the multiobjective optimization localization issues in wireless sensor networks. The multiobjective functions consist of the space distance constraint and the geometric topology constraint. The optimal solution is found by multiobjective particle swarm optimization algorithm. Dynamic method is adopted to maintain the archive in order to limit the size of archive, and the global optimum is obtained according to the proportion of selection. The simulation results show considerable improvements in terms of localization accuracy and convergence rate while keeping a limited archive size by a method using the global optimal selection operator and dynamically maintaining the archive.

1. Introduction

The research on localization issues is important to the practical application of wireless sensor networks (WSN) technology. Modern WSN applications pose increasingly complex and stringent performance requirements on localization in terms of accuracy. However, the localization methods by using traditional ranging technologies, such as received signal strength indication (RSSI), have the drawback of low accuracy. Some intelligent algorithms have been used to solve the problems of localization including the accuracy [1–3].

Traditionally, most studies focused on using single-objective optimization problem to solve localization problems. Among these methods, the particle swarm optimization (PSO) algorithm is concerned by many researchers for its fast convergence rate and simple implementation. By using particles to imitate the estimated coordinates of unknown nodes, some methods model the localization problem as a single-objective optimization model with the space distance constraint as the only fitness function. For example, the PSO localization algorithm based on log-barrier constraint function could accelerate the convergence speed and save energy [4], the PSO localization adopting crossover operator and the mutation operator could avoid the premature convergence [5], and the PSO localization algorithm based on quantum mechanics could enhance the global convergence and improve the accuracy [6]. However, it always happens that the results of estimated nodes' localizations meet the space distance constraint without meeting the geometric topology constraint because of ranging errors in some practical applications.

Recently, some works have proved the effectiveness of multiobjective optimization algorithms to solve conflict multiple objectives [7–14]. It is more reasonable to model the node localization as a multiobjective optimization problem, which can be described as solving a Pareto solution, rather than simply being described as a single-objective optimization problem. Based on this viewpoint, a multiobjective model was adopted to solve the node localization problem with fitness functions including the localization accuracy and the topological constraint, and the optimal solution was achieved by using the genetic algorithm [7]; however, there are still some problems which are not solved. (i) The estimation accuracy is affected by the selection and mutation operators. (ii) The convergence rate is slow. (iii) The storage consumption of nodes is very big due to the limitless size of the archive for storing the Pareto optimal solutions.

Multiobjective particle swarm optimization has been proved with outperformance in the accuracy and the convergence. A multiobjective multileader PSO was used to handle
an extra objective by constraint handling method with advantage in terms of efficiency [8]. A bare-bones PSO was combined with the sensitivity-based clustering to solve multiobjective reliability redundancy allocation problems as a Pareto optimal solution with high effectiveness [9]. Multiobjective PSO algorithm was used to optimize parameters of Stirling engine with more accurate results than genetic algorithms [10]. The multiobjective swarm optimization problem, which selected the global best particle from a set of Pareto optimal solutions to solve the convergence and the diversity of solutions, was solved by combining PSO with charge system search [11]. The multiswarm multiobjective PSO was decomposed into multiobjective decomposition to solve problems of local optimum, convergence, and accuracy of Pareto solution set [12]. The dynamic constrained multiobjective optimization problem was developed by introducing a local search operator based on attraction into PSO to speed up the optimization process or Pareto optimal solutions for obtaining a high convergence [13].

The main contribution of this paper is to propose a novel algorithm named multiobjective particle swarm optimization localization algorithm (MOPSOLA) for WSN localization to address the issues of the limitless archive and the slow convergence exiting in [7]. MOPSOLA constructs a multiobjective model with both the space distance constraint and the geometric topology constraint as multiobjective functions and solves the localization problem as a Pareto optimization problem. Three kinds of mechanisms are used to insure the higher localization accuracy, the lower storage consumption, and the higher convergence. (i) The geometric topology constraint is considered to reduce the inaccuracy of localization. (ii) The archive is dynamically maintained and limited in a maximum capacity to save the storage space for storing the Pareto optimal solutions. (iii) The global optimum solution is obtained to accelerate the convergence by adopting the mechanism of proportion of selection. The simulation results have proved the proposed algorithm can effectively find the multiobjective optimal solutions and achieve both the better convergence speed and the better localization accuracy.

2. Multiobjective Localization Model

The coordinates of unknown nodes need to meet both the space distance constraint and the geometric topology constraint. The purpose of meeting the space distance constraint is to make the estimated coordinates close to the real values, and meanwhile the purpose of meeting the geometric topology constraint is to make the network topology unique to avoid forming a topological structure which is not in conformity with the actual situation.

Assume that \( n \) nodes are deployed in two-dimensional space \( \mathbb{Z}^2 \) of WSN, including \( m \) anchor nodes and \( n - m \) unknown nodes with \( m < n \). Assuming the two nodes \( i \) and \( j \) being in the communication radius of each other, the internode ranging distance \( d_{ij} \) can be obtained by RSSI ranging technology and denoted by

\[
d_{ij} = r_{ij} + e_{ij},
\]

where \( r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \) is the actual distance between the two nodes and \( e_{ij} \) is the ranging error of RSSI which follows a zero mean Gaussian distribution with variance \( \sigma^2 \) [7].

The neighbour set \( N_i \) and the complement set \( \overline{N}_i \) of node \( i \) are defined as

\[
N_i = \{ j \in 1, \ldots, n, j \neq i : r_{ij} \leq R \},
\]

\[
\overline{N}_i = \{ j \in 1, \ldots, n, j \neq i : r_{ij} > R \},
\]

where \( R \) is the communication radius of node \( i \).

The objective function of the space distance constraint is denoted by

\[
f_1 = \sum_{i=m+1}^{n} \left( \sum_{j \in N_i} (d_{ij} - \tilde{d}_{ij})^2 \right),
\]

where \( \tilde{d}_{ij} \) is the estimated distance between node \( i \) and \( j \), denoted by

\[
\tilde{d}_{ij} = \begin{cases} 
\sqrt{(\hat{x}_i - \hat{x}_j)^2 + (\hat{y}_i - \hat{y}_j)^2} & \text{if } j \text{ is an anchor}, \\
\sqrt{(\hat{x}_i - \hat{x}_j)^2 + (\hat{y}_i - \hat{y}_j)^2} & \text{otherwise},
\end{cases}
\]

where \( (\hat{x}_i, \hat{y}_i) \) and \( (\hat{x}_j, \hat{y}_j) \) are estimated coordinates of unknown nodes \( i \) and \( j \).

The objective function of the geometric topology constraint is denoted by

\[
f_2 = \sum_{i=m+1}^{n} \left( \sum_{j \in N_i} \delta_{ij} + \sum_{j \in \overline{N}_i} (1 - \delta_{ij}) \right).
\]

The geometric topology constraint represents the connectivity constraint which dissatisfies the current estimated positions of nonanchor nodes [7]. And \( \delta_{ij} \) is denoted by

\[
\delta_{ij} = \begin{cases} 
1 & \text{if } \tilde{d}_{ij} > R, \\
0 & \text{otherwise}.
\end{cases}
\]

The space distance constraint and the geometric topology constraint imply the accuracy of the coordinates of the nodes. The high accuracy of estimated coordinates of the unknown nodes consequently lead to the small values of the two objective functions. Therefore, estimating coordinates of the unknown nodes can be modeled to search the optimum solution for the multiobjective optimization issues, which can be obtained by decreasing both values of the objective functions \( f_1 \) and \( f_2 \).

3. MOPSOLA

3.1. Mathematical Description of MOPSOLA. The optimal solution of multiobjective optimization issues is to find the Pareto optimal solution [14]. Multiobjective optimization
issue with \( u \)-dimensional decision vectors and \( v \) objectives is denoted by
\[
\text{Minimize } \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_v(\mathbf{x}))
\]
\[\text{s.t. } \mathbf{x} \in [\mathbf{x}_L, \mathbf{x}_U],\]
where \( \mathbf{x} = (x_1, x_2, \ldots, x_u) \in X \subset \mathbb{R}^u \) and \( \mathbf{F} = (f_1, f_2, \ldots, f_v) \subset Y \subset \mathbb{R}^v \). \( \mathbf{x} \) is called the decision vector belonging to the \( u \)-dimensional decision space \( X \), which is corresponding to \( u \)-particles in PSO. \( \mathbf{x}_L \) and \( \mathbf{x}_U \) are lower and upper bound constraints of the particle range, respectively. \( \mathbf{F} \) is the objective vector belonging to the \( v \)-dimensional objective space \( Y \). The objective function \( \mathbf{F}(\mathbf{x}) \) defines mapping functions from the decision space to the objective space. The set of all the particles meeting the constraints forms the decision space feasible set \( \Omega = \{ \mathbf{x} \in \mathbb{R}^u | \mathbf{x} \in [\mathbf{x}_L, \mathbf{x}_U] \} \).

Some concepts related with MOPSOLA are described as follows.

(i) For two feasible solutions \( \mathbf{x}_a, \mathbf{x}_b \in \Omega \) in the processing of PSO, if, \( \forall i = 1, 2, \ldots, v, f_i(\mathbf{x}_a) \leq f_i(\mathbf{x}_b) \land \exists j = 1, 2, \ldots, v, f_j(\mathbf{x}_b) < f_j(\mathbf{x}_a) \), then call \( \mathbf{x}_a \) dominate \( \mathbf{x}_b \), denoted by \( \mathbf{x}_a \prec \mathbf{x}_b \), and keep \( \mathbf{x}_b \) as optimization results.

(ii) The Pareto optimal solution \( \mathbf{x}^* \) is the decision vector which satisfies \( \forall \exists \mathbf{x} \in \Omega : \mathbf{x} \prec \mathbf{x}^* \).

(iii) \( S = \{ \mathbf{x} \in \Omega | \exists \exists \mathbf{x}^* \in \Omega, f_j(\mathbf{x}) \leq f_j(\mathbf{x}^*), j = 1, 2, \ldots, v \} \) constructs the set of Pareto optimal solution obtained from all of the Pareto optimal solutions.

(iv) \( \mathbf{F}(\mathbf{x}) = \{ \mathbf{F}(\mathbf{x}^*) = (f_1(\mathbf{x}^*), f_2(\mathbf{x}^*), \ldots, f_v(\mathbf{x}^*)) | \mathbf{x}^* \in S \} \) forms the Pareto front consisting of the values of objective functions corresponding to the Pareto optimal set.

(v) The purpose of optimization is to find the Pareto optimal solution.

In MOPSOLA, model the estimated coordinates of \( n - m \) unknown nodes as the decision vectors, and the two objective functions defined by (3)–(5) consist of the objective function \( \mathbf{F}(\mathbf{x}) \). Therefore, from formulas (3)–(5) and (7), MOPSOLA can be formulated in the optimum mathematical form as
\[
\text{Minimize } \mathbf{F}(\mathbf{x}) = (f_1(\tilde{x}, \tilde{y}), f_2(\tilde{x}, \tilde{y}))
\]
\[\text{s.t. } (\tilde{x}, \tilde{y}) \in (\tilde{x}_L, \tilde{y}_L), (\tilde{x}_U, \tilde{y}_U) \quad (8)\]
\[i = m + 1, m + 2, \ldots, n,\]
where decision vectors \( \mathbf{x} = (\tilde{x}, \tilde{y}) (i = m + 1, m + 2, \ldots, n) \) are the estimated coordinates corresponding to particles in PSO, \( (\tilde{x}_L, \tilde{y}_L) \) and \( (\tilde{x}_U, \tilde{y}_U) \) are the lower and upper bound constraint values, \( f_1 \) is the objective function of the space distance constraint, and \( f_2 \) is the objective function of the geometric topology constraint. Therefore the decision space is \( u = n - m \) dimension and the objective space is \( v = 2 \) dimension; that is, the coordinates of unknown nodes are \( u = n - m \) particles, \( f_1 \) and \( f_2 \) are the \( v = 2 \) objective functions.

Obtaining the multiobjective Pareto optimal solution is the ultimate goal of building a multiobjective optimal model for localization issues, which meets both the space distance constraint and the geometric topology constraint. Therefore the main essence of MOPSOLA can be described as determining the dominant relationship according to the decision space feasible set \( \Omega \) and the Pareto front \( \mathbf{F}(\mathbf{x}^*) = \{(f_1(\tilde{x}_1, \tilde{y}_1), f_2(\tilde{x}_1, \tilde{y}_1)) | \mathbf{x}_1 = (\tilde{x}_1, \tilde{y}_1) \in S, i = m + 1, m + 2, \ldots, n \} \), saving Pareto optimal solution set \( S \) in an archive and updating the position and the velocity of multiobjective PSO.

3.2. Describing of MOPSOLA

3.2.1. Overall Framework. Shown as Figure 1, the framework of the proposed multiobjective PSO algorithm includes some
key operators such as maintenance of archive, global optim-
mum selection, and the velocity and localization update. The
particle population relies on an archive to save Pareto optimal
solutions during the iterative process and selecting the global
optimum from these solutions, which is the key point that
the multiobjective PSO is different from the traditional single
objective localization.

Therefore, the localization issue is modeled as a multiob-
jective optimization model in MOPSOLA, and two operators,
which are the dynamic maintenance operator for the archive
and the global optimum selection operator based on propor-
tion of selection, are designed to be suitable for the limited
energy and the poor computing power of WSN nodes.

(i) In the multiobjective function calculation level, func-
tions as formulas (3)–(5) are calculated according to
to all particles, that is, all estimated nodes’ coordinates.

(ii) In the individual optimal selection level, the personal
optimum selection operator works. Based on the con-
cept of Pareto optimality, the pbest of each particle is
chosen between the particle’s current location and its
historical pbest dynamically.

(iii) In the global optimal selection level, the global opti-
mum selection operator works. The proportion of
selection is set for each Pareto optimal solution based
on intensive distance, and a global optimum for each
particle is selected by proportion of selection.

(iv) In the velocity and localization update level, the posi-
tion and velocity update operator works for all indi-
vidual particles. The update process is similar to the
traditional single objective PSO as

\[
v_i(t + 1) = \omega v_i(t) + c_1 r_1 (p_{best} - h_i(t)) + c_2 r_2 (g_{best} - h_i(t)),
\]

\[
h_i(t + 1) = h_i(t) + v_i(t + 1), \quad i = m + 1, m + 2, \ldots, n,
\]

where \(v_i\) is the velocity of the \(i\)th particle, \(h_i = (\bar{x}_i, \bar{y}_i)\)
is the estimated coordinates of the \(i\)th particle vector, \(p_{best}\)
is the best solution for the \(i\)th particle, \(g_{best}\) is the
best solution for the population, \(\omega\) is the inertia
weight, \(c_1\) and \(c_2\) are constants, and \(t\) is the iteration
time.

(v) In the maintenance of archive level, the archive main-
tenance operator works. The maximum capacity of
archive is set as ArcMax and the archive is dynami-
cally updated according to the density distance in the
objective space of the Pareto optimal solution to save
the storage space.

3.2.2. Archive Maintenance Operator. A multiobjective PSO
algorithm does not produce a unique solution but a set of
Pareto optimal solutions at the end of each iteration; therefore
an archive is used to save these Pareto optimal solutions and
the members in the archive become the final solution set
when the iteration stops. To adapt to the limited storage space
of a WSN node, the members in an archive are limited by
deleting some of the members when the maximum capacity
is reached. Consequently, a space will be maintained for the
new solutions entering into the archive.

However, due to nondominated property of Pareto opti-
mal solutions, namely, there is no difference between the solu-
tions, a concept of intensive distance is introduced to evaluate
the solution quality. Generally the point (solution) in a sparse
area which has bigger intensive distance is better than the
point in an intensive area in the objective space [15].

Definition 1. Given set \(S\) is the set of Pareto optimal solutions
in the archive and \(x_i \in S\) is a Pareto optimal solution, then
the intensive distance \(\text{Dis}_i\) of \(x_i\) is defined as the average value
of the minimum distance \(\text{dis}_i^1\) and the second minimum dis-
tance \(\text{dis}_i^2\) between \(x_i\) and other Pareto optimal solutions
\(x_j \in S\) in the archive. \(\text{dis}_i^1, \text{dis}_i^2,\) and \(\text{Dis}_i\) are computed as

\[
\text{dis}_i^1 = \min \{D_{ij} \mid x_i, x_j \in S, j = 1, 2, \ldots, c, j \neq i\},
\]

\[
\text{dis}_i^2 = \min \{D_{ij} \mid D_{ij} > \text{dis}_i^1, x_i, x_j \in S, j = 1, 2, \ldots, c, j \neq i\},
\]

\[
\text{Dis}_i = \left(\frac{\text{dis}_i^1 + \text{dis}_i^2}{2}\right), \quad (10)
\]

where \(D_{ij} = \sqrt{\sum_{k=1}^{2}(f_k(x_i) - f_k(x_j))^2}\) is the objective con-
straint distance between \(x_i\) and \(x_j\) and \(f_k (k = 1, 2)\) is the
objective function, \(c\) is the number of solutions.

The intensive distances of Pareto optimal solutions are
sorted and the smaller ones are deleted in order to limit the
number of members in the archive.

It is ensured that the members in the archive can be
limited in the maximum capacity by the archive maintenance
operator, which avoids the nondominated solutions increas-
ing infinitely to reduce the efficiency along with the evolution.
At the same time, the most intense individual is deleted, and
the uniform distribution of the Pareto front is ensured by
saving lots of scattered individuals.

3.2.3. Global Optimum Selection Operator. In the traditional
single objective PSO algorithm, it is tolerable to use the opti-
mal solution with the best fitness as the global optimum \(g_{best}\)
because the \(g_{best}\) of every particle is the same. However, in
multiobjective PSO the population generates more than one
nondominated \(g_{best}\) leading difference among some particle
global optimums, which results to necessarily choose the
appropriate global optimum. Based on the intensive distance,
the global optimum is selected for each particle by adopting
the proportion of selection.

Definition 2. Given that the number of Pareto optimal solu-
tions in the archive of the current iteration is \(c\) with \(c \leq
ArcMax\), a proportion of selection \(\xi_i\) of a Pareto optimal
solution \( x_i \in S \) is defined as the rate of its intensive distance to the sum of intensive distances in the archive according to

\[
\xi_i = \frac{\text{Dis}_i}{\sum_{k=1}^{c} \text{Dis}_k}, \quad i = 1, \ldots, c. \tag{11}
\]

Formula (11) implies that the larger intensive distance a Pareto optimal solution has, the bigger proportion of selection it has; therefore it has a bigger proportion to be selected as the global optimum.

3.2.4. The Pseudo Code of the MOPSOLA. The main components of MOPSOLA, including multiobjective function calculation, individual optimal selection, maintenance of archive, global optimal selection, and velocity and localization update, are described as function MopsoFuc with pseudocode in Algorithm 1. Where the node is the class of node information with four members of node.x, node.y, node.id, and node.anchorflag corresponding to the x-coordinate, y-coordinate, id number, and anchor flag of a node. \( t \) is the current iteration time and \( t_{\text{max}} \) is the maximum iteration time. The other variables have the consistent meaning with the same named variables aforementioned.

4. Simulations and Analysis

To validate the performance of the MOPSOLA, the simulations have been done to test MOPSOLA and the other two algorithms, the traditional PSO and PASE \([7]\), in Matlab 2010b. The simulations have been done using MOPSOLA with objective functions \( f_1 \) and \( f_2 \), and they also have been done using PSO with objective function \( f_1 \) and using PASE \([7]\) with objective functions \( f_1 \) and \( f_2 \).

4.1. The Simulation and Evaluation Parameters

4.1.1. The Simulation Parameters. Total \( n \) nodes are randomly deployed in 100 m x 100 m area with \( m \) anchor nodes being randomly generated from these nodes. Assuming that the RSSI ranging error \( e_{ij} \) follows a Gaussian distribution with a zero mean and variance \( \sigma^2 \), \( \sigma^2 = 0.01 \), \( \beta = 0.1 \). The maximum capacity of the archive \( ArcMax \) is 10 and the max iteration time \( t_{\text{max}} \) is 1000.

4.1.2. The Localization Errors. The localization error is generally related to the node communication radius. Based on the communication radius, two kinds of localization errors are defined to evaluate the localization performance of the proposed algorithm. One is the single localization error \( \text{SingleError} \) of an unknown node defined by formula (12), which is used to evaluate the performance of each node being estimated. The other is the average localization error \( \text{AverError} \) of the network defined as formula (13), which is further applied to evaluate the performance of all the nodes being estimated:

\[
\text{SingleError} = \frac{\sqrt{(\hat{x}_i - x_i)^2 + (\hat{y}_i - y_i)^2}}{R} \times 100\%, \tag{12}
\]

\[
\text{AverError} = \frac{\sum_{i=1}^{n-m} \sqrt{(\hat{x}_i - x_i)^2 + (\hat{y}_i - y_i)^2}}{(n - m) \times R} \times 100\%. \tag{13}
\]

4.2. The Efficiency of the Two Objective Functions

4.2.1. The Convergence Rate. Figure 2 is the comparison of objective function \( f_1 \) about the convergence rate between MOPSOLA and PAES. MOPSOLA and PAES start to converge at the 400th iteration and 700th iteration, respectively. The result shows that MOPSOLA retains more outstanding individuals to accelerate the algorithm convergence by avoiding the affection of selection and mutation operation on the next generation and adopting dynamic maintenance of archive combined with selecting the optimum by proportion of selection for maintaining much more outstanding individuals.

4.2.2. The Localization Accuracy. The single localization errors of 80 unknown nodes has been simulated to evaluate the localization accuracy under the condition of 20% anchor nodes by using MOPSOLA and PSO algorithm.

(i) The simulation results show that the maximum and the minimum single localization error are 81.13% and 1.52%, respectively, in MOPSOLA compared to 94.36% and 5.52%, respectively, in PSO.

(ii) The ranges of \( \text{SingleError} \) are counted to further evaluate the accuracy shown in Table 1. It can be seen that 24 nodes’ single localization errors fell into 0~5% in MOPSOLA compared to 10 nodes’ in PSO, and 5 nodes’ single localization errors are over 30% in MOPSOLA compared to 31 nodes’ in PSO.
\[ S = \text{MopsoFuc}(\text{node}, n, m, R) \]

\[ \text{node: class of node information} \]
\[ n: \text{total number of nodes} \]
\[ m: \text{anchor nodes} \]
\[ R: \text{node communication radius} \]
\[ S: \text{set of Pareto optimal solutions in the archive} \]

1. Initialize each parameter, \( \omega = 0.7298, c_1 = c_2 = 1.4962, \) \( \text{ArcMax} = 10 \)
2. Initialize \( h \) and \( v \) for each particle randomly then \( \text{pbest} \leftarrow \text{particle} \)
3. While \( t \leq t_{\text{max}} \) Do
   \[ \text{// Step 1. Multi-objective Function Calculation} \]
   4. For each particle Do
   5. Compute \( \text{particle}.f_1 \) and \( \text{particle}.f_2 \) using \text{class node}
   6. End

   \[ \text{// Step 2. Individual Optimal Selection} \]
   7. For each \( \text{pbest} \) Do
   8. Compute \( \text{pbest}.f_1 \) and \( \text{pbest}.f_2 \) using \text{class node}
   9. If \( \text{pbest} < \text{particle} \)
      10. \( \text{pbest} \leftarrow \text{pbest} \)
   11. Else If \( \text{particle} < \text{pbest} \)
   12. \( \text{pbest} \leftarrow \text{particle} \)
   13. Else \( a \leftarrow \text{rand} \)
   14. If \( a \leq 0.5 \) then \( \text{pbest} \leftarrow \text{pbest} \)
   15. Else \( \text{pbest} \leftarrow \text{particle} \)
   16. End

   \[ \text{// Step 3. Maintenance of Archive} \]
   17. While Receive \( \text{pbest} \) Do
   18. \( \text{TempArc} \leftarrow \text{Archive} \cup \text{pbest} \)
   19. Delete repetitive solution in \( \text{TempArc} \)
   20. For every member in \( \text{TempArc} \) Do
   21. IF dominated then Delete
   22. End
   23. \( c \leftarrow \text{number of TempArc} \)
   24. IF \( c \leq \text{ArcMax} \)
      25. \( \text{Archive} \leftarrow \text{TempArc} \)
   26. Else Compute \( \text{Dis of TempArc} \)
   27. Descending sort of \( \text{TempArc} \)
   28. \( \text{Archive} \leftarrow \text{TempArc} \)
   29. End

   \[ \text{// Step 4. Global Optimal Selection} \]
   30. For every member of \( \text{Archive} \) Do
   31. Compute \( \text{Dis} ; \)
   32. Compute \( \xi_i ; \)
   33. End
   34. For every particle Do
   35. Using roulette wheel to choose \( \text{gbest} \) according to \( \xi_i ; \)
   36. End

   \[ \text{// Step 5. Velocity and Localization Update} \]
   37. \( v(t + 1) \leftarrow \omega v(t) + c_1 r_1(p\text{best} - h(t)) + c_2 r_2(g\text{best} - h(t)) \)
   38. \( h(t + 1) \leftarrow h(t) + v(t + 1) \)
   39. End (While)
   40. \( S \leftarrow \text{Archive} \)

\text{Algorithm 1: The proposed MOPSOLA in pseudocode.}

\text{Table 1: The distribution of SingleError ranges.}

| SingleError | 0~5% | 5%~15% | 15%~30% | 30%~45% | 45%~60% | >60% |
|-------------|------|--------|---------|---------|---------|------|
| PSO         | 10   | 26     | 13      | 7       | 14      | 10   |
| MOPSOLA     | 24   | 34     | 17      | 3       | 0       | 2    |
Table 2: Localization errors with different node number.

| Total number of nodes | 60     | 80     | 100    | 120    | 140    | 160    |
|-----------------------|--------|--------|--------|--------|--------|--------|
| PSO                   | 37.20% | 26.43% | 21.02% | 19.58% | 19.39% | 18.38% |
| PAES                  | 22.52% | 21.59% | 15.65% | 13.77% | 12.56% | 11.70% |
| MOPSOLA               | 17.68% | 14.47% | 12.15% | 12.27% | 11.47% | 9.25%  |

Table 3: Localization errors with different anchor node proportion.

| Anchor node proportion | 5%    | 10%   | 15%   | 20%   | 20%   | 30%   |
|------------------------|-------|-------|-------|-------|-------|-------|
| PSO                    | 36.57%| 28.33%| 22.19%| 19.94%| 18.17%| 17.77%|
| PAES                   | 18.47%| 15.19%| 12.88%| 12.10%| 11.87%| 11.02%|
| MOPSOLA                | 15.57%| 13.24%| 12.17%| 11.55%| 10.99%| 10.54%|

(iii) From the comparison, it is obvious that the single localization errors by using MOPSOLA are lower than those by using PSO. This is because the geometric topology constraint is limited in a reasonable topology by the objective function $f_2$, which decreases the inaccuracy of the localization.

4.3. The Robust Performance of the MOPSOLA. The simulations focusing on the average localization errors have been done to evaluate the robustness of the MOPSOLA in case of different situations with varying nodes density, anchor nodes, and network connectivity.

4.3.1. The Effect of the Network Nodes Density. Table 2 and Figure 3 report the average localization errors, measured under the condition of changing the network nodes density and the total number of nodes, while holding on the anchor node proportion as 20% and the communication radius $R = 25$ m.

(i) All the average localization errors of three methods reduce as the nodes increasing, and the number of nodes does little effect on the errors when it is over 100. It is obvious that the errors created by using PAES and MOPSOLA are lower than the errors created by using PSO due to the geometric topology constraint being considered by the first two methods.

(ii) Furthermore, the MOPSOLA algorithm has better performance than PAES due to the estimation accuracy being affected by the selection and mutation operators in PAES, and the average error reduces 4.84%, 5.12%, 3.47%, 1.50%, 1.09%, and 2.45%, respectively, compared to PAES with node number of 60, 80, 100, 120, 140, and 160.

(iii) The further analysis shows that although the network topology changes following the nodes increasing or decreasing, the MOPSOLA is robustness in terms of having the average localization errors lower than 15% with nodes over 80.

4.3.2. The Effect of the Proportion of Anchor Nodes. Table 3 and Figure 4 show the relationship between the average localization errors and the anchor node proportion while...
Table 4: Localization errors with different network connectivity in different algorithms.

| Communication radius $R$ | 10  | 15  | 20  | 25  | 30  | 35  |
|--------------------------|-----|-----|-----|-----|-----|-----|
| PSO                      | 35.73% | 30.94% | 24.59% | 18.94% | 19.76% | 17.88% |
| PAES                     | 29.57% | 21.11% | 17.20% | 12.79% | 13.31% | 14.01% |
| MOPSOLA                  | 21.80% | 18.36% | 14.18% | 12.55% | 11.25% | 10.52% |

holding on the total number of nodes $n = 100$ and communication radius $R = 25$ m but with different anchor node proportion.

(i) All the average localization errors decrease as the anchor node proportion increases due to the increasing of anchor nodes around unknown nodes resulting in localization accuracy being improved.

(ii) MOPSOLA and PAES [7] have better performance in localization accuracy than the traditional PSO localization algorithm with the same anchor node proportion due to the two objective functions being considered in MOPSOLA and PAES compared to only one objective function being considered in PSO without the topology constraint. For example, the average localization error in MOPSOLA reduces 15.09% and 8.39% compared with the traditional PSO method, respectively, under the condition of 10% and 20% anchor node proportion.

(iii) MOPSOLA has slightly higher localization accuracy than PAES under the condition of the same anchor node proportions. Compared with PAES, MOPSOLA reduces the average localization error 1.95% and 0.05%, respectively under the condition of 10% and 20% anchor node proportion.

(iv) MOPSOLA is robust with no performance declining in terms of the average localization errors being lower than 15% when the anchor node proportion is over 10%. These results prove that varying the anchor node proportion will not affect the robust performance of MOPSOLA.

4.3.3. The Effect of the Network Connectivity. Table 4 and Figure 5 show the relationship between the average localization errors and the network connectivity while holding the total number nodes $n = 100$ and 20% anchor node proportion, but with different communication radius $R$ to change the network connectivity.

(i) The results show that the localization accuracy of each algorithm has the trend of rising as the increasing of the node communication radius. This is because the increasing of the communication radius will enable the unknown nodes to communicate with more neighbor nodes and further improve the search performance of intelligent optimization algorithms.

(ii) MOPSOLA can keep the average localization errors declining when the node communication radius reaches to 30 m, while the localization error using PAES begins to rising. The reason for this phenomenon is caused by rising of number of the first-level neighbor and the second-level neighbor [7] resulting from the bigger communication radios, which further leads to increasing of the ranging error of two classes unknown nodes defined in PAES.

(iii) The MOPSOLA is robust with no obvious performance decline in terms of the average localization errors kept lower than 15% when the node communication radius is over 20 m.

The simulation results analyzed in this section have proved that, because the model of multiobjective localization considers the influence of geometric topology constraint $f_2$, some localization results which violate the constraint will be modified to further improve the localization accuracy. Furthermore, the optimum by proportion of selection can obtain closer result to the real localization and further improve the localization accuracy.

4.4. The Time Complexity of the Algorithm. Table 5 shows the time complexity of three algorithms with unknown nodes number $M = n - m$, the size of particle swarm $N$, iterations number $t_{max}$, and archive size $ArcMax$. The time complexity of MOPSOLA is obviously higher than that of PSO and PAES.

Table 6 shows the operation times of three algorithms with total nodes $n = 100$, anchor nodes $m = 20$, and communication radius $R = 25$. It can more clearly be seen...
that the time complexity of MOPSOLA is higher than that of the two compared algorithms.

The reason is that the two main operating levels of archive maintenance operator and global optimum selection operator in MOPSOLA will cost more time to repeatedly calculate and compare the values of two objective functions and delete the dominated solutions to obtain the elitism archived strategy, which also result in higher time complexity.

5. Conclusions

MOPSOLA adopts multiobjective localization with objective functions consisting of space distance constraint and geometric topology constraint and solves optimal solutions by adopting both the dynamic maintenance operator for archive and global optimal selection operator based on proportion of selection. Compared with traditional PSO localization algorithm, MOPSOLA can improve the localization accuracy and decrease the convergence rate. Compared with similar multiobjective localization algorithm PAES, the convergence rate is greatly enhanced and the localization accuracy is slightly increased. Under the premise of ensuring localization accuracy, how to further improve the convergence rate and reduce the energy consumption will be the next research emphasis.

Conflict of Interests

The authors declare that they have no conflict of interests to this work.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grant no. 61373126, the Natural Science Foundation of Jiangsu Province of China under Grant no. BK2013107 and the State Scholarship Fund by China Scholarship Council.

References

[1] R. V. Kulkarni, A. Förster, and G. K. Venayagamoorthy, “Computational intelligence in wireless sensor networks: a survey; IEEE Communications Surveys & Tutorials, vol. 13, no. 1, pp. 68–96, 2011.
[2] A. F. Assis, L. F. M. Vieira, M. T. R. Rodrigues, and G. L. Pappa, “A genetic algorithm for the minimum cost localization problem in wireless sensor networks,” in Proceedings of the IEEE Congress on Evolutionary Computation (CEC ’13), pp. 797–804, IEEE, Cancun, Mexico, June 2013.
[3] M. Vecchio, R. López-Valcarce, and F. Marcelloni, “An effective metaheuristic approach to node localization in wireless sensor networks,” in Proceedings of the 8th IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS ’11), pp. 143–145, Valencia, Spain, October 2011.
[4] H. A. Nguyen, H. Guo, and K.-S. Low, “Real-time estimation of sensor node’s position using particle swarm optimization with log-barrier constraint,” IEEE Transactions on Instrumentation and Measurement, vol. 60, no. 11, pp. 3619–3628, 2011.
[5] X. Hu, S. Shi, and X. Gu, “An improved particle swarm optimization algorithm for wireless sensor networks localization,” in Proceedings of the 8th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM ’12), pp. 1–4, IEEE, Shanghai, China, September 2012.
[6] L. Gong, J. Sun, W. Xu, and J. Xu, “Research and simulation of node localization in WSN based on Quantum particle swarm optimization,” in Proceedings of the 11th International Symposium on Distributed Computing and Applications to Business, Engineering and Science (DCABES ’12), pp. 144–148, Guilin, China, October 2012.
[7] M. Vecchio, R. López-Valcarce, and F. Marcelloni, “A two-objective evolutionary approach based on topological constraints for node localization in wireless sensor networks,” Applied Soft Computing Journal, vol. 12, no. 7, pp. 1891–1901, 2012.
[8] M. Shokriani and K. A. High, “Application of a multi objective multi-leader particle swarm optimization algorithm on NLP and MINLP problems,” Computers and Chemical Engineering, vol. 60, pp. 57–75, 2014.
[9] E. Zhang, Y. Wu, and Q. Chen, “A practical approach for solving multi-objective reliability redundancy allocation problems using extended bare-bones particle swarm optimization,” Reliability Engineering and System Safety, vol. 127, pp. 65–76, 2014.
[10] C. Duan, X. Wang, S. Shu, C. Jing, and H. Chang, “Thermodynamic design of Stirling engine using multi-objective particle swarm optimization algorithm,” Energy Conversion and Management, vol. 84, pp. 88–96, 2014.
[11] A. Kaveh and K. Laknejadi, “A novel hybrid charge system search and particle swarm optimization method for multi-objective optimization,” Expert Systems with Applications, vol. 38, no. 12, pp. 15475–15488, 2011.
[12] H. Peng, R. Li, L. L. Cao, and L. X. Li, “Multiple swarms multi-objective particle swarm optimization based on decomposi- tion,” Procedia Engineering, vol. 15, pp. 3371–3375, 2011.
[13] J. Wei and M. Zhang, “Attraction based PSO with sphere search for dynamic constrained multi-objective optimization problems,” in Proceedings of the 13th Annual Genetic and Evolutionary Computation Conference (GECCO ’11), pp. 77–78, IEEE, Dublin, Ireland, July 2011.
[14] M. G. Gong, L. C. Jiao, D. D. Yang, and W. P. Ma, “Research on evolutionary multi-objective optimization algorithms,” Journal of Software, vol. 20, no. 2, pp. 271–289, 2009.
[15] W. Li and X. Zhang, “An improved multi-objective particle swarm optimization algorithm based on pareto,” Computer Simulation, vol. 27, pp. 96–99, 2010.