Sensors & Transducers, Vol. 168, Issue 4, April 2014, pp. 179-184

Sensors & Transducers

Improved Differential Evolution Algorithm for Wireless Sensor Network Coverage Optimization

[1, 2] Xing Xu, [3] Hao Hu, [4] Yu Wu, [5] Weiqin Ying, [1] Yang Zhou
[1] College of Information & Engineering, Jingdezhen Ceramic Institute, 333403, China
[2] State Key Laboratory of Software Engineering, Wuhan University, 430072, China
[3] Library of Huaihua University, 418000, China
[4] School of Computer Science and Education Software, Guangzhou University, 510006, China
[5] School of Software Engineering, South China University of Technology, Guangzhou 510006, China
[1] Tel.: 07988466611
E-mail: whuxx84@aliyun.com, 35434675@qq.com

Received: 15 January 2014 /Accepted: 7 April 2014 /Published: 30 April 2014

Abstract: In order to serve for the ecological monitoring efficiency of Poyang Lake, an improved hybrid algorithm, mixed with differential evolution and particle swarm optimization, is proposed and applied to optimize the coverage problem of wireless sensor network. And then, the affect of the population size and the number of iterations on the coverage performance are both discussed and analyzed. The four kinds of statistical results about the coverage rate are obtained through lots of simulation experiments.

Keywords: Differential evolution, Hybrid algorithm, Coverage optimization, Wireless sensor network, Particle swarm optimization.

1. Introduction

The problem of wireless network coverage optimization, which reflects the quality and effectiveness of network monitoring, is a core and challengingly important component of ecology sensor network research and application [1]. Then a series of sensor networks for ecological optimization algorithms and techniques research are particularly in need. Because of the importance of the wireless sensor network coverage problem, the coverage optimization has become a hot research direction. In recent years, the research work of wireless sensor network coverage control has made some progress. Many coverage control algorithms were proposed for different applications [2, 3].

Intelligence optimization algorithm has become an emerging technology [4], which has many advantages, such as, the decentralized control, multi-agent system, simplicity, implicit parallelism, and easiness for understanding and realizing. These advantages can effectively promote the application of intelligence optimization algorithm. And this intelligence algorithm can play an important role in optimizing the production process, improving production efficiency and effectiveness, and saving resources. Taking the complexity, constraints, nonlinear, multi-local minima and modeling difficulties of the practical engineering problems into account, looking for the suitable intelligence algorithm for the engineering practice is an important research direction. Intelligence algorithm is suitable for solving the complex, non-linear and multi-dimensional optimization problem; therefore, the intelligence algorithms have good application prospects in wireless sensor network coverage optimization problems. Singh et al. applied the approaches which mixed the genetic algorithm with the integer linear programming formulation to solve the Q-coverage problem versions in wireless sensor networks [5]. Sun et al. proposed a hybrid particle swarm optimization, which use the multi-swarm mechanism, to solve wireless sensor network coverage problem [6]. Wang et al. introduced a new copula-based estimation of distribution algorithms to solve the coverage problem [7]. A novel algorithm based on immune-swarm intelligence, which derived from the principle of particle swarm optimization and artificial immune system, is presented to solve the deterministic coverage problems [8]. Focusing on the differentiated or probabilistic coverage, the evolutionary multi-objective optimization algorithm MOEA/DFD is applied to schedule the nodes of a wireless sensor network [9]. Huang et al. proposed a coverage strategy based on fish swarm algorithm to improve the coverage rate and to reduce the redundancy and cost [10].

Differential Evolution (DE) is an evolutionary algorithm based on population differences [11], and DE was proposed by Storn and Price in 1996 to solve the Chebychev Polynomial fitting Problem. Differential evolution algorithm shows the superior performance in the first IEEE Evolutionary Computation competitions. Many improved variants of DE were proposed, and then DE has been widely used in various application fields. In searching some database, there were few papers about using difference evolution algorithm to solve the coverage optimization problem. In summary, a hybrid differential evolution algorithm combined with particle swarm intelligence is proposed to solve the coverage optimization of ecological sensor networks in this paper. Poyang Lake Ecological Economic Zone is the application target of the hybrid algorithm. So, the realistic theoretical significance and application value do exist.

2. Coverage Optimization Mathematical Models

The sensor nodes set S^* is the subset of the set S. $Area(S^*)$ represents the area covered by the nodes set S^*. Area coverage rate is defined as:

$$P_{acr} = {Area(S^*)} \big/ {Area(S)}, \quad (1)$$

However, in practice, it is very complex and difficult to calculate the $Area(S^*)$ [12]. To simplify the calculation, the monitoring area is rasterized into $M_1 \times M_2$ points. Supposing that the coordinates of the rasterized points are (x, y), the probability of the point (x, y) covered by the node s_i will be defined as:

$$p(x,y,s_i) = \begin{cases} 1 & \sqrt{(x-x_i)^2+(y-y_i)^2} \le r \\ 0 & otherwise \end{cases}, \quad (2)$$

Then the probability of the point (x, y) covered by the any node can be defined as:

$$\begin{aligned} p(x,y) = p(x,y,s_1) \mid p(x,y,s_2) \mid ... \\ \mid p(x,y,s_N) \end{aligned}, \quad (3)$$

where, "$|$" is the OR operation, so the $Area(S^*)$ is calculated as:

$$Area(S^*) = \sum_{M_1 * M_2} p(x,y), \quad (4)$$

and then the area coverage rate is redefined as:

$$\begin{aligned} P_{acr} &= {Area(S^*)} \big/ {Area(S)} \\ &= {\sum_{M_1*M_2} p(x,y)} \Big/ {(M_1 * M_2)} \end{aligned}, \quad (5)$$

3. Algorithms

3.1. Particle Swarm Optimization

In the standard PSO algorithm, each optimization problem solution is regarded as a "particle" of the search space. At first, the algorithm is initialized to a group of random particles (random solutions), and each particle has the properties of position and velocity. The particle is updated by tracking two extremes, i.e. the best previous position of the particle itself and the best particle among all the particles. The velocity and position of the particles are updated according to the equations as follows.

$$\begin{aligned} v_{id}^{t+1} = v_{id}^t + R_1 c_1 (P_{id} - x_{id}^t) \\ + R_2 c_2 (p_{gd} - x_{id}^t) \end{aligned}, \quad (6)$$

$$x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1}, \quad (7)$$

where t is the number of the iteration; R_1 and R_2 are the uniformly distributed numbers between 0 and 1; the parameter ω is called the inertia weight; c_1 and c_2 are the acceleration constants. From the perspective of sociology, the first part is the memory item, which represents the influence of the past on the present; the second part, which is associated with a local search, is called as a self-cognitive item, and it represents that the behavior of the particle derives from its own experience; the third part, which is associated with a global search, is known as a social-cognitive item, and it reflects the collaboration and the sharing knowledge between the particles. This means that the movement of the particle is decided by their own experience and the experience of the best companions.

3.2. Differential Evolution

The basic idea of the differential evolution algorithm is that the new temporary population generated from the current population by the mutation and crossover operation; then the two populations adopt the selection operation one-on-one to generate the next new population based on greedy idea. An offspring is generated by the mutation operation as follows:

$$x'_{ij}(t+1) = x_{p_1 j}(t) + \eta(x_{p_2 j}(t) - x_{p_3 j}(t)), \quad (8)$$

where t is the number of the iteration; η is the scale factor; $p_1 \neq p_2 \neq p_3$. The mutation scheme shown in Eq. (8) is also called as DE/rand/1. The extended modes of the mutation rule have been subsequently proposed in the literature [13]. Other versions of the mutation rule are listed as follows: DE/rand/2, DE/best/1, DE/best/2, DE/ current-to-rand/1, DE/current-to-best/1.

3.3. Hybrid Algorithm

By comparing and analyzing the individual update formulas of differential evolution algorithm and particle swarm optimization algorithm, it can be seen that there are some similarities between the two algorithms. And the formula $(P_{id} - x_{id}^t)$ and $(p_{gd} - x_{id}^t)$ can be seen as the differential vectors. The coefficients R_1c_1 and R_2c_2 can be seen as the scale factors. So, the particle swarm optimization can be regarded as a kind of differential evolution algorithm. In the DE algorithm, the differential vector is the difference of two random individuals or the random individual with the best individual. In the PSO algorithm, the differential vector is the difference of the current individual with the best personal individual in history and the current individual with the global best individual. Inspired by the DE/ current-to-rand/1 and the kernel concept of PSO, a novel mutation strategy is proposed.

$$\begin{aligned} x'_{ij}(t+1) = {} & \omega x_{ij}(t) + \eta(x_{mj}(t) - x_{nj}(t)) \\ & + F_1(P_{id} - x_{ij}(t)) + F_2(p_{gd} - x_{ij}(t)) \end{aligned}, \quad (9)$$

where F_1 and F_2 are the scale factors; F_1 and F_2 are equal to R_1c_1 and R_2c_2, respectively. It is named as DE/current-to-the-own-best; current-to-the-global-best/1. The main idea of the Eq. (9) is to generate the offspring according to the solution's own experience and the best experience of neighbors. The process of the hybrid DE algorithm with particle swarm intelligence is as follows:

Step1: Initialize the individuals of the population, and the solutions adopt the real coding method which is widely used in evolutionary algorithms;

Step2: Compute the fitness $f(x_i)$;

Step3: Update the extreme P_i;

Step4: Update the global extreme p_g;

Step5: Generate the offspring x'_{ij} according to the Eq. (9)

Step6: Generate the crossover offspring x_{off}.

for(j=0;j<DIM;j++)/*DIM represents the dimension of the solution*/

{

if rand(0,1) < crossover probability

$x_{off,j} = x'_{ij}$

else

$x_{off,j} = x_{ij}$

}

Step7: *if* $f(x_{off}) < f(x_i)$, Save the individual x_{off} and replace the x_i;

Step8: if the algorithm doesn't stop, go to Step2.

4. Experiments and Analysis

Assuming the wireless sensor nodes are arranged in the square area and the length of the side is 20. The experiments were executed on the 3.0 GHz computer; VC and Matlab were adopted as the simulation environment for wireless sensor network coverage optimization. In the hybrid algorithm, the crossover probability was set to 0.8; ω linearly decreased from 0.9 to 0.4; $c_1 = c_2 = 2$. To assess the performance of the algorithm, a total of 10 statistically independent runs of the algorithm have been performed for each situation.

4.1. The Influence of Population Size on the Coverage Performance

In this sub-section, the influence of population size on the coverage performance was discussed. The number of iteration is 200; the number of the wireless sensor node is 20; the sensing radius is 3; the raster size is 2×2. The population size is set to 10, 20, 30 and 40, respectively. In Table 1, the first column is the population size; the second is the best coverage rate of the 10 runs; the third is the average coverage rate of the 10 runs; the fourth is the worst coverage rate of the 10 runs; the last is the variance of the 10 runs. Fig. 1 is the coverage rate curves when the population size was set four kinds of different values; Fig. 2 is the line chart of the coverage rate when the population size was changing; Fig. 3 is the layout when the population size was 30. From the experimental results shown in the Table 1 and Figs. 1-3, it can be seen that the coverage rate is the optimal when the population size is 30. In other words, the population size is not of “the bigger the better” style. The increase of the population size can improve the coverage rate to a certain extent. However, the computational speed greatly reduces, therefore, in the algorithm the population size should be selected appropriately.

Table 1. The statistical results about the influence of the population size on coverage performance.

Population size	Best coverage rate	Average coverage rate	Worst coverage rate	Variance
10	6.50e-001	6.29e-001	6.10e-001	1.286684e-002
20	6.60e-001	6.38e-001	6.10e-001	1.398412e-002
30	6.80e-001	6.51e-001	6.40e-001	1.449138e-002
40	6.70e-001	6.50e-001	6.30e-001	1.154701e-002

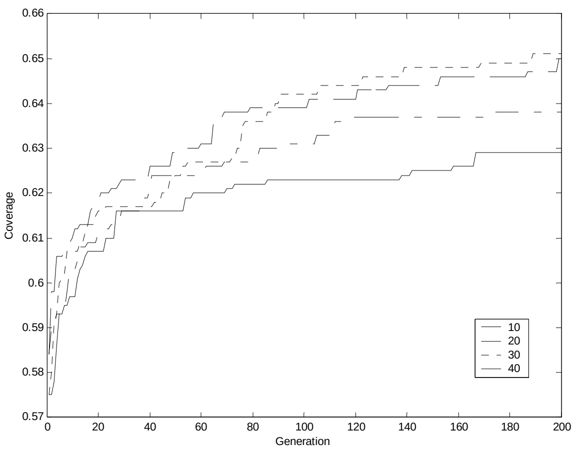

Fig. 1. The coverage rate curves when the population size was set different values.

4.2. The Influence of Iteration Number on the Coverage Performance

In this sub-section, the influence of iteration number on the coverage performance was discussed. The population size is 20; the number of the wireless sensor node is 20; the sensing radius is 3; the raster size is 2*2. The iteration number is set to 50, 100, 150, 200 and 250, respectively. Table 2 is the statistical results about the influence of the iteration number on coverage performance, and the meanings of the five columns are the same with the data in Table 1. Fig. 4 is the coverage rate curves when the iteration number was set with five kinds of different values; Fig. 5 is the line chart of the coverage rate when the iteration number is increasing; Fig. 6 is the wireless sensor nodes layout when the iteration number was 200. When the iteration number is 200, both the best coverage rate and the average rate are the biggest; when the iteration number is 250, the worst coverage rate is the biggest and the variance is the smallest.

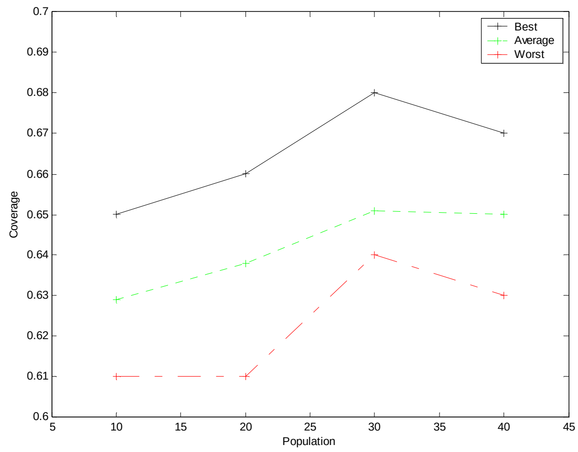

Fig. 2. Three kinds of the line chart of the coverage rate with the population size changing.

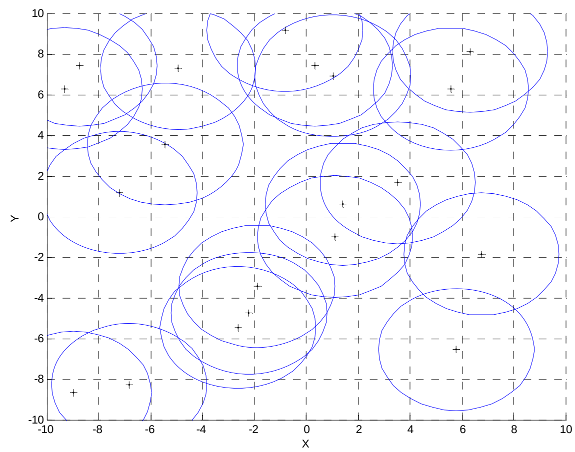

Fig. 3. The node layout when the population size was 30.

Table 2. The statistical results about the influence of the iteration number on coverage performance.

Iteration no.	Best coverage rate	Average coverage rate	Worst coverage rate	Variance
50	6.40e-001	6.21e-001	6.10e-001	1.100505e-002
100	6.50e-001	6.27e-001	6.10e-001	1.159502e-002
150	6.50e-001	6.32e-001	6.10e-001	1.316561e-002
200	6.60e-001	6.38e-001	6.10e-001	1.398412e-002
250	6.50e-001	6.37e-001	6.30e-001	8.232726e-003

It can be concluded that the greater the number of iteration is, the more stable the performance of the algorithm will be. However the increase of the iteration number didn't contribute to the improvement of the coverage rate. Additional iteration number increases the algorithm time on the contrary and the iteration number also needs to select the suitable value.

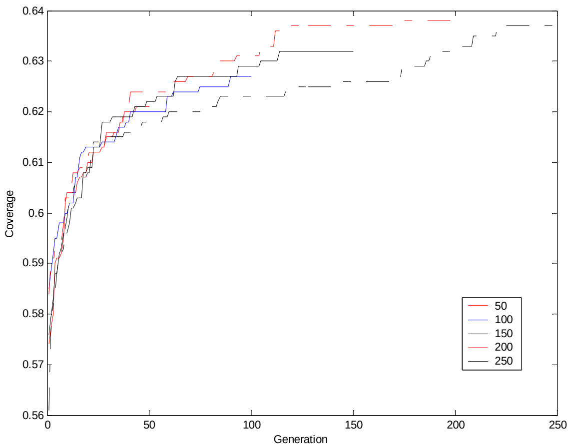

Fig. 4. The coverage rate curves when the iteration number was set with different values.

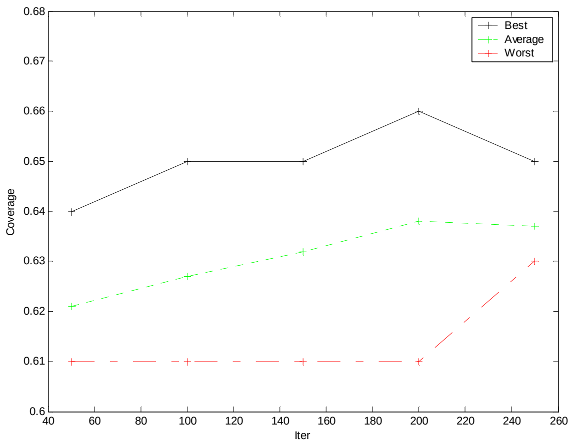

Fig. 5. Three kinds of the line charts of the coverage rate with the iteration number changing.

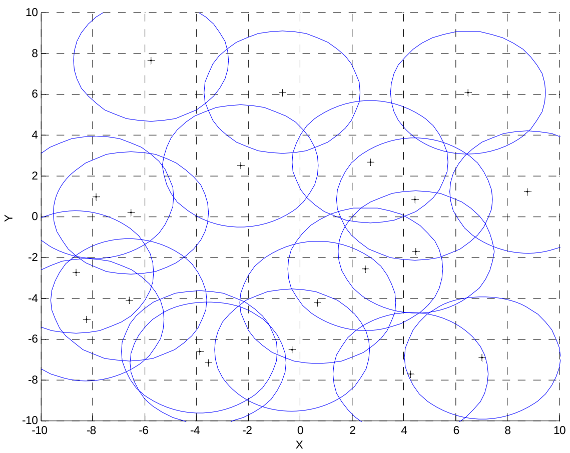

Fig. 6. The node layout when the iteration number was 200.

5. Conclusions

In order to improve the ecological monitoring efficiency of Poyang Lake, by comparing the differential evolution algorithm and particle swarm optimization algorithm, the modified differential evolution algorithm based on particle swarm intelligence for solving ecological sensor network coverage optimization problems is proposed. The factors which affect the coverage rate were analyzed from the two aspects. When the population size is 30, the number of iterations is 200, the coverage rate is the highest. The above analysis and discussion is about the effect of the change of one factor on the coverage rate, the further research will be focused on the influence of the other factors on the coverage performance.

Acknowledgements

This paper was supported by the National Science and Technology Support Plan (2012BAH25F02, 2013BAF02B01), State Key Laboratory of Software Engineering (SKLSE2012-09-35), the National Natural Science Foundation of Jiangxi Province (20122BAB211036, 20122BAB201044), the National Nature Science Foundation of China (61202313,11226225), the Natural Science Foundation of Guangdong Province (S2011040002472, S2012040007241), the Specialized Research Fund for the Doctoral Program of Higher Education (20110172120035), the Fundamental Research Funds for the Central Universities (2011ZM0107), Students Innovation and Entrepreneurship Education Project of Jiangxi Province (201310408034), Special Funds for Visiting Scholars Aiming at the Young and Middle-aged Teachers' Development Plan in Universities of Jiangxi.

References

[1]. Wei Wang, Feng Lin, Jiliu Zhou, Research progress on coverage problem in wireless sensor networks, *Application Research of Computers*, Vol. 27, Issue 1, 2010, pp. 32-35.

[2]. Yanheng Liu, Longxiang Suo, Dayang Sun, Aimin Wang, A virtual square grid-based coverage algorithm of redundant node for wireless sensor network, *Journal of Network and Computer Applications*, Vol. 36, Issue 2, 2013, pp. 811-817.

[3]. Yu-Chee Tseng, Po-Yu Chen, Wen-Tsuen Chen, k-Angle object coverage problem in a wireless sensor network, *IEEE Sensors Journal*, Vol. 12, Issue 12, 2012, pp. 3408-3416.

[4]. Xin-She Yang, Metaheuristic optimization, *Scholarpedia*, Vol. 6, Issue 8, 2011, pp. 11472.

[5]. Alok Singha, Andre Rossi, Marc Sevaux, Matheuristic approaches for Q-coverage problem versions in wireless sensor networks, *Engineering Optimization*, Vol. 45, Issue 5, 2013, pp. 609-626.

[6]. Hui Sun, Jun Li, Wenli Li, Hui Wang, A hybrid particle swarm optimization for wireless sensor network coverage problem, *Sensor Letters*, Vol. 10, Issue 8, 2012, pp. 1744-1750.

[7]. Xiaoping Wang, Huimin Gao, Jianchao Zeng, A copula-based estimation of distribution algorithms for coverage problem of wireless sensor network. *Sensor Letters*, Vol. 10, Issue 8, 2012, pp. 1892-1296.

[8]. Ji-Zhong Liu, Bao-Lei Wang, Jun-Yu Ao, S. H. Wang, Q. M. Jonathan Wu, An immune-swarm intelligence based algorithm for deterministic coverage problems of wireless sensor networks, *Journal of Central South University*, Vol. 19, Issue 11, 2012, pp. 3154-3161.

[9]. Soumyadip Sengupta, Swagatam Das, Md Nasir, Athanasios V. Vasilakos, Witold Pedrycz, An evolutionary multiobjective sleep-scheduling scheme for differentiated coverage in wireless sensor networks, *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews,* Vol. 42, Issue 6, 2012, pp. 1092-1102.

[10]. Yuyue Huang, Keqing Li, Coverage optimization of wireless sensor networks based on artificial fish swarm algorithm, *Application Research of Computers*, Vol. 30, Issue 2, 2013, pp. 554-556.

[11]. Rainer Storn, Kenneth Price, Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces, *Journal of Global Optimization*, Vol. 11, Issue 4, 1997, pp. 341-359.

[12]. Saeed Sedighian Kashi, Mohsen Sharifi, Coverage rate calculation in wireless sensor networks, *Computing*, Vol. 94, Issue 11, 2012, pp. 833-856.

[13]. Ferrante Neri, Ville Tirronen, Recent advances in differential evolution: a survey and experimental analysis, *Artificial Intelligence Review*, Vol. 33, Issue 1, 2010, pp. 61-106.
